ВИЩА ОСВІТА В УКРАЇНІ

ДОСЛІДЖЕННЯ ОПЕРАЦІЙ

ПІДРУЧНИК

Видавництво "Магнолія Плюс"

А. В. Катренко

Дослідження операцій

Підручник

За науковою редакцією доктора технічних наук, професора, завідувача кафедри " Інформаційні системи та мережі " Національного університету " Львівська політехніка " В.В. Пасічника

"Магнолія Плюс"
Львів – 2006

УДК 519.8
ББК 22.183.
К 29

Затверджено Міністерством освіти і науки України для студентів вищих навчальних закладів як підручник

(Лист №1/11 - 4684 від 6 листопада 2003 р.)

Рецензенти:

Цегелик Григорій Григорович – доктор фізико-математичних наук, професор, завідувач кафедри "Моделювання соціально-економічних процесів" Львівського національного університету ім. І. Франка.

Чаплига В'ячеслав Михайлович – доктор технічних наук, професор, завідувач кафедри "Економічна кібернетика" Львівського банківського інституту.

Федасюк Дмитро Васильович – доктор технічних наук, професор, заступник директора Інституту комп'ютерних наук та інформаційних технологій Національного університету "Львівська політехніка".

Катренко А. В.
К 29 Дослідження операцій. Підручник. – Львів: "Магнолія Плюс", 2006.–549 с.

ISBN 966-8340-18-3 "Магнолія Плюс"

У підручнику викладені основні поняття та методологія операційного дослідження, методи та задачі дослідження операцій, наведені алгоритми розв'язання задач пошуку оптимальних рішень. Розглянуті предмет та задачі дослідження операцій, задачі лінійного програмування та методи їх розв'язання, потокові задачі оптимізації на мережах, цілочисельні задачі, основні підходи до розв'язання задач багатокритерійної оптимізації. Викладені методи планування на мережах, теорії ігор, масового обслуговування та управління запасами, динамічного програмування, розв'язання нелінійних оптимізаційних задач.

Зміст підручника відповідає програмі обов'язкового курсу „Дослідження операцій" для базового напрямку „Комп'ютерні науки". З метою закріплення матеріалу з кожної теми наведені приклади розв'язання практичних задач, подано перелік питань та завдання для самостійного виконання.

Розрахований на бакалаврів, спеціалістів та магістрів. Ним можуть також скористатися аспіранти та викладачі ВЗО, де викладається навчальний предмет „Дослідження операцій".

УДК 519.8
ББК 22.183.

ISBN 966-8340-18-3 "Магнолія Плюс"

ЗМІСТ

ПЕРЕДМОВА НАУКОВОГО РЕДАКТОРА

Колектив кафедри „*Інформаційні системи та мережі*" Національного університету „*Львівська політехніка*" спільно з вітчизняними видавцями провадить роботу з підготовки сучасних україномовних матеріалів, які за змістом, формою подання та фаховістю викладення відповідали б сучасному світовому досвіду. Підручник з дисципліни "*Дослідження операцій*" є продовженням проекту з видання серії україномовних посібників та підручників для студентів та магістрів, що навчаються за базовим напрямком "*Комп'ютерні науки*".

Навчальна дисципліна "*Дослідження операцій*" є однією з основних фахово-орієнтованих дисциплін бакалаврату "*Комп'ютерні науки*", а тому й подається у підручнику у відповідності до змісту аналогічних курсів, які викладаються у провідних університетах світу. Приємно відзначити, що протягом останнього часу це вже друга книга автора – попередня – навчальний посібник з курсу "*Системний аналіз*" – видана 2003-го року.

Автор підручника одержав добру фахову підготовку з теорії систем в одному з провідних німецьких університетів, а його подальша наукова та педагогічна діяльність були безпосередньо пов'язані з проблематикою системного аналізу та дослідження операцій, а тому матеріал підручника в багатьох частинах містить авторські результати, які дозволяють краще зрозуміти суть дослідження операцій, а також засвоїти методи розв'язання конкретних задач. Багатий педагогічний досвід та відшліфована методика викладання предмета дозволили автору підручника звести чітку будівлю, яка базується на фундаменті високої математичної культури та досвіду практичних застосувань.

Сподіваюсь, що книгу з задоволенням опрацьовуватимуть як початківці, так і магістри та фахівці з дослідження операцій. Бажаю автору подальших творчих здобутків, а читачам – студентам успіхів у процесі вивчення, засвоєння та успішного складання іспитів та заліків з предмету "*Дослідження операцій*".

З глибокою повагою, д.т.н., завідувач кафедри
„Інформаційні системи та мережі"
Національного університету „Львівська політехніка"
Володимир Пасічник

ПЕРЕДМОВА

Дослідження операцій як наукова дисципліна виникло перед Другою світовою війною, виходячи з військових потреб, і надалі знайшло широке застосування до розв'язання практичних задач в економіці та інших галузях. Звідси і бере початок назва дисципліни - "*Дослідження операцій*". Найважливішим для майбутнього було те, що багато хто з фахівців побачив у цих військових розробках зародження нової науки про функціональні системи, а також можливості застосування отриманих знань у мирний час.

Дослідження операцій (ДО) – це теорія математичних моделей та методів отримання оптимальних розв'язків, що спрямована на обґрунтування доцільності вибору тієї чи іншої альтернативи з множини можливих в області цілеспрямованої діяльності людини. За словами відомого фахівця *Томаса Сааті*, "*Дослідження операцій – це мистецтво давати погані відповіді на ті практичні запитання, на які даються ще гірші відповіді за допомогою інших методів*". Це твердження в напівжартівливій формі стверджує факт, що практичні ситуації прийняття рішень у багатьох випадках настільки складні, що навіть невелика допомога з боку математичних методів суттєво полегшує процес пошуку та обґрунтування розв'язків.

Розвиток методів лінійного програмування в роботах *Л. В. Канторовича* та *Дж. Данціга* поклав початок практичному застосуванню методів дослідження операцій, а теоретичні результати *Г. Куна* і *А. Такера* заклали ґрунт для наступних досліджень в галузі нелінійного програмування.

Широко відомими в світі є роботи українських вчених *О.Г. Івахненка*, який запропонував та розробив метод групового врахування аргументів, величезним є вклад *В.М. Глушкова* в розроблення та впровадження оптимізаційних задач в АСУ, *В.С. Михалевича* та *І.В. Сергієнка* в галузі дискретної оптимізації та багатьох інших. Велику частку у розвиток динамічного програмування вніс американський математик *Річард Белман*, який сформулював основоположний принцип оптимальності.

Задачі планування та керування на мережах є моделями процесів планування та управління складними проектами. На основі методів СРМ та PERT розроблено і широко застосовується програмне забезпечення для планування та управління проектами (як лінійка продуктів фірми Primavera чи Microsoft Project фірми Microsoft). Важливі практичні застосування знайшли задачі управління запасами, ремонту і заміни обладнання та задачі масового обслуговування.

Задачі з активною протидією полягають у визначенні розумної стратегії поведінки за умови конфлікту інтересів учасників, які можуть бути або антагоністичними, або (що відповідає ситуаціям, поширеним в економіці) частково збіжними, а частково протилежними. Задачі транспортування продуктів є задачами вибору маршруту, які найчастіше зустрічаються при дослідженні різноманітних процесів на транспорті та в комп'ютерних мережах (транспорт інформації).

Підручник призначений для широкого кола студентів, які вивчають предмет "*Дослідження операцій*", насамперед комп'ютерних та економічних спеціальностей. Сподіваюся, він буде корисний для всіх, хто прагне пізнати основи теорії та практики знаходження оптимальних рішень.

ПЕРЕЛІК СКОРОЧЕНЬ

ДО – дослідження операцій
ДП – динамічне програмування
ДФП метод – метод Давідсона-Флетчера-Павела
ЛП – лінійне програмування
МП – математичне програмування
ОПР – особа, що приймає (ухвалює) рішення
ОС – оперуюча сторона
СМ – симплекс-метод
СМО – система масового обслуговування
СТ – симплекс-таблиця
ТЗ – транспортна задача
ЦЛП – цілочисельне лінійне програмування
CPM – Critical-Path Method
NP – Nondeterministically Polynomial
P – Polynomial
PERT – Programm Evaluation & Review Technique

РОЗДІЛ 1. ВСТУП ДО ПРОБЛЕМАТИКИ ДОСЛІДЖЕННЯ ОПЕРАЦІЙ ТА ПРАКТИЧНИХ ЗАСТОСУВАНЬ

ТЕМА 1. ПРЕДМЕТ ТА ЗАДАЧІ ДОСЛІДЖЕННЯ ОПЕРАЦІЙ

Дослідження операцій (ДО) – це наука, яка займається розробкою і практичним застосуванням методів оптимального управління організаційними системами. Предметом ДО є системи організаційного управління або організації, що складаються з великої кількості взаємодіючих між собою підрозділів, інтереси яких не завжди узгоджуються між собою і можуть бути повністю чи частково протилежними. ДО служить для кількісного обґрунтування рішень, які приймаються в організаціях, і виходить з того, що якість рішення можна кількісно оцінити за допомогою одного чи декількох критеріїв якості. Як наука, ДО виникло перед Другою світовою війною, виходячи з військових потреб, і надалі знайшло широке застосування до розв'язання практичних задач в економіці та інших галузях. Характерними рисами операційного підходу є: системність, комплексність, орієнтація на прийняття оптимального рішення, телеологічність та комп'ютеризація. Основними поняттями ДО є: операція, оперуюча сторона, активні засоби операції, стратегії оперуючої сторони, діючі фактори операції, стан операції, оптимальний розв'язок, критерій якості. Моделі ДО застосовуються для пошуку оптимальних рішень як в детермінованих, так і в стохастичних системах. За змістом задачі ДО поділяються на наступні: розподілу ресурсів; транспортування продуктів (вибору маршрутів); планування та керування на мережах; формування розкладів; планування та розміщення; управління запасами, ремонту та заміни обладнання; масового обслуговування; прийняття рішень у ситуаціях з активною протидією.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати ⇨предмет та мету дослідження операцій, історію виникнення ДО;⇨основні поняття, риси та послідовність реалізації операційного підходу;⇨основні типи задач ДО;

вміти ⇨розрізняти пряму та обернену задачі ДО; ⇨визначати характер невизначеності в ситуаціях прийняття рішень.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ дослідження операцій
- ☑ стратегії ОС
- ☑ математична модель операції
- ☑ системність
- ☑ стан операції
- ☑ задачі розподілу ресурсів
- ☑ комплексність
- ☑ критерій якості
- ☑ задачі вибору маршрутів
- ☑ оптимальне рішення
- ☑ адекватність моделі
- ☑ невизначеність
- ☑ телеологічність
- ☑ задачі планування на мережах
- ☑ задачі формування розкладів
- ☑ операція
- ☑ пряма задача ДО
- ☑ задачі планування та розміщення
- ☑ оперуюча сторона (ОС)
- ☑ обернена задача ДО
- ☑ задачі управління запасами
- ☑ активні засоби операції
- ☑ детермінованість
- ☑ задачі масового обслуговування
- ☑ діючі фактори операції
- ☑ стохастичність
- ☑ ігрові задачі

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

1.1. Предмет та історія виникнення дослідження операцій (ДО).
1.2. Основні поняття ДО та етапи операційного дослідження.
1.3. Пряма та обернена задачі ДО. Детерміновані задачі ДО.
1.4. Проблема вибору розв'язків в умовах невизначеності.
1.5. Основні класи задач дослідження операцій.

1.1. ПРЕДМЕТ ТА ІСТОРІЯ ВИНИКНЕННЯ ДОСЛІДЖЕННЯ ОПЕРАЦІЙ

Дослідження операцій (ДО) – це теорія математичних моделей та методів отримання оптимальних розв'язків, що спрямована на обґрунтування доцільності вибору тієї чи іншої альтернативи з множини можливих в області цілеспрямованої діяльності людини.

За визначенням Томаса Сааті, "Дослідження операцій – це мистецтво давати погані відповіді на ті практичні запитання, на які даються ще гірші відповіді за допомогою інших методів". Це твердження в напівжартівливій формі стверджує факт, що практичні ситуації прийняття рішень в багатьох випадках настільки складні, що навіть невелика допомога з боку математичних методів суттєво полегшує процес пошуку та обгрунтування розв'язків.

Початки ДО, що виявлялися в науковому мисленні, близькому до операційного, з'явилися досить давно. Гра в кості була однією з найпопулярніших ігор середньовіччя (слово „азарт" походить від арабського „al-zar" – кістка до гри). Вірогідно, найдавнішою книгою з теорії ігор є „Книга про гру в кості" („De Ludo Aleae") Джероламо Кардано (1501-1576), яка була опублікована лише в 1663р., майже через 100 років після написання, а пізніше цією ж проблематикою займався і Галілео Галілей.

Однією з перших ігрових задач, відомою з літератури, була задача про справедливий розподіл ставки між двома гравцями в кості, поставлена в кінці XV століття італійським математиком Фра Лукою Пачолі (1445–1509), що відомий також як винахідник двійкової системи числення та товариш Леонардо да Вінчі (який ілюстрував працю Пачолі „Про божественні пропорції" – „De Divina Proporcione"). Вперше ця задача була опублікована Пачолі в Венеції в 1494р. в огляді середньовічної математики – книзі „Сума знань з арифметики, геометрії, відношень та пропорційності" („Summa de arithmetica, geometria, proportioni et proportionalita"). Неправильний розв'язок запропонував Ніколо Тарталья (1499 -1557), який відомий як відкривач формули для знаходження коренів кубічного рівняння. Після декількох невдалих спроб Блез Паскаль (1623 – 1662) та П'єр де Ферма незалежно один від одного в 1654р. знайшли правильну відповідь для задачі про справедливий розподіл ставки між двома гравцями в кості.

В 1838р. була опублікована книга французького економіста Антуана Курно (1801–1847) "Дослідження математичних принципів теорії багатства", в якій вперше були сформульовані умови економічної

конкурентної рівноваги. Леон Вальрас у книзі "Елементи чистої політичної економії, або Теорія суспільного багатства" багато в чому повторив висновки Курно.

В 1906 р. італійський математик та соціолог, інженер за освітою Альфредо Парето (1848 –1923) видав у Мілані "Підручник політичної економії" та математичний додаток до нього, в якому викладено основні положення аналітичної теорії ігор. Подальший значний внесок у розвиток ігрових моделей був зроблений у 1937р. вченим-емігрантом з Угорщини Дж. фон Нойманом, який працював у США. В 1932р. Дж. фон Нойман запропонував багатосекторну модель економіки, що розширяється, за умови досконалої конкуренції, в якій вперше було введене поняття динамічної рівноваги.

Сам термін "Дослідження операцій" виник у роки Другої світової війни. В 1935р. у Великобританії розпочалася розробка системи виявлення літаків (радіолокаційної системи), що було викликане загрозою з боку військово-повітряних сил Німеччини. Протягом наступних трьох років підтвердилася технічна працездатність розвинутих засобів виявлення та створені практичні методи слідкування за літаками і повідомлень про їх появу. Однак, з метою підвищення ефективності операцій перехоплення, британська винищувальна авіація відчувала потребу в системі супроводу та наведення літаків-перехоплювачів. Тому незалежно (внаслідок секретності) від робіт зі створення системи виявлення на початку 1936р. в кінці 1937р. були початі експерименти з виявлення моделей ворожих літаків та з іншого боку, відслідковування та наведення власних літаків-перехоплювачів.

З кінця 1937р. обидві системи – виявлення та супроводу одиночного атакуючого літака суперника та система супроводу і наведення взаємодіючих перехоплювачів – почали розроблятися сумісно, з метою забезпечення узгоджених операцій всіх учасників бойових дій у повітрі та на землі. В цей же час (1938 р.) у групі дослідників, що керувалася А. Раувом, почав вживатися термін "операційне дослідження". В 1942-му р., коли у війну вступили США, аналогічна група була створена й там (керівником проекту був Ф. Морз – фізик з MTI, – а керівником дослідницької групи був призначений Уільям Шоклі з Bell Telephone Laboratories, який у майбутньому отримав Нобелівську премію як один із винахідників транзистора). Аналогічний відділ був сформований і в Канаді. Загальна кількість таких вчених у часи Другої світової війни налічувала у цих країнах близько 700 осіб.

Діяльність вчених не обмежувалася лише елементами технічних рішень.

а включала й застосування відповідних знань при плануванні тактичних операцій та опрацюванні стратегії військових операцій. Звідси бере початок і назва дисципліни – „Дослідження операцій". Найважливішим у цьому для майбутнього було те, що багато хто з фахівців побачив у цих військових розробках зародження нової науки про функціональні системи, а також можливості застосування отриманих знань у мирний час.

В 1939р. ленінградський математик (у майбутньому академік) Л. В. Канторович у роботі „Математичні методи організації та планування виробництва" в результаті аналізу багатьох проблем організації та планування виробництва сформулював клас умовно-екстремальних лінійних задач та запропонував методи їх розв'язування, що поклало початок лінійному програмуванню.

В 1947р. незалежно від Канторовича симплекс-метод для розв'язування задач лінійного програмування запропонував Дж. Данціг. В 1975 р. Л.В. Канторович отримав Нобелівську премію за видатний внесок у розробку теорії оптимального використання ресурсів в економіці.

Одночасно з розвитком лінійного програмування велика увага зверталася й на задачі нелінійного програмування, у яких або функція мети, або обмеження, або те й інше нелінійні. У 1951 р. була опублікована робота Г.М.Куна і А.В.Такера, у якій наведені необхідні і достатні умови оптимальності для розв'язання нелінійних задач. Ця робота послужила основою для наступних досліджень в галузі нелінійного програмування. А.Чарнес і Ч.Лемке (1954 р.) розглянули наближений метод розв'язання задач із сепарабельними опуклими функціями цілі і лінійними обмеженнями. Починаючи з 1955р., опубліковано багато робіт, присвячених квадратичному програмуванню (роботи А.Франка і Р.Вулфа, Ф.Баранкіна і Р.Дорфмана, Г.Марковича, Ф.Хілдрета й ін.). У роботах Л.Деніса, Б.Розена і Зойтендейка розроблені градієнтні методи розв'язання задач нелінійного програмування.

При дослідженні економічних процесів у більшості випадків їм відповідають задачі нелінійного програмування, а апроксимація їх лінійними задачами викликана тим, що останні добре вивчені і для них існують ефективні алгоритми розв'язання. Навіть найпростіша транспортна задача набуває нелінійного вигляду, якщо вартість перевезення одиниці вантажу залежить від його загальної кількості.

У багатьох задач лінійного і нелінійного програмування процес залежить від часу - від декількох періодів (етапів). При розв'язанні таких задач (вони називаються багатоетапними) необхідно враховувати

поетапний розвиток процесу. Це, наприклад, задача розподілу ресурсів між підрозділами корпорації на декілька років періоду планування. Метод розв'язання задач такого типу складає сутність динамічного програмування. Таким чином, динамічне програмування визначається як математична теорія пошуку оптимального розв'язку багатоетапних задач. Динамічне програмування як самостійна дисципліна сформувалася у п'ятдесятих роках. Великий внесок у його розвиток вніс американський математик Річард Белман, який сформулював принцип оптимальності. Подальший розвиток динамічне програмування одержало в працях Дрейфуса, Робертса, Ланге, Карра, Хоува та ін.

З метою координації та обміну інформацією між фахівцями різних країн в 1957 р. була створена Міжнародна федерація товариств дослідження операцій.

Широко відомими в світі є роботи українських вчених О.Г. Івахненка, який запропонував та розробив метод групового врахування аргументів, що знайшов застосування в прогнозуванні економічних процесів; великим є вклад В.М. Глушкова в розроблення та впровадження оптимізаційних задач в АСУ, В.С. Михалевича та І.В. Сергієнка в галузі дискретної оптимізації та багатьох інших.

Таким чином, ДО сформувалося як науковий підхід до розв'язування задач організаційного управління, що ґрунтується на побудові та дослідженні математичних моделей систем та процесів.

Внаслідок цього *операційний підхід має ряд наступних характерних рис.*

1. Системність. При реалізації операційного підходу важливе значення має системність досліджень, які проводяться, тобто будь-яка задача повинна розглядатися всебічно з різних точок зору, виходячи з ефективності функціонування системи, до якої входить задача, загалом.

2. Комплексність. Як наслідок системності, операційне дослідження повинне здійснюватися комплексно, операційною групою, до складу якої повинні входити фахівці з різних галузей знань: фахівці в галузі інформаційних технологій, математики, програмісти, соціологи, економісти, психологи.

3. Орієнтація на прийняття рішення. Отримані результати повинні визначати спосіб дій – стратегію або тактику, що орієнтований на досягнення оптимальних результатів. Враховуючи, що в багатьох випадках отримання точного оптимального розв'язку неможливо навіть із застосуванням сучасних комп'ютерів, обмежуються отриманням розв'язків, близьких до оптимальних.

4. Телеологічність. Оцінка якості отриманого розв'язку реалізується на основі критерію, що в кількісному вигляді відображає ступінь досягнення мети та якість того чи іншого розв'язку і дозволяє обрати найкращий.

5. Комп'ютеризація. Необхідність використання комп'ютерів пояснюється складністю тих задач, які розв'язуються.

1.2. ОСНОВНІ ПОНЯТТЯ ДОСЛІДЖЕННЯ ОПЕРАЦІЙ ТА ЕТАПИ ОПЕРАЦІЙНОГО ДОСЛІДЖЕННЯ

Основними поняттями ДО є: ⇨*операція,* ⇨*оперуюча сторона,* ⇨*активні засоби операції,* ⇨*стратегії оперуючої сторони,* ⇨*діючі фактори операції,* ⇨*стан операції,* ⇨*(оптимальне) рішення,* ⇨*критерій якості (ефективності).*

Операція - це сукупність взаємноузгоджених дій, що об'єднані єдиним задумом та скеровані на досягнення певної мети. Операція є завжди керованою, тобто деякі зі значень параметрів (змінних) ми можемо обирати самостійно.

Оперуюча сторона – це окремі особи та групи осіб, об'єднані межами операції, що активно прагнуть до досягнення поставленої мети.

Активні засоби проведення операції – це сукупність ресурсів усіх видів та організаційних можливостей, які використовуються оперуючою стороною для забезпечення успішного перебігу операції та її завершення.

Стратегії оперуючої сторони – це припустимі способи використання нею активних засобів.

Діючі фактори операції – це визначені та невизначені об'єктивні умови та обставини, які визначають її особливості та безпосередньо впливають на результат.

Стан операції – це сукупність значень характеристик операції в певний момент часу.

Прийняття рішення – дія, що полягає в виборі значень залежних від оперуючої сторони параметрів.

Оптимальний розв'язок – такий припустимий варіант реалізації операції, який з точки зору системи переваг оперуючої сторони є найліпшим.

Критерій ефективності – це міра очікуваної або досягнутої відповідності між результатом дій, які виконуються, та метою операції.

Математична модель операції – це формальне співвідношення, яке встановлює зв'язок критерію ефективності з діючими факторами операції та визначає припустимі стратегії оперуючої сторони.

Процес ДО включає ряд етапів, основними з яких є наступні.

1. Визначення мети дослідження.

2. Ідентифікація та формулювання проблеми.

3. Побудова моделі операції.

4. Синтез обчислювального методу та розв'язання поставленої задачі за допомогою моделі.

5. Перевірка адекватності моделі.

6. Реалізація результатів дослідження.

Цілі дослідження слід формулювати, виходячи з того, що сподівається отримати в результаті керівник або користувач. Мета роботи не повинна бути сформульована надто вузько, а так само широко, що може привести до безуспішної спроби відразу розв'язати всі проблеми в межах одного дослідження. Потрібно також передбачити, як цілі можуть змінитися в часі.

Ідентифікація проблеми включає формулювання задачі дослідження, виявлення можливих альтернатив стосовно досліджуваної проблемної ситуації, визначення властивих досліджуваній системі вимог, умов та обмежень.

Цей та попередній етапи здійснюються з використанням методів системного аналізу, особливо при дослідженні складних проблем.

На етапі побудови моделі операції розробник або група розробників обирає чи будує модель операції з врахуванням особливостей постановки задачі, причому модель повинна відповідати системі, що досліджується.

При побудові математичної моделі повинні бути встановлені кількісні співвідношення для обчислення значень критерію якості та обмежень у вигляді функцій, що залежать від керованих змінних та враховують дію випадкових факторів і нечіткості. Якщо модель відповідає деякому загальному класу математичних моделей ДО, то для отримання розв'язку використовують відомі математичні методи. В складніших випадках застосовують методи імітаційного моделювання систем.

Розв'язок при використанні математичних моделей отримують за допомогою випробуваних оптимізаційних методів. На цьому етапі, крім визначення оптимального розв'язку, прагнуть отримати інформацію про те, яким чином зміниться оптимальний розв'язок при можливих змінах параметрів моделі, тобто аналізують модель на чутливість. Така інформація необхідна і є цінною в тому випадку, коли деякі з характеристик системи не піддаються точній оцінці.

Загальний метод перевірки адекватності (відповідності) моделі операції полягає в співставленні отриманих результатів із характеристиками системи, які вона мала за тих самих умов у минулому. Модель можна вважати адекватною, якщо вона здатна забезпечити достатньо надійне передбачення поведінки системи. Частково ця проблема розв'язується шляхом проведення експериментів на реальній системі. При розробці нових та складних систем цей метод непридатний і застосовуються методи моделювання систем.

На етапі реалізації отримані результати оформляються у вигляді детальних інструкцій з експлуатації, що забезпечується сумісною роботою всіх членів операційної групи.

1.3. ПРЯМА ТА ОБЕРНЕНА ЗАДАЧІ ДО. ДЕТЕРМІНОВАНІ ЗАДАЧІ ДО

Задачі ДО поділяються на прямі та обернені.

Розв'язок прямої задачі ДО відповідає на запитання: "Що буде, якщо за заданих умов ми оберемо конкретний розв'язок із множини припустимих розв'язків $x \in X$?" і є математичною моделлю критерію якості в залежності від керованих змінних $Q(x)$. Отже, розв'язок прямої задачі дозволяє розрахувати значення критерію якості (чи критеріїв якості, якщо їх декілька) для обраного конкретного розв'язку x.

Обернена задача покликана дати відповідь на питання: " Яке значення x необхідно обрати, щоб $Q(x) \Rightarrow Max$?"

Зрозуміло, що пряма задача є простішою, ніж обернена, і для розв'язання оберненої задачі спочатку необхідно розв'язати пряму. Якщо для одних типів операцій пряма задача є достатньо тривіальною, то для інших побудова математичних моделей та обчислення значень критеріїв якості це не так – для прикладу можуть послужити прямі задачі теорії масового обслуговування.

Щодо оберненої задачі – у випадку, коли кількість можливих варіантів розв'язків невелика, можна застосувати метод прямого перебору, тобто обчислити значення критерію якості для кожного варіанту і обрати один чи декілька з максимальним значенням якості. Однак в практично важливих задачах кількість варіантів розв'язків, на жаль, є значною, і використання прямого перебору є неможливим навіть із використанням найсучасніших комп'ютерів, а тому в цих випадках використовуються методи спрямованого перебору, а також, як варіант, шукається не строго оптимальний розв'язок, а такий (один із таких), що гірші, ніж оптимальний, не більш ніж на певне значення (субоптимальний розв'язок), тобто будь-який з тих, що належать певному околу оптимального розв'язку.

У тому випадку, коли дія невизначеностей відсутня, задача є детермінованою. Всі фактори, від яких залежить успіх (чи невдача) операції, поділимо на дві групи: задані, відомі фактори – умови виконання операції – детерміновані параметри a ; елементи розв'язку x, вибір значень яких залежить від нас як оперуючої сторони. Перша група включає й обмеження, що накладаються на розв'язок, тобто визначає область припустимих розв'язків X, $x \in X$.

Критерій якості розв'язку для детермінованих задач залежить від множини детермінованих параметрів a та керованих змінних x, і в загальному обернена задача ДО формулюється наступним чином: $Q(a,x) \Rightarrow Max,\ x \in X$. Пошук максимального значення критерію не звужує умови задачі, оскільки, якщо задача формулюється у вигляді $Q'(a,x) \Rightarrow Min,\ x \in X$, то її легко перетворити до еквівалентної задачі максимізації шляхом підстановки $Q(a,x) = -Q'(a,x)$.

Розв'язком оберненої задачі ДО є таке значення вектору керованих змінних, яке забезпечує досягнення найбільшого значення критерію ефективності серед всіх можливих значень вектора керованих змінних – це

$$x^* = \arg \underset{x \in X}{Max}\ Q(a,x).$$

Для розв'язування таких задач методи класичного математичного аналізу застосовні лише в незначній кількості випадків. Це пояснюється тим, що навіть за умови неперервності змінних та відсутності обмежень у випадку пошуку екстремуму шляхом прирівнювання до нуля часткових похідних по кожній з керованих змінних у більшості випадків отримуємо систему нелінійних рівнянь, розв'язати яку можливо лише за допомогою чисельних методів, а при наявності обмежень та цілочисельності змінних задача ускладнюється ще більше.

1.4. ПРОБЛЕМА ВИБОРУ РОЗВ'ЯЗКІВ В УМОВАХ НЕВИЗНАЧЕНОСТІ

У багатьох випадках на результат операції впливають неконтрольовані фактори, тобто має місце вибір розв'язків в умовах часткової або повної невизначеності.

Критерій ефективності операції має в цьому випадку вигляд $Q(a,x,y)$, де y – невідомі фактори, і обернена задача ДО формулюється таким чином: при заданих постійних або детермінованих умовах з врахуванням невідомих факторів y необхідно знайти такий розв'язок $x \in X$, який забезпечує отримання оптимального значення Q.

Для розв'язування задач в умовах невизначеності існує ряд можливостей, які залежать від природи випадкових факторів та можливостей їх контролю.

У загальному розрізняються два основні види невизначеності:

☑ "доброякісна" – коли невідомі фактори підкоряються законам теорії вірогідності і дослідникам відомі значення та вид законів розподілу цих факторів (стохастичні задачі);

☑ "погана" – коли фактори не підкоряються законам вірогідності, або параметри законів невідомі.

Шляхи розв'язування стохастичних задач є наступними:

☑ заміна випадкових факторів значеннями їх математичних сподівань чи (у випадку експериментального визначення) – середніх значень та розв'язування задачі як детермінованої:

$Q(a,x,M[y]) \Rightarrow Min,\ x \in X$, де $M[y]$ – вектор математичних сподівань випадкових факторів;

☑ пошук екстремуму математичного сподівання критерію якості: $Q = M[Q(a,x,y)] \Rightarrow Min,\ x \in X$, де $M[Q(a,x,y)]$ – математичне сподівання критерію якості;

☑ введення стохастичних обмежень.

Слід мати на увазі, що перші два прийоми не завжди виправдані, хоча на практиці використовуються доволі широко.

Гірша ситуація виникає в тому випадку, коли невизначені фактори не можуть бути вивчені та описані статистичними методами, а саме:

☑ розподіл вірогідностей для y в принципі існує, однак до моменту прийняття рішення невідомий;

☑ розподіл вірогідностей не існує.

В ДО при розв'язуванні задач з невизначеністю в ряді випадків розглядається ситуація з позиції "скрайнього песимізму", тобто рішення приймається з розрахунку на найгірший збіг обставин, і з розв'язків, які можливі за найгірших умов, обирається найкращий – використовується так званий "принцип гарантованого результату". Якщо у випадку проектування системи захисту від повеней такий підхід цілком виправданий – гребля мусить встояти і за умов найбільших повеней, – то в інших така позиція приводить до перестрахувальних рішень, оскільки будується з розрахунку на те, що "розумний активний супротивник" завжди буде обирати найрозумніші, найкращі розв'язки. Звичайно, воєначальник, який буде виходити з таких позицій, в більшості випадків не виграє битви, і в цьому випадку необхідним є елемент ризику.

1.5. ОСНОВНІ ТИПИ ЗАДАЧ ДОСЛІДЖЕННЯ ОПЕРАЦІЙ

Задачі дослідження операцій класифікують за їх змістовною постановкою наступним чином (рис. 1.1):

☑ задачі розподілу ресурсів;

☑ задачі транспортування продуктів (вибору маршрутів);

☑ задачі планування та керування на мережах;

☑ задачі формування розкладів (календарного та об'ємно-календарного планування);

☑ задачі планування та розміщення;

☑ задачі управління запасами, ремонту та заміни обладнання;

☑ задачі масового обслуговування;

☑ задачі прийняття рішень в ситуаціях з активною протидією (конфліктні ситуації).

Задачі розподілу ресурсів зводяться до розподілу обмежених ресурсів, призначених для виконання певної множини робіт, між ними найефективнішим чином.

Задачі транспортування продуктів – це по суті задачі вибору маршруту, які найчастіше зустрічаються при дослідженні різноманітних процесів на транспорті та в комп'ютерних мережах (транспорт інформації). Розв'язання задач цього типу зводиться до визначення деякого маршруту (чи множини маршрутів) з числа можливих, які є найекономічнішими з точки зору критерію якості (наприклад, найменша вартість чи найкоротший шлях). В цих задачах можуть накладатися обмеження на те, щоб побувати в кожному пункті лише один раз і повернутися в пункт, з якого почався маршрут (задача про комівояжера) або можливість затримки в вершинах (пунктах) мережі (ця ситуація є характерною при моделюванні комп'ютерних мереж зв'язку з проміжними пунктами, де можуть зберігатися протягом певного часу пакети інформації).

Рис.1.1. Класифікація задач ДО за змістом

Задачі планування та керування на мережах є моделями процесів планування та управління складними проектами, що включають до свого складу певну множину наліввпорядкованих робіт, для виконання яких необхідно використати певні об'єми ресурсів різних типів. З одного боку, за відсутності інформації про споживані ресурси можливе планування часових характеристик, з іншого – набагато цікавішого з практичної точки зору - тривалість кожної з робіт проекту залежить від кількості вкладеного ресурсу (ресурсів), і необхідно визначити такий розподіл ресурсів між роботами з врахуванням відношення передування чи слідування на множині робіт, щоб загальна тривалість виконання комплексу робіт (проекту) була мінімальною. Для розв'язування цих практично важливих задач існують ряд відомих програмних пакетів (наприклад, є Microsoft Project фірми Microsoft чи ряд продуктів фірми Primavera).

Задачі календарного та об'ємно-календарного планування тісно пов'язані з задачами планування на мережах і полягають в наступному. Наявна певна множина деталей, кожна з яких має свій заданий технологічний

маршрут проходження множини верстатів та задані часи опрацю-вання на кожному верстаті для кожної з деталей. Необхідно визначити календарні строки опрацювання кожної з деталей на верстатах її техноло-гічного маршруту таким чином, щоб загальний час опрацювання був мінімальним (водночас на одному верстаті може опрацьовуватися лише одна деталь). В сучасних умовах ці задачі, що розглядаються теорією розкладів, використовуються для планування роботи багатопроцесорних систем.

Задачі планування та розміщення об'єктів полягають у визначенні найкращих місць розміщення нових об'єктів (споживачів) за умови наявності в певних місцях продуцентів (наприклад, в якому місці найвигідніше розмістити збагачувальну фабрику, коли відомі місця розміщення енергетичних підприємств та родовищ корисних копалин, в яких видобувається сировина). При цьому об'єкти можуть розглядатися як точки, або ж як такі, що мають певну довжину в просторі.

Задачі управління запасами, ремонту та заміни обладнання полягають у визначенні оптимальних моментів (тобто таких, які забезпечуватимуть мінімальні сумарні витрати) поновлення запасів в коморах (або моментів планово-профілактичних ремонтів та заміни обладнання) при заданому детермінованому чи стохастичному рівні попиту, характеристики якого теж можуть змінюватися в часі.

Задачі масового обслуговування спрямовані на визначення оптимальних характеристик систем обслуговування з чергами заявок на обслуговування (наприклад, черги клієнтів на міжміських телефонних станціях, покупців в крамниці). Якщо кількість обслуговуючих пристроїв буде великою, то великими будуть і їх простоювання в очікуванні клієнтів, якщо ж малою, то в черзі буде багато клієнтів, і деякі можуть бути загубленими внаслідок відмови стояти в черзі.

Задачі з активною протидією полягають у визначенні розумної стратегії поведінки за умови активної протидії, тобто конфлікту інтересів. В задачах цього типу інтереси учасників можуть бути або антагоніс-тичними, або (що відповідає ситуаціям, поширеним в економіці) частково збіжними, а частково протилежними. Моделі прийняття рішень у конфліктних ситуаціях розглядаються в теорії ігор.

ПРИКЛАДИ

Приклад 1.1. Задача про розподіл ставки в грі (Фра Лука Пачолі).

Два гравці, що грають у гру з однаковими шансами на перемогу, домовились про те, що приз (певну суму грошей) отримає той гравець, хто першим виграє 6 партій. Гра припинилася до моменту виграшу одним із гравців 6 партій, а саме – перший гравець виграв 5 партій, а другий – лише 3. Яким чином за цих умов справедливо розділити приз між цими двома гравцями?

Ця задача стосується не лише гри в кості – неважко уявити шаховий матч між двома шахістами з певним призовим фондом до шести перемог, який за умовами весь дістається переможцю. Це ж питання виникне, якщо з певних об'єктивних причин такий матч зупиниться за рахунку 5:3 та не зможе бути продовженим.

Розв'язання.

Існувало декілька відповідей - згідно з однією (Фра Лука Пачолі), приз слід розділити у відношенні 5:3 (пропорційно виграним партіям), а Ніколо Тарталья запропонував поділити у відношенні 2:1 (оскільки перший гравець виграв на дві партії більше, а це складає третю частину від необхідних для перемоги 6 партій, то перший гравець повинен спочатку отримати третину призу, а те, що залишиться після цього, слід розділити порівну).

Насправді ж справедливим є розподіл у відношенні 7:1 !!!

Природно, що справедливим буде розподіл, який пропорційний шансам першого гравця виграти приз. Покажемо, що розподіл у відношенні 7:1 відповідає цій вимозі. Першому гравцеві для перемоги в матчі залишилося виграти одну партію, а другому – три. Продовжимо гру, згідно з ідеєю Ферма, трьома фіктивними партіями, навіть якщо деякі з них виявляться зайвими (тобто коли один з гравців уже виграв приз). У цьому випадку рівноймовірними є всі $2\times2\times2=8$ можливих результатів, при цьому лише в одному випадку (коли 2-й гравець виграє всі три партії) 2-й гравець отримає приз, а в усіх 7 інших – переможе 1-й гравець, внаслідок чого справедливим є розподіл у відношенні 7:1.

Загальний розв'язок для випадку, коли 1-му гравцю для перемоги необхідно виграти n партій, а другому – m, був знайдений Паскалем та Ферма.

Приклад 1.2. Формулювання задач з невизначеністю.

1. Планується асортимент товарів для продажу на книжковій виставці-ярмарку. Бажано обрати такий асортимент та кількості виставленого товару, щоб максимізувати прибуток, але невідома ні кількість відвідувачів-покупців, ані потреби кожного з них.

2. Проектується система споруд та гребель, які охоронятимуть район від повеней. Ні моменти повеней, ні їх розміри невідомі.

3. Розробляється державний план озброєнь на декілька років наперед. Невідомі ні конкретний супротивник, ні його озброєння в майбутньому.

Приклад 1.3. Задача з стохастичною („доброякісною") невизначеністю.

Організується система профілактичного та аварійного ремонту технічних пристроїв з метою зменшити простоїв техніки за рахунок несправностей та ремонтів. Необхідні стохастичні характеристики (середнє напрацювання на відмову за видами обладнання та ін.) можуть бути отримані шляхом збирання відповідної статистики, і на ґрунті цієї інформації можна визначити оптимальні строки профілактичного ремонту для кожного з видів обладнання.

Приклад 1.4. Введення стохастичних обмежень.

Розглянемо як приклад проблематику створення інформаційної системи (ІС) швидкої допомоги. Необхідно розробити такий алгоритм роботи ІС, щоб служба швидкої допомоги функціонувала найефективніше. Для оцінювання якості роботи такої системи в принципі логічним видається вибір критерію T – часу очікування на допомогу, адже чим менше значення часу доїзду до хворого, тим ефективніше функціонує система, і лікарі можуть оперативно надати допомогу.

Однак в такій постановці задача не має сенсу, оскільки величина T є випадковою. Мінімізація середнього часу в цьому випадку приведе до того, що довгий час очікування в одному випадку компенсуватиметься коротким часом в іншому, і ситуація нагадуватиме ситуацію "середньої температури хворих в лікарні" – 36.6°, в той час як пацієнти можуть мати як високу (41°), так і низьку (32 °) температуру. Тому величина T не повинна бути більшою, ніж гранично припустима t_0, $T \le t_0$. Однак безпосередньо таке обмеження ввести в задачу неможливо, так як T – випадкова величина. Оминути цю складність можна у випадку, коли ми задамо рівень граничної вірогідності того, що величина T не буде більшою, ніж β – $P(T \le t_0) \ge \beta$

В цьому випадку задача буде сформульована коректно, натомість введення такого обмеження сильно ускладнить алгоритм і трудомісткість її рішення.

Приклад 1.5. Задача з невідомим розподілом вірогідностей, який в принципі існує.

Проектується інформаційна система для обслуговування випадкових запитів, вірогідні характеристики яких могли би бути отримані за умови дослідження функціонування її протягом певного часу. Однак визначити ці характеристики ми зможемо, експлуатуючи вже побудовану систему. Таким чином виникає хибне замкнене коло – для побудови системи ми повинні володіти даними про її функціонування, а система ще не побудована, і отримати цю інформацію немає де. Підхід до рішення — залишити деякі параметри системи вільними, з можливостями зміни їх значень, та при накопиченні реальної статистики корегувати їх значення в процесі експлуатації.

Приклад 1.6. Задача з неіснуючим розподілом вірогідностей.

"Погана невизначеність". Необхідно спрогнозувати, який одяг будуть одягати жінки через декілька років. В цьому випадку ми можемо лише розглянути всі можливі альтернативи і відкинути ті, які за будь-яких умов будуть гіршими – тобто множина розв'язків звужується та знаходяться такі компромісні розв'язки, які є відносно стійкими.

РЕЗЮМЕ

1.1. Дослідження операцій (ДО) – це теорія математичних моделей та методів отримання оптимальних розв'язків, що спрямована на обґрунтування доцільності вибору тієї чи іншої альтернативи з множини можливих у галузі цілеспрямованої діяльності людини. Найперші дослідження в цій галузі стосувалися вибору оптимальних стратегій в іграх. В 1906 році італійський математик та соціолог, інженер за освітою Альфредо Парето (1848—1923) видав в Мілані "Підручник політичної економії" та математичний додаток до нього, в якому викладено основні положення аналітичної теорії ігор. Сам же термін "Дослідження операцій" виник у роки Другої світової війни. У 1951 р. була опублікована робота Куна і Такера, у якій наведені необхідні і достатні умови оптимальності для розв'язання нелінійних задач.

Операційний підхід має ряд наступних характерних рис: системність, комплексність, орієнтація на прийняття рішення, телеологічність, комп'ютеризація.

1.2. Основними поняттями ДО є: операція, оперуюча сторона, активні засоби операції, стратегії оперуючої сторони, діючі фактори операції, стан операції, (оптимальне) рішення, критерій якості (ефективності). Процес ДО включає ряд етапів: визначення мети дослідження; ідентифікація та формулювання проблеми; побудова моделі операції; синтез обчислювального методу та розв'язання поставленої задачі за допомогою моделі; перевірка адекватності моделі; реалізація результатів дослідження.

1.3. Розв'язок прямої задачі ДО відповідає на запитання: "Що буде, якщо при заданих умовах ми оберемо конкретний розв'язок з множини припустимих розв'язків $x \in X$?" і є математичною моделлю критерію якості в залежності від керованих змінних $Q(x)$. Обернена задача покликана дати відповідь на питання: "Яке значення x необхідно обрати, щоб $Q(x) \Rightarrow Max$?" Для розв'язання оберненої задачі спочатку необхідно розв'язати пряму.

1.4. В багатьох випадках на результат операції впливають неконтрольовані фактори, тобто має місце вибір розв'язків в умовах часткової або повної невизначеності. В загальному розрізняється два основні види невизначеності: "доброякісна" та "погана". Шляхами розв'язування стохастичних задач є наступні: заміна випадкових факторів значеннями їх математичних сподівань; пошук екстремуму математичного сподівання критерію якості; введення стохастичних обмежень. В ДО при розв'язуванні задач з невизначеністю в ряді випадків розглядається ситуація з позиції "скрайнього песимізму", тобто рішення приймається з розрахунку на найгірший збіг обставин, і з розв'язків, які можливі за найгірших умов, обирається найкращий — використовується принцип гарантованого результату.

*1.5. Задачі ДО класифікуються за двома основними класифікаційними ознаками: **формальними моделями та методами** (детерміновані-випадкові, неперервні – дискретні, лінійні – нелінійні); **змістовною постановкою** (задачі розподілу ресурсів, задачі транспортування продуктів (вибору маршрутів), задачі планування та керування на мережах, задачі формування розкладів (календарного та об'ємно-календарного планування), задачі планування та розміщення, задачі управління запасами, ремонту та заміни обладнання, задачі масового обслуговування, задачі прийняття рішень у конфліктних ситуаціях).*

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Назвіть риси операційного підходу та розкрийте їх зміст.
2. Дайте визначення операції, оперуючої сторони, активних засобів операції, стратегії оперуючої сторони, діючих факторів операції, стану операції, оптимального розв'язку, критерію якості.
3. Поясніть зміст основних етапів процесу дослідження операцій.
4. Дайте визначення прямої та оберненої задач дослідження операцій.
5. Сформулюйте проблему вибору розв'язків в умовах невизначеності та вкажіть шляхи розв'язання стохастичних задач.
6. Перелічіть основні типи задач дослідження операцій.
7. Розкрийте зміст задач розподілу ресурсів та транспортування продуктів.
8. Які проблеми розв'язуються за допомогою задач формування розкладів та планування на мережах?
9. В яких галузях знаходять застосування задачі масового обслуговування та управління запасами?

ТЕМА 2. БАГАТОКРИТЕРІЙНІ ЗАДАЧІ ДО ТА ОСНОВНІ ПІДХОДИ ДО ЇХ РОЗВ'ЯЗУВАННЯ

Невизначеність мети є важливим видом невизначеності і виявляється у наявності декількох, в більшості випадків незбіжних аспектів оцінки якості припустимих розв'язків задачі. У формальному вигляді аспекти оцінки якості відображаються за допомогою множини критеріїв, і таким чином природно виникає багатокритерійна задача ДО. Знайти розв'язок, який одночасно був би найкращим за всіма критеріями, неможливо, тому що в загальному випадку покращення значення одного з критеріїв приводить до погіршення значення іншого. В загальному випадку розв'язком багатокритерійної задачі є множина недомінованих (Парето-оптимальних) розв'язків, побудова якої для задач ДО в більшості випадків є неможливою внаслідок значних обчислювальних труднощів. Тому для розв'язування багатокритерійних задач використовуються як методи згортання критеріїв, що не вимагають втручання особи, що приймає рішення (ОПР), так і гнучкіші діалогові методи, які в процесі розв'язання потребують отримання певної інформації від ОПР.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати: ⇨що є в загальному розв'язком багатокритерійної задачі; ⇨методи зведення багатокритерійної задачі до однокритерійної; ⇨переваги та недоліки діалогових методів розв'язання багатокритерійних задач;

вміти: ⇨застосовувати діалогові методи та методи згортання критеріїв до розв'язування багатокритерійних задач; ⇨будувати множину Парето-оптимальних розв'язків для найпростіших задач із заданою скінченою множиною варіантів.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

☑невизначеність мети	☑простір змінних
☑багатокритерійна задача ДО	☑діалоговий метод
☑простір критеріїв	☑ідеальна точка
☑згортання критеріїв	☑поступка за критерієм
☑метрика простору	☑контрольний показник
☑множина Парето-оптимальних розв'язків	☑переведення критерію в обмеження

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

2.1. Основні поняття та постановка задачі.

2.2. Методи згортання критеріїв. Метод "ідеальної точки".

2.3. Переведення критеріїв в обмеження. Контрольні показники. Метод послідовних поступок.

2.4. Поняття про діалогові методи.

2.1. ОСНОВНІ ПОНЯТТЯ ТА ПОСТАНОВКА ЗАДАЧІ

На практиці задачі, що не мають невизначеностей, є скоріше винятком, аніж правилом. Поряд із розглянутими вище існує ще один важливий вид невизначеності – невизначеність мети, що виявляється у наявності декількох, в більшості випадків незбіжних аспектів оцінки якості того чи іншого розв'язку з множини припустимих. У формальному вигляді аспекти оцінки якості відображаються за допомогою множини критеріїв.

Таким чином виникає багатокритерійна задача дослідження операцій, загальний вигляд якої наступний:

$$Q_1(a,x)\Rightarrow Max,\quad Q_2(a,x)\Rightarrow Max,\ \ldots,\ Q_n(a,x)\Rightarrow Max,\quad x\in X.$$

Знайти розв'язок, який одночасно був би найкращим за всіма критеріями, неможливо, тому що в загальному випадку покращення значення одного з критеріїв приводить до погіршення значення іншого.

Проілюструємо геометрично задачу оптимізації за двома критеріями. При цьому вважатимемо (як і всюди надалі, окрім окремих випадків), що критерії якості максимізуються.

Розглянемо загальну задачу оптимізації за двома критеріями з двома змінними:

$$Q_1(x_1,x_2)\Rightarrow Max,\ Q_2(x_1,x_2)\Rightarrow Max,\ x=(x_1,x_2),\ x\in X. \qquad (2.1)$$

Зобразимо область припустимих розв'язків у просторі змінних (x_1,x_2). Значення критеріїв Q_1, Q_2 відображатимемо у просторі критеріїв (Q_1,Q_2). Кожній конкретній точці множини припустимих рішень $\left(x_1^{(k)},x_2^{(k)}\right)$ відповідатиме одне і лише одне значення кожного з критеріїв $Q_1\left(x_1^{(k)},x_2^{(k)}\right),Q_2\left(x_1^{(k)},x_2^{(k)}\right)$, хоча обернене твердження не завжди буде відповідати дійсності (декілька розв'язків можуть бути рівноцінними з точки зору значень критеріїв), тобто відповідне відображення буде гомоморфним. Здійснивши таку операцію для всіх точок припустимої

області в просторі змінних, отримаємо її образ в просторі критеріїв:

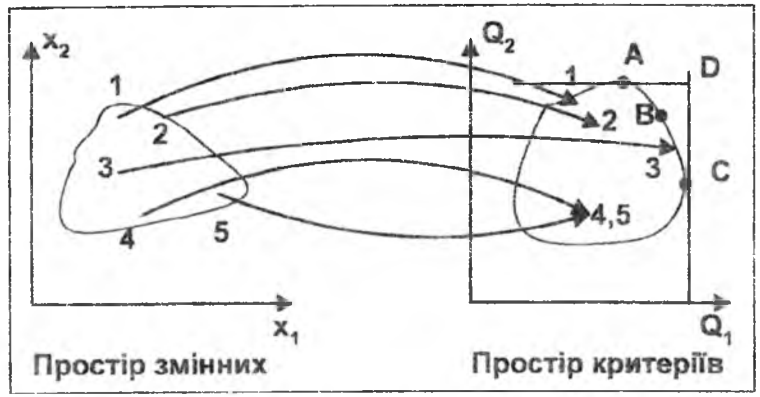


Рис. 2.1. Відображення припустимої області з простору змінних в простір критеріїв

На рис. 2.1 розв'язки 4 та 5 відображаються в одну й ту ж саму точку в просторі критеріїв, тобто є ідентичними з точки зору їх якості. Крім того, вони є гіршими, ніж розв'язки 2 та 3, у яких значення кожного з критеріїв є більші, ніж у 4 та 5. Розв'язки 1, 2, та 3 є непорівняльними, тобто без додаткової інформації неможливо визначити, який із них є кращий – значення за одним з критеріїв для них є більші, а за іншим – менші.

В той же час, аналізуючи розв'язки, що знаходяться на кривій А-В-С, можна зробити висновок, що вони є множиною "найкращих" розв'язків: для будь-якого іншого розв'язку з множини припустимих завжди знайдеться хоча б один із розв'язків, що знаходяться на А-В-С та кращий за нього (тобто такий, що його домінує). Таким чином, розв'язки, що лежать на А-В-С, не домінуються ніякими іншими розв'язками, що належать до припустимої області.

Множина недомінованих розв'язків багатокритерійної задачі називається множиною Парето-оптимальних розв'язків (саме Вільфредо Парето одним із перших досліджував задачі такого типу) і є, таким чином, у загальному випадку розв'язком багатокритерійної задачі. В свою чергу розв'язок належить до множини Парето-оптимальних, якщо він не домінується ніяким іншим.

Побудова множини Парето-оптимальних розв'язків для задач ДО в більшості випадків є неможливою внаслідок значних обчислювальних труднощів. Крім того, в більшості випадків завдання полягає в знаходженні одного розв'язку. Такий розв'язок повинен належати до множини Парето-оптимальних, а ось яким він повинен бути, може виявитися лише в процесі його побудови. Тому був розроблений та застосовується на практиці цілий ряд методів, деякі з них розглядаються нижче.

2.2. МЕТОДИ ЗГОРТАННЯ КРИТЕРІЇВ. МЕТОД "ІДЕАЛЬНОЇ ТОЧКИ"

Одним із найрозповсюдженіших способів є приведення множини критеріїв до одного глобального та розв'язування класичної однокритерійної задачі. Однак застосування цього підходу має суттєві вади, і однією з них є те, що отриманий розв'язок для деяких специфічних задач може навіть не належати до множини Парето-оптимальних.

Методи згортання критеріїв приводять первісну задачу до однокритерійної задачі такого вигляду:

$Q(Q_1(a,x),...,Q_n(a,x)) \Rightarrow Max,\ x \in X$. Найвживанішими є:

☑ лінійне згортання

$$Q = \sum_{i=1}^{n}(c_i \times Q_i(a,x)) \Rightarrow Max,\ x \in X,\ \sum_{i=1}^{n} c_i = 1,\ c_i > 0; \tag{2.2}$$

☑ лінійне згортання нормованих критеріїв:

$$Q = \sum_{i=1}^{n}\left(c_i \times \frac{Q_i(a,x) - Q_i^{min}}{Q_i^{max} - Q_i^{min}}\right) \Rightarrow Max,\ x \in X,\ \sum_{i=1}^{n} c_i = 1,\ c_i > 0. \tag{2.3}$$

В цих методах c_i — вагові коефіцієнти критеріїв, які повинні відображати їх важливість. Q_i^{min}, Q_i^{max} — мінімальне та максимальне значення i-го критерію.

Основною проблемою цих методів є проблема виявлення точних значень вагових коефіцієнтів – ця процедура в більшості випадків є суб'єктивною. Окрім того, коефіцієнти в методі лінійного згортання повинні бути розмірними величинами, тому що критерії в більшості випадків мають різну розмірність. З метою позбавлення від цього недоліку в згортанні нормованих критеріїв окремі критерії спочатку нормуються (нормовані критерії є безрозмірними та змінюються в інтервалі від 0 до 1). Але внаслідок такого "вдосконалення" з'являються нормовані критерії, які не мають змістовного навантаження, і тому об'єктивне визначення вагових коефіцієнтів ще більш ускладнюється. Таким чином довільність (що викликана багатокритерійністю) переноситься в іншу інстанцію

(визначення числових значень вагових коефіцієнтів).

Інший підхід до розв'язання проблеми багатокритерійності – аксіоматичний – полягає в формулюванні множини аксіом з наступним формальним виведенням виду функції корисності (глобального критерію), за допомогою якого й здійснюватиметься остаточний вибір. У цьому випадку виявляються всі обмеження, які побічно накладаються при еврістичному застосуванні того чи іншого методу. Виявляється, що лінійне згортання обґрунтоване при достатньо жорстких аксіоматичних умовах, які в багатьох випадках не виконуються.

Окрім того, існують й інші методи згортання – такі, як метод ідеальної точки. Метод ідеальної точки базується на тому, що постулюється існування "ідеальної точки" для розв'язку задачі, у якій досягається екстремум усіх критеріїв (принцип Джофріона). Так, на рис. 2.1 ідеальною є точка D в просторі критеріїв, якій не відповідає жоден припустимий розв'язок простору змінних. Оскільки ідеальна точка в абсолютній кількості випадків не знаходиться серед припустимих, виникає проблема знаходження точки, що „найближча" до ідеальної і належить до множини припустимих. Все було би добре, якщо б існувало єдине об'єктивне поняття "віддалі", однак це не так – якщо на площині ми можемо з тим чи іншим обґрунтуванням застосовувати Евклідову метрику, то, наприклад, на поверхні кулі (земної також!) найкоротшою віддаллю буде дуга, а не пряма.

Таким чином, для розв'язання задачі за допомогою методу "ідеальної точки" необхідно насамперед визначити її координати, і надалі визначити метрику, за допомогою якої можна було б виміряти віддаль до оптимальної точки. Для визначення координат "ідеальної точки" розв'язуємо *n* однокритерійних задач за кожним з критеріїв оптимізації $Q_i(a,x) \Rightarrow Max,\ x \in X,\ i = \overline{1,n}$. Сукупність оптимальних значень критеріїв кожної з однокритерійних задач $Q_i^* = Max\ Q_i(a,x), x \in X$ і визначить координати ідеальної точки $Q^* = (Q_1^*,...,Q_n^*)$ в просторі критеріїв. Якщо "ідеальна точка" належить до множини припустимих (що зустрічається вкрай рідко), то розв'язок отриманий.

В іншому випадку визначаємо "віддаль" до ідеальної точки, вводячи метрику, і розв'язуємо однокритерійну задачу знаходження точки з числа припустимих, яка найменш віддалена від ідеальної. Таким чином задача матиме вигляд $Q(x) = \rho(Q(a,x) - Q^*) \Rightarrow Min,\ x \in X$. Якщо обрана

Евклідова метрика, то критерій буде мати вигляд:

$$Q(x) = \sqrt{\sum_{i=1}^{n}(Q_i(x) - Q_i^*)^2} \Rightarrow Min. \tag{2.4}$$

2.3. ПЕРЕВЕДЕННЯ КРИТЕРІЇВ В ОБМЕЖЕННЯ. КОНТРОЛЬНІ ПОКАЗНИКИ. МЕТОД ПОСЛІДОВНИХ ПОСТУПОК

Одним із найзрозуміліших змістовно є **метод переведення критеріїв в обмеження**, що полягає у виділенні головного критерію $Q_1(x)$, за яким проводитиметься оптимізація, нормативних значень Q_i^N для кожного з критеріїв, що залишилися (значення критерію не може бути меншим за нормативне), та розв'язуванні отриманої таким чином однокритерійної задачі оптимізації:

$$Q_1(x) \Rightarrow Max,\ Q_2(x) \ge Q_2^N, \dots Q_n(x) \ge Q_n^N,\ x \in X. \quad (2.5)$$

Основними проблемами при застосуванні цього методу є складнощі з визначенням головного критерію та нормативних значень для інших критеріїв. Якщо нормативні значення обрані недостатньо великими, то не всі резерви покращення їх значень будуть використані, якщо ж ці значення будуть завеликими, то задача взагалі не буде мати розв'язків, оскільки область припустимих рішень виявиться пустою.

Метод контрольних показників дозволяє позбутися деяких проблем, притаманних методу переведення критеріїв в обмеження. Система нормативів задається для всіх критеріїв, і критерій якості представляється у вигляді:

$$Q(x) = Min \frac{Q_i}{Q_i^N} \Rightarrow \underset{x \in X}{Max}. \quad (2.6)$$

Але й у цьому випадку залишається проблема обґрунтування значень нормативів і додається проблема знаходження розв'язку максимінної задачі.

Метод послідовних поступок є одним із найобґрунтованіших змістовно, і за умови відсутності суперечностей в перевагах особи, що приймає рішення, може дати добрий результат. Насамперед критерії впорядковуються ОПР за важливістю в порядку її спадання:

$Q_1 \succ Q_2 \succ \dots \succ Q_n$. Після цього на кожному і-му кроці алгоритму розв'язується задача оптимізації за критерієм Q_i та призначається поступка ΔQ_i, на яку ми готові зменшити отримане оптимальне значення критерію Q_i^*, щоб покращити значення інших критеріїв, менш важливих, ніж Q_i. Значення цих критеріїв розраховуються за відомими координатами оптимуму x^*. Призначення поступки означає введення на кожному кроці ще одного додаткового обмеження $Q_i \ge Q_i^* - \Delta Q_i$, і таким чином на (і+1)-му кроці розв'язуватиметься задача:

$$Q_{i+1}(x) \Rightarrow Max, x \in X, Q_1(x) \ge Q_1^* - \Delta Q_1, \dots Q_i(x) \ge Q_i^* - \Delta Q_i. \quad (2.7)$$

Процес розв'язування закінчується у випадках, коли або досягнуто останнього критерію, або ж призначення поступки недоцільне. У випадку необхідності процес можна повторити, здійснивши аналіз попередніх результатів. Таким чином, метод послідовних поступок є достатньо гнучким і дозволяє уникнути багатьох проблем, властивих іншим методам. Для його реалізації достатньо мати ефективний метод розв'язування однокритерійної задачі потрібного типу.

2.4. ПОНЯТТЯ ПРО ДІАЛОГОВІ МЕТОДИ

Діалогові методи належать до групи найгнучкіших методів пошуку розв'язку багатокритерійних задач. Характерною рисою цих методів є участь у процесі розв'язування особи, що приймає рішення, а це дозволяє скорегувати перебіг рішення та врахувати деякі неформальні моменти. У принципі, момент діалогу присутній уже в методі послідовних поступок – на кожному кроці алгоритм звертається до ОПР з метою отримання значення поступки для того чи іншого критерію. Надалі з метою ілюстрації розглянемо схеми деяких з діалогових методів.

Доволі розповсюдженим є алгоритм розв'язування, запропонований Джофріоном та модифікований багатьма дослідниками, що використовує ідеї добре відомого градієнтного методу. Робота алгоритму починається з будь-якої точки припустимої області. На кожному етапі з залученням ОПР визначається напрям руху в просторі критеріїв та довжина кроку в цьому напрямку. Напрямок руху (еквівалент градієнту) визначається шляхом опитування ОПР щодо значень коефіцієнтів заміщення критеріїв в біжучій точці (проводиться опитування – яким значенням зміни по одному з критеріїв можна скомпенсувати зміну іншого критерію). Звичайно, що напрямок руху залежатиме від координат точки в просторі критеріїв. Після цього ОПР задає величину кроку в заданому напрямку і здійснюється крок – якщо його значення приводить до виходу за межі припустимої області в просторі змінних, величина кроку зменшується, щоб отримана точка належала до області припустимих значень. Процедура повторюється, поки ОПР не зупинить її виконання, або поки не будуть виконані формальні умови зупинки. Цей метод висуває високі вимоги до ОПР відносно виявлення значень коефіцієнтів заміщень критеріїв.

Іншу групу методів утворюють методи поступового звуження множини розв'язків, що належать до множини Парето-оптимальних.

Розглянемо діалоговий алгоритм розв'язування двокритерійної задачі оптимізації:

$$Q_1(x_1,x_2) \Rightarrow Max, Q_2(x_1,x_2) \Rightarrow Max, x=(x_1,x_2), x \in X. \qquad (2.8)$$

Розв'яжемо пару однокритерійних задач оптимізації по кожному з критеріїв і підстановкою відповідних оптимальних значень змінних визначимо іншу координату в просторі критеріїв:

$Q_1(x_1,x_2) \Rightarrow Max, x=(x_1,x_2), x \in X.$

Результат - координати оптимальної точки $x^{(1)} = \left(x_1^{(1)}, x_2^{(1)}\right)$ та оптимальне значення критерію $Q_1^{(1)} = Q_1\left(x^{(1)}\right)$ і обчислене значення критерію $Q_2^{(1)} = Q_2\left(x^{(1)}\right)$.

$Q_2(x_1,x_2) \Rightarrow Max, x=(x_1,x_2), x \in X.$ Результат–координати оптимальної точки $x^{(2)} = \left(x_1^{(2)}, x_2^{(2)}\right)$ та оптимальне значення критерію $Q_2^{(2)} = Q_2\left(x^{(2)}\right)$ і обчислене значення критерію $Q_1^{(2)} = Q_1\left(x^{(2)}\right)$.

Таким чином, отримуємо дві граничні точки множини Парето-оптимальних розв'язків в просторі критеріїв: $Q^{(1)} = \left(Q_1\left(x^{(1)}\right), Q_2\left(x^{(1)}\right)\right)$ – та $Q^{(2)} = \left(Q_1\left(x^{(2)}\right), Q_2\left(x^{(2)}\right)\right)$.

Надалі обираємо середину відрізка: $\left(Q_1\left(x^{(1)}\right), Q_1\left(x^{(2)}\right)\right)$ — $Q_1^{(3)} = \left(Q_1\left(x^{(1)}\right) - Q_1\left(x^{(2)}\right)\right)/2$ і розв'язуємо задачу: $Q_2(x_1,x_2) \Rightarrow Max$, $x=(x_1,x_2), Q_1(x_1,x_2) \geq Q_1^{(3)}$. Розв'язавши її і підставивши координати оптимальної точки в просторі змінних у вирази для критеріїв, отримаємо координати середньої точки: $Q^{(3)} = \left(Q_1\left(x^{(3)}\right), Q_2\left(x^{(3)}\right)\right)$.

Ці кроки виконуються без втручання ОПР, і ОПР пред'являється графічне зображення (рис. 2.2) з координатами трьох точок в області критеріїв, а також ставиться запитання: В якому напрямку від середньої точки необхідно рухатися по осі критерію Q_1 ? В залежності від відповіді інтервал пошуку звужується, переіндексовуються крайні точки, шукаються координати середньої точки і процедура опитування повторюється. Цікаво відзначити, що в цьому випадку по суті звужується область Парето-оптимальних розв'язків, але при цьому ОПР не повинен знати її конфігурацію.

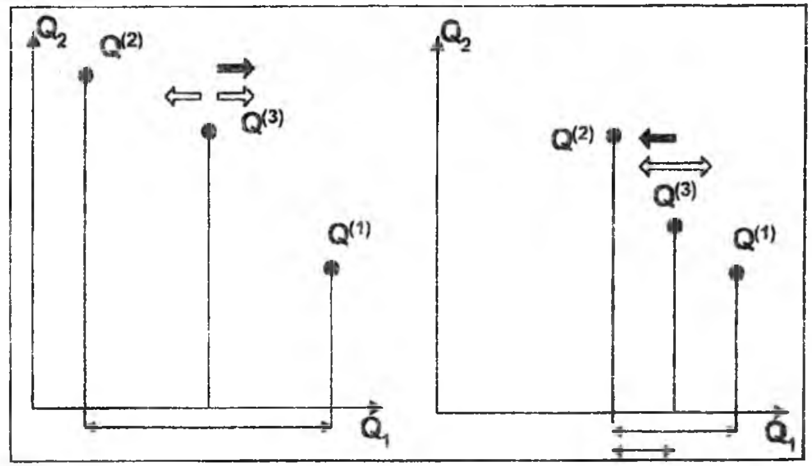


Рис. 2.2. Послідовність кроків при пошуку оптимального розв'язку двокритерійної задачі за допомогою діалогового методу

ПРИКЛАДИ

Приклад 2.1. Багатокритерійна задача.

Необхідно придбати автомобіль, враховуючи декілька аспектів, що відображаються критеріями якості. Важливими критеріями в цьому випадку є питомі витрати пального, максимальна швидкість, комфортність, прохідність, зовнішній вигляд, вартість, гарантія, наявність сервісного обслуговування, ціна. Звичайно, придбати авто, яке б витрачало найменше пального, було б при цьому найдешевшим, найкомфортнішим та найшвидшим водночас неможливо.

Приклад 2.2. Багатокритерійна задача.

Необхідно обрати місце зупинки для мандрівників у горах. Важливими критеріями є наявність джерела або річки, наявність палива, можливість купання, відсутність комарів, близькість до населених пунктів, відсутність хижих звірів.

Приклад 2.3. Побудова множини Парето-оптимальних розв'язків, вибір кращої альтернативи згортанням критеріїв.

Необхідно визначити множину Парето-оптимальних альтернатив, обрати найкращу з використанням лінійної згортки критеріїв з вагами 0.3 та 0.7 та за методом ідеальної точки, якщо критерії задані наступним чином:

$$Q_1(x_1,x_2) = 2x_1 + x_2^2 \Rightarrow Max,$$

$$Q_2(x_1,x_2) = -2x_1^2 + x_2 \Rightarrow Max,$$

а координати альтернатив в просторі змінних задані наступною таблицею:

№	1	2	3	4	5	6
x1	1	3	0	-1	6	8
x2	2	1	3	2	1	-2

Розв'язання.

Послідовність розв'язання цієї задачі в загальному є наступною: спочатку розраховуємо значення двох критеріїв для кожної з 6 альтернатив (тобто будуємо образи кожної з альтернатив в просторі критеріїв); далі – для побудови множини Парето – оптимальних альтернатив виключаємо з наведеної множини послідовно доміновані альтернативи – поки не дійдемо до останньої. Лінійну згортку будуємо, використовуючи образи альтернатив в просторі критеріїв. Всі необхідні розрахунки зведені в наступну таблицю:

№ альтернативи	1	2	3	4	5	6
Q1	6	7	9	2	13	20
Q2	0	-17	3	0	-71	-130
Парето-оптимальні альтернативи	-	-	+	-	+	+
Qзгортки	1,8	-9,8	4,8	0,6	-45,8	-85.0

Парето-оптимальні альтернативи визначаємо, порівнюючи біжучу альтернативу зі всіма наступними. Якщо зустрічається альтернатива, що домінується біжучою, то вона виключається з подальшого розгляду. Якщо ж така альтернатива домінує біжучу, то біжуча виключається з розгляду, здійснюється перехід до альтернативи, наступної за біжучою і не виключеної з розгляду. Процес повторюється до моменту, поки біжучу альтернативу не буде з чим порівнювати.

Починаємо з альтернативи 1. Вона непорівняльна з альтернативою 2. Порівнюємо 1 з наступною – 3. 3 домінує 1 – тому переходимо до наступної невиключеної за 1 альтернативи – 2, а 1 виключаємо з розгляду. Альтернативу 2 порівнюємо з наступною невиключеною – 3. 3 домінує 2, тому виключаємо

2 і переходимо до 3 як до біжучої альтернативи. 3 домінує 4 – тому 4 виключаємо. 3 непорівняльна з 5, та 3 непорівняльна з 6 – і отже – оскільки наступні альтернативи відсутні – 3 належить до множини Парето-оптимальних. Наступна біжуча альтернатива – 5, яка непорівняльна з 6. Таким чином, множину Парето-оптимальних альтернатив складають рішення за номерами 3, 5, та 6.

Обчислимо значення критерія-згортки для кожної з 6 альтернатив Наприклад, для альтернативи 1:

$$Q^{(1)} = 0.3\times Q_1^{(1)} + 0.7\times Q_2^{(1)} = 0.3\times 6 + 0.7\times 0 = 1.8 .$$

Обираючи максимальне значення, вважаючи, що аргументом є номер альтернативи, отримаємо: $x^* = \arg \underset{x\in X}{Max}\ Q(a,x) =$

$$= \arg \underset{x\in X}{Max}\left\{Q^{(1)} = 1.8,\ Q^{(2)} = -9.8,\ Q^{(3)} = 4.8,\ Q^{(4)} = 0.6, Q^{(5)} = -45.8,\ Q^{(6)} = -85\right\} = 3$$

тобто за критерієм згорткою кращою є альтернатива 3.

Приклад 2.4. Визначення найкращого розв'язку методом переведення критеріїв в обмеження.

Визначити найкращий розв'язок при оцінюванні шести можливих розв'язків (A_1-A_6) за трьома критеріями, образи яких у просторі критеріїв задані в таблиці нижче, шляхом переведення критеріїв в обмеження, за умови, що шукається максимальне значення критерію Q_1, для наступного випадку: $Q_2 \geq 5,\ Q_3 \geq 4$.

	Q1	Q2	Q3
A1	2	4	8
A2	4	3	14
A3	7	8	2
A4	5	6	6
A5	8	4	4
A6	3	6	12

Розв'язання.

Виключимо з переліку альтернатив ті, для яких не виконуються обмеження, і серед тих, що залишаться, визначимо найкращі. Так, $Q_2 \geq 5$ для альтернатив A_3, A_4, A_6, $Q_3 \geq 4$ для A_1, A_2, A_4, A_5, A_6, тобто одночасно ці обмеження виконуються для A_4, A_6. Для цих альтернатив максимальне значення критерію $Q_1^{(4)} = 5$, тобто обираємо A_4.

Приклад 2.5. Визначення найкращого розв'язку методом контрольних показників.

Визначити найкращий розв'язок при оцінюванні за трьома критеріями шляхом використання контрольних показників:

$$Q_1^K = 6,\ Q_2^K = 8,\ Q_3^K = 10 .$$

Розв'язання.

	Q_1	Q_2	Q_3
A_1	2	4	8
A_2	4	3	14
A_3	7	8	2
A_4	5	6	6
A_5	8	4	4
A_6	3	6	12

Згідно до співвідношення $Q(x) = \underset{i}{Min}\frac{Q_i}{Q_i^K} \Rightarrow Max$ розрахуємо спочатку значення Q_i / Q_i^K і заповнимо цими значеннями перші 3 колонки таблиці рішення. На наступному кроці визначимо мінімальні значення в кожному з рядків таблиці рішення і запишемо їх у четверту колонку.

	Q_1	Q_2	Q_3	Min	
A_1	2/6	4/8	8/10	2/6	
A_2	4/6	3/8	14/10	3/8	
A_3	7/6	8/8	2/10	2/10	
A_4	5/6	6/8	6/10	6/10	Max
A_5	8/6	4/8	4/10	4/10	
A_6	3/6	6/8	12/10	3/6	

Серед мінімальних значень обираємо максимальне, що відповідає альтернативі A_4.

РЕЗЮМЕ

2.1. Важливий вид невизначеності – невизначеність мети, що виявляється у наявності декількох, в більшості випадків незбіжних аспектів оцінки якості того чи іншого розв'язку з множини припустимих. У формальному вигляді аспекти оцінки якості відображаються за допомогою множини критеріїв. Знайти розв'язок, який одночасно був би найкращим за всіма критеріями, неможливо, тому що в загальному випадку покращення значення одного з критеріїв приводить до погіршення значення іншого. Множина недомінованих розв'язків багатокритерійної задачі називається множиною Парето-оптимальних розв'язків і є в загальному випадку розв'язком багатокритерійної задачі.

2.2. Одним із найрозповсюдженіших способів є приведення множини критеріїв до одного глобального та розв'язування класичної однокритерійної задачі. Однак застосування цього підходу має суттєві вади, і однією з них є те, що отриманий розв'язок для деяких специфічних задач може навіть не належати до множини Парето-оптимальних. Основною проблемою цих методів є проблема виявлення точних значень вагових коефіцієнтів – ця процедура в більшості випадків є суб'єктивною. Інший підхід до розв'язання проблеми багатокритерійності – аксіоматичний – полягає в формулюванні множини аксіом з наступним формальним виведенням виду функції корисності (глобального критерію), за допомогою якого й здійснюватиметься остаточний вибір.

*2.3. Одним із найзрозуміліших змістовно є **метод переведення критеріїв в обмеження**. Основними проблемами при застосуванні цього методу є складнощі з визначенням головного критерію та нормативних значень для інших критеріїв. **Метод контрольних показників** дозволяє позбутися деяких проблем, притаманних методу переведення критеріїв в обмеження, але й у цьому випадку залишається проблема обґрунтування значень нормативів і додасться проблема знаходження розв'язку максимінної задачі. **Метод послідовних поступок** є одним з найобґрунтованіших змістовно, і за умови відсутності суперечностей у перевагах особи, що приймає рішення, може дати добрий результат.*

2.4. Діалогові методи належать до групи найгнучкіших методів пошуку розв'язку багатокритерійних задач. Характерною рисою цих методів є участь у процесі розв'язування особи, що приймає рішення, а це дозволяє скорегувати перебіг рішення та врахувати деякі неформальні моменти.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 2.1.

Необхідно визначити множину Парето-оптимальних альтернатив, обрати найкращу з використанням лінійної згортки критеріїв з вагами 0.6 та 0.4 і за методом ідеальної точки, якщо критерії задані наступним чином:

$Q_1(x_1, x_2) = 2x_1 + x_2^2 \Rightarrow Max, Q_2(x_1, x_2) = -x_1^2 + x_2 \Rightarrow Max$, а координати альтернатив в просторі змінних задані наступною таблицею:

№	1	2	3	4	5	6
x_1	1	3	0	-1	4	6
x_2	2	1	3	2	1	-2

Завдання 2.2.

Визначити найкраще рішення при оцінюванні шести можливих рішень (A_1-A_6) за трьома критеріями, образи яких в просторі критеріїв задані в

таблиці нижче, шляхом переведення критеріїв в обмеження, за умови, що шукається максимальне значення критерію Q_1,
для наступних випадків:

	Q1	Q2	Q3
A1	2	4	8
A2	4	3	14
A3	7	8	2
A4	5	6	6
A5	8	4	6
A6	3	6	12

a) $Q_2 \geq 4,\ Q_3 \geq 6$; b) $Q_2 \geq 3,\ Q_3 \geq 11$;

c) $Q_2 \geq 3,\ Q_3 \geq 11$.

Завдання 2.3.

Визначити найкраще рішення при оцінюванні за трьома критеріями, шляхом використання контрольних показників:

$Q_1^K = 7,\ Q_2^K = 10,\ Q_3^K = 12$.

	Q1	Q2	Q3
A1	10	3	8
A2	8	3	11
A3	7	8	2
A4	5	6	6
A5	18	4	4
A6	3	6	14

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Чому виникають багатокритерійні задачі дослідження операцій?
2. Які властивості мають Парето-оптимальні рішення?
3. В чому полягають переваги та недоліки методів згортання критеріїв?
4. Яким чином розв'язуються багатокритерійні задачі методом ідеальної точки?

5. Які переваги та недоліки методу переведення критеріїв в обмеження порівняно з методом контрольних показників?
6. В чому суть методу послідовних поступок?
7. Сформулюйте переваги діалогових методів розв'язання багатокритерійних задач ДО.

РОЗДІЛ 2.

ЗАДАЧА ЛІНІЙНОГО ПРОГРАМУВАННЯ

ТЕМА 3. ПОСТАНОВКА ЗАДАЧІ ЛІНІЙНОГО ПРОГРАМУВАННЯ

Задачі лінійного програмування (ЛП) є підкласом ширшого класу задач математичного програмування (МП), для яких існують ефективні алгоритми знаходження оптимальних розв'язків. Це пояснюється тим, що задачі ЛП належать до класу P-складних, тобто в загальному для них існують поліноміальні алгоритми, що суттєво економніші, аніж повний перебір. Задача ЛП в загальному вигляді завжди може бути зведена до канонічної форми, а тому й розглядається в більшості випадків в канонічній формі. Для того, щоб реальна економічна проблема чи задача могла бути представлена у вигляді задачі ЛП, необхідно, щоб для неї виконувалися умови пропорційності та адитивності. Задача ЛП з двома змінними може бути наочно представлена в площині та розв'язана геометрично. Аналіз задачі ЛП на чутливість є важливим з точки зору економічних застосувань, так як дозволяє визначити дефіцитні та недефіцитні ресурси, доцільні межі зміни дефіцитних ресурсів та їх цінність, а також чутливість отриманого оптимального розв'язку до змін значень коефіцієнтів функції мети, які звичайно розглядаються як ціни чи прибуток на одиницю продукції.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати ⇨загальну постановку задачі МП та мати поняття про складність задач МП;⇨умови, за яких задача може бути приведена до задачі ЛП; ⇨основні задачі аналізу на чутливість;

вміти ⇨привести задачу ЛП до канонічної форми:⇨представити та розв'язати геометрично задачу ЛП;⇨побудувати модель реальної економічної ситуації у вигляді задачі ЛП;⇨розв'язати задачу ЛП геометрично та проаналізувати її на чутливість.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ лінійне програмування
- ☑ геометричне представлення задачі ЛП
- ☑ математичне програмування
- ☑ цінність ресурсу
- ☑ P- та NP- повні задачі
- ☑ канонічна форма задачі ЛП

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

3.1. Місце лінійного програмування в математичному програмуванні.

3.2. Формальна постановка задачі лінійного програмування.

3.3. Побудова моделей задач лінійного програмування.

3.4. Геометричне представлення задач лінійного програмування.

3.5. Задачі аналізу лінійних моделей на чутливість.

3.1. МІСЦЕ ЛІНІЙНОГО ПРОГРАМУВАННЯ В МАТЕМАТИЧНОМУ ПРОГРАМУВАННІ

Термін "лінійне програмування" з'явився вперше в 1951р. в працях американських вчених Дж. Данціга та Т. Купманса. Задачі ЛП входять як важлива складова до ширшого класу задач математичного програмування (рис 3.1).

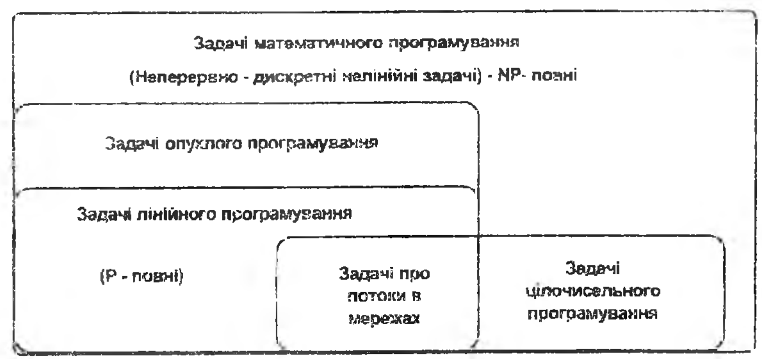


Рис.3.1. Місце задач лінійного програмування в системі задач математичного програмування

У загальному задача математичного програмування представляється наступним чином:

$$f(x) \Rightarrow Max,\ g(x) \le 0,\ h(x) = 0\ ;$$
$$x = (x_1, \ldots, x_n),\ g = (g_1(x), \ldots, g_m(x)),\ h = (h_1(x), \ldots, h_l(x)). \qquad (3.1)$$

Задачі на максимізацію та мінімізацію еквівалентні, також це стосується і знаків $\le$, $\ge$.

Задачі математичного програмування за складністю поділяються на P (Polynomial) – та NP (Nondeterministically Polynomial) – повні. Природним є прагнення розвинути якомога ефективніші методи розв'язування якомога ширших класів задач. Однак, як доводить багаторічний досвід, загальність методу та його ефективність знаходяться в певному антагонізмі. Разом із тим дуже важливо знати, **чи можна в принципі** сподіватися на створення достатньо загальних та ефективних методів, чи необхідно свідомо розбивати задачі на вужчі класи та, використовуючи їх специфіку, розробляти для них ефективні алгоритми.

Більшість із дискретних та комбінаторних задач математичного програмування в принципі можуть бути розв'язані за допомогою повного перебору варіантів. Однак число кроків методу перебору зростає експоненційно в залежності від розміру задачі. Для деяких задач такого типу вдається побудувати ефективні методи розв'язування, однак кількість їх невелика. Аналіз труднощів синтезу ефективних алгоритмів розв'язування задач привів до постановки центральної проблеми дискретної математики – пошуку відповіді на запитання: "Чи можна виключити перебір при розв'язуванні дискретних задач?". В даний час ця проблема відкрита.

Аналогом алгоритмічної нерозв'язності в скінченій області є перебір експоненційного числа варіантів, а аналогом алгоритмічної розв'язності – існування суттєво більш економічного алгоритму, ніж перебір. Загальноприйнятим вважається, що задача перебірного типу розв'язується ефективно, якщо наявний алгоритм, що розв'язує її за час, обмежений поліномом від "розміру задачі".

Вважають, що перебірна задача Π_1 зводиться до перебірної задачі Π_2, якщо метод розв'язування задачі Π_2 можна перетворити в метод розв'язування задачі Π_1. Приведення є поліноміальним, якщо його можна здійснити за поліноміальний час. Головними об'єктами є клас NP – всіх перебірних задач та клас P – перебірних задач, які можуть бути розв'язаними за поліноміальний час машиною Тьюринга – однією з найпростіших формальних моделей обчислювача. Очевидно, що $P \subseteq NP$. Центральним є питання про те, чи тотожні ці класи.

Основні результати теорії NP-повних задач отримані в роботі С. Кука „Складність процедур виведення теорем" (1971 р., „The complexity of theorem-proving procedures"). В цій роботі наголошено на важливості поняття зведення за поліноміальний час однієї задачі до іншої. Це дозволяє стверджувати, що якщо задача може бути поліноміально зведена до іншої, складність розв'язування якої є поліноміальною, то вона теж є поліноміальною. Іншим важливим результатом було звернення уваги на певний клас задач розпізнавання властивостей (це задача, розв'язками якої можуть бути „так" чи „ні", які можуть бути розв'язані за поліноміальний час, але на недермінованому обчислювальному пристрої – клас NP-складних задач).

В класі NP виявлені так звані універсальні (NP – повні) задачі, до яких поліноміально зводиться довільна задача з NP , тобто вони є еталоном складності. Якщо б вдалося довести, що хоча б одна з NP - повних задач належить до класу P , то це означало б, що $P = NP$, і можна було б в принципі сподіватися на побудову ефективних алгоритмів для різних класів дискретних задач. Хоча практичний досвід розв'язування дискретних задач дозволяє вважати, що задачі цих двох класів значною мірою розрізняються за складністю, в строгому сенсі ця різниця не доведена.

Важливим результатом математика Л. Хачіяна є доведення факту P - повноти задач лінійного програмування. Окрім того, оскільки задачі про потоки в мережах є підкласом задач лінійного програмування та одночасно підкласом задач дискретної оптимізації, для них розвинуті спеціалізовані ефективні алгоритми розв'язування. Тому ці задачі посідають особливе місце серед задач математичного програмування – для них доведений факт можливості побудови ефективних алгоритмів пошуку оптимальних розв'язків з поліноміальною складністю, і такі алгоритми побудовані.

3.2. ФОРМАЛЬНА ПОСТАНОВКА ЗАДАЧІ ЛІНІЙНОГО ПРОГРАМУВАННЯ

В загальному вигляді задача лінійного програмування формулюється наступним чином:

$$Q(x) = \sum_{j=1}^{n} c_j x_j \Rightarrow Max\ (Min),$$

$$\sum_{j=1}^{n} a_{ij}x_j (\le,=,\ge) b_i,\; i=\overline{1,m},\; j=\overline{1,n},\; x_j (\le,=,\ge) 0. \quad (3.2)$$

Таким чином, і критерій якості, і обмеження є лінійними функціями від змінних, звідки й походить назва „задача лінійного програмування".

Задача ЛП в канонічній формі є наступною:

$$Q(x)=\sum_{j=1}^{n} c_j x_j \Rightarrow Max\;,$$

$$\sum_{j=1}^{n} a_{ij}x_j = b_i,\; i=\overline{1,m},\; j=\overline{1,n},\; x_j \ge 0. \quad (3.3)$$

У векторній формі запису обмеження задачі лінійного програмування у вигляді нерівностей можна записати так:

$P_1x_1 + P_2x_2 + \ldots + P_nx_n \le b = P_0$,

де

$$P_1=\begin{bmatrix} a_{11} \\ a_{21} \\ \ldots\ldots \\ a_{m1} \end{bmatrix}; P_2=\begin{bmatrix} a_{12} \\ a_{22} \\ \ldots\ldots \\ a_{m2} \end{bmatrix}; \ldots\ldots; P_n=\begin{bmatrix} a_{1n} \\ a_{2n} \\ \ldots\ldots \\ a_{mn} \end{bmatrix}; P_0=b=\begin{bmatrix} b_1 \\ b_1 \\ \ldots\ldots \\ b_m \end{bmatrix}. \quad (3.4)$$

Розглянемо припустиму множину R у просторі даних векторів. Так як $x_i \ge 0, i=1,2,\ldots,n$, то усі позитивні комбінації векторів $P_1,P_2,\ldots,P_n$ утворюють конус. Тому питання про існування припустимого розв'язку рівнозначне питанню про належність вектора b до цього конуса. Оскільки $P_1,P_2,\ldots,P_n$ – m-вимірні вектори (n>m), то серед них завжди знайдеться m лінійно незалежних векторів, що утворюють базу m-вимірного простору і містять конус, утворений векторами $P_1,P_2,\ldots,P_n$.

Тому справедливе наступне твердження. Якщо задача ЛП містить n змінних і m обмежень (n>m), записаних у формі нерівностей, не враховуючи обмеження невід'ємності $x_j \ge 0$, то в оптимальний розв'язок входить не більше, ніж m ненульових компонент вектора *x*.

На перший погляд, задача ЛП в загальній формі, в якій припускаються обмеження будь-якого вигляду – і у вигляді рівностей, і у вигляді нерівностей „більше" чи „менше", і не накладаються ніякі обмеження на значення змінних – вони можуть бути як позитивними, так і негативними, а також необмеженими в знаку, видається загальнішою, ніж задача в канонічній формі – лише задача максимізації, обмеження існують лише у вигляді рівностей, а всі змінні не повинні бути негативними. Однак ці обидві постановки є еквівалентними.

Задачу ЛП в загальній формі завжди можна перетворити в задачу в канонічній формі за допомогою наступних підстановок:

$$Q(x)\Rightarrow Min,\; -Q(x)\Rightarrow Max\;. \quad (3.5)$$

$$\sum_{j=1}^{n} a_{ij}x_j \le b_i\;,\; \sum_{j=1}^{n} a_{ij}x_j + s_i = b_i\;, s_i \ge 0. \quad (3.6)$$

$$\sum_{j=1}^{n} a_{ij}x_j \ge b_i\;,\; \sum_{j=1}^{n} a_{ij}x_j - s_i = b_i\;, s_i \ge 0. \quad (3.7)$$

$$x_j \le 0,\; x_j^c = -x_j,\; x_j^c \ge 0. \quad (3.8)$$

$$x_j >< 0,\; x_j = x_j^+ - x_j^-,\; x_j^+, x_j^- \ge 0\;. \quad (3.9)$$

Мінімізація функції заміняється максимізацією її ж зі знаком „мінус" (3.5), що відповідає симетричному відбиттю її відносно осей ординат, нерівності „більше" (3.7) чи „менше" (3.6) приводяться до рівностей шляхом введення додаткових ненегативних змінних, непозитивні змінні заміняються ненегативними (3.8) з оберненим знаком, а змінна, необмежена в знаку (з областю зміни значень від „мінус безмежності" до „плюс безмежності") заміняється різницею пари ненегативних додаткових змінних (3.9, див.: Приклад 3.1).

3.3. ПОБУДОВА МОДЕЛЕЙ ЗАДАЧ ЛП

Економіко-математична модель є математичним описанням економічного процесу чи об'єкту, що досліджується. Ця модель відбиває закономірності економічного процесу в абстрактному вигляді за допомогою математичних співвідношень. Розглянемо деякі економічні процеси та об'єкти, що з достатньою точністю описуються за допомогою моделі лінійного програмування.

Методи математичного програмування використовуються в економічних, організаційних, військових і ін. системах для вирішення **розподільчих задач**. Розподільчі задачі (РЗ) виникають у разі, коли наявних ресурсів не вистачає для виконання кожної з передбачених робіт ефективним чином і необхідно найкращим чином розподілити ресурси за роботами відповідно до обраного критерію оптимальності.

Лінійне програмування (ЛП) є найпростішим і найкраще вивченим розділом математичного програмування. Характерні риси задач ЛП наступні:

1) критерій оптимальності є **лінійною** функцією від залежних змінних $X=(x_1,x_2,\ldots,x_n)$;

2) обмеження, що накладаються на можливі розв'язки, мають вигляд лінійних рівностей або нерівностей.

При описанні реальної ситуації за допомогою лінійної моделі слід перевіряти наявність у моделі таких властивостей, як пропорційність і адитивність. **Пропорційність** означає, що внесок кожної змінної в критерій якості і загальний об'єм споживання відповідних ресурсів повинен бути **прямо пропорційний** значенню цієї змінної. Тобто, якщо продаючи j-й товар у загальному випадку за ціною 100 гривень, фірма робитиме знижку при певному рівні закупівлі до рівня ціни 95 гривень, то буде відсутня пряма пропорційність між доходом фірми і величиною змінної x_j - порушуватиметься таким чином умова пропорційності, а саме в різних ситуаціях **одна** одиниця j-го товару приноситиме **різний** дохід. **Адитивність** означає, що критерій якості і обмеження повинні бути сумою внесків від різних змінних. Прикладом порушення адитивності є ситуація, коли збільшення збуту одного з конкуруючих видів продукції, що виробляються однією фірмою, впливає на об'єм реалізації іншого.

3.4. ГЕОМЕТРИЧНЕ ПРЕДСТАВЛЕННЯ ЗАДАЧ ЛП

Задачі лінійного програмування з двома змінними можуть бути розв'язані графічно на площині у просторі 2 змінних.

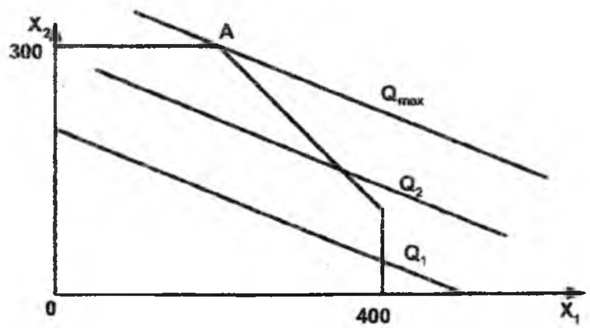


Рис. 3.2. Геометрична інтерпретація задачі ЛП

Розглянемо наступний приклад:

$Q(x_1, x_2) = (2x_1 + 5x_2) \Rightarrow Max$

за умов $x_1 \le 400; x_2 \le 300; x_1 + x_2 \le 500; x_1 \ge 0; x_2 \ge 0$. (3.10)

Кожна з цих нерівностей-обмежень визначає півплощини, перетин яких дає багатокутник. Цей багатокутник (опуклий) і є припустимою множиною розв'язків R задачі ЛП.

Тепер розглянемо функцію мети $Q(x_1, x_2) = 2x_1 + 5x_2$. Нехай $Q(x_1, x_2) = 1000 = Q_1$. Графік рівняння $2x_1 + 5x_2 = 1000$ є прямою з відрізками на осях $x_1 = 500, x_2 = 200$.

При $Q(x_1, x_2) = 1500$ отримаємо Q_2 пряму, рівняння якої

$$\frac{2x_1}{1500} + \frac{5x_2}{1500} = \frac{x_1}{750} + \frac{x_2}{300} = 1 \qquad (3.11)$$

Лінія рівня Q_2 паралельна до лінії рівня Q_1, але розташована вище за неї. Рухаючи її вгору, паралельно до самої себе, приходимо до такого положення Q_{max}, коли лінія рівня і множина R будуть мати лише одну спільну точку A. Очевидно, що точка A ($x_1 = 200, x_2 = 300$) - оптимальний розв'язок, так як вона лежить на прямій з максимально можливим значенням Q_{max}. Зазначимо, що ця точка виявилася крайньою точкою множини R.

3.5. ЗАДАЧІ АНАЛІЗУ ЛІНІЙНИХ МОДЕЛЕЙ НА ЧУТЛИВІСТЬ

Аналіз лінійних моделей на чутливість здійснюється після того, як отриманий оптимальний розв'язок, і має за мету виявлення чутливості оптимального розв'язку до певних змін значень параметрів первісної моделі. Це дозволяє дослідити, які збурення зовнішнього середовища – зменшення чи збільшення запасів ресурсів чи їх цін, цін на остаточну продукцію, прибутку від реалізації продуктів є суттєвими та будуть вимагати повторного розв'язування задачі. Таким чином, розв'язання задач аналізу на чутливість є важливим з економічної точки зору. Виділяються 3 основні задачі аналізу на чутливість.

1-а задача аналізу на чутливість.

А). На скільки доцільно збільшити запас деякого ресурсу для покращення отриманого розв'язку при незмінних значеннях інших ресурсів?

В). На скільки можна зменшити запас деякого ресурсу за умови збереження (не погіршення) оптимального значення критерію якості задачі?

Перша задача дозволяє виявити та дослідити чутливість до правої частини обмежень задачі ЛП.

2-а задача аналізу на чутливість.

Якому з ресурсів слід віддати перевагу при вкладенні додаткових засобів (тобто значення запасу якого з ресурсів найвигідніше збільшувати)? Розв'язання цієї задачі дозволяє визначити цінність ресурсу, його тіньову ціну.

3-я задача аналізу на чутливість.

В яких межах можуть змінюватись значення коефіцієнтів функції мети без зміни координат оптимального розв'язку в просторі змінних? Розв'язання цієї задачі дозволяє отримати відповідь на питання, які зміни цін змушуватимуть до зміни пропорцій виробництва.

ПРИКЛАДИ

Приклад 3.1. Приведення задачі ЛП до канонічної форми.

Привести до канонічного вигляду наступну задачу лінійного програмування.

$$\begin{aligned} 3x_1 + 2x_2 - 5x_3 &\Rightarrow Min \\ 2x_1 - 4x_2 + 3x_3 &= 15 \\ -x_1 + 5x_2 + 2x_3 &\le 21 \\ 5x_1 - 3x_2 - 4x_3 &\ge 17 \end{aligned}$$

$x_1 \ge 0, \quad x_2 >< 0, \quad x_3 \le 0.$

Розв'язання.

Для того, щоб уникнути плутанини з індексами, здійснимо підстановку-перетворення всіх змінних первісної задачі, а також додаткових змінних у змінні y з відповідними порядковими значеннями індексів. Таким чином реалізуємо наступні підстановки:

$y_1 = x_1; y_2 - y_3 = x_2; -y_4 = x_3,$

додамо та віднімемо додаткові змінні y_5, y_6 відповідно до другого та третього обмеження, поміняємо знак у критерія якості і отримаємо задачу в канонічній формі:

$$\begin{array}{llllll} -3y_1 & -2y_2 & +2y_3 & -5y_4 & \Rightarrow & Max \\ 2y_1 & -4y_2 & +4y_3 & -3y_4 & & =15 \\ -y_1 & +5y_2 & -5y_3 & -2y_4 & +y_5 & =21 \\ 5y_1 & -3y_2 & +3y_3 & +4y_4 & & -y_6 = 17 \end{array}$$

$\forall\left(j=\overline{1,6}\right): y_j \ge 0$

Відповідне представлення у матричній формі матиме вигляд:

$$\begin{bmatrix} -3 & -2 & 2 & 5 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix} \Rightarrow Max, \begin{bmatrix} 2 & -4 & 4 & 3 & 0 & 0 \\ -1 & 5 & -5 & -2 & 1 & 0 \\ 5 & -3 & 3 & 4 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix} = \begin{bmatrix} 15 \\ 21 \\ 17 \end{bmatrix}, y \ge 0.$$

Приклад 3.2. Задача визначення оптимального асортименту.

Наявно p видів ресурсів в кількостях $a_1, a_2, ..., a_p$ та q видів виробів. Задана матриця $A = \left[a_{ik}\right]$, де a_{ik} характеризує норми витрат i-го ресурсу на одиницю k-го виробу (k=1,2,...,q).

Ефективність випуску одиниці k-го виробу характеризується показником c_k, що задовільняє умові лінійності.

Необхідно визначити план випуску виробів (оптимальний асортимент), за яким сумарна ефективність набуває найбільшого значення.

Розв'язання.

Кількість одиниць k-го виробу, що випускається підприємством, позначимо x_k. В цьому випадку математична модель задачі матиме наступний вигляд:

$$\sum_k c_k x_k \Rightarrow Max$$

$$\sum_k a_{ik} x_k \le a_i, i = 1,2,...,p, \forall(i=\overline{1,p}): x_i \ge 0.$$

Крім обмеження за ресурсами, в модель можуть бути введені додаткові обмеження на плановий випуск продукції $x_j > x_{j0}$, умови комплектності для складання $x_i : x_j : x_k = b_i : b_j : b_k$ для усіх i, j, k та ін.

Слід зауважити, що умова лінійності і для норм витрат, і для ефективності випуску продукції не завжди є справедливою. Як показує досвід, зі збільшенням об'ємів випуску продукції питома трудомісткість виготовлення на один виріб (деталь) має тенденцію до спадання, а не залишається сталою. Це ж стосується і питомої ефективності, мірою якої може бути прибуток на одиницю виробу – з насиченням ринку ціна виробу падає, і відповідно зменшується питомий прибуток. Звичайно, можуть виникати ситуації, в яких ці тенденції певною мірою компенсуватимуться, і лінійна модель даватиме непогані результати. В інших випадках добрим наближенням буде кусочно-лінійна апроксимація нелінійностей, що приводить до зростання розмірності відповідної лінійної моделі.

Приклад 3.3. Задача про суміші.

Наявні р компонентів i = 1,2,...,р, при сполученні яких в різних пропорціях отримують різні суміші. До складу кожного компоненту, а відповідно і до суміші входить q речовин. Кількість k-ї речовини k = 1,2,...,q, що входить до складу одиниці i-го компоненту і до складу одиниці суміші, позначимо, відповідно a_{ik} і a_k. Вважатимемо, що a_k залежить від a_{ik} лінійно, тобто якщо суміш складається з x_1 одиниць першого компоненту; x_2 - одиниць другого компоненту і т.д., то $a_k = \sum a_{ik} x_i$.

Задано р значень c_i, що характеризують ціну, масу або калорійність одиниці i-го компоненту і q величин b_k, що вказують мінімально необхідний процентний склад k-ї речовини у суміші.

Необхідно визначити склад суміші, при якому сумарна характеристика (ціна, маса або калорійність) виявиться найкращою.

Розв'язання.

Позначимо через $x_1, x_2, \dots, x_p$ величину компоненту р-го виду, що входить до суміші.

Математична модель задачі матиме наступний вигляд:

$$\sum_{i=1}^{p} c_i x_i \Rightarrow Min$$

$$\sum_{i=1}^{p} a_{ik} x_i \geq b_k, k = 1,2,..,q, \forall (i = \overline{1,p}) : x_i \geq 0.$$

Кожна умова – обмеження означає, що процентний склад k-ї речовини в одиниці суміші повинен бути не менше величини b_i .

До цієї ж моделі зводиться, наприклад, задача визначення оптимального раціону.

Приклад 3.4. Оптимальне розподілення взаємнозамінних ресурсів.

Наявно m видів взаємозамінних ресурсів $a_1, a_2, \dots, a_m$, що використовуються при виконанні n різних робіт в об'ємах $b_1, b_2, \dots, b_n$.

Задано числа λ_{ij} , що показують, скільки одиниць j-ї роботи можна отримати з одиниці i-го ресурсу, а також c_{ij} – витрати при виготовленні одиниці j-го продукту з i-го ресурсу.

Необхідно розподілити ресурси за роботами таким чином, щоб сумарна ефективність була найбільшою (або сумарні затрати - найменшими).

Кількість одиниць i-го ресурсу, яка виділена для виконання робіт j-го виду, позначимо x_{ij} .

Розв'язання.

Математична модель задачі буде наступною:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \Rightarrow Min$$

$$\sum_{i=1}^{m} \lambda_{ij} x_{ij} > b_j, j = 1,2,\dots,n; \sum_{j=1}^{n} x_{ij} = a_i, i = 1,2,\dots,m; \forall (i = \overline{1,p}) : x_i \geq 0$$

Перше обмеження означає, що план усіх робіт повинен бути виконаний повністю, а друге – що ресурси повинні бути використані повністю.

Прикладом такої задачі може служити задача про розподілення літаків за авіалініями, але в цій задачі змінні будуть цілочисельними, так як 0.4 літака не можна закріпити за авіалінією.

Приклад 3.5. Задача про розкрій.

Паперова фабрика випускає рулони шириною 2 м. За замовленнями постачаються рулони шириною 0.5, 0.7, 0.9 м. Наявні наступні замовлення:

№ замовл.	Потрібна ширина	Потрібна к-ть рулонів
1	0.5	150
2	0.7	200
3	0.9	300

Необхідно визначити такий спосіб розрізання рулонів, щоб кількість відходів була мінімальною.

Розв'язання.

Для формалізації цієї задачі необхідно виконати декілька попередніх дій. Спочатку сформуємо можливі доцільні варіанти розрізання двометрового рулону на рулони, які вимагаються за замовленням, і розрахуємо втрати для кожного варіанту розрізання рулону. Вважатимемо, що виконання плану замовлень можливо у вигляді певної комбінації основних способів розрізання. В цьому випадку необхідно буде визначити оптимальну кількість розрізаних кожним способом рулонів довжиною 2 м.

Ширина	1	2	3	4	5	6	Мінімальна к-ть рулонів	
0.5	0	2	2	4	1	0	150	y1
0.7	1	1	0	0	2	0	200	y2
0.9	1	0	1	0	0	2	300	y3
Втрати	0.4	0.3	0.1	0	0.1	0.2		
	x1	x2	x3	x4	x5	x6		

Нехай x_j – кількість стандартних рулонів довжиною 2 м, що розрізаються за допомогою j-го способу; y_i – залишкова к-ть рулонів і-го типу.

В цьому випадку до втрат належатимуть як „обрізки" з основних способів розрізання, так і зайві рулони стандартного для замовлення розміру, а кожне з обмежень відображатиме той факт, що кількість зайвих рулонів кожної зі замовлених довжин (кількість обмежень дорівнює кількості типорозмірів нарізаних рулонів) є різницею між загальною кількістю нарізаних рулонів даного типорозміру та кількістю замовлених рулонів цього ж типорозміру. Таким чином, формальна модель задачі матиме наступний вигляд:

$$0.4x_1 + 0.3\,x_2 + 0.1\,x_3 + 0\,x_4 + 0.1\,x_5 + 0.2\,x_6 + 0.5y_1 + 0.7y_2 + 0.9y_3 \Rightarrow Min$$

$$0\,x_1 + 2\,x_2 + 2\,x_3 + 4\,x_4 + 1\,x_5 + 0\,x_6 - 150 = y_1$$

$$1\,x_1 + 1\,x_2 + 0\,x_3 + 0\,x_4 + 2\,x_5 + 0\,x_6 - 200 = y_2$$

$$1\,x_1 + 0\,x_2 + 1\,x_3 + 0\,x_4 + 0\,x_5 + 2\,x_6 - 300 = y_3 \qquad \forall x, y \ge 0$$

Приклад 3.6. Загальна задача про розкрій матеріалів.

На розкрій надходить m різних матеріалів. Необхідно виготовити з них k різних комплектів виробів в кількостях, пропорційних $b_1, b_2, \ldots, b_k$ (умова комплектності).

Кожна одиниця j-го матеріалу, j = 1,2,...,m, може бути розкроєна n різними способами, так що при використанні і-го способу розкрою, і = 1,2,...,n. вийде a_{ij}^{k} одиниць k-го виробу.

Визначити план розкрою, що забезпечує максимальну кількість комплектів, якщо відомо, що обсяг запасу j-го матеріалу рівний a_j одиниць.

Розв'язання.

Кількість одиниць j-го матеріалу, що розкроюються і-тим способом, позначимо x_{ij}, а кількість комплектів виробів, що виготовляються, – x.

Математична модель задачі виглядає наступним чином:

$$x \rightarrow Max$$

$$\sum_{i=1}^{n} x_{ij} \le a_j;\ \sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij} a_{ij}^{(k)} = b_k x; \forall (i = \overline{1,n} \wedge j = \overline{1,m}) : x_i \ge 0;\ x \ge 0.$$

Перша умова означає обмеження запасу j-го матеріалу, а друга - умову комплектності.

Приклад 3.7. Оптимальні балансові моделі.

Розглянемо n-галузеву балансову модель з постійними технологічними коефіцієнтами, що задаються матрицею затрат $A = \left[a_{ij} \right]$, де a_{ij} - затрати продуктів і-ї галузі на виробництво одиниці продукції j-ї галузі. Виробничі потужності і-ї галузі обмежують її валовий випуск величиною d_i (i=1,...,n) і ціна остаточного продукту і-ї галузі складає c_i.

Необхідно визначити оптимальний валовий випуск продукції кожної галузі, при якому буде досягнено максимальний сумарний випуск остаточного продукту в грошовому представленні.

Розв'язання.

Позначимо вектор валової продукції усіх галузей $X = \left[x_1, x_2, \ldots, x_n\right]$, а вектор остаточного продукту $Y = \left[y_1, y_2, \ldots, y_n\right]$. Тоді y_i - обсяг продукту і-ї галузі, що йде на накопичення.

Між векторами X та Y існує наступний зв'язок: X = AX + Y, де AX продукт, що витрачається на споживання.

Звідси $Y = X \times \left[E - A\right], X = \left[E - A\right]^{-1} \times Y$.

Математична модель задачі має вигляд:

$$C^T Y = \sum_i c_i y_i \Rightarrow Max,$$

при умовах $X = \left[E - A\right]^{-1} Y \le d, Y \ge 0$.

Крім того, в задачі можуть бути додатково вказані обмеження, що накладаються на остаточні продукти, наприклад:

а) $y_1 : y_2 : \ldots : y_n = b_1 : b_2 : \ldots : b_n$ – умова комплектності;

б) $d_i \ge y_i \ge b_i$ - умова обмеженості випуску остаточного продукту.

Приклад 3.8. Геометричне розв'язання задачі ЛП.

Нафтопереробне підприємство виготовляє 2 види пального: бензин вищого сорту та бензин нижчого сорту, для виробництва яких використовується два види нафтопродуктів (сировини) – А та В. Максимально можливі добові запаси цих продуктів: А – 6, В – 8 тонн.

Витрати А та В на 1 тонну пального наведені в таблиці:

	Бензин вищого сорту	Бензин нижчого сорту	Запас
A	1	2	6
B	2	1	8

Добовий попит на бензин нижчого сорту не перевищує попиту на бензин вищого сорту більш, ніж на 1 тонну, а попит на бензин нижчого сорту не перевищує 2 тонни. Гуртові ціни на пальне за тонну становлять: 3 тис. – вищого сорту, 2 тис. – нижчого сорту.

Розв'язання.

Формальна модель задачі має наступний вигляд:

$Q = 3x_1 + 2x_2 \rightarrow Max$

$x_1 + 2x_2 \le 6$ [1], $\quad x_2 \le 2$ [4]

$2x_1 + x_2 \le 8$ [2], $\quad x_1 \ge 0$ [5],

$-x_1 + x_2 \le 1$ [3], $\quad x_2 \ge 0$ [6]

Розв'яжемо задачу геометрично.

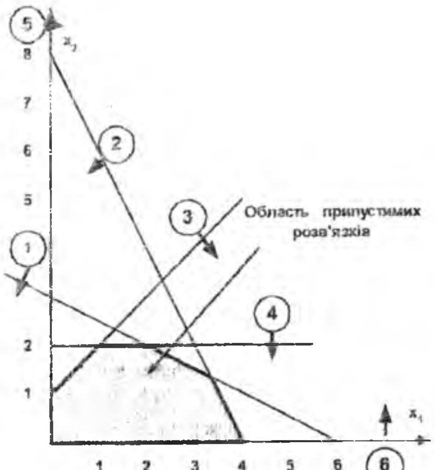


Рис 3.3. Область припустимих розв'язків прикладу

Оптимальний розв'язок:
$x_1 = 3\ 1/3$, $x_2 = 1\ 1/3$,
$Q = 3x_1 + 2x_2 = 12\ 2/3$.

Область припустимих розв'язків утворюється як область спільної дії всіх обмежень, а кожне з обмежень – це півплощина, до складу якої входить лінія, що відображає відповідне обмеження у випадку рівності правої та лівої частин. Таким чином, область припустимих розв'язків будується в просторі змінних задачі.

Функція мети в просторі змінних зображається у вигляді лінії рівня, і дві лінії рівня дозволяють визначити напрямок зростання (спадання)

Рис 3.4. Геометричне розв'язання задачі

значень функції мети. Побудуємо лінії рівня функції мети. Рухаючись у напрямку зростання функції мети паралельно до ліній рівня, вийдемо на кутову точку області припустимих розв'язків С, С(3 1/3; 1 1/3) з відповідним значенням функції мети Q=12 2/3, що й буде оптимальним розв'язком.

Приклад 3.9. Аналіз лінійної моделі на чутливість.

Як приклад, розглянемо задачу з попереднього прикладу. Розв'яжемо основні задачі аналізу на чутливість.

Задача 1. Визначення статусу ресурсів.

Обмеження 1 та 2 є зв'язуючими, оскільки оптимальна лінія рівня функції мети проходить через точку їх перетину, і збільшуючи запаси цих ресурсів, можна сподіватися покращити значення функції мети. Обмеженнями на ресурси в задачі максимізації вважатимемо обмеження типу $\le$. Ресурси, що відповідають зв'язуючим обмеженням, називаються дефіцитними. Таким чином, ресурси, що відповідають обмеженням 1 та 2 є дефіцитними, і відповідно, ресурси 3 та 4 – недефіцитними

Визначення меж збільшення дефіцитних ресурсів.

Координати точки К визначимо як розв'язок системи рівнянь:

$2x_1 + x_2 = 8$ (обмеження 2)

$x_2 = 2$ (обмеження 4) К(3; 2)

Збільшення запасу ресурсу 1 (зсув обмеження 1 паралельно самого до себе) доцільно до досягнення точки К, оскільки надалі діятимуть обмеження 4 та 2, які й визначатимуть положення нового оптимуму.

Рис 3.5. Межі зміни дефіцитного ресурсу 1

Обмеження за ресурсом 1: $x_1+2x_2<=6$. Підставляючи координати K, отримаємо: 3+2*2=7, тобто запас ресурсу 1 доцільно збільшувати від 6 до 7 тонн, при цьому значення функції мети збільшиться від 12 2/3 (в точці C) до $Q=3x_1+2x_2=13$ (в точці K).

Аналогічно для ресурсу 2 доцільні межі збільшення обмежені точкою J, координати якої знаходимо, розв'язавши систему рівнянь:

$x_1+2x_2=6$ (обмеження 1),

$x_2=0$ (обмеження 6). J(6; 0)

Таким чином, ресурс 2 доцільно збільшувати з 8 до $2x_1+x_2=12$ тон (обмеження 2). При цьому значення функції мети зросте з 12 2/3 до $Q=3x_1+2x_2=18$ (в точці J).

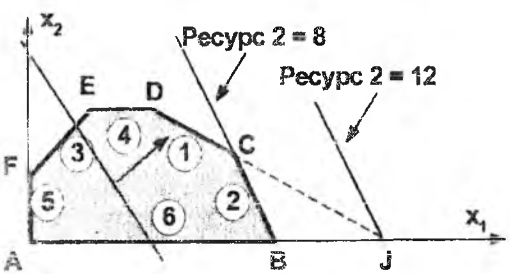


Рис 3.6. Визначення меж зміни значення ресурсу 2

Визначення меж зменшення значень недефіцитних ресурсів.

Запас ресурсу 3 можна зменшувати, не погіршивши значення оптимуму, до точки C(3 1/3; 1 1/3), тобто до значення

$-x_1+x_2=-3\ 1/3+1\ 1/3=-2$.

Аналогічно можна зменшити запас ресурсу 4, тобто $x_2=1\ 1/3$.

Значення функції мети не змінювалося – це була умова зменшення запасів ресурсів 3 та 4.

Зведемо результати рішення 1-ї задачі аналізу на чутливість до таблиці:

Ресурс	Тип	Макс. зміна запасу ресурсу	Відповідна зміна Q	Цінність ресурсу
1	деф.	7-6=1	13-12 2/3=1/3	1/3
2	деф.	12-8=4	18-12 2/3=5 1/3	4/3
3	нед.	-2-1=-3	12 2/3-12 2/3=0	0
4	нед.	1 1/3 - 2 = -2/3	12 2/3-12 2/3=0	0

Задача 2. Визначення цінності ресурсів.

У (Цінність ресурсу) = (Приріст Q)/(Макс. приріст ресурсу) - див. попередню таблицю. Таким чином, за рівних умов ресурс 2 купувати є вигідніше.

Задача 3. Визначення меж зміни коефіцієнтів функції мети.

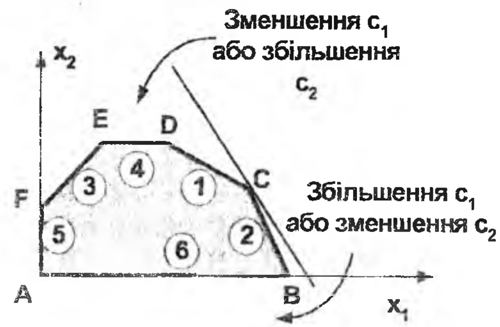


Рис 3.7. Визначення інтервалу зміни значень коефіцієнтів функції мети

Визначимо інтервал зміни c_1 при значенні

$c_2=2$, $Q=3x_1+2x_2$

Припустимі межі зміни визначаємо за рівністю кутів нахилу лінії рівня функції мети при її нахилянні до кутів нахилу обмежень 1 та 2.

$Q=c_1x_1+2x_2$, $x_2=-c_1/2*x_1+Q$

Обмеження 1:

$x_1+2x_2=6$

$x_2=-x_1/2+3$ $c_1/2=1/2$

Обмеження 2:

$2x_1+x_2=8$

$x_2=-2x_1+8$ $c_1/2=2$

Звідси межі для зміни $1<c_1<4$.

Аналогічно визначаємо межі зміни значень для коефіцієнта c_2.

РЕЗЮМЕ

3.1. Задачі ЛП входять як важлива складова до ширшого класу задач математичного програмування. Задачі математичного програмування за складністю поділяються на P (Polynomial) – та NP (Nondeterministically Polynomial) – повні. Більшість з дискретних та комбінаторних задач математичного програмування можуть бути в принципі розв'язані за допомогою повного перебору варіантів. Однак число кроків методу перебору зростає експоненційно в залежності від розміру задачі. Для деяких задач такого типу вдається побудувати ефективні методи розв'язування, однак кількість їх невелика. Аналогом алгоритмічної нерозв'язності в скінченій області є перебір експоненційного числа варіантів, а аналогом алгоритмічної розв'язності – існування суттєво більш економічного алгоритму, ніж перебір. Важливим результатом є доведення факту P – повноти задач лінійного програмування.

3.2. У лінійній задачі і критерій якості, і обмеження є лінійними функціями від змінних, звідки й походить назва „задача лінійного програмування". Задачу ЛП в загальній формі завжди можна перетворити в задачу в канонічній формі за допомогою підстановок.

3.3. Методи математичного програмування використовуються в економічних, організаційних, військових і ін. системах для вирішення розподільчих задач, які виникають у разі, коли наявних ресурсів не вистачає для виконання кожної з передбачених робіт ефективним чином і необхідно найкращим чином розподілити ресурси за роботами відповідно до обраного критерію оптимальності. При описанні реальної ситуації за допомогою лінійної моделі слід перевіряти наявність у моделі таких властивостей, як пропорційність і адитивність.

3.4. Задачі лінійного програмування з двома змінними можуть бути розв'язані графічно на площині у просторі двох змінних. Якщо задача ЛП містить n змінних і m обмежень (n>m), записаних у формі нерівностей, не враховуючи обмеження невід'ємності $x_j \geq 0$, то в оптимальний розв'язок входить не більше, ніж m ненульових компонент вектора x.

3.5. Аналіз лінійних моделей на чутливість здійснюється після того, як отриманий оптимальний розв'язок, і має за мету виявлення чутливості оптимального розв'язку до певних змін значень параметрів первісної моделі. Виділяють 3 основні задачі аналізу на чутливість: визначення доцільних меж зміни ресурсів; визначення цінності ресурсів; визначення меж зміни значень коефіцієнтів функції мети, при яких положення оптимуму не зміниться.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 3.1. Приведення загальної задачі ЛП до канонічного вигляду.

$$
\begin{array}{llll}
-4x_1 - & 3x_2 - & 6x_3 & \Rightarrow Min \\
4x_1 + & 2x_2 - & 3x_3 & \geq 15 \\
4x_1 - & 5x_2 + & 2x_3 & = 21 \\
-5x_1 + & 3x_2 - & 4x_3 & \leq 30 \\
\multicolumn{4}{l}{x_1 >< 0, \quad x_2 \leq 0, \quad x_3 \geq 0.}
\end{array}
$$

Приведіть до канонічного вигляду та представте у матричному вигляді наступну задачу лінійного програмування.

Завдання 3.2. Побудова математичної моделі задачі ДО.

Побудуйте математичну модель наступної задачі.

Для виготовлення брусів довжиною 1.2м, 3м та 5м у співвідношенні 2:1:3 для розпилювання наявно 195 брусів довжиною 6м. Визначити план розпилювання, що забезпечує максимальне число комплектів.

Завдання 3.3. Геометричне розв'язання задачі ЛП.

$$
\begin{array}{llll}
2x_1 + & 3x_2 & & \Rightarrow Max \\
x_1 + & 3x_2 & \leq 18 & \\
2x_1 + & x_2 + & \leq 16 & \\
 & x_2 & \leq 5 & \\
\multicolumn{4}{l}{x_1 \geq 0, \quad x_2 \geq 0 \quad .}
\end{array}
$$

Розв'яжіть геометрично та проаналізуйте на чутливість наступну задачу лінійного програмування:

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Наведіть загальний вигляд задачі математичного програмування.
2. Яке місце посідає задача лінійного програмування серед задач математичного програмування і чому?
3. Назвіть основні класи задач математичного програмування.
4. В чому полягає поняття складності задач математичного програмування ?
5. Наведіть формальну постановку задачі лінійного програмування в загальній та канонічній формі.
6. Які умови повинні виконуватися для задачі, щоб вона могла бути представленою у вигляді задачі лінійного програмування?
7. Розкрийте економічний сенс задач аналізу на чутливість.

ТЕМА 4. РОЗВ'ЯЗУВАННЯ ЗАДАЧІ ЛІНІЙНОГО ПРОГРАМУВАННЯ СИМПЛЕКС - МЕТОДОМ

Для розв'язування задачі лінійного програмування (ЛП) за допомогою симплекс-методу (СМ) насамперед її слід представити у канонічній формі, а це можна зробити завжди. Наступним кроком є пошук початкового базового розв'язку, що може бути здійснене або безпосередньо шляхом введення додаткових змінних для незначної кількості задач, або ж застосовуючи двохетапний метод чи метод великих штрафів – для довільної задачі ЛП в канонічній формі. Головна особливість задачі ЛП, на основі якої побудований метод їх розв'язування, це те, що повний простір припустимих рішень такої задачі визначається за допомогою скінченої множини точок – базових припустимих розв'язків. Основні результати, наведені у вигляді теорем, стверджують, що оптимум слід шукати у вершинах многогранника обмежень, а отримати координати цих вершин можна як множину базових розв'язків системи m рівнянь з n змінними (n>>m). Однак безпосереднє використання цих результатів приводить до повного перебору вершин з визначенням тієї, в якій критерій якості отримує екстремальне значення, а тому внаслідок опуклості многогранника-симплекса достатньо в процесі розв'язування просуватися від однієї вершини до іншої сусідньої таким чином, щоб не погіршувати розв'язок. Симплекс-метод використовує ідею переходу до сусідньої вершини по „найкрутішому" ребру, що дозволяє досягнути значно кращих результатів, аніж повний перебір. Під час розв'язування задач ЛП можливе виникнення особливих випадків, а саме: відсутність припустимих розв'язків, необмеженість значення критерію якості, зациклення, які потрібно вміти вчасно виявляти і які, зазвичай, у більшості випадків виникають внаслідок недостатньо коректної формалізації реальної задачі. На основі результатів розв'язання задачі ЛП за допомогою симплекс-методу задача інтерпретується економічно – визначаються дефіцитні та недефіцитні ресурси, їх цінність, доцільні межі зміни їх запасів та межі зміни значень коефіцієнтів критерію оптимальності, при яких координати оптимального розв'язку не змінюються. Окрім того, існує клас задач, важливих з економічної точки зору – задачі дробово-лінійного програмування, які шляхом підстановки приводяться до задачі лінійного програмування та розв'язуються за допомогою симплекс-методу.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати:⇨основні теореми лінійного програмування;⇨алгоритм симплекс-методу;⇨теоретичне обґрунтування симплекс-методу;⇨переваги та недоліки методу великих штрафів та двохетапного методу;⇨особливі випадки симплекс-методу;

вміти:⇨розв'язувати задачі лінійного програмування за допомогою симплекс-методу;⇨знаходити початкові базові розв'язки задачі ЛП методом великих штрафів та двохетапним;⇨виявляти особливі випадки при розв'язуванні задач ЛП;⇨інтерпретувати симплекс-таблицю оптимального розв'язку задачі ЛП з економічної точки зору;⇨розв'язувати задачі дробово-лінійного програмування.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ канонічна форма задачі ЛП
- ☑ симплекс
- ☑ кутова точка
- ☑ многогранник обмежень
- ☑ ведучий стовпчик
- ☑ базовий розв'язок
- ☑ лінійна комбінація точок
- ☑ ведучий рядок
- ☑ штучна змінна
- ☑ метод великих штрафів
- ☑ симплекс-таблиця
- ☑ додаткова змінна
- ☑ критерії оптимальності біжучого розв'язку задачі ЛП
- ☑ двохетапний метод
- ☑ задача дробово-лінійного програмування

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

4.1. Основні теоретичні відомості про задачу лінійного програмування. Загальна схема алгоритму симплекс-методу та його таблична форма.

4.2. Теоретичне обґрунтування СМ.

4.3. Методи знаходження початкового базового розв'язку: метод великих штрафів та двохетапний метод.

4.4. Особливі випадки СМ та відображення їх в симплекс-таблицях. Інтерпретація симплекс-таблиць.

4.5. Задачі дробово-лінійного програмування.

4.1. ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ ПРО ЗАДАЧУ ЛІНІЙНОГО ПРОГРАМУВАННЯ

Як ми розглядали раніше, задача лінійного програмування завжди може бути приведена до канонічної форми:

$$Q(x)=\sum_{j=1}^{n} c_j x_j \Rightarrow Max \ ,$$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i,\ b_i \geq 0,\ i=\overline{1,m},\ j=\overline{1,n},\ x_j \geq 0. \qquad (4.1)$$

Тому надалі при розгляді симплекс-методу ми вважатимемо, що задача ЛП представлена в канонічній формі. В матричній формі задача має вигляд:

$$c^T \times x \Rightarrow Max,\ A\times x = b,\ x \geq 0,\ b \geq 0, \qquad (4.2)$$

де x - вектор-стовпчик змінних задачі ЛП, c - вектор- стовпчик значень коефіцієнтів критерія якості – функції мети задачі, A - матриця значень коефіцієнтів обмежень, b - вектор-стовпчик значень правих частин системи обмежень, ненегативність елементів якого досягається помноженням лівої та правої частини відповідного обмеження при потребі на (-1), верхній індекс T біля вектора c означає операцію транспонування.

Обмеження $A\times x=b$ можна записати також у векторному вигляді таким чином:

$$\sum_{j=1}^{n} P_j x_j = P_0,\ P_0=b,\ P_0 \geq 0,\ P_j=\left(a_{1j},\ldots,a_{mj}\right)^T, \qquad (4.3)$$

Головна особливість задачі ЛП, на основі якої побудований метод її розв'язування, це те, що повний простір припустимих рішень такої задачі визначається за допомогою скінченої множини точок – базових припустимих розв'язків.

Симплекс-метод грунтується на двох основних теоремах ЛП. В більшості випадків метод доведення теорем є методом „від протилежного" – припускається, що правильним є протилежне твердження (яке разом з прямим твердженням утворює повну систему), і доводиться, що воно неправильне. З цього факту робимо висновок, що правильним є пряме твердження.

Наступна теорема дозволяє обмежитися при пошуку оптимального розв'язку лише вершинами симплексу – многогранника обмежень – тобто пошук на безмежній множині розв'язків замінюється пошуком на скінченій (хоча й дуже великій) множині вершин многогранника обмежень. Для означення – що таким чином теорема доведена чи це й

доводить теорему – вживатимемо загальноприйняте позначення ■

Теорема 4.1. Якщо задача ЛП має оптимальний розв'язок, то максимальне значення функція мети набуває в одній з вершин (кутових точок) многогранника, що утворений обмеженнями. Якщо функція мети отримує це значення більш ніж в одній кутовій точці, то вона має таке ж значення в довільній їх лінійній комбінації.

Доведення. Нехай $x_1,\ldots,x_m \in R^n$ – кутові точки (вершини многогранника) множини $X \subset R^n$. Тоді довільна точка $x\in X$ є опуклою лінійною комбінацією точок $x_1,\ldots,x_m \in R^n$, якщо

$$x=\sum_{i=1}^{m}\lambda_i x_i,\ \sum_{i=1}^{m}\lambda_i = 1,\ \lambda_i \geq 0.$$

Нехай $x^* = \arg\max_{x\in X} Q(x)$ — точка, в якій досягається оптимальне значення (максимум) функції мети. Припустимо, що x^* не є кутовою точкою множини X, тобто $c^T x^* \geq c^T x$. Функція мети $Q(x)$ є лінійною, тобто $Q\left(\sum_{i=1}^{m}\lambda_i x_i\right)=\sum_{i=1}^{m}\lambda_i Q(x_i)$, внаслідок чого

$$c^T x^* = \sum_{i=1}^{m} c^T \lambda_i x_i \leq \sum_{i=1}^{m}\lambda_i \max_{1\leq i\leq m}\left\{c^T x_i\right\} = \max_{1\leq i\leq m}\left\{c^T x_i\right\}\sum_{i=1}^{m}\lambda_i = \max_{1\leq i\leq m}\left\{c^T x_i\right\} \qquad (4.4)$$

тобто $c^T x^* \leq \max\left\{c^T x_i\right\}$. Однак x^* — оптимальний розв'язок, і справедливе співвідношення $c^T x^* \leq \max\left\{c^T x_i\right\}$. Таким чином $c^T x^* \leq \max\left\{c^T x_i\right\}$, тобто існує хоча б одна кутова точка x_k, $k=\arg\max_{1\leq i\leq m}\left\{c^T x_i\right\}$, в якій функція мети набуває максимального значення.

Припустимо, що оптимальні розв'язки знаходяться в кутових точках $x_1, x_2,\ldots,x_u$. Розрахуємо значення функції мети для їх довільної лінійної комбінації $x'=\sum_{i=1}^{u}\lambda_i x_i,\ \sum_{i=1}^{u}\lambda_i=1,\ \lambda_i\geq 0$. $Q(x')=c^T x'=\sum_{i=1}^{u} c^T\lambda_i x_i$. Оскільки x_i за умовою є оптимальними,

то $Q(x^*)=c^T x^* = c^T x_i,\ i=\overline{1,u}$, та

$$Q(x')=c^T x^* \sum_{i=1}^{u}\lambda_i = Q(x^*) \ ■$$

Таким чином, координати вершин многогранника припустимої області дають достатній об'єм інформації для знаходження оптимального розв'язку.

Наступна теорема надає механізм для обчислення координат вершин многогранника обмежень – а маючи ці значення, обчислюються значення функції мети – критерію якості – і з'являється можливість порівняти ці розв'язки за якістю і обрати кращий.

Базовий розв'язок системи $A \times x = b$ визначається базою – m незалежними векторами P_i. В довільній екстремальній точці змінна x_j, якій відповідає P_j, що не входить до бази, має нульове значення. Таким чином, загальна кількість базових розв'язків становить:

$$C_n^m = \frac{n!}{m!(n-m)!} \tag{4.5}$$

Теорема 4.2. Базові розв'язки системи $A \times x = b$ повністю визначають всі її кутові точки.

Доведення. Припустимо, що базовий розв'язок x (тобто розв'язок рівняння $P_1x_1 + P_2x_2 + ... + P_kx_k = P_0$, $k \le m$, $\forall x_i \ge 0$, $(x^D = x_1,...,x_k,\underbrace{0,...,0}_{n-k})$ — точка припустимої області) визначає точку $x^D \in R^n$, що не є кутовою (базовим розв'язком). У цьому випадку $x^D = \lambda x_1 + (1-\lambda)x_2$, $0 < \lambda < 1$, x_1 та x_2 — різні кутові точки.

Оскільки останні *n-k* компонент x^D є нулями, то й відповідні компоненти x_1 та x_2 — також нулі. Тому для інших компонент $x_1 = \left(x_1^{(1)},...,x_n^{(1)}\right)$ та $x_2 = \left(x_1^{(2)},...,x_n^{(2)}\right)$ справедливі співвідношення:

$$\begin{aligned} &P_1x_1^{(1)} + P_2x_2^{(1)} + ... + P_kx_k^{(1)} = P_0, \\ &P_1x_1^{(2)} + P_2x_2^{(2)} + ... + P_kx_k^{(2)} = P_0. \end{aligned} \tag{4.6}$$

Віднявши одну рівність від іншої, отримаємо:

$$P_1\left(x_1^{(1)} - x_1^{(2)}\right) + P_2\left(x_2^{(1)} - x_2^{(2)}\right) + ... + P_k\left(x_k^{(1)} - x_k^{(2)}\right) = 0 \tag{4.7}$$

Оскільки $P_1, P_2 ... P_k$ – незалежні, то різниці компонент

$$\left(x_1^{(1)} - x_1^{(2)}\right) = 0, \left(x_2^{(1)} - x_2^{(2)}\right) = 0, ..., \left(x_k^{(1)} - x_k^{(2)}\right) = 0$$

лише в тому випадку, коли $x_1 = x_2$. Це суперечить припущенню, що ці точки є різними кутовими точками множини X. Так як x^D не може бути представлена в вигляді опуклої комбінації інших точок, вона є кутовою точкою множини X ■

У принципі, координати цих точок можна розрахувати, відсіюючи всі

неприпустимі точки та залишаючи ту, для якої значення функції мети максимальне. Але такий шлях є найменш ефективним, оскільки ґрунтується на ідеї повного перебору.

Симплекс-метод ґрунтується на ідеї послідовного просування від однієї вершини многогранника обмежень до сусідньої іншої, в якій значення функції мети є кращим (або принаймні не гіршим), ніж у попередній. Для того ж, щоб мати можливість застосувати симплекс-метод та знайти оптимальний розв'язок задачі ЛП, потрібно задачу в канонічній формі привести до вигляду з початковим базовим розв'язком.

Таким чином, задача в канонічній формі приводиться до форми з початковим базовим розв'язком, тобто до форми, в якій матриця коефіцієнтів обмежень $A = \{a_{ij}\}$ має n одиничних стовпчиків. Таке перетворення здійснюється шляхом застосування методу великих штрафів або двохетапного методу, що розглядаються нижче (п. 4.3).

Загальна схема алгоритму симплекс-методу та його таблична форма виглядає наступним чином і включає наступні кроки.

Укрупнена форма алгоритму симплекс-методу.

Крок 1. Знаходження початкового припустимого базового розв'язку.

Крок 2. Перевірка на зупинку алгоритму: чи знайдений розв'язок є оптимальним? Якщо так, то стоп.

Крок 3. Пошук наступної вершини многогранника, до якої необхідно перейти: визначення змінної, що вводиться до бази (ведучого стовпчика, ведучої колонки); визначення змінної, що виводиться з бази (ведучого рядка).

Крок 4. Перехід до нової вершини з застосуванням методу Жордана-Гауса. Перехід до п.2.

Таблична форма алгоритму симплекс-методу.

Вважаємо, що крок 1 укрупненої форми алгоритму вже реалізовано – знайдений початковий базовий розв'язок – тобто є форма з m одиничними векторами.

***Крок 1.* Побудова початкової симплекс-таблиці.**

Будуємо початкову симплекс-таблицю і розраховуємо значення:

$$Q = \sum_{i \in I^B} c_i b_i, \; z_j = \sum_{i \in I^B} c_i a_{ij}, \; \Delta_j = z_j - c_j, \tag{4.8}$$

де Δ_j – критерії оптимальності знайденого розв'язку, Q – біжуче значення функції мети для знайденого базового розв'язку.

***Крок 2.* Перевірка на оптимальність:**

a) $\forall \Delta_j \geq 0$ - розв'язок оптимальний. Тобто, якщо всі значення критеріїв досягнення оптимального розв'язку не є негативними, то досягнутий оптимум. **Стоп.**

b) $(\exists \Delta_j < 0) : \forall a_{ij} \leq 0$ – функція мети необмежено зростає. **Стоп.** У цьому випадку область обмежень відкрита „згори" (для задачі максимізації) – і функція мети необмежено зростає. Для реальних економічних задач це, зазвичай, свідчить про те, що задача була некоректно сформульована.

c) $(\forall j \in I) \wedge (\Delta_j < 0) : \exists a_{ij} > 0$ - в цьому випадку існують можливості покращення значення функції мети, а тому переходимо до наступного кроку.

***Крок 3.* Визначення ведучого стовпчика та ведучого рядка.**

З визначенням ведучого стовпчика визначається, яка змінна вводиться до бази. Індекс цієї змінної визначаємо наступним чином:

$$j^* = \arg \max_{j \in I^N \wedge \Delta_j < 0} |\Delta_j|, \tag{4.9}$$

тобто обирається стовпець з максимальним за абсолютною величиною значенням критерію досягнення оптимального розв'язку серед усіх від'ємних значень.

Визначення ведучого рядка (яка змінна виводиться з бази):

$$i^* = \arg \min_{j \in I^B \wedge a_{ij} > 0} \left(\frac{b_i}{a_{ij^*}} \right), \tag{4.10}$$

тобто обирається змінна x_{i^*}, для якої досягається мінімальне значення частки від ділення $\frac{b_i}{a_{ij^*}}$, серед усіх додатних значень a_{ij^*}. Ведучим елементом є елемент $a_{i^*j^*}$, що знаходиться на перетині ведучого стовпця та ведучого рядка.

***Крок 4.* Перехід до нової симплекс-таблиці.**

Заповнюємо СТ в наступній послідовності: заповнюємо стовпчик базових змінних та відповідні значення c_b; заповнюємо стовпчики базових змінних (одиничні вектори!), відповідні значення $\Delta_i = 0,\ i \in I^B$; заповнюємо рядок, що відповідає ведучому рядкові в попередній СТ –

$$a'_{i^*j} = a_{i^*j} / a_{i^*j^*};\ b'_{i^*} = b_{i^*} / a_{i^*j^*}, \tag{4.11}$$

таким чином усі елементи старого ведучого рядка ділимо на ведучий елемент і записуємо на відповідне місце в новій СТ; значення інших

елементів СТ ($i \neq i^*$) обчислюємо за формулами –

$$a'_{ij} = a_{ij} - a_{ij^*} \times a'_{i^*j},\ b'_i = b_i - a_{ij^*} \times \frac{b_{i^*}}{a_{i^*j^*}}, \tag{4.12}$$

обчислюємо Q та Δ_j за формулами (4.8). Перехід до п.2.

Обчислення значень елементів нової симплекс-таблиці зручно здійснювати за допомогою „правила трикутника" (рис. 4.1) – вершинами трикутника є значення обчислюваного елемента в попередній симплекс-таблиці *a*, значення елементу в ведучому стовпчику попередньої симплекс-таблиці, в який „впирається" горизонтальна стрілка від перераховуваного елемента в старій симплекс-таблиці *b*, та значення в перерахованому рядку нової таблиці, що за положенням у новій таблиці відповідає ведучому рядку старої таблиці *c*, значення перерахованого елементу a' відповідає *a* в попередній таблиці.

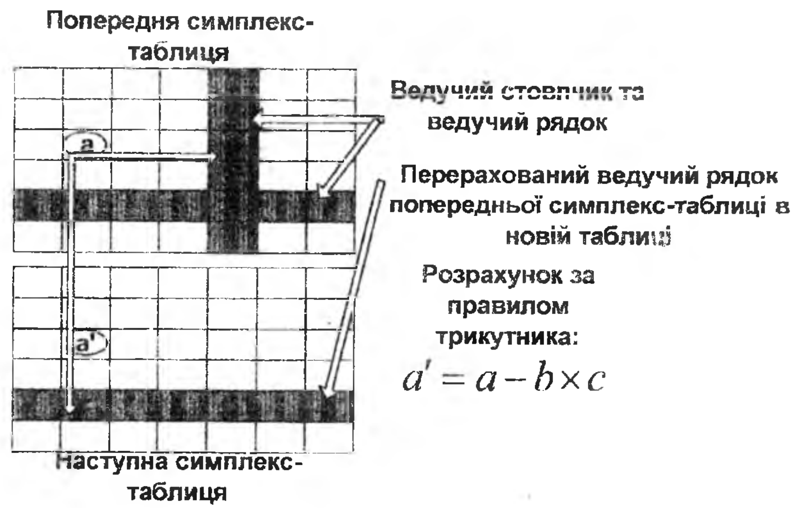


Рис. 4.1. Ілюстрація правила трикутника

Наочно порядок обчислень проілюстрований на рис. 4.2.

Якщо задача має вироджені опорні плани (базові розв'язки), то на одній з ітерацій одна або декілька змінних базового розв'язку рівні 0, що може приводити до зациклювання.

При здійсненні обчислень зручно починати та продовжувати їх у межах однієї сторінки (звичайно, це стосується лише розрахунків „*вручну*").

x_b	c_b	P_0	c_1	c_2	…	c_m	…	c_j^*		c_n
			P_1	P_2	…	P_m	…	P_j^*		P_n
x_1	c_1	b_1	1	0		1		a_{1j}^*		a_{1n}
x_2	c_2	b_2	0	1	…	0	…	a_{2j}^*		a_{2n}
.	.	.	.	.	…	.	…	.	.	.
x_i^*		b_i^*	0	0	…	0	…	a_{ij}^{**}		a_{in}
.	.	.	.	.	…	.	…	.	.	.
x_m	0	b_m	0	0	…	0	…	a_{mj}		a_{mn}
Q	=	Q_0	0	0	…	0	…	Δ_j^*		Δ_n
x_1	c_1	b_1'	1	0	…	1	…	0		a_{1n}'
x_2	c_2	b_2'	0	1	…	0	…	0		a_{2n}'
.	.	.	.	.	…	.	…	.	.	.
x_j^*		b_i^* / a_{ij}^{**}	0	0	…	0	…	1		a_{in} / a_{ij}^{**}
			.	.	…	.	…	.	.	.
x_m	0	b_m'	0	0	…	0	…	0		a_{mn}'
Q	=	Q_0	0	0	…	Δ_m'	…	0		Δ_n'

Перша симплекс-таблиця

Друга симплекс-таблиця

Рис. 4.2. Порядок обчислень при застосуванні симплекс-методу

4.2. ТЕОРЕТИЧНЕ ОБҐРУНТУВАННЯ СМ

Обґрунтуємо, чому і яким чином відбувається перехід до сусідньої вершини симплекса в термінах симплекс-таблиці та чому умова $\forall \Delta_j \geq 0$ є необхідною та достатньою умовою того, що оптимальний розв'язок досягнутий.

Задача ЛП в матричній формі, як ми вже переконалися, виглядає наступним чином:

$$Q = c^T \times x \Rightarrow Max, A \times x = b, x \geq 0. \tag{4.13}$$

Розіб'ємо A на дві підматриці B, N відповідно до поділу змінних на змінні бази x_B та небазові змінні x_N, $A(m \times n), B(m \times m)$, $N(m \times (n-m))$.

Представимо задачу у канонічній формі з врахуванням цього розбиття:

$$A = [B|N],\ x = \left[\frac{x_B}{x_N}\right], Q = \left[c_B^T | c_N^T\right] \times \left[\frac{x_B}{x_N}\right] \Rightarrow Max$$

$$[B|N] \times \left[\frac{x_B}{x_N}\right] = b. \tag{4.14}$$

Таким чином, $Q = c_B^T x_B + c_N^T x_N$. Розв'яжемо систему рівнянь відносно x_B та підставимо отримане значення в вираз для Q:

$$Bx_B + Nx_N = b,\ Bx_B = b - Nx_N,\ B^{-1}Bx_B = B^{-1}b - B^{-1}Nx_N.$$

Оскільки $B^{-1}B = E$ - одинична матриця, то звідси $x_B = B^{-1}b - B^{-1}Nx_N$, і підставляючи, отримуємо:

$$Q = c_B^T\left(B^{-1}b - B^{-1}Nx_N\right) + c_N^T x_N. \tag{4.15}$$

В результаті отримуємо задачу ЛП у перетвореному вигляді:

$$Q = c_B^T B^{-1} b - \left(c_B^T B^{-1} N - c_N^T\right) x_N \Rightarrow Max,$$

$$x_B = B^{-1}b - B^{-1}Nx_N, x_B \geq 0, x_N \geq 0. \tag{4.16}$$

Проаналізуємо отримані результати.

$$\text{При } x_N = 0 \ \Rightarrow \ Q = c_B^T B^{-1} b,\ x_B = B^{-1} b. \tag{4.17}$$

Якщо $\forall \Delta_j \geq 0$, то, оскільки $\Delta = c_B^T B^{-1} N - c_N^T$, введення будь-якої з небазових змінних x_N зменшить значення Q. Таким чином, ця умова є необхідною для того, щоб знайдений базовий розв'язок був оптимальним.

Вона ж є і достатньою. Нехай одне або декілька $\Delta_j < 0$, $x_B = B^{-1}b - B^{-1}Nx_N$. Якщо змінну x_j вводимо в базу, то повинна виконуватися умова $x_B = B^{-1}b - B^{-1}Nx_N \geq 0$, а тому з бази виводиться змінна x_{i^*}, яка першою досягне значення 0, а саме змінна з індексом

$$i^* = \arg\min_i \left\{ \frac{\left(B^{-1}b\right)_i}{\left(B^{-1}P_j\right)_i} \middle| \left(B^{-1}P_j\right)_i \geq 0 \right\}, \quad x_j = \frac{\left(B^{-1}b\right)_i}{\left(B^{-1}P_j\right)_i}. \tag{4.18}$$

4.3. МЕТОДИ ЗНАХОДЖЕННЯ ПОЧАТКОВОГО БАЗОВОГО РОЗВ'ЯЗКУ: МЕТОД ВЕЛИКИХ ШТРАФІВ ТА ДВОХЕТАПНИЙ МЕТОД

Метод великих штрафів та двохетапний метод служать для можливості розв'язування задачі ЛП в тому випадку, коли початкового базового розв'язку з умови задачі отримати безпосередньо неможливо. В цьому випадку початковий базовий розв'язок будується шляхом введення штучних змінних у тій кількості, щоб отримати початковий базовий розв'язок. Звичайно, це приводить до необхідності введення додаткових обмежень – в початковому базовому розв'язку допускається від'ємність штучних змінних, тоді як в оптимальному розв'язкові їх не повинно бути, так як це порушуватиме деякі з обмежень первісної задачі.

Існує два шляхи позбавлення від штучних змінних в остаточному оптимальному розв'язкові – це або введення штрафів за відхилення значень штучних змінних від нуля та включення відповідних складових до функції мети (метод великих штрафів), або розв'язання на першому етапі допоміжної задачі, що дозволить отримати початковий базовий розв'язок без штучних змінних з подальшим розв'язанням класичним шляхом (двохетапний метод).

В методі великих штрафів критерій якості модифікується і має вигляд

$$Q(x)=\sum_{j=1}^{n}c_jx_j-\sum_{k=1}^{u}MR_k \Rightarrow Max,\ M>>0, R_k\ge 0\ , \qquad (4.19)$$

де u – кількість штучних змінних, що є мінімально необхідною для побудови початкового базового розв'язку, M – значення штрафів, R – штучні змінні. Таким чином, навіть невеликі відхилення штучних змінних від нуля приводитимуть до значного зменшення значення критерію якості, яке потрібно максимізувати, а тому при достатньо великих значеннях M значення штучних змінних мали би бути в оптимальному розв'язку нульовими, тобто штучні змінні виводитимуться з бази.

Метод великих штрафів приводить до необхідності розв'язання проблеми визначення конкретних значень штрафів. Якщо значення штрафів завеликі, то похибки заокруглення дуже суттєво можуть вплинути на результат (довжина машинного слова в комп'ютері має певне скінчене значення), і знайдений розв'язок буде неоптимальним внаслідок мізерного вкладу значень первісних коефіцієнтів функції мети, тобто ці коефіцієнти будуть приблизно нульовими порівняно зі значеннями штрафів. Якщо ж значення штрафів невеликі, то штучні змінні в оптимальному розв'язку

будуть ненульовими, тобто знайдений розв'язок буде неприпустимим.

Цих недоліків позбавлений двохетапний метод, в якому на першому етапі розв'язується задача з введеними штучними змінними (тобто з тими ж модифікованими обмеженнями), але з функцією мети

$$\sum_i R_i \Rightarrow Min. \qquad (4.20)$$

Якщо в результаті розв'язування задачі отримуємо розв'язок, у якому всі значення штучних змінних нульові, то первісна задача має припустимі розв'язки, в іншому випадку задача не має розв'язків. Якщо такий розв'язок знайдений, то це означає, що всі штучні змінні виведені з числа базових, і переходимо до наступного етапу, на якому оптимальний розв'язок першого етапу використовуємо як початковий базовий розв'язок (всі штучні змінні виведені з бази!), виключаємо штучні змінні з таблиці і розв'язуємо задачу за допомогою звичайного симплекс-методу. При цьому не забуваємо повернутися до первісного критерію, тобто відновити в симплекс-таблиці первісні значення коефіцієнтів функції мети.

4.4. ОСОБЛИВІ ВИПАДКИ СМ ТА ВІДОБРАЖЕННЯ ЇХ У СИМПЛЕКС-ТАБЛИЦЯХ. ІНТЕРПРЕТАЦІЯ СИМПЛЕКС-ТАБЛИЦЬ

Особливі випадки виникають внаслідок певних аномалій в умові задачі ЛП і поділяються на наступні.

a_1). **Виродженість** – наявність надлишкових обмежень – якщо ця ситуація виникає в вершині, де знаходиться оптимальний розв'язок, то такі обмеження несуттєві.

a_2). **Виродженість** – наявність надлишкових обмежень в неоптимальних

Рис. 4.3. Графічне представлення виродженості

вершинах. Цьому випадку відповідає відсутність однозначного вибору в симплекс-таблиці – існує щонайменше два однакових значення Δ, і хоча б одна з базових змінних рівна нулю. Не існує способу виявлення надлишкових обмежень безпосередньо з симплекс-таблиці.

В цьому випадку можливе виникнення зациклювання програми. Хоча й існують спеціальні прийоми, які запобігають зациклюванню, в практичних алгоритмах вони не застосовуються, оскільки суттєво подовжують час розв'язування задачі.

б) **Безмежна множина оптимальних розв'язків**. Якщо небазова змінна x_j зі значенням c_j=0 включається в базу, то її включення не змінює значення функції мети Q, але дозволяє отримати наступну кутову точку.

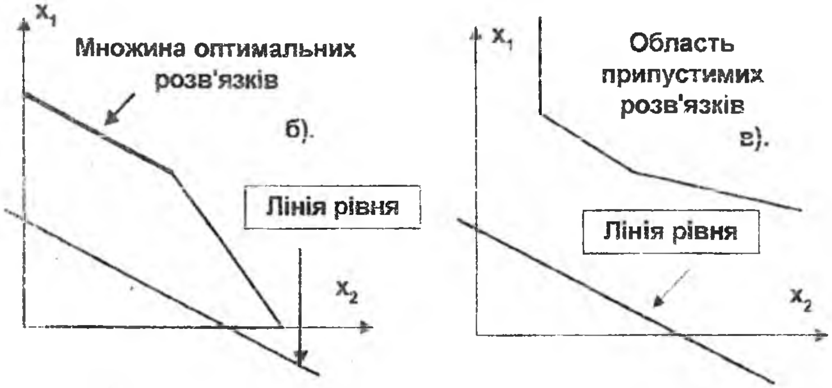


Рис. 4.4. Безмежна множина оптимальних розв'язків та необмежені розв'язки

в) **Необмежені розв'язки**: якщо знайдеться $\Delta_j < 0$, для якого всі значення $a_{ij} \leq 0$ у відповідному j-му стовпці симплекс-таблиці, то максимальне (мінімальне) значення функції мети буде необмеженим.

г) **Необмежена область розв'язків**: в симплекс-таблиці $a_{ij} \leq 0$, але відповідне значення $\Delta_j \geq 0$.

д) **Відсутність припустимих розв'язків**: хоча б одна штучна змінна в оптимальному розв'язку задачі після першого етапу двохетапного методу не рівна нулеві.

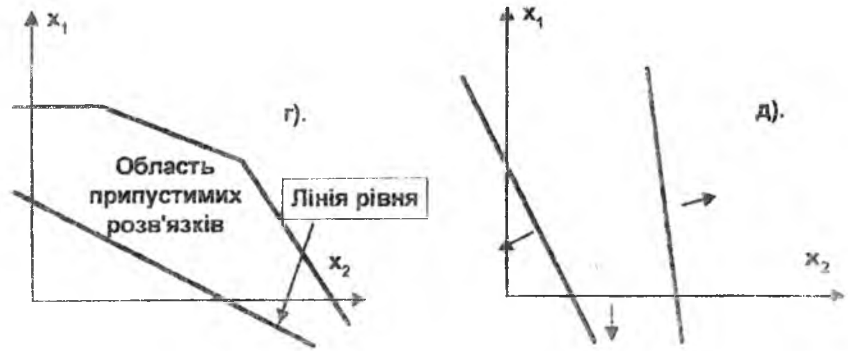


Рис. 4.5. Необмежена область розв'язків та відсутність припустимих

Інтерпретація симплекс-таблиць.

За допомогою симплекс-таблиць безпосередньо або за допомогою нескладних обчислень можна отримати значно більшу інформацію, аніж лише координати оптимального розв'язку, а саме:

- ⇨ *оптимальний розв'язок;*
- ⇨ *статус ресурсів;*
- ⇨ *цінність кожного з ресурсів;*
- ⇨ *чутливість оптимального розв'язку до зміни запасів ресурсів, варіацій значень коефіцієнтів функції мети та інтенсивності споживання ресурсів.*

Конкретна реалізація цих пунктів розглядається на прикладі 4.4.

4.5. ЗАДАЧІ ДРОБОВО-ЛІНІЙНОГО ПРОГРАМУВАННЯ

Задача дробово-лінійного програмування в загальному випадку формулюється наступним чином:

$$Q(x) = \frac{\sum_{j=1}^{n} c_j x_j}{\sum_{j=1}^{n} d_j x_j} \Rightarrow Max, \sum_{j=1}^{n} a_{ij} x_j = b_i,\ i = \overline{1,m},\ j = \overline{1,n},\ \forall x_j \geq 0. \quad (4.21)$$

Окрім того. $\sum_{j=1}^{n} d_j x_j > 0,\ \sum_{j=1}^{n} d_j x_j \neq 0$ в області невід'ємних розв'язків системи рівнянь $\sum_{j=1}^{n} a_{ij} x_j = b_i$. Таким чином, в задачі дробово-лінійного програмування критерій якості на відміну від задачі лінійного програмування є часткою від ділення двох різних лінійних функцій від змінних задачі, а обмеження залишаються лінійними.

Задача дробово-лінійного програмування завжди може бути приведена до задачі лінійного програмування за допомогою наступних підстановок:

$$y_0 = \left(\sum_{j=1}^{n} d_j x_j\right)^{-1},\ y_j = y_0 \times x_j. \tag{4.22}$$

В результаті отримаємо

$$Q = \sum_{j=1}^{n} c_j y_j,\ \sum_{j=1}^{n} a_{ij} y_j - b_i y_0 = 0,\ \sum_{j=1}^{n} d_j y_j = 1,\ \forall y_j \geq 0,\ j = \overline{1,n}. \tag{4.23}$$

$$\sum_{j=1}^{n} d_j y_j = 1 = \sum_{j=1}^{n} d_j y_0 x_j = y_0 \sum_{j=1}^{n} d_j x_j = \frac{y_0}{y_0} = 1. \tag{4.24}$$

Таким чином, для розв'язання задачі дробово-лінійного програмування її необхідно спочатку привести до еквівалентної задачі лінійного програмування, розв'язати її і за допомогою обернених підстановок розрахувати оптимальні значення змінних первісної задачі.

ПРИКЛАДИ

Приклад 4.1. Задача ЛП в канонічному вигляді та рівняння для знаходження її базових розв'язків.

Нехай задана наступна задача лінійного програмування в канонічній формі:

$$Q = 3x_1 + 2x_2 + 1x_3 + 3x_4 \Rightarrow Max$$

$$2x_1 + 2x_2 + 3x_3 + 4x_4 = 10$$

$$1x_1 + 2x_2 + 0x_3 + 1x_4 = 4$$

$$\forall x_i \geq 0.$$

Необхідно представити її систему обмежень у векторній формі та виписати рівняння для знаходження координат вершин многогранника обмежень.

Розв'язання.

Випишемо систему обмежень у векторній формі

$$\begin{pmatrix}2\\1\end{pmatrix}x_1 + \begin{pmatrix}2\\2\end{pmatrix}x_2 + \begin{pmatrix}3\\0\end{pmatrix}x_3 + \begin{pmatrix}4\\1\end{pmatrix}x_4 = \begin{pmatrix}10\\4\end{pmatrix}$$

$$\forall x_i \geq 0.$$

Визначимо координати вершин многогранника умов (За теоремою 4.1 максимум досягається в одній з вершин многогранника обмежень) Оскільки кількість невідомих n=4, а кількість рівнянь m=2, то кількість вершин многогранника обмежень становитиме

$$C_n^m = \frac{n!}{m!(n-m)!} = \frac{4!}{2!(4-2)!} = \frac{1\times2\times3\times4}{1\times2\times1\times2} = 6.$$

Таким чином, необхідно розв'язати 6 систем рівнянь з двома невідомими, (m=2) надаючи іншим m-n=4-2=2 нульові значення. Системи рівнянь та їх розв'язки - координати вершин, подані нижче:

$$\begin{array}{llll}
x_1 = x_2 = 0 & x_1 = x_3 = 0 & x_1 = x_4 = 0 & x_2 = x_3 = 0 \\
3x_3 + 4x_4 = 10 & 2x_2 + 4x_4 = 10 & 2x_2 + 3x_3 = 10 & 2x_1 + 4x_4 = 10 \\
0x_3 + 1x_4 = 4 & ,2x_2 + 1x_4 = 4 & ,2x_2 + 0x_3 = 4 & ,1x_1 + 1x_4 = 4 \quad , \\
\forall x_i \geq 0 & \forall x_i \geq 0 & \forall x_i \geq 0 & \forall x_i \geq 0
\end{array}$$

$$\begin{array}{ll}
x_2 = x_4 = 0 & x_3 = x_4 = 0 \\
2x_1 + 3x_3 = 10 & 2x_1 + 2x_2 = 10 \\
1x_1 + 0x_3 = 4 & ,1x_1 + 2x_2 = 4 \\
\forall x_i \geq 0 & \forall x_i \geq 0
\end{array}$$

$$x^{(1)} = \begin{pmatrix}0\\0\\-2\\4\end{pmatrix}, x^{(2)} = \begin{pmatrix}0\\1\\0\\2\end{pmatrix}, x^{(3)} = \begin{pmatrix}0\\2\\2\\0\end{pmatrix}, x^{(4)} = \begin{pmatrix}3\\0\\0\\1\end{pmatrix}, x^{(5)} = \begin{pmatrix}4\\0\\2/3\\0\end{pmatrix}, x^{(6)} = \begin{pmatrix}6\\-1\\0\\0\end{pmatrix}.$$

Вершини $x^{(1)}$ та $x^{(6)}$ виключаємо з розгляду, оскільки для них порушуються умови на невід'ємність змінних. Для інших, підставляючи координати у формулу для критерію Q, обчислюємо його значення:

$$Q\left(x^{(2)}\right)=8,\ Q\left(x^{(3)}\right)=6,\ Q\left(x^{(4)}\right)=12,\ Q\left(x^{(5)}\right)=12\frac{2}{3}.$$

Таким чином, оптимальним розв'язком задачі є:

$$x_1=4,\ x_2=0,\ x_3=\frac{2}{3},\ x_4=0,\ Q^*=12\frac{2}{3}.$$

Зрозуміло, що задачу більшої розмірності розв'язати таким чином неможливо, оскільки кількість вершин симплекса зростає дуже швидко.

Приклад 4.2. Розв'язування задачі ЛП за допомогою СМ.

Необхідно за допомогою симплекс-методу розв'язати наступну задачу ЛП.

$$Q=3x_1+2x_2 \quad \Rightarrow Max$$

$$x_1+2x_2\le 6$$

$$2x_1+x_2\le 8$$

$$-x_1+x_2\le 1$$

$$x_2\le 2,\ x_1\ge 0,\ x_2\ge 0.$$

Розв'язання.

Оскільки всі нерівності мають вигляд „менше або рівне", і значення правих частин обмежень є невід'ємними, то при приведенні до канонічного вигляду отримаємо відразу ж і початковий базовий розв'язок. Нижче наведена також система обмежень у векторній формі

$$P_1x_1+P_2x_2+\ldots+P_nx_n=P_0,\ P_0\ge 0,\ \forall x_j\ge 0,$$

і, як видно, початкову базу утворюють вектори P_3,P_4,P_5,P_6.

$$Q=3x_1+2x_2+0x_3+0x_4+0x_5+0x_6\Rightarrow Max$$

$$1x_1+2x_2+1x_3+0x_4+0x_5+0x_6=6$$

$$2x_1+1x_2+0x_3+1x_4+0x_5+0x_6=8$$

$$-1x_1+1x_2+0x_3+0x_4+1x_5+0x_6=1$$

$$0x_1+1x_2+0x_3+0x_4+0x_5+1x_6=2,$$

$$\forall x_j\ge 0.$$

$$\begin{array}{ccccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 & P_0 \\ \begin{pmatrix}1\\2\\-1\\0\end{pmatrix}x_1+ & \begin{pmatrix}2\\1\\1\\1\end{pmatrix}x_2+ & \begin{pmatrix}1\\0\\0\\0\end{pmatrix}x_3+ & \begin{pmatrix}0\\1\\0\\0\end{pmatrix}x_4+ & \begin{pmatrix}0\\0\\1\\0\end{pmatrix}x_5+ & \begin{pmatrix}0\\0\\0\\1\end{pmatrix}x_6= & \begin{pmatrix}6\\8\\1\\2\end{pmatrix}\end{array}$$

Таким чином, небазовими змінними є x_1 та x_2, які внаслідок цього рівні нулю, і з розгляду системи рівнянь (обмежень) $x_3=6,\ x_4=8,\ x_5=1,$ $x_6=2.$

			c_1	c_2	c_3	c_4	c_5	c_6	
x_b	c_b	P_0	3	2	0	0	0	0	
			P_1	P_2	P_3	P_4	P_5	P_6	
x_3	0	6	1	2	1	0	0	0	6/1=6
x_4	0	8	2	1	0	1	0	0	8/2=4; 4<6
x_5	0	1	-1	1	0	0	1	0	$a_{ij^*}\le 0$
x_6	0	2	0	1	0	0	0	1	$a_{ij^*}\le 0$
Q		0	-3	-2	0	0	0	0	$\max\limits_{\Delta_j<0}\lvert\Delta_j\rvert=3$
x_3	0	2	0	3/2	1	-1/2	0	0	
x_1	3	4	1	1/2	0	1/2	0	0	
x_5	0	5	0	3/2	0	1/2	1	0	
x_6	0	2	0	1	0	0	0	1	
Q		12	0	-1/2	0	3/2	0	0	
x_2	2	4/3	0	1	2/3	-1/3	0	0	
x_1	3	10/3	1	0	-1/3	2/3	0	0	
x_5	0	3	0	0	-1	1	1	0	
x_6	0	2/3	0	0	-2/3	1/3	0	1	
Q		38/3	0	0	1/3	4/3	0	0	

Будуємо першу симплекс-таблицю згідно з алгоритмом симплекс-методу. Отриманий базовий розв'язок (розглянутий вище) не є оптимальним, так як 2 значення $\Delta_1=-3,\ \Delta_2=-2$ є від'ємними.

Далі обираємо ведучий стовпчик, що відповідає вектору P_j (найбільше за абсолютним значенням від'ємне значення дельта). Наступним обираємо ведучий рядок серед двох кандидатів, що відповідають базовим змінним x_3,x_4 (інші дві виключаємо з розгляду, оскільки для них не виконується умова додатності відповідних значень a_{ij} ведучого стовпчика. Ведучий рядок обираємо серед мінімальних значень частки (див. алгоритм симплекс-методу), що відповідає змінній x_4.

Таким чином, з числа базових змінних виводиться x_4, а вводиться замість неї змінна x_1. Здійснивши цю дію з використанням перетворення Гауса-Жордана, ми перейдемо до сусідньої вершини симплекса, в якій значення функції мети краще, ніж для першого базового розв'язку, і таким чином отримаємо наступну, другу симплекс-таблицю.

Отриманий базовий розв'язок (значення базових змінних зафіксовані в стовпчику P_0, а небазових рівні 0) є кращим, ніж попередній (значення критерію якості збільшилося з 0 до 12), однак не оптимальний (є від'ємне значення $\Delta_2 = -1/2$). Обираємо ведучий стовпчик, що відповідає вектору $\mathbf{P}_2$, ведучий рядок, що відповідає x_3 і переходимо до наступної, третьої таблиці. Оскільки в ній усі значення Δ є невід'ємними, то отриманий оптимальний розв'язок.

Таким чином, оптимальний розв'язок $x = (10/3; 4/3; 0; 0; 3; 2/3)$, і оптимальне значення критерію $Q = 38/3$.

Приклад 4.3. Розв'язування задачі ЛП за допомогою методу великих штрафів та двохетапного методу.

Необхідно знайти оптимальний розв'язок наступної задачі ЛП.

$$Q = 4x_1 + x_2 \Rightarrow Min$$
$$3x_1 + x_2 = 3$$
$$4x_1 + 3x_2 \geq 6$$
$$x_1 + 2x_2 \leq 4$$
$$x_1 \geq 0,\ x_2 \geq 0$$

Розв'язання. Спочатку приводимо задачу до канонічної форми.

$$Q = -4x_1 - x_2 \Rightarrow \quad Max$$
$$3x_1 + x_2 \qquad = 3$$
$$4x_1 + 3x_2 - x_3 \qquad = 6$$
$$x_1 + 2x_2 \qquad + x_4 = 4$$
$$\forall x_i \geq 0.$$

Вводимо штучні змінні R_1 та R_2, і отримуємо систему обмежень, яка використовуватиметься в обох методах:

$$3x_1 + x_2 + 0x_3 + 1R_1 + 0R_2 + 0x_4 = 3$$
$$4x_1 + 3x_2 - x_3 + 0R_1 + 1R_2 + 0x_4 = 6$$
$$x_1 + 2x_2 - 0x_3 + 0R_1 + 0R_2 + 1x_4 = 4$$

Для методу великих штрафів необхідно, щоб в оптимальному розв'язку штучні змінні були виведеними з бази, тому функція мети матиме наступний вигляд: $Q = -4x_1 - x_2 - MR_1 - MR_2 \Rightarrow Max$, а початкову базу утворюватимуть (R_1, R_2, x_4).

Для двохетапного методу функція мети першого етапу матиме вигляд: $-R_1 - R_2 \Rightarrow Max$.

Послідовність симплекс-таблиць для методу великих штрафів та двохетапного відповідно наведена нижче.

Метод великих штрафів

			c_1	c_2	c_3	c_4	c_5	c_6
x_b	c_b	P_0	-4	-1	0	0	-M	-M
			P_1	P_2	P_3	P_4	R_1	R_2
R_1	-M	3	3	1	0	0	1	0
R_2	-M	6	4	3	-1	0	0	1
x_4	0	4	1	2	0	1	0	0
Q		0	4	1	0	0	0	0
	*M	-9	-7	-4	1	0	0	0
x_1	-4	1	1	1/3	0	0		0
R_2	-M	2	0	5/3	-1	0		1
x_4	0	3	0	5/3	0	1		0
Q		4	0	-1/3	0	0		0
	*M	-2	0	-5/3	1	0		0
x_1	-4	3/5	1	0	1/5	0		
x_2	-1	6/5	0	1	-3/5	0		
x_4	0	1	0	0	1	1		
Q		-18/5	0	0	-1/5	0		
x_1	-4	2/5	1	0	0	-1/5		
x_2	-1	9/5	0	1	0	3/5		
x_3	0	1	0	0	1	1		
Q		-17/5	0	0	0	1/5		

У цій симплекс - таблиці для представлення значень функції мети та критеріїв оптимальності отриманого базового розв'язку використовується два рядки, оскільки ці значення будуть мати складову з M та без штрафного коефіцієнту. Наприклад, у першій симплекс-таблиці $\Delta_2 = 1 - 4M$ є від'ємним. Взагалі, при порівнянні таких значень слід пам'ятати про те, що $M >> 0$, і внаслідок цього в першу чергу розглядаються коефіцієнти при M. Тому, наприклад, $0.001M + 1 > 100000$. Коли змінна з коефіцієнтом –M в функції мети виводиться з бази, відповідний стовпчик в симплекс-таблиці може не обчислюватися, оскільки ця змінна більше до числа базових не потрапить. Саме тому в 2-й таблиці не обчислюються значення в стовпчику, що відповідає R_1, а в наступних таблицях не обчислюються два стовпчики, що відповідають штучним змінним. Хід процесу розв'язування просувається від неприпустимих розв'язків до припустимих, що досягається виведенням з бази штучних змінних.

Двохетапний метод.

У двохетапному методі починаємо з симплекс-таблиці, подібної на таблицю методу великих штрафів, але з іншою функцією мети – мінімізуємо суму штучних змінних – або що еквівалентне – максимізуємо суму штучних змінних зі знаком „мінус" перед кожною. Хід розв'язування не відрізняється від класичного симплекс-методу, і в результаті першого етапу отримуємо оптимальний розв'язок

$x_1 = 3/5,\ x_2 = 6/5,\ x_3 = 0,\ x_4 = 1,\ R_1 = 0,\ \ R_2 = 0.$

Оскільки значення штучних змінних нульові, то задача має розв'язок, і переходимо до другого етапу.

Симплекс-таблиці першого етапу

			c_1	c_2	c_3	c_4	c_5	c_6
x_b	c_b	P_0	0	0	0	0	-1	-1
			P_1	P_2	P_3	P_4	R_1	R_2
R_1	-1	3	3	1	0	0	1	0
R_2	-1	6	4	3	-1	0	0	1
x_4	0	4	1	2	0	1	0	0
Q		-9	-7	-4	1	0	0	0
x_1	0	1	1	1/3	0	0	1/3	0
R_2	-1	2	0	5/3	-1	0	-4/3	1
x_4	0	3	0	5/3	0	1	-1/3	0
Q		-2	0	-5/3	1	0	-1/3	0
x_1	0	3/5	1	0	1/5	0	3/5	-1/5
x_2	0	6/5	0	1	-3/5	0	-4/5	3/5
x_4	0	1	0	0	1	1	1	-1
Q		0	0	0	2/5	0	1	1
x_1	-4	3/5	1	0	1/5	0		
x_2	-1	6/5	0	1	-3/5	0		
x_4	0	1	0	0	1	1		
Q		-18/5	0	0	-1/5	0		

Симплекс-таблиці другого етапу

В результаті отриманий такий же розв'язок, як і при застосуванні методу великих штрафів.

			c_1	c_2	c_3	c_4
x_b	c_b	P_0	-4	-1	0	0
			P_1	P_2	P_3	P_4
x_1	-4	3/5	1	0	1/5	0
x_2	-1	6/5	0	1	-3/5	0
x_4	0	1	0	0	1	1
Q		-18/5	0	0	-1/5	0
x_1	-4	2/5	1	0	0	-1/5
x_2	-1	9/5	0	1	0	3/5
x_3	0	1	0	0	1	1
Q		-17/5	0	0	0	1/5

Приклад 4.4. Інтерпретація симплекс-таблиць.

Розглянемо останню СТ прикладу 4.2 цієї теми. Необхідно знайти оптимальний розв'язок, визначити статус ресурсів, цінність та межі зміни запасів ресурсів, межі зміни значень коефіцієнтів функції мети.

Розв'язання.

Оптимальний розв'язок. Значення отримуємо безпосередньо з СТ як компоненти стовпчика P_0: x_1*=10/3, x_2*=4/3, Q*=38/3.

			c_1	c_2	c_3	c_4	c_5	c_6
x_b	c_b	P_0	3	2	0	0	0	0
			P_1	P_2	P_3	P_4	P_5	P_6
x_2	2	4/3	0	1	2/3	-1/3	0	0
x_1	3	10/3	1	0	-1/3	2/3	0	0
x_5	0	3	0	0	-1	1	1	0
x_6	0	2/3	0	0	-2/3	1/3	0	1
Q		38/3	0	0	1/3	4/3	0	0

Статус ресурсів. Статус ресурсів також визначається безпосередньо з СТ за значеннями додаткових змінних $x_3 - x_6$:

Ресурс	Дод. змінна	Статус ресурсу
1	x_3=0	Дефіцитний
2	x_4=0	Дефіцитний
3	x_5=3	Недефіцитний
4	x_6=2/3	Недефіцитний

Цінність та межі зміни запасів ресурсів.

Цінність ресурсу розглядається як величина збільшення оптимального значення функції мети, що припадає на одиницю приросту об'єму цього ресурсу та визначається за значеннями коефіцієнтів при змінних початкової бази. З останнього рядка СТ записуємо:

$$Q = 12\frac{2}{3} - (\frac{1}{3}x_3 + \frac{4}{3}x_4 + 0x_5 + 0x_6)$$

Рівняння для ресурсу 1: $x_1 + 2x_2 + x_3 = 6$.

Збільшення значення x_3 еквівалентне зменшенню запасу ресурсу на 1 одиницю (тобто це є невикористаний ресурс). Найціннішим є ресурс 2, збільшення запасу якого (і відповідно зменшення значення x_4) на одиницю приводить до збільшення значення функції мети на 4/3.

Межі зміни значень запасів ресурсів визначаються з умови невід'ємності значень змінних задачі: $x = P_0 + P_j\delta_j \geq 0$. Розв'язуючи систему нерівностей для кожного ресурсу, що відповідає змінній x_j, отримуємо межі δ_j.

Визначимо межі зміни ресурсу, що відповідає x_3 (першого ресурсу).

$$\begin{pmatrix} x_2 \\ x_1 \\ x_5 \\ x_6 \end{pmatrix} = \begin{pmatrix} 4/3 \\ 10/3 \\ 3 \\ 2/3 \end{pmatrix} + \begin{pmatrix} 2/3 \\ -1/3 \\ -1 \\ -2/3 \end{pmatrix} \times \delta_3 \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \begin{matrix} (1) \\ (2) \\ (3) \\ (4) \end{matrix}$$

а). Якщо $\delta_3 \geq 0$, то відповідні нерівності виконуються, коли:

(1) – завжди ; (2) — за умови $\delta_3 \leq 10$; (3)— за умови $\delta_3 \leq 3$; (4) — за умови $\delta_3 \leq 1$.

Таким чином, $\delta_3 \leq 1$.

б). Якщо $\delta_3 < 0$, то відповідні нерівності виконуються, коли:

(1) – $\delta_3 \geq -2$, (2)-(4) – завжди. Таким чином, $\delta_3 \geq -2$.

Об'єднуючи отриманий результат, отримуємо $1 \geq \delta_3 \geq -2$.

Оскільки запас цього ресурсу становив 6, то припустимі межі зміни знаходитимуться в інтервалі [4; 7], що відповідає результату, отриманому шляхом графічного розв'язування. Аналогічно визначаються й межі зміни запасу іншого ресурсу.

Визначення меж зміни значень коефіцієнтів функції мети.

Цей крок відповідає на запитання: Яка можлива максимальна зміна коефіцієнтів питомого прибутку, при якій координати розв'язку залишаються незмінними? Розглядаючи зміну коефіцієнту c_1, запишемо $Q = (3 + \delta_1) \times x_1 + 2 \times x_2$. Відповідне співвідношення запишемо і для зміни c_2: $Q = 3 \times x_1 + (2 + \delta_2) \times x_2$.

Якщо б ми виконали всі розрахунки від першої до останньої симплекс-таблиці з відповідними значеннями δ, то отримали б наступний результат:

				c_1	c_2	c_3	c_4	c_5	c_6
	x_b	**c_b**	**P_0**	**3**	**2**	**0**	**0**	**0**	**0**
				P_1	**P_2**	**P_3**	**P_4**	**P_5**	**P_6**
δ_2	x_2	2	4/3	0	1	2/3	-1/3	0	0
δ_1	x_1	3	10/3	1	0	-1/3	2/3	0	0
	x_5	0	3	0	0	-1	1	1	0
	x_6	0	2/3	0	0	-2/3	1/3	0	1
	Q		38/3	0	0	1/3	4/3	0	0
δ_1	Q		38/3+10/3 δ_1	0	0	1/3-1/3 δ_1	4/3+2/3 δ_1	0	0
δ_2	Q		38/3+4/3 δ_2	0	0	1/3+2/3 δ_2	4/3-1/3 δ_2	0	0

Значення δ розраховуємо, приймаючи до уваги те, що значення небазових змінних не можуть бути від'ємними, тобто

$$1)\quad \begin{aligned} &1/3 - \delta_1/3 \geq 0,\ \delta_1 \leq 1, \\ &4/3 + 2/3\delta_1 \geq 0,\ \delta_1 \geq -2, \\ &-2 \leq \delta_1 \leq 1,\ 1 \leq c_1 \leq 4. \end{aligned}$$

$$2)\quad \begin{aligned} &1/3 + 2/3\delta_2 \geq 0,\ \delta_2 \geq -1/2, \\ &4/3 - 1/3\delta_2 \geq 0,\ \delta_2 \leq 4, \\ &-1/2 \leq \delta_2 \leq 4,\ 3/2 \leq c_2 \leq 6. \end{aligned}$$

Приклад 4.5. Змістовне формулювання критерію, що приводить до задачі дробово-лінійного програмування.

Нехай c_j – затрати на виробництво j-го виробу, x_j – кількість виробів j-го типу.

В цьому випадку , якщо необхідно мінімізувати середню собівартість (собівартість на один виріб), критерій якості матиме наступний вигляд:

$$Q = \frac{\sum_{j=1}^{n} c_j x_j}{\sum_{j=1}^{n} x_j} \Rightarrow Min.$$

РЕЗЮМЕ

4.1. Задача лінійного програмування завжди може бути приведена до канонічної форми, а тому при розгляді симплекс-методу вважається, що задача ЛП представлена в канонічній формі. Головна особливість задачі ЛП, на основі якої побудований метод її розв'язування, це те, що повний простір припустимих рішень такої задачі визначається за допомогою скінченої множини точок – базових припустимих розв'язків. Симплекс-метод ґрунтується на двох основних теоремах ЛП, що дозволяє обмежитися при пошуку оптимального розв'язку лише вершинами симплексу – многогранника обмежень – тобто пошук на безмежній множині розв'язків замінюється пошуком на скінченій (хоча й дуже великій) множині вершин многогранника обмежень.

4.2. Алгоритм симплекс-методу включає наступні кроки: 1.Знаходження початкового припустимого базового розв'язку; 2.Перевірка на зупинку алгоритму: чи знайдений розв'язок є оптимальним? Якщо так, то стоп. 3.Пошук наступної вершини многогранника, до якої необхідно перейти: визначення змінної, що вводиться до бази (ведучого стовпчика, ведучої колонки); визначення змінної, що виводиться з бази (ведучого рядка). 4.Перехід до нової вершини з застосуванням методу Жордана-Гауса.

4.3. Метод великих штрафів та двохетапний метод служать для можливості розв'язування задачі ЛП в тому випадку, коли початкового базового розв'язку з умови задачі отримати безпосередньо неможливо. В цьому випадку початковий базовий розв'язок будується шляхом введення штучних змінних у тій кількості, щоб отримати початковий базовий розв'язок. Існує два шляхи позбавлення від штучних змінних в остаточному оптимальному розв'язкові – це або введення штрафів за відхилення значень штучних змінних від нуля та включення відповідних складових до функції мети (метод великих штрафів), або розв'язання на першому етапі допоміжної задачі, що дозволить отримати початковий базовий розв'язок без штучних змінних з подальшим розв'язанням класичним шляхом (двохетапний метод).

4.4. Особливі випадки виникають внаслідок певних аномалій в умові задачі ЛП і є наступними: виродженість – наявність надлишкових обмежень; безмежна множина оптимальних розв'язків; необмежені розв'язки; необмежена область розв'язків; відсутність припустимих розв'язків. За допомогою симплекс-таблиць безпосередньо або за допомогою нескладних обчислень можна отримати наступну інформацію: оптимальний розв'язок; статус ресурсів; цінність кожного з ресурсів; чутливість оптимального розв'язку до зміни запасів ресурсів, варіацій значень коефіцієнтів функції мети та інтенсивності споживання ресурсів.

4.5. В задачі дробово-лінійного програмування критерій якості на відміну від задачі лінійного програмування є часткою від ділення двох різних лінійних функцій від змінних задачі, а обмеження залишаються лінійними. Задача дробово-лінійного програмування завжди може бути приведена до задачі лінійного програмування за допомогою підстановок.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 4.1.

Розв'язати наступну задачу ЛП за допомогою симплекс-методу. Здійснити економічну інтерпретацію симплекс-таблиць – знайти оптимальне значення, визначити статус, цінність та доцільні межі зміни ресурсів, межі зміни значень коефіцієнтів функції мети.

$$Q = 9x_1 + 10x_2 + 16x_3 \Rightarrow Max$$
$$18x_1 + 15x_2 + 12x_3 \le 360$$
$$6x_1 + 4x_2 + 8x_3 \le 192$$
$$5x_1 + 3x_2 + 3x_3 \le 180$$
$$\forall x_i \ge 0.$$

Завдання 4.2.

Розв'язати наступну задачу ЛП за допомогою методу великих штрафів та двохетапного методу.

$$Q = -x_1 - 2x_2 - 3x_3 + x_4 \Rightarrow Max$$
$$x_1 + 2x_2 + 3x_3 = 15$$
$$2x_1 + x_2 + 5x_3 = 20$$
$$x_1 + 2x_2 + x_3 + x_4 = 10$$
$$\forall x_j \ge 0.$$

Завдання 4.3.

Розв'язати наступну задачу дробово-лінійного програмування.

$$Q = \frac{2x_1 + 3x_2}{x_1 + x_2}$$

$$2x_1 + 8x_2 \le 26$$

$$x_1 + x_2 \ge 4 \qquad x_1, x_2 \ge 0\ .$$

$$12x_1 + 3x_2 \le 39$$

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Наведіть формулювання задачі лінійного програмування в матричній та векторній формі.
2. Який висновок дозволяє зробити теорема 4.1?
3. Що називається базовим розв'язком задачі лінійного програмування та яким чином можна знайти всі базові розв'язки для задачі ЛП?
4. На якій ідеї грунтується симплекс-метод?
5. Опишіть укрупнену схему алгоритму симплекс-методу та дайте її змістовну інтерпретацію.
6. Яким чином визначаються ведучий стовпчик та ведучий рядок симплеск-таблиці і навіщо це робиться?
7. Доведіть необхідність та достатність умови досягнення оптимального розв'язку в задачі ЛП.
8. З якою метою використовуються методи пошуку початкового базового розв'язку?
9. В чому полягає метод великих штрафів та які його недоліки?
10. Розкрийте суть двохетапного методу.
11. Як відображаються особливі випадки симплекс-методу в симплекс-таблицях?
12. Яку додаткову інформацію і яким чином можна отримати з симплекс-таблиць?
13. Яким чином задача дробово-лінійного програмування приводиться до задачі лінійного програмування?
14. Яким чином змістовно інтерпретується задача дробово-лінійного програмування?

ТЕМА 5. ДВОЇСТІСТЬ В ЛІНІЙНОМУ ПРОГРАМУВАННІ. МОДИФІКОВАНИЙ СИМПЛЕКС-МЕТОД. БЛОЧНІ ЗАДАЧІ ЛІНІЙНОГО ПРОГРАМУВАННЯ

Поняття двоїстості є одним з найважливіших у лінійному програмуванні. Кількість змінних двоїстої задачі рівна кількості обмежень прямої, а кількість обмежень – кількості змінних прямої задачі. Значення функції мети прямої задачі для довільного припустимого розв'язку не перевищує значення функції мети двоїстої задачі. Якщо одна з пари двоїстих задач має розв'язок, то й інша має розв'язок, і оптимальні значення функцій мети рівні між собою. Теорема про доповнюючу нежорсткість дозволяє отримати розв'язки двоїстої задачі, знаючи оптимальний розв'язок прямої задачі, що дозволяє не розв'язувати двоїсту задачу, а її результати використати як оцінки ступеня наближення до оптимуму. Економічна інтерпретація задачі лінійного програмування дозволяє оцінити ряд важливих економічних ресурсів, звернути увагу на те, які з них є дефіцитними і чому, а також в яких межах можуть коливатися значення коефіцієнтів функції мети, щоб положення оптимуму не змінилося. В ряді випадків використовується двоїстий симплекс-метод, особливо як крок в методах, що побудовані на ідеї відсічень (наприклад, метод Гоморі для розв'язування мішаних задач лінійного програмування). Ще одним з широко застосовуваних в практиці комп'ютерних розрахунків є модифікований симплекс-метод, що суттєво пришвидшує отримання оптимального рішення. Для розв'язування блочних задач лінійного програмування використовується ряд спеціальних методів, як, наприклад, метод Данціга – Вулфа та інші.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати: ⇨поняття двоїстості та його практичне економічне значення; ⇨зв'язок між прямою та двоїстою задачею та його теоретичне обґрунтування; ⇨економічну інтерпретацію прямих задач; ⇨застосування принципу декомпозиції до побудови алгоритмів розв'язання задач лінійного програмування значної розмірності;

вміти: ⇨побудувати умову двоїстої – до прямої задачі лінійного програмування та знайти оптимальний розв'язок двоїстої задачі за розв'язком прямої; ⇨розв'язувати задачі лінійного програмування за допомогою двоїстого та модифікованого симплекс-методу;⇨економічно інтерпретувати розв'язок задачі лінійного програмування; ⇨розв'язувати задачі з використанням методу Данціга-Вулфа.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

☑двоїста задача ЛП
☑доповнююча нежорсткість
☑економічна інтерпретація прямої та двоїстої задач ЛП
☑опукла комбінація екстремальних точок
☑двоїстий симплекс-метод
☑блочна задача ЛП
☑модифікований симплекс-метод
☑координуюча задача
☑розрідженість матриці обмежень
☑розмірність задачі ЛП
☑метод Данціга-Вулфа
☑опукла комбінація екстремальних точок
☑декомпозиція задачі ЛП

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

5.1. Пряма та двоїста задачі лінійного програмування.

5.2. Зв'язок між розв'язками прямої та двоїстої задач. Отримання оптимального розв'язку двоїстої задачі за допомогою симплекс-методу.

5.3. Економічна інтерпретація задач лінійного програмування. Двоїстий симплекс-метод.

5.4. Модифікований симплекс-метод.

5.5. Блочні задачі лінійного програмування. Метод Данціга - Вулфа.

5.1. ПРЯМА ТА ДВОЇСТА ЗАДАЧІ ЛІНІЙНОГО ПРОГРАМУВАННЯ

Поняття двоїстості є одним із найважливіших у лінійному програмуванні. Оскільки задачу ЛП завжди можна привести до канонічної форми, то пряму задачу ми розглядатимемо завжди в канонічній формі. Таким чином, пряма задача ЛП має вигляд:

$$Q(x)=\sum_{j=1}^{n} c_j x_j \Rightarrow Max\ ,\ \sum_{j=1}^{n} a_{ij} x_j = b_i,\ i=\overline{1,m},\ j=\overline{1,n},\ x_j \ge 0\ . \quad (5.1)$$

Двоїстою до цієї прямої задачі ЛП буде наступна:

$$G(y)=\sum_{i=1}^{m} y_i b_i \Rightarrow Min\ ,\ \sum_{i=1}^{m} y_i a_{ij} \ge c_j,\ i=\overline{1,m},\ j=\overline{1,n},\ y_i <> 0 \quad (5.2)$$

Таким чином, змінні двоїстої задачі необмежені в знаку коефіцієнтами критерію якості, котрий потрібно мінімізувати, є вектор правих частин обмежень прямої задачі, а вектор правих частин обмежень двоїстої задачі складають коефіцієнти функції мети прямої задачі. Кількість змінних двоїстої задачі рівна кількості обмежень прямої, а кількість обмежень - кількості змінних прямої задачі.

Відповідно в матричній формі пряма та двоїста задачі відображаються наступним чином (ми вважаємо, що всі вектори є векторами-стовпчиками):

$c^T \times x \Rightarrow Max,\ A \times x = b,\ x \ge 0$ – пряма задача, (5.3)

$y^T \times b \Rightarrow Min,\ y^T \times A \ge c^T,\ y <> 0,$ (5.4)

або

$b^T \times y \Rightarrow Min,\ A^T \times y \ge c$ - двоїста.

Зв'язок між прямою та двоїстою задачею наочно відображений за допомогою наступної таблиці:

Змінні прямої задачі

x_1	x_2	…	x_j	…	x_n		
c_1	c_2	…	c_j	…	c_n		
a_{11}	a_{12}	…	a_{1j}	…	a_{1n}	b_1	y_1
a_{21}	a_{22}	…	a_{2j}	…	a_{2n}	b_2	y_2
…	…	…	…	…	…	…	…
a_{m1}	a_{m2}	…	a_{mj}	…	a_{mn}	b_m	y_m

Змінні двоїстої задачі

Рис. 5.1. Взаємний зв'язок між прямою та двоїстою задачами

Двоїста задача до двоїстої буде первісною прямою задачею, тобто, якщо A – пряма задача, D{A} – двоїста до A, то A=D{D{A}}. Покажемо, що це насправді так:

$$\begin{matrix} b^T \times y \to Min, \\ A^T \times y \ge c \end{matrix} \Rightarrow \begin{matrix} -b^T \times y \to Max. \\ -A^T \times y \le -c \end{matrix} \Rightarrow \begin{matrix} -b^T \times (y'-y'') + 0z \to Max, \\ -A^T \times (y'-y'') + Ez = -c \\ y = y'-y'',\ y' \ge 0,\ y'' \ge 0 \end{matrix} \quad (5.5)$$

Тепер переходимо до двоїстої задачі:

$$\begin{array}{l} -c^T \times x \to Min, \\ -A\times x \ge -b,\ Ax \ge b,\ E^T x \ge 0 \end{array} \Rightarrow \begin{array}{l} c^T \times x \to Max, \ \blacksquare \\ A\times x = b,\ x \ge 0. \end{array} \quad (5.6)$$

5.2. ЗВ'ЯЗОК МІЖ РОЗВ'ЯЗКАМИ ПРЯМОЇ ТА ДВОЇСТОЇ ЗАДАЧ. ОТРИМАННЯ ОПТИМАЛЬНОГО РОЗВ'ЯЗКУ ДВОЇСТОЇ ЗАДАЧІ ЗА ДОПОМОГОЮ СИМПЛЕКС-МЕТОДУ

Зв'язок між розв'язками прямої та двоїстої задач встановлюється за допомогою двох теорем двоїстості. Для їх доведення використаємо наступну лему.

Лема 5.1.

Значення функції мети $Q(x)$ прямої задачі для довільного припустимого розв'язку $x \in X$ не перевищує значення функції мети $G(y)$ двоїстої задачі для довільного її припустимого розв'язку $y \in Y$, тобто $c^T x \le b^T y$. Якщо для деяких $x^* \in X$ та $y^* \in Y$ справедлива рівність $c^T x^* = b^T y^*$, то x^* та y^* – оптимальні розв'язки.

Доведення.

Нехай x та y – припустимі розв'язки прямої та двоїстої задач, тобто $A\times x = b,\ y^T A \ge c^T$. Помножимо перший вираз на y^T зліва, а другий – на x справа: $y^T Ax = y^T b$, $y^T Ax \ge c^T x$. Порівнюючи ці вирази, отримаємо $y^T b = b^T y \ge c^T x$.

Нехай для деяких припустимих x, y $y^T b = c^T x$. Припустимо, що - x не оптимальний розв'язок. У такому випадку для оптимального розв'язку x^* справедливе співвідношення $c^T x^* > c^T x = y^T b$, що суперечить вже доведеному вище твердженню. Припустимо, що y – не оптимальний розв'язок. У такому випадку для оптимального розв'язку y^* справедливе співвідношення $y^{*T} b < y^T b = c^T x$, що також суперечить вже доведеному вище твердженню ■

Доведемо тепер першу теорему двоїстості.

Теорема 5.1. (1-а теорема двоїстості).

Якщо одна з пари двоїстих задач має розв'язок, то й інша має розв'язок, і оптимальні значення функцій мети рівні між собою – $\underset{x\in X}{Max}\ Q(x) = Q^* = \underset{y\in Y}{Min}\ G(y) = G^*$. Якщо функція мети для однієї з задач не обмежена , двоїста до неї задача взагалі не має припустимих розв'язків.

Доведення.

Запишемо задачу ЛП у перетвореному вигляді:

$$Q = c_B^T B^{-1} b - \left(c_B^T B^{-1} N - c_N^T\right) x_N \Rightarrow Max, \quad (5.7)$$

$$x_B = B^{-1} b - B^{-1} N x_N,\ x_B \ge 0, x_N \ge 0, \text{ де } \Delta = c_B^T B^{-1} N - c_N^T.$$

На останній ітерації – в останній симплекс-таблиці розв'язок є оптимальним, $\forall \Delta_j \ge 0$, тобто $c_B^T B^{-1} N \ge c_N^T$. Введемо позначення $y^T = c_B^T B^{-1}$. Тоді $y^T N \ge c_N^T$, $y^T B = c_B^T B^{-1} B = c_B^T$, і об'єднуючи ці обидва вирази $y^T \times \left[B \middle| N\right] \ge \left[c_B^T \middle| c_N^T\right]$, отримаємо $y^T \times A \ge c^T$.

Таким чином, y – припустимий розв'язок двоїстої задачі. Доведемо, що y – оптимальний розв'язок двоїстої задачі: $y^T b = c_B^T B^{-1} b = c_B^T x_B$. Таким чином, значення функції мети для двоїстої задачі для її припустимого розв'язку y дорівнює значенню функції мети прямої задачі для її припустимого розв'язку x. Згідно з лемою, це означає, що y є оптимальним розв'язком двоїстої задачі.

Доведемо другу частину теореми.

Нехай відомо, що функція мети прямої задачі не обмежена згори на множині припустимих розв'язків. Припустимо, що двоїста задача при цьому має припустимі розв'язки і y – один із них. У такому випадку внаслідок необмеженості функції мети прямої задачі знайдеться такий її розв'язок, для якого $c^T x > y^T b$, що суперечить лемі ■

Примітка.

Твердження, обернене до 2-ї частини теореми 5.1, взагалі кажучи, є недійсним, тобто, якщо одна з пари двоїстих задач має порожню множину припустимих розв'язків, то інша не обов'язково повинна мати необмежену функцію мети. Обидві задачі можуть бути неприпустимими, наприклад:

$$\begin{array}{llll} Q = 2x_1 - x_2 \Rightarrow Max, & & G = y_1 + y_2 \Rightarrow Min, & \\ x_1 - x_2 \ge 1 & \Rightarrow y_1 & y_1 - y_2 \ge -1 & \\ -x_1 + x_2 \ge 1 & \Rightarrow y_2 & -y_1 + y_2 \ge 2 & (5.8) \\ x_1 \ge 0,\ x_2 \ge 0 & & y_1 \le 0, y_2 \le 0 & \end{array}$$

Область припустимих розв'язків прямої задачі $X = \varnothing$.

Область припустимих розв'язків двоїстої задачі $Y = \varnothing$.

Теорема 5.2. (2-а теорема двоїстості, теорема про доповнюючу нежорсткість).

Нехай $x^* = (x_1^*, \dots, x_n^*)$, $y^* = (y_1^*, \dots, y_m^*)$ – припустимі розв'язки прямої та двоїстої задач відповідно. Якщо для кожного j, для якого відповідне j-те обмеження двоїстої задачі перетворюється при $y = y^*$ в строгу нерівність, $x_j^* = 0$, то $x^* = (x_1^*, \dots, x_n^*)$ та $y^* = (y_1^*, \dots, y_m^*)$ є оптимальними розв'язками прямої та двоїстої задач.

Ця теорема може бути також сформульована у наступному еквівалентному вигляді. Розв'язки $x^* = (x_1^*, \dots, x_n^*)$, $y^* = (y_1^*, \dots, y_m^*)$ є оптимальними тоді і лише тоді, якщо виконуються співвідношення:

$$x_j^* \times (\sum_{i=1}^{m} a_{ij} y_i^* - c_j) = 0, \quad j = \overline{1, n}, \tag{5.9}$$

$$y_i^* \times (\sum_{j=1}^{n} a_{ij} x_j^* - b_i) = 0, \quad i = \overline{1, m}. \tag{5.10}$$

Ця теорема дає можливість за оптимальним розв'язком однієї з задач знайти оптимальний розв'язок двоїстої до неї.

Доведення.

Нехай для кожного j, для якого $\sum_{i=1}^{m} y_i a_{ij} > c_j$, $i = \overline{1, m}$, $j = \overline{1, n}$, $y_i <> 0$, маємо $x_j^* = 0$. Тоді $c^T x = \sum_{j=1}^{n} x_j^* \times \left(\sum_{i=1}^{m} y_i^* \times a_{ij} \right)$. Справді, ті складові, для яких $\sum_{i=1}^{m} y_i^* \times a_{ij} > c_j$, фактично рівні нулю, оскільки для них $x_j^* = 0$. Залишаються до розгляду лише ті складові, для яких $\sum_{i=1}^{m} y_i^* \times a_{ij} = c_j$, що доводить справедливість рівності. Після перегрупування суми отримаємо

$c^T x = \sum_{i=1}^{m} y_i^* \times \left(\sum_{j=1}^{n} x_j^* \times a_{ij} \right)$. Оскільки x^* - припустимий розв'язок прямої задачі, то суми в дужках рівні відповідним значенням b_i, тобто $c^T x^* = \sum_{i=1}^{m} y_i^* \times b_i = y^T b$ і згідно до леми ці розв'язки є оптимальними.

Нехай тепер навпаки, відомо, що $x^* = (x_1^*, \dots, x_n^*)$, $y^* = (y_1^*, \dots, y_m^*)$ — оптимальні розв'язки прямої та двоїстої задач. Тоді за першою теоремою двоїстості $c^T x^* = y^{*T} b$. Оскільки x^* – припустимий розв'язок задачі, то $b_i = \sum_{j=1}^{n} a_{ij} x_j^*$, $\forall i = \overline{1, m}$. Підставляючи ці вирази в рівність та перегрупувуючи складові в лівій частині, отримаємо:

$$\sum_{j=1}^{n} \left(\sum_{i=1}^{m} y_i^* \times a_{ij} \right) x_j^* = c^T x^*.$$

Якщо хоча б для одного j, для якого $\sum_{i=1}^{m} y_i^* \times a_{ij} > c_j$, було б $x_j^* \neq 0$, то попереднє співвідношення не виконувалося б. Тобто дійсно, для кожного j, для якого б виконувалася попередня строга нерівність, отримуємо $x_j^* = 0$, що й потрібно було довести ■

Розглянемо можливості отримання оптимального розв'язку двоїстої задачі з оптимального розв'язку (за допомогою останньої симплекс-таблиці) прямої задачі, використовуючи отримані внаслідок доведення теорем результати.

Для того, щоб отримати оптимальний розв'язок двоїстої задачі, маючи оптимальний розв'язок прямої задачі, зовсім не обов'язково розв'язувати двоїсту задачу. Використовуючи другу теорему двоїстості, це можна зробити безпосередньо з останньої симплекс-таблиці розв'язування прямої задачі.

Відомо, що кожне обмеження прямої задачі відповідає змінній двоїстої задачі, кожна змінна прямої задачі відповідає обмеженню двоїстої. Запишемо умову двоїстої задачі та останню симплекс-таблицю для прямої задачі, в якій наявний оптимальний розв'язок, а також змінні першого базового розв'язку прямої задачі.

Для визначення оптимальних значень змінних двоїстої задачі складемо систему рівнянь, використовуючи теорему про доповнюючу нежорсткість. Таким чином, отримаємо наступне співвідношення:

Коефіцієнт Δ при початковій базовій змінній в рядку прямої задачі	**=**	**Різниця між лівою та правою частинами обмеження двоїстої задачі, яке асоційоване з цією початковою змінною**

Це ж співвідношення в звичній формі:

$$\Delta_j = \sum_{i=1}^{m} a_{ij} y_i - c_j, \quad \sum_{i=1}^{m} a_{ij} y_i \geq c_j. \tag{5.11}$$

Окрім того, наслідком з 1-ї теореми двоїстості є те, що для будь-якої пари припустимих розв'язків прямої та двоїстої задачі на будь-якому етапі розв'язування виконується наступне співвідношення:

Значення функції мети в задачі максимізації	$\le$	**Значення функції мети в задачі мінімізації**

Це співвідношення має важливе практичне значення: розв'язуючи пряму та отримуючи на кожному кроці розв'язок двоїстої задачі, можна в будь-який момент припинити хід розв'язування за умови досягнення необхідної точності (інтервал значень функції мети).

5.3. ЕКОНОМІЧНА ІНТЕРПРЕТАЦІЯ ЗАДАЧ ЛІНІЙНОГО ПРОГРАМУВАННЯ. ДВОЇСТИЙ СИМПЛЕКС-МЕТОД

Розглянемо пряму та двоїсту задачі ЛП:

Пряма задача:

$$Q(x)=\sum_{j=1}^{n} c_j x_j \Rightarrow Max \ , \ \sum_{j=1}^{n} a_{ij} x_j = b_i, \ i=\overline{1,m}, \ j=\overline{1,n}, \ x_j \ge 0 \ . \quad (5.13)$$

Двоїста задача:

$$G(y)=\sum_{i=1}^{m} y_i b_i \Rightarrow Min \ , \ \sum_{i=1}^{m} y_i a_{ij} = c_j, \ i=\overline{1,m}, \ j=\overline{1,n}, \ y_i <> 0 \ . \quad (5.14)$$

Пряму задачу розглядатимемо, як задачу виробництва певної продукції за умови наявності певних обмежених ресурсів, що можуть витрачатися на їх виробництво. Проблема полягає в тому, щоб отримати максимальний прибуток від реалізації продукції, тобто необхідно визначити кількість продуктів, які забезпечуватимуть максимальне значення функції мети.

В цій інтерпретації:

$i=\overline{1,m}$ – індекс видів ресурсів, що застосовуються для виробництва всіх продуктів;

$j=\overline{1,n}$ – індекс видів продуктів, що можуть продукуватися;

x_j – кількість продукції j – го типу, яка випускається;

c_j – прибуток від реалізації одиниці j – го продукту;

b_i – величина запасу i – го ресурсу;

a_{ij} – кількість i – го ресурсу, що витрачається на виробництво одиниці j – го продукту;

$Q(x)=\sum_{j=1}^{n} c_j x_j$ – прибуток від реалізації продукції.

Для подальшого аналізу використаємо результати теорем двоїстості, власне співвідношення $\Delta_j = \sum_{i=1}^{m} a_{ij} y_i - c_j$, тобто змістовно,

Коефіцієнт Δ при початковій базовій змінній в рядку прямої задачі	=	**Різниця між лівою та правою частинами обмеження двоїстої задачі, яке асоційоване з цією початковою змінною**

а також рівність значень функцій мети для прямої та двоїстої задачі в оптимальній точці.

Використаємо співвідношення $\underset{x \in X}{Max}\ Q(x)=Q^*=\underset{y \in Y}{Min}\ G(y)=G^*$ для змістовної інтерпретації змінних двоїстої задачі. Перепишемо це співвідношення у вигляді $\sum_{j=1}^{n} c_j x_j^* = \sum_{i=1}^{m} b_i y_i^*$ і визначимо розмірність у:

$$\frac{[\text{грошова одиниця}]}{[\text{кількість}]} \times [\text{кількість}] = [\text{одиниця ресурсу}] \times [y]. \quad (5.15)$$

Звідси розмірність змінних двоїстої задачі:

$[y]=\dfrac{[\text{грошова одиниця}]}{[\text{одиниця ресурсу}]}$, тобто y є цінністю ресурсу (його тіньовою ціною). Для неоптимального розв'язку $Q(x)< G(y)$, тобто (прибуток)<=(цінність ресурсу), іншими словами – цінні ресурси не використані повністю, і можливе збільшення прибутку за рахунок використання відповідних видів виробничої діяльності (випуску продукції певних видів).

Обмеження двоїстої задачі можна записати у вигляді: $\Delta_j = \sum_{i=1}^{m} a_{ij} y_i - c_j$,

де змістовно $\sum_{i=1}^{m} a_{ij} y_i$ – сумарна оцінка ресурсів, що витрачаються на виробництво одиниці j-го продукту, і таким чином, і

$$\frac{[\text{прибуток або цінність}]}{[\text{од. продукції}]} = \frac{[\text{цінність}]}{[\text{од. продукції}]} - \frac{[\text{прибуток}]}{[\text{од. продукції}]} \quad (5.16)$$

Тобто, якщо $\Delta_j < 0$, то відповідний вид виробничої діяльності повинен бути введений в базу (з $x_j > 0$ випливає, що $\sum_{i=1}^{m} a_{ij} y_i - c_j < 0$).

На співвідношеннях двоїстості ґрунтується різновид симплекс-методу – двоїстий симплекс-метод, який може використовуватись як самостійний метод, хоча знаходить широке застосування як крок методу Ґоморі внаслідок специфічного способу формування додаткових обмежень під час роботи алгоритму методу Гоморі.

Двоїстий симплекс-метод.

При розв'язуванні прямої задачі на довільній ітерації значення коефіцієнта Δ_j в Q -рядку симплекс-таблиці дорівнює різниці між лівою та правою частинами відповідного обмеження двоїстої задачі:

$$\Delta_j = z_j - c_j = \sum_{i=1}^{m} a_{ij} y_i - c_j,\ z_j = (c, P_j). \tag{5.17}$$

Якщо $\Delta_j < 0$, то $\sum_{i=1}^{m} a_{ij} y_i < c_j$, іншими словами, коли розв'язок прямої задачі неоптимальний, то розв'язок двоїстої неприпустимий. З іншого боку, якщо $z_j - c_j \ge 0$, то $\sum_{i=1}^{m} a_{ij} y_i - c_j \ge 0$, тобто оптимальному розв'язкові прямої задачі відповідає припустимий розв'язок двоїстої.

В двоїстому симплекс-методі (СМ) спочатку отримуємо розв'язок "кращий, ніж оптимальний" (надоптимальний), але неприпустимий. Двоїстий СМ забезпечує виконання умови оптимальності та наближення кожного наступного біжучого розв'язку до області припустимих розв'язків. В момент, коли отриманий розв'язок виявляється припустимим, процес розв'язування закінчується, так як останній розв'язок є оптимальним і припустимим.

Всі значення $\Delta_j \ge 0$, і це співвідношення зберігається на кожній ітерації двоїстого СМ. Деякі елементи стовпчика P_0 є від'ємними. Перехід від одного біжучого розв'язку до іншого здійснюється до моменту, поки з P_0 не будуть виключені від'ємні елементи.

Алгоритм двоїстого СМ включає наступні кроки:

Крок 1. Знаходження початкового надоптимального розв'язку.

Крок 2. Перевірка біжучого надоптимального розв'язку на припустимість (якщо в стовпчику P_0 відсутні від'ємні значення, то знайдений оптимальний розв'язок. **Стоп**.).

Крок 3. Обираємо найбільше за абсолютною величиною від'ємне значення в стовпчикові P_0 і визначаємо його індекс – $k = \arg\max(-b_i)$, $b_i < 0$ – цим шляхом визначаємо змінну x_k, яку виключаємо з бази. Визначаємо індекс змінної, що включається до бази, $r = \arg\min_j \left(\dfrac{-\Delta_j}{a_{kj}} \right)$, $a_{kj} < 0$. Таким чином, ведучий елемент є a_{kr}.

Крок 4. Переходимо до наступного надоптимального розв'язку так само, як і в звичайному СМ, використовуючи схему Гауса-Жордана. Перехід до кроку 2.

5.4. МОДИФІКОВАНИЙ СИМПЛЕКС-МЕТОД

У варіантах симплекс-методу, що розглядалися раніше, при реалізації послідовних ітерацій – переході від попередньої до наступної симплекс-таблиці – використовувався метод Гауса-Жордана, при застосуванні якого для автоматизації розрахунків необхідний великий об'єм пам'яті. З метою усунення такого недоліку застосовується модифікований симплекс-метод, що являє собою різновид симплекс-методу, який, крім того, дозволяє в багатьох випадках зменшити ще й загальну кількість арифметичних операцій. Суттєвою в цьому методі є реалізація процедури обчислення оберненої матриці B^{-1}.

Основні кроки модифікованого симплекс-методу не відрізняються від класичного (різниця полягає в деталях реалізації) і утворюють наступну послідовність:

Крок 1.Знаходження початкового базового розв'язку.

Крок 2.Розрахунок матриці B^{-1}, що являє собою обернену до матриці B, яка складається з векторів початкової бази.

Крок 3.Розрахунок компонент вектору $\Omega = c_B^T B^{-1}$.

Крок 4. Обчислення значень критеріїв оптимальності досягнутого біжучого розв'язку задачі ЛП $\Delta_j = \Omega P_j - c_j$. Якщо $\forall \Delta_j \ge 0$, то знайдений біжучий розв'язок є оптимальним – робота алгоритму зупиняється. В іншому випадку обирається Δ_k – найбільше за абсолютним значенням від'ємне число.

Крок 5. Обчислення компонент вектору P_k у вихідній базі. Якщо серед цих компонент немає додатніх, то функція мети не обмежена на множині припустимих розв'язків – **стоп**. Якщо ж серед компонент вектору є додатні, то перехід до наступного базового розв'язку.

Крок 6. Знаходження ведучого рядка за відомими правилами симплекс-методу та обчислення компонент нового базового розв'язку і матриці B^{-1}, що є тепер матрицею, оберненою до матриці компонент векторів нового базового розв'язку.

Крок 7.Перевірка нового розв'язку на оптимальність. Якщо розв'язок неоптимальний, то перехід до кроку 3.

Розрахунки проводяться виходячи з того, що для задачі знайдений початковий базовий розв'язок, тобто відома матриця B, до якої можна знайти обернену B^{-1}. Подальші обчислення реалізуються у вигляді таблиць, при цьому для переходу від однієї основної таблиці до наступної використовується допоміжна таблиця, що має наступний вигляд.

x_b	c_b	P_0	c_1	c_2	...	c_n	$\Omega^{(1)}$	...	$\Omega^{(k)}$
			P_1	P_2	...	P_n			
x_1	c_1	b_1	x_{11}	x_{12}	...	x_{1n}	$\omega_1^{(1)}$	...	$\omega_1^{(k)}$
x_2	c_2	b_2	x_{21}	x_{22}	...	x_{2n}	$\omega_2^{(1)}$	...	$\omega_2^{(k)}$
.	.	.	.	.	...	.	...	...	...
x_i	c_i	b_i	x_{i1}	x_{i2}	...	x_{in}	$\omega_i^{(1)}$	...	$\omega_i^{(k)}$
.	.	.	.	.	...	.	...	...	...
x_m	c_m	b_m	x_{m1}	x_{m2}	...	x_{mn}	$\omega_m^{(1)}$	...	$\omega_m^{(k)}$
$\Delta^{(1)}$			$\Delta_1^{(1)}$	$\Delta_2^{(1)}$	...	$\Delta_n^{(1)}$			
$\Delta^{(2)}$			$\Delta_1^{(2)}$	$\Delta_2^{(2)}$	...	$\Delta_n^{(2)}$			
...			...	...	...	...			
$\Delta^{(k)}$			$\Delta_1^{(k)}$	$\Delta_2^{(k)}$	...	$\Delta_n^{(k)}$			

Таб. 5.1. Допоміжна таблиця модифікованого симплекс-методу

Допоміжна таблиця відрізняється від звичайної тим, що в ній наявні додаткові стовпці та рядки, в яких зберігаються вектори Δ та Ω, значення компонент яких обчислюються при розв'язуванні задачі.

Нижче наведена основна таблиця, що відрізняється від звичайної наступним:

☑ замість стовпчиків векторів Р з відповідними значеннями c_j записуються стовпчики векторів A_j, координатами яких є відповідні стовпчики матриці B^{-1};

☑ в (m+1)- му рядку записуються компоненти вектору Ω, а не значення Δ;

☑ в таблиці є один додатковий стовпчик, в якому записують координати вектора P_s, розкладеного за біжучою базою, який включатиметься до бази на наступній ітерації.

Для того, щоб знайти вектор P_s, спочатку знаходимо вектор Ω, компоненти якого визначаються як скалярний добуток вектору c_B на відповідні вектори A_i (знайдені значення записуємо в останньому рядку основної таблиці та у відповідному стовпчику додаткової таблиці). Після

x_b	c_b	P_0	A_1	A_2	...	A_n	P_s
x_1	c_1	b_1	a_{11}	a_{12}	...	a_{1n}	x_{1s}
x_2	c_2	b_2	a_{21}	a_{22}	...	a_{2n}	x_{2s}
.	.	.	.	.	...	.	
x_i	c_i	b_i	a_{i1}	a_{i2}	...	a_{in}	x_{is}
.	.	.	.	.	...	.	
x_m	c_m	b_m	a_{m1}	a_{m2}	...	a_{mn}	x_{ms}
$\Omega^{(i)}$		Q	$\omega_1^{(i)}$	$\omega_2^{(i)}$	...	$\omega_m^{(i)}$	$\Delta_s^{(i)}$

Таб. 5.2. Основна таблиця модифікованого симплекс-методу

цього розраховуємо елементи відповідного Δ -рядка допоміжної таблиці. Якщо всі компоненти цього рядка невід'ємні, то знайдено оптимальний розв'язок. В іншому випадку , якщо задача має розв'язок, здійснюється перехід до наступного кроку. В останньому стовпчику основної таблиці записуємо компоненти вектора P_s, розкладені за векторами біжучої бази. Ці значення отримуємо в результаті множення матриці B^{-1} основної таблиці на вектор P_s допоміжної таблиці. Далі знаходимо ведучий рядок і за відомими правилами переходимо до наступної основної таблиці.

5.5. БЛОЧНІ ЗАДАЧІ ЛІНІЙНОГО ПРОГРАМУВАННЯ. МЕТОД ДАНЦІГА-ВУЛФА

Для більшості практичних задач ЛП характерною є їх велика розмірність (наявність багатьох змінних та обмежень) і розрідженість (наявність великої кількості нульових елементів) матриці обмежень. Класичні методи в цьому випадку виявляються малопридатними, що привело до розроблення як точних, так і наближених методів для розв'язу-вання задач великої розмірності такого типу.

В основному ці методи використовують прийом декомпозиції, тобто розбиття первісної задачі великої розмірності на ряд підзадач меншої розмірності, які незалежно розв'язуються за допомогою класичних методів, та координації, або узгодження незалежних розв'язків, виходячи з наявності спільних обмежень. Цей процес ітеративно повторюється до моменту отримання оптимального розв'язку задачі загалом, або ж до того моменту, коли будуть виконані умови зупинки.

Методи розв'язування задач великої розмірності ґрунтуються також на вивченні структури матриці обмежень і застосовуються в більшості до матриць блочно-діагональної структури, один з підвидів яких виглядає наступним чином:

$$A = \begin{bmatrix} A_1 & A_2 & \dots & A_g \\ B_1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & B_g \end{bmatrix},$$

де A_i, B_i - підматриці. Змістовно задачу такого типу можна розглядати наступним чином. До складу фірми, що випускає певні вироби, входить g відділень, кожне з яких для виробництва використовує незалежні від інших відділень автономні ресурси (для i -го відділення норми витрат незалежних ресурсів відображаються підматрицею B_i . Окрім того, в розпорядженні керівництва фірми є спільні ресурси, норми витрат яких задані зв'язуючим рядком $\begin{bmatrix} A_1 & A_2 & \dots & A_g \end{bmatrix}$ матриці обмежень. Необхідно побудувати такий план виробництва, для якого б досягався максимальний прибуток фірми загалом.

Відповідна задача ЛП формулюється наступним чином:

$$\begin{aligned} Q = c_1x_1 + c_2x_2 + \dots + c_gx_g & \Rightarrow Max \\ A_1x_1 + A_2x_2 + \dots + A_gx_g &= b_0 \\ B_1x_1 &= b_1 \\ B_2x_2 &= b_2 \\ \dots & \\ B_gx_g &= b_g \end{aligned} \qquad (5.18)$$

Вважаючи, що множини $R_j = \{x:\ B_jx_j = b_j,\ x_j \ge 0\}$, $j = \overline{1,g}$, — непусті та обмежені (ця умова не є обмежуючою), а x_j^k, $k = 1,\dots,K_j$ — екстремальні точки цієї множини, будь-яка з точок множини R_j може бути представленою у вигляді опуклої комбінації крайніх точок, тобто $x_j = \sum_{k=1}^{K_j} \beta_j^k \bar{x}_j^k$, $\sum_{k=1}^{K_j} \beta_j^k = 1$. Підставивши отримані співвідношення в умову первісної задачі, отримаємо:

$$\begin{aligned} Q = \sum_{k=1}^{K_1} c_1\bar{x}_1^k\beta_1^k + \sum_{k=2}^{K_2} c_2\bar{x}_2^k\beta_2^k + \dots + \sum_{k=1}^{K_g} c_g\bar{x}_g^k\beta_g^k & \Rightarrow Max \\ \sum_{k=1}^{K_1} A_1\bar{x}_1^k\beta_1^k + \sum_{k=2}^{K_2} A_2\bar{x}_2^k\beta_2^k + \dots + \sum_{k=1}^{K_g} A_g\bar{x}_g^k\beta_g^k &= b_0 \\ \sum_{k=1}^{K_1} \beta_1^k &= 1 \\ \sum_{k=1}^{K_2} \beta_2^k &= 1 \\ \dots & \\ \sum_{k=1}^{K_g} \beta_g^k &= 1 \end{aligned} \qquad (5.19)$$

Ця задача є головною - координуючою задачею, в якій невідомі значення β (припускаючи, що відомі значення координат усіх крайніх точок кожної з множин R_j). Якщо ми знайдемо ці значення, то цим самим розв'яжемо задачу.

Оптимальний розв'язок знаходиться за допомогою дворівневої процедури, яка має назву методу Данціга-Вулфа. В координуючій задачі кількість обмежень становить $m_A + g$, що суттєво менше, ніж кількість обмежень $m_A + \sum m_{B_i}$, де m_A , m_{B_i} - кількості рядків у матрицях A , B_i відповідно. Але кількість стовпчиків у координуючій задачі надзвичайно велика - рівна числу всіх крайніх точок множин R_j . Якщо б розв'язувати цю задачу безпосередньо, то ніякого виграшу в порівнянні з первісною задачею не було б. У методі Данціга-Вулфа не зберігаються всі стовпці матриці обмежень координуючої задачі — вони отримуються по мірі необхідності з використанням методу генерації стовпців та розв'язуванням множини допоміжних підзадач з використанням первісних підматриць B_j .

Представимо координуючу задачу в еквівалентній формі:

$$\begin{aligned} Q = \sum_{k=1}^{K_1} d_1^k\beta_1^k + \sum_{k=2}^{K_2} d_2^k\beta_2^k + \dots + \sum_{k=1}^{K_g} d_g^k\beta_g^k & \Rightarrow Max \\ \sum_{k=1}^{K_1} P_1^k\beta_1^k + \sum_{k=2}^{K_2} P_2^k\beta_2^k + \dots + \sum_{k=1}^{K_g} P_g^k\beta_g^k &= b_0 \\ \sum_{k=1}^{K_1} \beta_1^k = 1,\ \sum_{k=1}^{K_2} \beta_2^k = 1, \dots \sum_{k=1}^{K_g} \beta_g^k &= 1.\ \forall \beta_j^k \ge 0 \end{aligned} \qquad (5.20)$$

де $d_j^k = c_j x_j^k$, $P_j^k = A_j x_j^k$.

Розв'язання цієї задачі відносно невідомих β_j^k дозволяє знайти остаточний розв'язок, тому що $x_j = \sum_{k=1}^{K_j} \beta_j^k \hat{x}_j^k$. При розв'язуванні головної задачі за допомогою модифікованого симплекс-методу потрібно ідентифікувати змінні, що вводяться та виводяться з бази. На перший погляд може здатися, що для цього потрібно знання всіх кутових точок $\hat{x}_j^k$. Однак сенс побудови головної задачі при використанні методу декомпозиції полягає в тому, що цю задачу можна розв'язати, маючи лише одну екстремальну точку, що відповідає змінній, яка вводиться до бази. Після знаходження цієї точки можна чисельно визначити всі елементи вектора, що включається до бази, а також відповідне значення Δ. Після цього вектор, що виключається з бази, знаходиться звичайним шляхом у відповідності до умов припустимості симплекс-методу.

Таким чином, укрупнена схема алгоритму декомпозиції включає два етапи:

- перетворення первісної задачі в координуючу шляхом непрямого представлення простору розв'язків кожної підзадачі у вигляді опуклої комбінації його екстремальних точок;
- формування вектора-стовпчика, що асоційований зі змінною, яка включається до бази, та використання отриманого результату для визначення вектору, що виключається з бази.

Останній етап алгоритму реалізується за допомогою процедури генерації стовпчиків, за допомогою якої й отримують значення компонент, що включаються до бази.

Розглянемо деталі реалізації цієї процедури. Нехай B — біжуча база, а c_B — вектор коефіцієнтів функції мети при базових змінних. Згідно зі схемою обчислень модифікованого симплекс-методу, біжучий розв'язок є оптимальним, якщо для всіх небазових векторів P_j^k виконується умова оптимальності $\Delta_j = z_j - c_j^k = c_B B^{-1} P_j^k - c_j^k \ge 0$, де згідно з визначенням координуючої задачі $c_j^k = c_j \widehat{X}_j^k$ та $P_j^k = \left[\underbrace{A_j X_j^k;}_{r_0}\ \underbrace{0;\ldots;\overset{r_0+j}{1};\ldots;0}_{n} \right]^T$, одиничний вектор має n компонент, одиниця знаходиться в (r_0+j)-ій позиції. Здійснимо наступні перетворення. Нехай

$B^{-1} = \left[\overbrace{R_0}^{r_0} \middle| \overbrace{V_1, V_2, \ldots, V_j, \ldots, V_n}^{n} \right]$, де R_0 — матриця $(r_0+n)\times r_0$, що складається з перших r_0 стовпчиків матриці B^{-1}, а V_j — (r_0+j)-й стовпчик цієї ж матриці. У цьому випадку

$$z_j^k - c_j^k = \left(\left(c_B R_0 A_j X_j^k + c_B V \right)_j - c_j V_j \right) - c_j X_j^k = \left(c_B R_0 A_j - c_j \right) X_j^k + c_B V_j. \quad (5.21)$$

Якщо біжучий розв'язок неоптимальний, то вектор P_j^k, для якого значення Δ від'ємне і найбільше за абсолютним значенням повинен бути включений до бази. У цьому випадку суттєвим є те, що $z_j^k - c_j^k$ не можна обчислити до тих пір, поки не відомі всі елементи P_j^k, що є неможливим, поки не відома відповідна кутова точка X_j^k. Найсуттєвіша складова алгоритму власне й пов'язана з розв'язанням цієї проблеми.

Замість того, щоб знаходити всі кутові точки всіх n підмножин, задачу можна звести до визначення кутової точки $X_j^{k^*}$ кожної з підмножин, якій відповідатиме найменше значення Δ.

Нехай $\rho_j = \min_k \{ z_j^k - c_j^k \} = z_j^{k^*} - c_j^{k^*}$ та $\rho = \min_j \rho_j$. Таким чином, якщо $\rho < 0$, змінна $\beta_j^{k^*}$, що відповідає ρ, повинна бути введена до бази. В іншому випадку, якщо $\rho \ge 0$, отриманий розв'язок є оптимальним. Кожна з екстремальних кутових точок визначається внаслідок розв'язання наступної задачі ЛП:

$$z_j - c_j \Rightarrow Min,\ B_j X_j = b_j,\ X_j \ge 0. \quad (5.22)$$

В цій задачі індекс k не вживається з наступної причини. Внаслідок припущення про обмеженість множини $B_j X_j = b_j$, $X_j \ge 0$ мінімальне значення $z_j - c_j$ також обмежене і повинно відповідати деякій кутовій точці цієї множини. Оскільки $z_j^k - c_j^k = \left(c_B R_0 A_j - c_j \right) X_j^k + c_B V_j$ і $c_B V_j$ постійна, що не залежить від k, задача переформульовується наступним чином.

$$\left(c_B R_0 A_j - c_j \right) X_j \Rightarrow Min,\ B_j x_j = b_j,\ x_j \ge 0. \quad (5.23)$$

Звідси визначаємо: $\rho_j = \omega^*_j + c_B V_j$, де ω^*_j — оптимальне значення ω_j. Змінна, що виводиться з бази, знаходиться згідно з умовою припустимості, що використовується в модифікованому симплекс-методі.

Таким чином, алгоритм декомпозиції складається з наступних кроків:

Крок 1. Перетворюємо первісну задачу таким чином, щоб в її модифікованому формулюванні фігурували нові змінні β_j^k.

Крок 2. Знаходимо початковий базовий розв'язок модифікованої задачі. (На цьому кроці доцільно застосувати перший етап двохетапного симплекс-методу).

Крок 3. На біжучій ітерації знаходимо $\rho_j = \omega^*_j + c_B V_j$ для кожної з підзадач та визначаємо ρ. Якщо $\rho \geq 0$, то знайдений розв'язок оптимальний — **стоп**.

Крок 4. Включаємо змінну β^k_j, що відповідає ρ, до бази. Визначаємо змінну, що виключається з бази, виключаємо її та обчислюємо нову матрицю B^{-1}. Перехід до кроку 3.

Таким чином, врахування структури задачі лінійного програмування дозволяє конструювати спеціалізовані алгоритми, які здатні розв'язувати задачі великої розмірності.

ПРИКЛАДИ

Приклад 5.1. Побудова двоїстої задачі.

Задана умова наступної задачі ЛП. Необхідно побудувати двоїсту до неї.

$$Q = 5x_1 + 12x_2 + 4x_3 \quad \Rightarrow Max$$
$$x_1 + 2x_2 + x_3 \leq 10$$
$$2x_1 - x_2 + 3x_3 = 8$$
$$\forall x_j \geq 0.$$

Розв'язання.

Приводимо задачу до канонічної форми та від неї переходимо до двоїстої.

$$Q = 5x_1 + 12x_2 + 4x_3 + 0x_4 \quad \Rightarrow Max \qquad G = 10y_1 + 8y_2 \Rightarrow Min$$
$$x_1 + 2x_2 + x_3 + x_4 = 10 \rightarrow y_1 \qquad y_1 + 2y_2 \geq 5$$
$$2x_1 - x_2 + 3x_3 + 0x_4 = 8 \rightarrow y_2 \qquad 2y_1 - y_2 \geq 12$$
$$\forall x_j \geq 0. \qquad y_1 + 3y_2 \geq 4$$
$$y_1 + \quad \geq 0$$
$$y_2 <> 0$$

Приклад 5.2. Отримання розв'язку двоїстої задачі з розв'язку прямої.

Необхідно, використовуючи наявний розв'язок прямої задачі, побудувати двоїсту та знайти її розв'язок.

$$Q = 3x_1 + 2x_2 \quad \Rightarrow Max$$
$$x_1 + 2x_2 \leq 6$$
$$2x_1 + x_2 \leq 8$$
$$-x_1 + x_2 \leq 1$$
$$x_2 \leq 2,\ x_1 \geq 0,\ x_2 \geq 0.$$

Розв'язання.

Приводимо задачу до канонічного вигляду та будуємо двоїсту до неї:

$$Q = 3x_1 + 2x_2 + 0x_3 + 0x_4 + 0x_5 + 0x_6 \Rightarrow Max$$
$$1x_1 + 2x_2 + 1x_3 + 0x_4 + 0x_5 + 0x_6 = 6 \rightarrow y_1$$
$$2x_1 + 1x_2 + 0x_3 + 1x_4 + 0x_5 + 0x_6 = 8 \rightarrow y_2$$
$$-1x_1 + 1x_2 + 0x_3 + 0x_4 + 1x_5 + 0x_6 = 1 \rightarrow y_3$$
$$0x_1 + 1x_2 + 0x_3 + 0x_4 + 0x_5 + 1x_6 = 2 \rightarrow y_4$$
$$\forall x_j \geq 0.$$

Кожне обмеження прямої задачі відповідає змінній двоїстої задачі, кожна змінна прямої задачі відповідає обмеженню двоїстої. Запишемо умову двоїстої задачі:

$$G(y) = 6y_1 + 8y_2 + y_3 + y_4 \Rightarrow Min$$
$$y_1 + 2y_2 - y_3 \qquad \geq 3$$
$$2y_1 + y_2 + y_3 + y_4 \geq 2$$
$$y_1 \geq 0,\ y_2 \geq 0,\ y_3 \geq 0,\ y_4 \geq 0.$$

Оскільки пряма задача розв'язана в темі 4, приклад 2, запишемо останню симплекс-таблицю для прямої задачі, в якій наявний оптимальний розв'язок:

			c_1	c_2	c_3	c_4	c_5	c_6
x_b	c_b	P_0	**3**	**2**	**0**	**0**	**0**	**0**
			P_1	P_2	P_3	P_4	P_5	P_6
x_2	2	4/3	0	1	2/3	-1/3	0	0
x_1	3	10/3	1	0	-1/3	2/3	0	0
x_5	0	3	0	0	-1	1	1	0
x_6	0	2/3	0	0	-2/3	1/3	0	1
Q		38/3	0	0	1/3	4/3	0	0

Базові змінні першої симплекс-таблиці візьмемо там же: x_3, x_4, x_5, x_6.

Для визначення оптимальних значень змінних двоїстої задачі згідно з теоремами двоїстості складаємо рівняння:

Коефіцієнт Δ при початковій базовій змінній в рядку прямої задачі	**=**	**Різниця між лівою та правою частинами обмеження двоїстої задачі, яке асоційоване з цією початковою змінною**

або $\Delta_j = \sum_{i=1}^{m} a_{ij} y_i - c_j$.

Таким чином, отримуємо систему рівнянь відносно змінних першого базового розв'язку:

$$1/3 = y_1^* - 0 \quad \rightarrow x_3 \quad y_1^* = 1/3$$

$$4/3 = y_2^* - 0 \quad \rightarrow x_4 \quad y_2^* = 4/3$$

$$0 = y_3^* - 0 \quad \rightarrow x_5 \quad y_3^* = 0$$

$$0 = y_4^* - 0 \quad \rightarrow x_6 \quad y_4^* = 0$$

$$G^* = 6y_1^* + 8y_2^* + y_3^* + y_4^* = 6\times\frac{1}{3} + 8\times\frac{4}{3} + 1\times 0 + 1\times 0 = 12\frac{1}{3} = Q^*$$

Приклад 5.3. Двоїстий симплекс-метод.

Наступну задачу розв'язати за допомогою двоїстого симплекс-методу.

$$-4x_1 - 7x_2 - 8x_3 - 5x_4 \Rightarrow Max,$$

$$x_1 + x_2 + 2x_4 >= 4$$

$$2x_1 + x_2 + 2x_3 >= 2$$

$$x_1 >= 0 \quad x_2 >= 0 \quad x_3 >= 0 \quad x_4 >= 0.$$

Розв'язання.

Приводимо задачу до канонічно вигляду і домножуємо кожне з двох обмежень-рівнянь на -1, щоб отримати початковий базовий розв'язок.

$$-4x_1 - 7x_2 - 8x_3 - 5x_4 \Rightarrow Max,$$

$$x_1 + x_2 + 2x_4 - x_5 = 4 \quad \times(-1)$$

$$2x_1 + x_2 + 2x_3 - x_6 = 2 \quad \times(-1)$$

$$\forall x_j \ge 0.$$

$$-4x_1 - 7x_2 - 8x_3 - 5x_4 \Rightarrow Max,$$

$$-x_1 - x_2 - 2x_4 + x_5 = -4$$

$$-2x_1 - x_2 - 2x_3 + x_6 = -2$$

$$\forall x_j \ge 0.$$

Таким чином, отримаємо початкову базу, але з від'ємними складовими вектора b правої частини системи обмежень, внаслідок чого звичайний симплекс-метод буде незастосовний. Якщо при цьому всі Δ будуть невід'ємними, то можна застосувати двоїстий симплекс-метод. Будуємо першу симплекс-таблицю.

			c_1	c_2	c_3	c_4	c_5	c_6
x_b	c_b	P_0	-4	-7	-8	-5	0	0
			P_1	P_2	P_3	P_4	P_5	P_6
x_5	0	-4	-1	-1	0	-2	1	0
x_6	0	-6	-2	-1	-2	0	0	1
Q		0	4	7	8	5	0	0
x_5	0	-1	0	-1/2	1	-2	1	1/2
x_1	-4	3	1	1/2	1	0	0	-1/2
Q		-2	0	5	4	5	0	2
x_4	-5	1/2	0	1/4	-1/2	1	-1/2	-1/4
x_1	-4	3	1	1/2	1	0	0	-1/2
Q		-29/2	0	2	13/2	0	5/2	13/4

$b_i < 0 \quad \underset{i}{Max}(-b_i) = Max(4;6) = 6$

$\underset{j}{Min}\left(\frac{-\Delta_j}{a_{kj}}\right) = Min(2;4;7) = 2.$

Згідно до двоїстого симплекс-методу, всі Δ невід'ємні, але порушуються обмеження задачі (є від'ємні компоненти вектора b). Обираємо спочатку ведучий рядок, що відповідає змінній x_6, а потім ведучий стовпчик P_1 (формули згідно з алгоритмом наведені при таблиці).

Таким чином, виводимо з бази x_6, і на її місце вводимо x_1 та переходимо до наступної таблиці, розраховуючи значення в ній так само, як і в звичайному симплекс-методі, використовуючи схему Гауса-Жордана (правило трикутника).

Оскільки в другій таблиці умова оптимальності не виконується (не всі компоненти b невід'ємні), переходимо до третьої таблиці, і досягаємо оптимального розв'язку ($x_1^* = 3$, $x_2^* = 0$, $x_3^* = 0$, $x_4^* = \frac{1}{2}$, $Q^* = -\frac{29}{2}$), який є першим припустимим розв'язком за умовами роботи алгоритму двоїстого симплекс-методу.

Приклад 5.4. Модифікований симплекс-метод.

Розв'яжемо за допомогою модифікованого симплекс-методу задачу, яку ми вже розв'язали звичайним симплекс-методом.

$$Q = 3x_1 + 2x_2 \Rightarrow Max$$
$$x_1 + 2x_2 \le 6$$
$$2x_1 + x_2 \le 8$$
$$-x_1 + x_2 \le 1$$
$$x_2 \le 2,\ x_1 \ge 0,\ x_2 \ge 0.$$

$$Q = 3x_1 + 2x_2 + 0x_3 + 0x_4 + 0x_5 + 0x_6 \Rightarrow Max$$
$$1x_1 + 2x_2 + 1x_3 + 0x_4 + 0x_5 + 0x_6 = 6 \quad \rightarrow y_1$$
$$2x_1 + 1x_2 + 0x_3 + 1x_4 + 0x_5 + 0x_6 = 8 \quad \rightarrow y_2$$
$$-1x_1 + 1x_2 + 0x_3 + 0x_4 + 1x_5 + 0x_6 = 1 \quad \rightarrow y_3$$
$$0x_1 + 1x_2 + 0x_3 + 0x_4 + 0x_5 + 1x_6 = 2 \quad \rightarrow y_4$$
$$\forall x_j \ge 0.$$

Ця задача має початковий базовий розв'язок x=(0; 0; 6; 8; 1; 2), що визначається векторами P_3, P_4, P_5, P_6. Складові цих векторів визначають одиничну матрицю B , обернена до якої теж є одиничною.

Складаємо основну та допоміжну таблиці. В допоміжній таблиці заповнюємо 4 рядки стовпчиків векторів, а в основній — перші 4 рядки, враховуючи, що обернена до одиничної матриці є одиничною матрицею.

			c_1	c_2	c_3	c_4	c_5	c_6			
x_b	c_b	P_0	3	2	0	0	0	0			
			P_1	P_2	P_3	P_4	P_5	P_6	$\Omega^{(1)}$	$\Omega^{(2)}$	$\Omega^{(3)}$
x_3	0	6	1	2	1	0	0	0	0	0	1/3
x_4	0	8	2	1	0	1	0	0	0	3/2	4/3
x_5	0	1	-1	1	0	0	1	0	0	0	0
x_6	0	2	0	1	0	0	0	1	0	0	0
$\Delta^{(1)}$			-3	-2	0	0	0	0			
$\Delta^{(2)}$			0	-1/2	0	3/2	0	0			
$\Delta^{(3)}$			0	0	1/3	4/3	0	0			

Допоміжна таблиця.

Визначаємо вектор $\Omega^{(1)}$, записуючи його в основну таблицю:

$$\omega_1^{(1)} = c_B A_1 = 0\times1 + 0\times0 + 0\times0 + 0\times0 = 0$$
$$\omega_2^{(1)} = c_B A_2 = 0\times0 + 0\times1 + 0\times0 + 0\times0 = 0$$
$$\omega_3^{(1)} = c_B A_3 = 0\times0 + 0\times0 + 0\times1 + 0\times0 = 0$$
$$\omega_4^{(1)} = c_B A_4 = 0\times0 + 0\times0 + 0\times0 + 0\times1 = 0$$

Розрахуємо також значення функції мети та запишемо його в 1-у основну таблицю. Після заповнення 1-го додаткового рядка основної таблиці значення компонент $\Omega^{(1)}$ перепишемо в 1-й додатковий стовпчик додаткової таблиці.

Після цього за формулою $\Delta_j = \Omega P_j - c_j$ розраховуємо відповідні значення критеріїв оптимальності та записуємо їх в додатковий рядок додаткової таблиці. Оскільки серед цих значень є від'ємні, то переходимо до наступного базового розв'язку.

В базу необхідно ввести вектор P_1. Тому цей стовпчик з додаткової таблиці, розкладений за векторами наявної бази (для цього множимо обернену матрицю, наявну в стовпчиках A 1-ї основної таблиці, на вектор P_1 в додатковій таблиці), дописуємо в 1-у основну таблицю.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 2 \\ -1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ -1 \\ 0 \end{bmatrix}$$

1-а основна таблиця.

x_b	c_b	P_0	A_1	A_2	A_3	A_4	P_1	
x_3	0	6	1	0	0	0	1	6/1=6
x_4	0	8	0	1	0	0	2	8/2=4
x_5	0	1	0	0	1	0	-1	—
x_6	0	2	0	0	0	1	0	—
$\Omega^{(1)}$		0	0	0	0	0	-3	

З бази за правилами звичайного симплекс-методу виводимо x_4. Число 2 є ведучим елементом. Виконуємо перехід до нової основної таблиці симплекс-методу.

2-а основна таблиця.

x_b	c_b	P_0	A_1	A_2	A_3	A_4	P_2	
x_3	0	2	1	-1/2	0	0	3/2	4/3
x_1	3	4	0	1/2	0	0	1/2	8
x_5	0	5	0	1/2	1	0	3/2	10/3
x_6	0	2	0	0	0	1	1/2	4
$\Omega^{(2)}$		12	0	3/2	0	0	-1/2	

Визначаємо вектор $\Omega^{(2)}$, записуючи його в основну таблицю:

$$\omega_1^{(2)} = c_B A_1 = 0,\ \omega_2^{(2)} = c_B A_2 = 3/2$$

$$\omega_3^{(2)} = c_B A_3 = 0,\ \omega_4^{(2)} = c_B A_4 = 0$$

Обчислюємо $\Delta^{(2)}$ в основній таблиці. До бази вводимо P_2, записуємо в 2-гу основну таблицю його представлення за векторами нової бази:

$$\begin{bmatrix} 1 & -1/2 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 1/2 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 2 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 3/2 \\ 1/2 \\ 3/2 \\ 1/2 \end{bmatrix}$$

З бази виводимо x_3, а вводимо P_2. Переходимо до наступної основної симплекс-таблиці.

3-я основна таблиця.

x_b	c_b	P_0	A_1	A_2	A_3	A_4
x_2	2	4/3	2/3	-1/3	0	0
x_1	3	10/3	-1/3	2/3	0	0
x_5	0	3	-1	1	1	0
x_6	0	2/3	-2/3	1/3	0	1
$\Omega^{(2)}$		38/3	1/3	4/3	0	0

Розраховуємо значення $\Delta^{(3)}$. Оскільки всі його компоненти невід'ємні, то знайдений оптимальний розв'язок.

РЕЗЮМЕ

5.1. Поняття двоїстості є одним із найважливіших у лінійному програмуванні. Оскільки задачу ЛП завжди можна привести до канонічної форми, то пряму задачу ми розглядатимемо завжди в канонічній формі. Відповідно до цього змінні двоїстої задачі необмежені в знаку, коефіцієнтами критерію якості, котрий потрібно мінімізувати, є вектор правих частин обмежень прямої задачі, а вектор правих частин обмежень двоїстої задачі складають коефіцієнти функції мети прямої задачі. Кількість змінних двоїстої задачі рівна кількості обмежень прямої, а кількість обмежень – кількості змінних прямої задачі. Двоїста задача до двоїстої буде первісною прямою задачею.

5.2. Зв'язок між розв'язками прямої та двоїстої задач встановлюється за допомогою двох теорем двоїстості. Якщо одна з пари двоїстих задач має розв'язок, то й інша має розв'язок, і оптимальні значення функцій мети рівні між собою. Якщо ж функція мети для однієї з задач не обмежена, двоїста до неї задача взагалі не має припустимих розв'язків. Друга теорема двоїстості (теорема про доповнюючу нежорсткість) дає можливість за оптимальним розв'язком однієї з задач знайти оптимальний розв'язок двоїстої до неї.

5.3.Змінні двоїстої задачі мають важливий економічний сенс – їх значення є значеннями цінностей ресурсів (тіньовими цінами). Для неоптимального розв'язку (прибуток)<=(цінність ресурсу), іншими словами цінні ресурси не використані повністю, і можливе збільшення прибутку за рахунок використання відповідних видів виробничої діяльності (випуску продукції певних видів). На співвідношеннях двоїстості ґрунтується різновид симплекс-методу – двоїстий симплекс-метод, який може використовуватись як самостійний метод, хоча знаходить широке застосування як крок методу Гоморі внаслідок специфічного способу формування додаткових обмежень під час роботи алгоритму методу Гоморі.

5.4. У варіантах симплекс-методу при реалізації послідовних ітерацій – переході від попередньої до наступної симплекс-таблиці – використовується метод Гауса-Жордана, при застосуванні якого для автоматизації розрахунків необхідний великий об'єм пам'яті. З метою усунення такого недоліку був розроблений модифікований симплекс-метод, що є різновидом симплекс-методу, який, крім того, дозволяє в багатьох випадках зменшити ще й загальну кількість арифметичних операцій.

5.5. Для більшості практичних задач ЛП характерною є їх велика розмірність (наявність багатьох змінних та обмежень) і розрідженість (наявність великої кількості нульових елементів) в матриці обмежень. Класичні методи в цьому випадку виявляються малопридатними, що привело до розроблення як точних, так і наближених методів для розв'язування задач великої розмірності такого типу. В основному ці методи використовують прийом декомпозиції, тобто розбиття первісної задачі великої розмірності на ряд підзадач меншої розмірності, які незалежно розв'язуються за допомогою класичних методів, та координації, або узгодження незалежних розв'язків, виходячи з наявності спільних обмежень. Цей процес ітеративно повторюється до моменту отримання оптимального розв'язку задачі загалом, або ж до того моменту, коли будуть виконані умови зупинки.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 5.1.

Побудуйте двоїсту задачу до заданої.

$$
\begin{aligned}
&Q = 6x_1 - 2x_2 + 9x_3 \Rightarrow Max \\
&2x_1 + x_2 + x_3 \le 10 \\
&2x_1 - x_2 + 3x_3 = 8 \\
&-2x_1 + x_2 - 7x_3 \ge 30 \\
&\forall x_j \ge 0.
\end{aligned}
$$

Завдання 5.2.

Розв'яжіть пряму задачу лінійного програмування, побудуйте двоїсту задачу до неї та за оптимальним розв'язком прямої задачі знайдіть оптимальний розв'язок двоїстої.

$$
\begin{aligned}
&Q = x_1 + x_2 \Rightarrow Max \\
&-3x_1 + 2x_2 \le 1 \\
&x_1 + 2x_2 \le 14 \\
&2x_1 + x_2 \le 13 \\
&3x_1 \qquad x_2 \le 12 \\
&\forall x_j \ge 0.
\end{aligned}
$$

Завдання 5.3.

Розв'яжіть наступну задачу лінійного програмування за допомогою двоїстого симплекс-методу.

$$
\begin{aligned}
&3x_1 + 2x_2 + 5x_3 \Rightarrow Min, \\
&2x_1 - 3x_2 + x_3 \ge 10 \\
&4x_1 + x_2 + 2x_3 \ge 15 \\
&\forall x_j \ge 0.
\end{aligned}
$$

Завдання 5.4.

Розв'яжіть наступну задачу лінійного програмування за допомогою модифікованого симплекс-методу.

$$
\begin{aligned}
&x_1 + x_2 + 2x_3 + 8x_4 \Rightarrow Min, \\
&2x_1 - x_2 + 3x_3 - 2x_4 = 3 \\
&-x_1 + 3x_2 - 4x_3 + 4x_4 = 1 \\
&\forall x_j \ge 0.
\end{aligned}
$$

Завдання 5.5.

Розв'яжіть наступну задачу лінійного програмування за допомогою методу Данціга-Вулфа.

$$
\begin{aligned}
&Q = x_1 + x_2 + 2x_3 + x_4 \Rightarrow Max \\
&x_1 + 2x_2 + x_3 - x_4 \le 10 \\
&2x_1 + x_2 \le 4 \\
&x_1 + 2x_2 \le 6 \\
&x_3 + x_4 \le 5 \\
&-2x_3 + 3x_4 \le 6 \\
&\forall x_j \ge 0.
\end{aligned}
$$

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Як будується двоїста до прямої задачі лінійного програмування?
2. Доведіть, що двоїста до двоїстої задачі є прямою задачею.
3. В чому полягає зв'язок між прямою та двоїстою задачею лінійного програмування та яке значення він має?
4. Яке практичне значення першої теореми двоїстості ?
5. Що таке умови доповнюючої нежорсткості та де вони застосовуються?
6. Яким чином економічно інтерпретуються пряма та двоїста задачі лінійного програмування?
7. Чим відрізняється двоїстий симплекс-метод від прямого?
8. В чому полягають переваги модифікованого симплекс-методу порівняно зі звичайним?
9. В яких випадках застосовується метод Данціга-Вулфа?

ТЕМА 6. ТРАНСПОРТНА ЗАДАЧА ЛІНІЙНОГО ПРОГРАМУВАННЯ

Транспортні задачі належать до класу задач лінійного програмування. і знаходять широке застосування на практиці в економіці. Цьому сприяє також їх структура, що дозволило розробити ефективні алгоритми їх розв'язування. Для розв'язування транспортна задача завжди повинна бути приведена до задачі закритого типу. Закрита транспортна задача завжди має припустимий розв'язок, і, крім того, якщо значення всіх коефіцієнтів у транспортній задачі закритого типу є цілими числами, то й оптимальний розв'язок є цілочисельним. Алгоритм розв'язування транспортної задачі складається з двох основних етапів: знаходження початкового опорного плану транспортної задачі та покрокове покращення плану перевезень до моменту досягнення оптимального. Метод потенціалів є особливим випадком симплекс-методу, що враховує особливості транспортної задачі. Для уникнення незручностей, які виникають у методі потенціалів і пов'язані з виродженістю, застосовується метод диференційних рент, він і використовується частіше для програмних реалізацій. Задача про призначення є особливим видом транспортної, для якої виявилося можливим з врахуванням специфічних її особливостей побудувати ефективні алгоритми розв'язу-вання, одним з яких є угорський алгоритм.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати:⇨формальні постановки транспортної задачі та задачі про призначення;⇨ властивості транспортної задачі;⇨алгоритми розв'язування транспортної задачі та задачі про призначення;⇨інтерпретацію транспортної задачі в термінах симплекс - методу та теорему про потенціали;

вміти:⇨знаходити початковий опорний план та розв'язувати транспортну задачу методом потенціалів;⇨розв'язувати транспортні задачі з ускладненням у постановці та з виродженням;⇨застосовувати алгоритм методу диференційних рент до розв'язування транспортної задачі; ⇨розв'язувати задачу про призначення за допомогою угорського методу.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ відкрита транспортна задача (ТЗ)
- ☑ теорема про потенціали
- ☑ пряма та двоїста ТЗ
- ☑ закрита ТЗ
- ☑ метод потенціалів
- ☑ рента
- ☑ метод північно-західного кута
- ☑ метод диференційних рент
- ☑ задача про призначення
- ☑ метод мінімального елемента
- ☑ цикл
- ☑ угорський метод
- ☑ евристичний метод Фойгеля
- ☑ ТЗ з ускладненнями в постановці
- ☑ викреслення
- ☑ опорний план ТЗ
- ☑ виродженість
- ☑ редукція

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

6.1. Математична та змістовна постановка транспортної задачі (ТЗ). Методи знаходженння опорного плану ТЗ.
6.2. Метод потенціалів. Розв'язування транспортних задач з ускладненнями в постановці.
6.3. Інтерпретація методу потенціалів як симплекс-методу.
6.4. Метод диференційних рент.
6.5. Задача про призначення.

6.1. МАТЕМАТИЧНА ТА ЗМІСТОВНА ПОСТАНОВКА ТРАНСПОРТНОЇ ЗАДАЧІ. МЕТОДИ ЗНАХОДЖЕННЯ ОПОРНОГО ПЛАНУ ТРАНСПОРТНОЇ ЗАДАЧІ

Змістовно транспортна задача формулюється наступним чином. Необхідно перевезти однорідний продукт з пунктів зберігання (комор) у пункти споживання таким чином, щоб загальна вартість перевезень була мінімальною. Наявно m пунктів зберігання $A_1,\dots,A_m$ та n пунктів споживання $B_1,\dots,B_n$. Запас продукту в кожному і-му пункті зберігання A_i становить a_i, а потреби в кожному j-му пункті споживання B_j рівні b_j. Вартість перевезення одиниці продукту з кожного і-го пункту зберігання в кожний j-й пункт споживання також відома і становить c_{ij}. Якщо сумарні потреби рівні сумарним запасам продукту в коморах - $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$, то транспортна задача є задачею закритого типу, формальна математична модель якої виглядає наступним чином:

$$Q(x)=\sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}x_{ij} \Rightarrow Min \ , \sum_{i=1}^{m} x_{ij}=b_j, \ \sum_{j=1}^{n} x_{ij}=a_i,$$

$$i=\overline{1,m}, \ j=\overline{1,n}, \quad x_{ij}\ge 0. \qquad (6.1)$$

Для відкритої транспортної задачі характерним є те, що сумарні запаси та сумарні потреби є незбалансованими. Відкрита задача завжди приводиться до закритої за допомогою наступних підстановок.

У випадку, коли $\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j$, тобто наявні запаси перевищують потреби, необхідно ввести додатковий пункт споживання B_{n+1}, потреби якого становитимуть $b_{n+1}=\sum_{i=1}^{m} a_i - \sum_{j=1}^{n} b_j$ та вартості перевезень у цей пункт становитимуть $c_{i,n+1}=0, \ i=\overline{1,m}$. В іншому випадку $\sum_{i=1}^{m} a_i < \sum_{j=1}^{n} b_j$, і необхідно ввести додатковий пункт зберігання A_{m+1} з запасом $a_{m+1}=\sum_{j=1}^{n} b_j - \sum_{i=1}^{m} a_i$ та нульовими вартостями перевезень $c_{m+1,j}=0, \ j=\overline{1,n}$.

Для розв'язування транспортна задача завжди повинна бути приведена до задачі закритого типу.

Основними властивостями закритої транспортної задачі є наступні.

1. Закрита транспортна задача завжди має припустимий розв'язок.

Для доведення припустимості закритої транспортної задачі розглянемо наступний розв'язок: $x_{ij}=\dfrac{a_i b_j}{d}$, $d=\sum_{i=1}^{m} a_i=\sum_{j=1}^{n} b_j$. Для цього розв'язку виконуються обмеження задачі - $x_{ij}\ge 0$,

$$\sum_{j=1}^{n}\frac{a_i b_j}{d}=\frac{a_i}{d}\sum_{j=1}^{n} b_j=a_i, \ i=1,\dots,m, \quad \sum_{i=1}^{m}\frac{a_i b_j}{d}=\frac{b_j}{d}\sum_{i=1}^{m} b_i=a_i, \ j=1,\dots,n$$

Таким чином, для закритої транспортної задачі завжди існує хоча б один припустимий розв'язок. Окрім того, розв'язок транспортної задачі не є необмеженим, тобто функція мети приймає скінчене значення, що є наслідком обмеженості многогранника — області припустимих розв'язків, так як справедливе співвідношення $x_{ij}\le Min\{a_i;b_j\}$.

2. Ранг матриці системи обмежень-рівностей закритої невиродженої транспортної задачі $r(A)=m+n-1$.

Матриця коефіцієнтів обмежень транспортної задачі за умови приведення задачі до класичного вигляду задачі лінійного програмування з впорядкуванням змінних $x_{11},x_{12},\dots,x_{1n},x_{21},x_{22},\dots,x_{2n},\dots,x_{m1},x_{m2},\dots,x_{mn}$ у загальному випадку має вигляд:

$$A=\left[\begin{array}{cccccccccccccc|c}
1 & 1 & \dots & 1 & 0 & 0 & \dots & 0 & \dots & \dots & 0 & 0 & \dots & 0 & a_1\\
0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 & \dots & \dots & 0 & 0 & \dots & 0 & a_2\\
\dots & & & & & & & & & & & & & & \dots\\
0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \dots & \dots & 1 & 1 & \dots & 1 & a_m\\
1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & \dots & \dots & 1 & 0 & \dots & 0 & b_1\\
0 & 1 & \dots & 0 & 0 & 1 & \dots & 0 & \dots & \dots & 0 & 1 & \dots & 0 & b_2\\
\dots & & & & & & & & & & & & & & \dots\\
0 & 0 & \dots & 1 & 0 & 0 & \dots & 1 & \dots & \dots & 0 & 0 & \dots & 1 & b_n
\end{array}\right] \qquad (6.2)$$

Доведемо, що ранг матриці A рівний $r(A)=m+n-1$.

1-й рядок цієї матриці є лінійною комбінацією інших: додамо поелементно рядки цієї матриці, починаючи з $(m+1)$-го до $(m+n)$-го та віднімемо від отриманого результату суму рядків від 2-го до m -го отримаємо 1-й рядок. Таким чином ми показали, що $r(A)\le m+n-1$. Доведемо тепер, що рядки від 2-го до $(m+n)$-го є лінійно незалежними. Для цього слід показати, що будь-який з цих рядків представляється лінійною комбінацією інших лише за умови рівності нулю всіх лінійних коефіцієнтів.

Помножимо 2-й рядок матриці на α_2, і так далі m -й - на α_m, $(m+1)$-й — на β_1,..., $(m+n)$-й - на β_n, додамо їх і припустимо, що значення суми рівне нулю. Тоді для перших n координат цього нульового рядка отримаємо рівності:

$$0=0\times\alpha_2+\dots+0\times\alpha_m+\ \ 1\times\beta_1+0\times\beta_2+\dots+0\times\beta_n=0$$

$$\dots\dots\dots$$

$$0=0\times\alpha_2+\dots+0\times\alpha_m+0\times\beta_1+\dots+1\times\beta_k+\dots+0\times\beta_n=0 \qquad (6.3)$$

$$\dots\dots\dots$$

$$0=0\times\alpha_2+\dots+0\times\alpha_m+\ \ 0\times\beta_1+0\times\beta_2+\dots+1\times\beta_n=0$$

Звідси $\beta_1=\beta_2=\dots=\beta_n=0$.

Для інших з $(n+1)$-го до до $((m-1)n+1)$-го отримаємо:

$$\begin{matrix} 0 = 1\times\alpha_2 + 0\times\alpha_3 ... + 0\times\alpha_m + 1\times\beta_1 + 0\times\beta_2 ... + 0\times\beta_n \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ 0 = 0\times\alpha_2 + 0\times\alpha_3 ... + 1\times\alpha_m + 1\times\beta_1 + 0\times\beta_2 ... + 0\times\beta_n \end{matrix} . \quad (6.4)$$

Враховуючи, що $\beta_1 = \beta_2 = ... = \beta_n = 0$, отримаємо:

$\alpha_2 = \alpha_3 = ... = \alpha_m = 0$,

тобто рядки матриці A з 2-го по $(m+n)$-й є лінійно незалежними і $r(A) = m+n-1$.

3. Якщо значення всіх коефіцієнтів a_i, b_j в транспортній задачі закритого типу є цілими числами, то й оптимальний розв'язок є цілочисельним.

Це безпосередньо випливає з унімодулярності матриці A (в кожному стовпчику є по два одиничних елементи, а інші рівні нулю). З іншого боку – на кожній ітерації методу потенціалів перевезення змінюються на ціле число, і, крім того, початковий опорний план є також цілочисельним.

Алгоритм розв'язування транспортної задачі складається з двох основних етапів:

1.Знаходження початкового опорного плану транспортної задачі (початкового базового розв'язку).

2.Покрокове покращення плану перевезень до моменту досягнення оптимального.

Для знаходження початкового опорного плану транспортної задачі використовується один із трьох методів: метод північно-західного кута; метод мінімального елемента; евристичний метод Фойгеля.

При розв'язуванні транспортної задачі використовується представлення її у вигляді наступної таблиці:

B_j / A_i	B_1	...	B_n	Запаси
A_1	c_{11} x_{11}	...	c_{1n} x_{1n}	a_1
...	...	...	...	...
A_m	c_{m1} x_{m1}		c_{mn} x_{mn}	a_m
Потреби	b_1	...	b_n	

Табл. 6.1. Таблиця транспортної задачі

Для знаходження початкового базового розв'язку можна використати метод північно-західного кута, метод мінімального елемента або евристичний метод Фойгеля. Найпродуктивнішим є використання методу Фойгеля, який, не зважаючи на свою складність порівняно з іншими методами, дозволяє в середньому отримати початковий базовий розв'язок (опорний план транспортної задачі), найближчий до оптимуму і цим самим суттєво зменшити кількість наступних ітерацій, необхідних для отримання оптимального розв'язку (оптимального плану транспортної задачі).

Метод північно-західного кута.

При знаходженні розв'язку задачі за допомогою методу північно-західного кута заповнення клітинок значеннями перевезень починається від верхнього лівого ("північно-західного") кута таблиці; надалі рух та заповнення відповідних клітинок відбувається зліва направо до моменту вичерпання запасу відповідного пункту; якщо запас вичерпаний, опускаємося на клітинку вниз і продовжуємо рух до моменту вичерпання всіх запасів.

Метод мінімального елемента.

На кожному кроці з числа незаповнених клітинок таблиці транспортної задачі обираємо клітинку з мінімальним значенням тарифу, заповнюємо її та виключаємо з подальшого розгляду стовпчик або рядок, в якому знаходиться ця клітинка, в залежності від того, чи задоволені відповідні потреби, чи вичерпані відповідні запаси.

Евристичний метод Фойгеля.

На кожній ітерації методу Фойгеля для кожного стовпчика та рядка таблиці обчислюється різниця між значеннями мінімального та найближчого до нього тарифів відповідного стовпчика чи рядка та записується у відповідні клітинки додаткового стовпчика та рядка. Серед обчислених різниць обираємо максимальну і у відповідному стовпчику чи рядку знаходимо мінімальний тариф з призначенням перевезення цій клітинці.

Якщо є декілька однакових тарифів, то для заповнення обирається клітинка, що відповідає максимальній різниці.

6.2. МЕТОД ПОТЕНЦІАЛІВ. РОЗВ'ЯЗУВАННЯ ТРАНСПОРТНИХ ЗАДАЧ З УСКЛАДНЕННЯМИ В ПОСТАНОВЦІ

Метод потенціалів є особливим випадком симплекс-методу, що враховує особливості транспортної задачі. Метод потенціалів ґрунтується на наступній теоремі.

Теорема 6.1. (Теорема про потенціали).

Якщо для деякого опорного плану $x^* = \left[x^*_{ij} \right], i = \overline{1,m}, \ j = \overline{1,n}$ транспортної задачі існують такі числа $\alpha_1,...,\alpha_m, \ \ \beta_1,...,\beta_n$, що $\alpha_i + \beta_j = c_{ij}$ при $x_{ij} > 0$, $\alpha_i + \beta_j \le c_{ij}$ при $x_{ij} = 0$ для всіх, $i = \overline{1,m}, \ j = \overline{1,n}$ то $x^* = \left[x^*_{ij} \right]$ – оптимальний план транспортної задачі.

Величини α_i та β_j називаються потенціалами пунктів постачання та пунктів споживання відповідно.

Алгоритм методу потенціалів включає наступні кроки.

Крок 1. Розраховуємо значення потенціалів рядків та стовпців α_i та β_j, розв'язуючи систему рівнянь $\alpha_i + \beta_j = c_{ij}$, що складені для заповнених клітинок транспортної задачі. Число заповнених клітинок дорівнює $n+m-1$, а система має $n+m$ невідомих, тому для визначеності приймаємо $\alpha_1 = 0$ та послідовно з рівнянь знаходимо інші значення α_i та β_j.

Крок 2. Для кожної з вільних клітинок розраховуємо значення $\alpha_{ij} = \beta_j + \alpha_i - c_{ij}$. Якщо $\forall \alpha_{ij} \le 0$, то план оптимальний. В іншому випадку обираємо максимальне значення α_{ij} та заповнюємо відповідну клітинку, змінюючи об'єми поставок в інших заповнених клітинках, що пов'язані з нею циклом.

а). Побудова циклу.

Цикл будується у вигляді ламаної лінії в таблиці транспортної задачі, вершини якого розташовані в заповнених клітинках таблиці, а ланки ламаної – вздовж рядків та стовпчиків, причому в кожній вершині циклу зустрічається рівно дві ланки – одна з них проходить вздовж стовпчика, а інша - вздовж рядка. Точки самоперетину ламаної не є вершинами. Однією з вершин циклу є обрана незаповнена клітинка.

б). Переміщення перевезень в межах клітинок, пов'язаних з вільною клітинкою, циклом.

Починаючи з вільної клітинки, якій приписується знак +, іншим вершинам почергово, просуваючись за циклом, приписуємо знаки – та +.

У вільну клітинку переносимо найменше з чисел x_{ij}, що знаходиться в мінусових клітинках. Це число додається до клітинок зі знаком + та віднімається від клітинок зі знаком – . В результаті вільна клітинка стає заповненою, а заповнена з мінімальним значенням та +-ом — вільною. Баланс перевезень не змінюється, оскільки в кожному стовпчику та рядку додається і віднімається одне й те ж значення.

Перехід до кроку 1.

Нижче наведені приклади можливих конфігурацій циклів у транспортній таблиці.

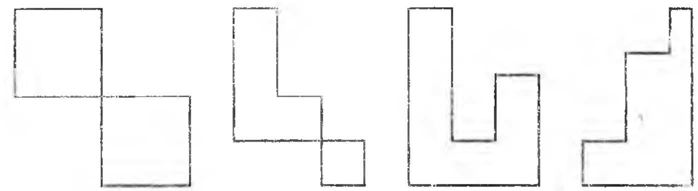


Рис.6.1. Можливі конфігурації циклів у транспортній таблиці

Розв'язування транспортних задач з ускладненнями в постановці.

Метод потенціалів може бути застосований також до розв'язування транспортних задач з додатковими умовами. Цими додатковими умовами, або ускладненнями в постановці, можуть бути наступні:

☑ заборона на перевезення продукту з конкретних пунктів зберігання в конкретні пункти споживання;

☑ перевезення продукту з конкретного пункту зберігання в конкретний пункт споживання лише строго визначеної кількості;

☑ завезення з конкретного пункту зберігання в конкретний пункт споживання не менше визначеної кількості продукту;

☑ завезення з конкретного пункту зберігання в конкретний пункт споживання не більше визначеної кількості продукту.

1. У тому випадку, коли поставки з A_i в B_j не можуть бути здійснені, відповідний тариф на перевезення $c_{ij} = M$, де M – дуже велике додатне число.

2. У тому випадку, коли з A_i в B_j необхідно перевезти точно d_{ij} одиниць продукту, це число записують у відповідну клітинку, яку надалі вважаємо вільною з тарифом M – якнайбільшим.

3. Якщо з пункту A_i до пункту B_j необхідно перевезти не менш ніж a_{ij} продукту, то вважаємо, що запаси A_i та потреби B_j зменшені на a_{ij} одиниць.

4. Якщо з пункту A_i до пункту B_j необхідно перевезти не більш ніж a_{ij} одиниць, то для кожного обмеження такого типу вводиться додатковий стовпчик: у стовпчику B_j потреби $= a_{ij}$, у додатковому стовпчику потреби $b_j - a_{ij}$, а вартість перевезення $c_{ij} = M$.

6.3. ІНТЕРПРЕТАЦІЯ МЕТОДУ ПОТЕНЦІАЛІВ ЯК СИМПЛЕКС-МЕТОДУ

Метод потенціалів є не чим іншим, як варіантом симплекс – методу, що орієнтований на максимальне врахування особливостей транспортної задачі. Метод потенціалів інтерпретується як симплекс – метод наступним чином:

☑ заповнені клітинки транспортної таблиці (де є перевезення) відповідають базовим змінним, а їх значення – значенням базових змінних, а не заповнені – небазовим змінним;

☑ знаходження клітинки, що буде заповнюватися, відповідає пошуку змінної, що вводитиметься до бази;

☑ знаходження клітинки в циклі, відміченої знаком "–" з найменшим значенням перевезення, відповідає пошуку змінної, що виключатиметься з бази;

☑ переміщення перевезення в межах циклу відповідає переходу до нової симплекс-таблиці в симплекс-методі;

☑ для включення в базу в симплекс-методі обирається змінна з найбільшим за абсолютною величиною від'ємним значенням Δ_j, тобто x_j, а в транспортній задачі – незаповнена клітинка з найбільшим додатним значенням потенціалу α_{ij} (транспортна задача – це задача на знаходження мінімуму);

☑ значення потенціалів незаповнених клітинок (небазових змінних) α_{ij} в транспортній задачі відповідають коефіцієнтам Δ_{ij} у Q-рядку симплекс-таблиці (за умови відповідної переіндексації змінних);

☑ потенціали рядків та потенціали стовпчиків у транспортній таблиці відповідають значенням змінних двоїстої задачі; знаючи значення двоїстих змінних, на кожній ітерації визначається α_{ij} як різниця між правою та лівою частиною відповідного обмеження двоїстої задачі;

☑ теорема про потенціали є не чим іншим, як видозміною теореми про доповнюючу нежорсткість (другої теореми двоїстості).

Розглянемо транспортну задачу з двома пунктами зберігання та трьома пунктами споживання в загальному вигляді:

$$
\begin{array}{l}
c_{11}x_{11} + c_{12}x_{12} + c_{13}x_{13} + c_{21}x_{21} + c_{22}x_{22} + c_{23}x_{23} \Rightarrow Min \\
x_{11} + x_{21} = b_1 \to \beta_1 \\
x_{12} + x_{22} = b_2 \to \beta_2 \\
x_{13} + x_{23} = b_3 \to \beta_3 \\
x_{11} + x_{12} + x_{13} = a_1 \to \alpha_1 \\
x_{21} + x_{22} + x_{23} = a_2 \to \alpha_2 .
\end{array}
\qquad (6.5)
$$

Позначимо двоїсті змінні, що відповідають обмеженням на пункти споживання (потреби кожного пункту споживання повинні бути задоволені), як β_j, а двоїсті змінні, що відповідають обмеженням на пункти зберігання (весь продукт, що знаходиться на кожному пункті зберігання, повинен бути перевезений), як α_i. Помножимо ліву та праву частину кожного з рівнянь прямої задачі на -1 та поміняємо знак у функції мети, щоб отримати канонічну форму задачі. Будуючи двоїсту до неї, отримаємо:

$$
\begin{array}{l}
a_1 \times \alpha_1 + a_2 \times \alpha_2 + b_1 \times \beta_1 + b_2 \times \beta_2 + b_3 \times \beta_3 \Rightarrow Min \\
\alpha_1 + \beta_1 \le c_{11} \\
\alpha_1 + \beta_2 \le c_{12} \\
\alpha_1 + \beta_3 \le c_{13} \\
\alpha_2 + \beta_1 \le c_{21} \\
\alpha_2 + \beta_2 \le c_{22} \\
\alpha_2 + \beta_3 \le c_{23} .
\end{array}
\qquad (6.6)
$$

Таким чином, для закритої транспортної задачі в загальному вигляді

$$Q(x) = \sum_{j=1}^{n}\sum_{i=1}^{m} c_{ij}x_{ij} \Rightarrow Min . \qquad (6.7)$$

$$\sum_{i=1}^{m} x_{ij} = b_j, \ \sum_{j=1}^{n} x_{ij} = a_i, \ \overline{1,m}, \ j = \overline{1,n}, \ x_{ij} \ge 0$$

двоїста буде наступною:

$$G(\alpha,\beta) = \sum_{i=1}^{m} a_i\alpha_i + \sum_{j=1}^{n} b_j\beta_j \Rightarrow Max, \quad \alpha_i + \beta_j \le c_{ij} \qquad (6.8)$$

$$i = \overline{1,m}, \ j = \overline{1,n} .$$

Позначимо $\alpha_{ij} = \alpha_i + \beta_j - c_{ij}$ та застосуємо результати теорем двоїстості. Дійсно, значення критеріїв оптимальності рівне різниці між лівою та правою частинами відповідного обмеження двоїстої задачі, і у випадку, коли змінна прямої задачі є базовою, значення $\alpha_{ij} = 0$, і розв'язок є оптимальним, коли $\forall \alpha_{ij} \le 0$, оскільки пряма задача є задачею мінімізації. Таким чином, якщо в якомусь розв'язку транспортної задачі для базових змінних x_{ij} – $\alpha_i + \beta_j = c_{ij}$, а для небазових – $\alpha_i + \beta_j \le c_{ij}$, то цей розв'язок є оптимальним – що й доводить теорему про потенціали. Таким чином, метод потенціалів є варіантом симплекс-методу, який враховує специфіку транспортної задачі і працює ефективно. Однак у випадку виродження розв'язування задачі ускладнюється.

Опорний план (базовий розв'язок) називається виродженим, якщо число заповнених клітинок k в таблиці менше за $m+n-1$, тобто рангу матриці обмежень транспортної задачі. Вироджений опорний план може виникнути як на початку розв'язку, коли вироджений початковий базовий розв'язок, так і при переході від одного до наступного опорного плану.

Якщо початковий опорний план вироджений, то обираємо деякі нульові елементи матриці обмежень (кількістю $m+n-1-k$) в якості базових, заміняємо їх на довільні, безмежно малі перевезення $\varepsilon > 0$ так, щоб при цьому не порушилась умова опорності (відсутність циклу з ненульових перевезень – тобто лише з базових змінних). Задачу розв'язуємо як невироджену, пишучи в оптимальному плані замість $\varepsilon > 0$ нулі.

Вироджений план може бути отриманий також у випадку, якщо до циклу входить не менш ніж два мінімальні елементи зі знаком "-" в клітинках. У цьому випадку вважають нульовим лише один з цих елементів, інші такі в процесі руху циклом заміняємо на $\varepsilon > 0$ та продовжуємо розв'я-зувати задачу як невироджену. Якщо на деякому кроці обраний для виведення з бази елемент зі значенням перевезення $\varepsilon > 0$, то значення функції мети не змінюється, а перевезення для елементу, що вводиться до бази, буде $\varepsilon > 0$.

6.4. МЕТОД ДИФЕРЕНЦІЙНИХ РЕНТ

Для уникнення незручностей, які виникають в методі потенціалів і пов'язані з виродженістю, застосовується метод диференційних рент.

Алгоритм методу диференційних рент складається з двох етапів:

☑ початковий розподіл частини продукту між пунктами призначення найкращим чином (побудова умовно оптимального розподілу);

☑ зменшення на наступних кроках величини нерозподілених поставок, щоб прийти до оптимального плану перевезень.

Алгоритм методу диференційних рент.

Крок 1. У кожному стовпчику транспортної таблиці знаходимо мінімальний тариф. Знайдені тарифи позначаємо колами, що їх оточують, і заповнюємо відповідні клітинки – записуємо максимально можливі перевезення. Таким чином, отримуємо розподіл, що в загальному може не задовільняти обмеженням транспортної задачі. Якщо розподіл задовольняє обмеженням задачі, то він оптимальний – **стоп**.

Крок 2. Скорочуємо нерозподілені поставки продукту так, щоб при цьому загальна вартість перевезень залишалась мінімальною:

☑ визначаємо надлишкові та недостатні рядки – рядок є недостатнім (**від'ємним**), якщо запаси відповідного пункту зберігання розподілені повністю, а потреби не задоволені; рядок є надлишковим (**додатним**), якщо потреби задоволені, і залишився продукт у відповідному пункті зберігання. Для уникнення неоднозначностей у визначенні знаку рядка, в якого нерозподілений залишок рівний 0, необхідно користатися наступним правилом: рядок вважається додатним за умови, що друга заповнена клітинка, яка знаходиться в стовпчику, що зв'язаний з таким рядком однією заповненою клітинкою, розташована в додатному рядку;

☑ для кожного стовпчика знаходимо різницю між тарифом у колі та найближчим до нього тарифом, який записаний в надлишковому рядку. Якщо тариф у колі знаходиться в позитивному (надлишковому) рядку, то різницю не визначаємо; серед різниць знаходимо найменшу – проміжну ренту;

☑ переходимо до нової таблиці – додаємо до відповідних тарифів, які знаходяться у від'ємних (недостатніх) рядках, проміжну ренту. Інші елементи не змінюємо.

Всі клітинки нової таблиці вважаємо вільними. Заповнюємо клітинки нової таблиці – до заповнення тепер буде на одну клітинку більше. Ця додаткова клітинка знаходиться в стовпчику, в якому записана проміжна рента. Всі інші клітинки знаходяться по одній в кожному зі стовпчиків і в них записані найменші числа в колах для кожного стовпчика. Так само в колах є два однакові числа, що знаходяться в стовпчику, в якому в попередній таблиці була записана проміжна рента.

Оскільки в новій таблиці число клітинок, що заповнюються, є більшим, ніж число стовпчиків, при заповненні клітинок користуємося наступним правилом. Обираємо деякий стовпчик (рядок), в якому наявна одна клітинка з колом. Цю клітинку заповнюємо і виключаємо з розгляду даний стовпчик (рядок). Продовжуючи цю процедуру, заповнюємо всі клітинки з колами. Якщо план припустимий, то він оптимальний - **стоп**, в іншому випадку переходимо до кроку 2.

Оскільки метод диференційних рент має простішу логічну структуру, ніж метод потенціалів, він і використовується частіше для програмних реалізацій.

6.5. ЗАДАЧА ПРО ПРИЗНАЧЕННЯ

Задача про призначення змістовно формулюється наступним чином. Необхідно розподілити m робіт серед n виконавців таким чином, щоб сумарні витрати на виконання робіт були мінімальні. Відомі витрати c_{ij} – на виконання i-ї роботи j-м виконавцем для всіх робіт та виконавців. Окрім того, відомо, що кожен з виконавців може виконувати не більш ніж одну роботу, і кожна робота може виконуватись не більш ніж одним виконавцем.

Ця задача є не чим іншим, як специфічним випадком транспортної – роботи можна розглядати, як пункти зберігання, виконавців – як пункти споживання, запаси в кожному з пунктів зберігання та потреби в пунктах споживання рівні одиниці, тариф на перевезення – c_{ij}.

В залежності від початкових умов для приведення задачі до задачі закритого типу додаємо фіктивні роботи або фіктивних виконавців і отримуємо задачу з n роботами та n виконавцями. Формальна постановка задачі виглядатиме наступним чином:

$$Q(x)=\sum_{j=1}^{n}\sum_{i=1}^{n} c_{ij}x_{ij} \Rightarrow Min \ , \qquad (6.9)$$

$$\sum_{i=1}^{n} x_{ij}=1,\ \sum_{j=1}^{n} x_{ij}=1,\ i=\overline{1,n},\ j=\overline{1,n},\ x_{ij}\in\{0;1\}$$

$x_{ij}=0$, якщо i-та робота не виконується j-м виконавцем, та 1 – якщо виконується. Слід зауважити, що розв'язок задачі про призначення завжди буде виродженим, оскільки для невиродженого розв'язку ранг матриці коефіцієнтів С повинен становити $2\times n-1$, в той час як призначеними можуть бути n робіт. Таким чином, задачу про призначення можна розв'язати за допомогою методів розв'язування транспортних задач, але при цьому буде необхідне заповнення в транспортній таблиці $2\times n-1-n=n-1$ порожніх клітинок безмежно малими перевезеннями ε.

В той же час виявилося можливим із врахуванням специфічних особливостей задачі про призначення побудувати ефективні алгоритми її розв'язування, одним з яких є так званий угорський алгоритм. Для обгрунтування редукції в цьому алгоритмі покажемо, що оптимальний розв'язок задачі не зміниться, якщо до будь-якого рядка чи будь-якого стовпчика додати чи відняти постійну величину. Віднімемо від кожного i-го стовпчика та кожного j-го рядка постійні для відповідного рядка чи стовпчика значення p_i та q_j:

$$Q(x)=\sum_{j=1}^{n}\sum_{i=1}^{n} c_{ij}x_{ij} \ .$$

$$Q'(x)=\sum_{j=1}^{n}\sum_{i=1}^{n}\left(c_{ij}-p_i-q_j\right)x_{ij}=\sum_{j=1}^{n}\sum_{i=1}^{n} c_{ij}x_{ij}-\sum_{i=1}^{n}p_i\sum_{j=1}^{n}x_{ij}-$$

$$-\sum_{j=1}^{n}q_j\sum_{i=1}^{n}x_{ij}=\sum_{j=1}^{n}\sum_{i=1}^{n}c_{ij}x_{ij}-\sum_{i=1}^{n}p_i-\sum_{j=1}^{n}q_j=Q(x)-const. \qquad (6.10)$$

Таким чином, ця процедура зменшує значення функції мети на постійну величину, що не приводить до зміни положення оптимального розв'язку.

Угорський алгоритм для задачі про призначення складається з наступних кроків.

Крок 1. Редукція рядків та стовпчиків матриці С.

Послідовно в кожному з рядків матриці визначаємо мінімальний елемент та зменшуємо на це значення всі інші елементи відповідного рядка. В отриманій таким чином матриці таку ж процедуру виконуємо з кожним стовпчиком. Метою цього кроку є отримання максимальної кількості нульових елементів в редукованій матриці.

Крок 2. Реалізація призначень.

a) Послідовно переглядаючи рядки, визначаємо ті з них, в яких знаходиться рівно по одному невикресленому нульовому елементу. В кожному з таких рядків виконуємо призначення, яке відповідає цьому невикресленому нульовому елементу (призначені нулі відзначаємо кружечками). При виконанні призначення в кожному стовпчику, який відповідає призначеному нулеві, викреслюємо всі невикреслені раніше елементи.

b) Послідовно переглядаємо стовпчики, визначаємо ті з них, в яких знаходиться рівно по одному нульовому елементу. В кожному з таких стовпчиків виконуємо призначення, що відповідає невикресленому нульовому елементу. При виконанні призначення в кожному рядкові, який відповідає призначеному нулеві, викреслюємо всі невикреслені раніше елементи.

c) Перевірка: Якщо знайдене таким чином призначення повне (тобто призначено n нульових елементів), то воно й буде оптимальним розв'язком задачі – **стоп**. Якщо деякі з нулів не викреслені (тобто залишилися рядки та стовпчики з кількістю нулів, більшою ніж 1), то обираємо рядок чи стовпчик з мінімальною кількістю нулів, довільно призначаємо один з них, викреслюємо всі нулі в цьому рядку (стовпчику) та відповідному стовпчикові (рядку) і далі виконуємо крок 2. Якщо викреслені та призначені всі нулі в матриці і знайдене призначення неповне, то переходимо до кроку 3.

Крок 3. Модифікація редукованої матриці.

З метою модифікації редукованої матриці без зміни положення оптимального розв'язку виконуємо наступні дії. Відміняємо всі призначення та викреслення:

a) обчислюємо число нулів у кожному невикресленому рядку та кожному невикресленому стовпчику;

b) викреслюємо рядок чи стовпчик з максимальною кількістю нулів;

c) виконуємо a та b до того часу, поки всі нулі не будуть викреслені;

d) від усіх невикреслених елементів віднімаємо значення мінімального невикресленого елементу та додаємо його до елементів, що знаходяться на перетині викреслюючих прямих. **Перехід до кроку 2.**

Задача про призначення може розглядатися ще й у такому змістовному формулюванні. На n вакантних місць претендують m осіб, причому задана ефективність виконання праці кожним з претендентів на кожній посаді. Необхідно обрати таких претендентів, щоб сумарна ефективність їх функціонування була максимальною. На відміну від попередньої ця задача є задачею максимізації, яка приводиться до вище розглянутої домноженням всіх елементів матриці ефективностей на -1 та додаванням значення максимального елемента матриці умови задачі.

ПРИКЛАДИ

Приклад 6.1. Пошук початкового опорного плану транспортної задачі.

Знайти початковий опорний план транспортної задачі методом північно-західного кута, методом мінімального елемента та евристичним методом Фойгеля і порівняти отримані результати для наступної транспортної задачі, що задана таблицею:

	B_1	B_2	B_3	B_4	Зап.
A_1	7	8	1	2	160
A_2	4	5	9	8	140
A_3	9	2	3	6	170
Потр.	120	50	190	110	470

Розв'язання.

а).Застосовуємо метод північно-західного кута:

	B_1	B_2	B_3	B_4	Зап.
A_1	7 120	8 40	1	2	160
A_2	4	5 10	9 130	8	140
A_3	9	2	3 60	6 110	170
Потр.	120	50	190	110	470

Сумарна вартість перевезень становитиме:

$$7\times120+8\times40+5\times10+9\times130+3\times60+6\times110=3220.$$

б).Застосовуємо метод мінімального елемента:

	B_1	B_2	B_3	B_4	Зап.
A_1	7	8	1 160	2	160
A_2	4 120	5	9	8 20	140
A_3	9	2 50	3 30	6 90	170
Потр.	120	50	190	110	470

Сумарна вартість перевезень становитиме:

$$1\times160+4\times120+8\times20+2\times50+3\times30+6\times90=1530.$$

в). Застосовуємо евристичний метод Фойгеля:

	B_1	B_2	B_3	B_4	Зап.				
A_1	7	8	1 50	2 110	160	1	6	-	-
A_2	4 120	5 20	9	8	140	1	1	1	1
A_3	9	2 30	3 140	6	170	1	1	1	7
Потр.	120	50	190	110	470				
	3	3	2	4					
	3	3	2	-					
	5	3	6	-					
	5	3	-	-					

Сумарна вартість перевезень становитиме:

$1\times 50+2\times 110+4\times 120+5\times 20+2\times 30+3\times 140=1330$.

Таким чином, найкращий початковий опорний план отриманий з використанням евристичного методу Фойгеля.

Приклад 6.2. Транспортна задача.

На трьох пунктах зберігання наявні наступні запаси: $a_1=100$, $a_2=150$, $a_3=50$. Потреби чотирьох пунктів споживання становлять $b_1=75$, $b_2=80$, $b_3=60$, $b_4=85$. Вартості перевезень одиниці продукції задані матрицею C,

$$C=\begin{bmatrix} 6 & 7 & 3 & 5 \\ 1 & 2 & 5 & 6 \\ 8 & 10 & 20 & 1 \end{bmatrix}.$$

Розв'язати задачу за допомогою методу потенціалів, для знаходження початкового опорного плану застосувати метод північно-західного кута.

Розв'язання.

Прямою перевіркою впевнюємося, що задача закритого типу, оскільки сумарні потреби рівні сумарним запасам. Застосуємо на першому кроці для знаходження початкового опорного розв'язку метод північно-західного кута.

	B1	B2	B3	B4	Зап.
A1	6 75	7 25	3	5	**100**
A2	1	2 55	5 60	6 35	**150**
A3	8	10	20	1 50	**50**
Потр.	**75**	**80**	**60**	**85**	**300**

	$\beta_1=6$	$\beta_2=7$	$\beta_3=10$	$\beta_4=11$
$\alpha_1=0$	6 75	7 25	3	5
$\alpha_2=-5$	1	2 55	5 60	6 35
$\alpha_3=-10$	8	10	20	1 50

Розраховуємо значення потенціалів рядків та стовпчиків, використовуючи для цього заповнені клітинки таблиці та присвоюємо значення потенціалу $\alpha_1=0$.

Заповнена клітинка з c_{11}, тому $\alpha_1+\beta_1=6$, $\beta_1=6$. На наступних кроках обираємо заповнену клітинку, для якої відомо значення одного потенціалу, а інший невідомий. Таким чином, далі обираємо c_{12}, $\alpha_1+\beta_2=7$, $\beta_2=7$.

Після цього c_{22}, $\alpha_2+\beta_2=2$, $\alpha_2=-5$. Наступна клітинка c_{23}, $\alpha_2+\beta_3=5$, $\beta_3=10$ і далі c_{24}, $\alpha_2+\beta_4=6$, $\beta_4=11$, і, нарешті, c_{34}, $\alpha_3+\beta_4=1$, $\alpha_3=-10$.

	6	7	10	11
0	6 75	7 − 25	3 + $\boxed{7}$	5 $\boxed{6}$
-5	1 $\boxed{0}$	2 + 55	5 − 60	6 35
-10	8 $\boxed{-12}$	10 $\boxed{-13}$	20 $\boxed{-20}$	1 50

Після цього розраховуємо потенціали незаповнених клітинок і, якщо оптимум не досягнутий, будуємо цикл. Оскільки серед потенціалів незаповнених клітинок є додатні, то знайдений опорний розв'язок не оптимальний – клітинка з найбільшим позитивним значенням потенціалу – 7 обирається як вершина циклу, а всі інші клітинки – вершини циклу повинні бути заповненими.

	6	0	3	4
0	6 75	7 $\boxed{-7}$	3 + 25	5 $\boxed{-1}$
2	1 + $\boxed{7}$	2 80	5 − 35	6 35
-3	8 $\boxed{-5}$	10 $\boxed{-13}$	20 $\boxed{-20}$	1 50

Здійснюємо переміщення перевезення 25 в межах циклу та переходимо до нової транспортної таблиці з повторенням розрахунків.

Процес побудови циклу повторюємо, так як оптимального розв'язку ще не досягнуто.

В процесі переходу від однієї таблички до іншої зберігається опорність знайденого перевезення та зменшується його сумарна вартість, тобто йде процес наближення до оптимуму. Переходимо до наступної таблиці.

	6	7	3	11
0	6 − 40	7 $\boxed{0}$	3 60	5 + $\boxed{6}$
-5	1 + 35	2 80	5 $\boxed{-7}$	6 − 35
-10	8 $\boxed{-12}$	10 $\boxed{-13}$	20 $\boxed{-17}$	1 50

Оскільки серед потенціалів незаповнених клітинок немає ні одного додатного, то знайдений план перевезень є оптимальним. Сумарна вартість перевезень для нього становитиме:

$6\times 5+3\times 60+5\times 35+1\times 70+2\times 80+$

$+1\times 50=665$.

	6	7	3	5
0	6 5	7 $\boxed{0}$	3 60	5 35
-5	1 70	2 80	5 $\boxed{-7}$	6 $\boxed{-6}$
-4	8 $\boxed{-6}$	10 $\boxed{-7}$	20 $\boxed{-21}$	1 50

Приклад 6.3. Метод диференційних рент.

Розв'язати наступну транспортну задачу за допомогою методу диференційних рент.

	B1	B2	B3	B4	B5	Запаси
A1	7	12	4	8	5	180
A2	1	8	6	5	3	350
A3	6	13	8	7	4	20
Потреби	110	90	120	80	150	550

Розв'язання.

Задача закритого типу, тому почнемо розв'язування за методом диференційних рент. Перейдемо до першої таблиці методу, додавши до первісної таблиці стовпчик для вказання надлишку та недостачі за рядками та рядок для запису відповідних різниць.

	B1	B2	B3	B4	B5	Запаси	Недостача (-), Надлишок (+)
A1	7	12	4 120	8	5	**180**	60
A2	1 110	8 90	6	5 80	3 70	**350**	-80
A3	6	13	8	7	4	**20**	20
Потреби	**110**	**90**	**120**	**80**	**150**	**550**	
Різниця	5	4	-	2	1		

У кожному зі стовпчиків цієї таблиці знаходимо мінімальні тарифи та беремо їх в коло. Заповнюємо клітинки, в яких знаходяться ці тарифи, записуючи в кожну з них максимально припустимі числа. Наприклад, в клітинку A_1B_3 записуємо 120 – більше значення записати неможливо, оскільки при цьому були б перевищені потреби B_3. В результаті такого заповнення отримуємо умовно оптимальний план перевезень, в якому повністю задоволені потреби пунктів призначення B_1, B_2, B_3, B_4 та частково – B_5. При цьому повністю розподілені запаси складу A_2, частково – A_1 і зовсім не розподілені запаси A_3.

Після визначення умовно оптимального плану визначаємо рядки з надлишками та недостачами. Рядок A_2 є з недостачею, оскільки запаси A_2 повністю використані, а потреби B_5 задоволені частково, значення недостачі становить 80. Рядки A_1 та A_3 є з надлишками, оскільки їх запаси розподілені не повністю, надлишок A_1 становить 60, A_3 – 20. Сума недостач

дорівнює сумі надлишків.

Наступним знаходимо для кожного зі стовпчиків різниці між мінімальними тарифами, що записані в рядках з надлишками, та тарифами, що знаходяться в заповнених клітинках. Для стовпчика B_3 різниця не визначена, оскільки число в колі цього стовпчика знаходиться в позитивному рядку. В стовпчику B_1 число в колі, рівне 1, а в рядках з надлишками в клітинках цього стовпчика найменшим є значення 6, тобто різниця для цього стовпчика становить 6-1=5. Аналогічно знаходиться різниця і для інших стовпчиків.

Обираємо найменшу з цих різниць – 1 (стовпчик B_5), яка й буде проміжною рентою, і переходимо до наступної таблиці:

	B1	B2	B3	B4	B5	Запаси
A1	7	12	4 120	8	5	**180**
A2	2 110	9 90	7	6 80	4 70	**350**
A3	6	13	8	7	4 20	**20**
Потреби	**110**	**90**	**120**	**80**	**150**	**550**
Різниця	5	3	-	2	1	

В рядки A_1 та A_3 цієї таблиці (з надлишками) переписуємо тарифи з попередньої таблиці, а до тарифів рядка A_2 додаємо значення проміжної ренти – 1. Число заповнених клітинок збільшилося на одну, так як кількість мінімальних тарифів, що знаходяться в кожному зі стовпчиків цієї таблиці, зросло на одиницю, а саме: в стовпчику B_5 знаходиться тепер два мінімальні елементи – 4. Заповнюємо ці клітинки, а також клітинки, в яких знаходяться мінімальні тарифи інших стовпчиків (в таблиці обведені кружечками). Спочатку встановлюємо послідовність їх заповнення, знаходимо стовпчики (рядки), в яких до заповнення є лише одна клітинка. Після її визначення та заповнення виключаємо з подальшого розгляду відповідний стовпчик (рядок) і переходимо до заповнення наступної клітинки. Таким чином заповнюємо послідовно клітинки A_1B_3, A_2B_1, A_2B_2, A_2B_4, оскільки вони є єдиними до заповнення в відповідних стовпчиках B.

Після заповнення цих клітинок заповнюємо клітинку A_3B_5, так як вона є єдиною до заповнення в рядку A_3, після чого виключаємо з розгляду рядок A_3, і в стовпчику B_5 залишається до заповнення одна клітинка A_1B_5, яку теж заповнюємо, після чого встановлюємо рядки з недостачами та надлишками. Як бачимо, залишився нерозподілений залишок, і таким чином отримуємо наступний умовно оптимальний план і переходимо до наступної таблиці. Знаходимо різниці та проміжну ренту як в попередньому випадку (проміжна рента рівна 1) і переходимо до наступної таблиці:

	B1	**B2**	**B3**	**B4**	**B5**	**Запаси**	**Недостача (-), Надлишок (+)**
A1	7	12	4 120	8	5 60	180	0
A2	3 110	10 90	8	7 80	5 70	350	0
A3	7	14	9	8	5 20	20	0

Внаслідок додавання до значень в рядках A_2 та A_3 (які є з недостачами) проміжної ренти 1 число клітинок для заповнення збільшилося ще на одну і стало 6. Послідовно заповнюємо клітинки A_1B_3, A_2B_1, A_2B_2, A_2B_4, потім A_3B_5, A_2B_5, A_1B_5. Таким чином, отриманий оптимальний план перевезень:

$$X^* = \begin{bmatrix} 0 & 0 & 120 & 0 & 60 \\ 110 & 90 & 0 & 80 & 70 \\ 0 & 0 & 0 & 0 & 20 \end{bmatrix}.$$

Вартість перевезень становить:

$$Q^* = 4\times120+5\times60+1\times110+8\times90+5\times80+3\times70+4\times20=2300.$$

Приклад 6.4. Задача про призначення.

Розв'язати задачу про призначення деталей для опрацювання на верстатах з наступною матрицею C за допомогою угорського методу.

$$C = \begin{pmatrix} 2 & 10 & 9 & 7 \\ 15 & 4 & 14 & 8 \\ 13 & 14 & 16 & 11 \\ 4 & 15 & 13 & 19 \end{pmatrix}$$

Розв'язання.

Редукуємо матрицю C за рядками:

$$\begin{pmatrix} 2 & 10 & 9 & 7 \\ 15 & 4 & 14 & 8 \\ 13 & 14 & 16 & 11 \\ 4 & 15 & 13 & 19 \end{pmatrix}\begin{pmatrix} -2 \\ -4 \\ -11 \\ -4 \end{pmatrix} \Rightarrow \begin{pmatrix} 0 & 8 & 2 & 5 \\ 11 & 0 & 5 & 4 \\ 2 & 3 & 0 & 0 \\ 0 & 11 & 4 & 15 \end{pmatrix}$$

та за стовпчиками: Матриця після кроку 2:

$$\begin{pmatrix} 0 & 8 & 7 & 5 \\ 11 & 0 & 10 & 4 \\ 2 & 3 & 5 & 0 \\ 0 & 11 & 9 & 15 \end{pmatrix} \Rightarrow \begin{pmatrix} 0 & 8 & 7 & 5 \\ 11 & 0 & 10 & 4 \\ 2 & 3 & 5 & 0 \\ 0 & 11 & 9 & 15 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & -5 & 0 \end{pmatrix}$$

Здійснюємо призначення, послідовно переглядаючи рядки та стовпчики редукованої матриці з призначенням нулів та їх викреслюванням згідно з алгоритмом. Таким чином, у результаті отримаємо наступну матрицю призначень:

$$\begin{matrix} [0] & 8 & 2 & 5 \\ 11 & [0] & 5 & 4 \\ 2 & 3 & [0] & \cancel{0} \\ \cancel{0} & 11 & 4 & 5 \end{matrix}.$$

Оскільки призначення неповне (матриця розміром 4×4, а призначень не 4, а лише 3), переходимо до кроку 3 алгоритму.

Після завершення викреслювань (згідно з алгоритмом угорського методу викреслюємо нулі матриці мінімальною кількістю прямих) визначаємо зміни в елементах матриці. Визначаємо мінімальний серед невикреслених елементів – 2, його значення додаємо до значень елементів, що знаходяться на перетині викреслюючих прямих і віднімаємо від значень невикреслених елементів:

0	8(−2)	2(−2)	5(−2)
11(+2)	0	5	4
2(+2)	3	0	0
0	11(−2)	4(−2)	15(−2)

Формуємо модифіковану матрицю та виконуємо крок 2 алгоритму (призначення та викреслювання нульових елементів)з модифікованою матрицею.

$$\begin{pmatrix} 0 & 6 & 0 & 3 \\ 13 & 0 & 5 & 4 \\ 4 & 3 & 0 & 0 \\ 0 & 9 & 2 & 13 \end{pmatrix} \begin{pmatrix} \cancel{0} & 6 & \boxed{0} & 3 \\ 13 & \boxed{0} & 5 & 4 \\ 4 & 3 & \cancel{0} & \boxed{0} \\ \boxed{0} & 9 & 2 & 13 \end{pmatrix}.$$

Таким чином, отримане повне призначення, яке й є оптимальним розв'язком задачі про призначення. Повертаючись до умови задачі, розраховуємо оптимальне значення функції мети для призначення:

9+4+11+4=28.

РЕЗЮМЕ

6.1. Для відкритої транспортної задачі характерним є те, що сумарні запаси та сумарні потреби є незбалансованими. Відкрита задача завжди приводиться до закритої. Основними властивостями закритої транспортної задачі є наступні: закрита транспортна задача завжди має припустимий розв'язок, ранг матриці системи обмежень-рівностей закритої невиродженої транспортної задачі дорівнює $r(A) = m + n - 1$; якщо значення всіх коефіцієнтів a_i, b_j в транспортній задачі закритого типу є цілими числами, то й оптимальний розв'язок є цілочисельним. Алгоритм розв'язування транспортної задачі складається з двох основних етапів: знаходження початкового опорного плану транспортної задачі (початкового базового розв'язку); покрокове покращення плану перевезень до моменту досягнення оптимального. Для знаходження початкового базового розв'язку використовуються методи: північно-західного кута, мінімального елемента, евристичний метод Фойгеля.

6.2. Метод потенціалів є особливим випадком симплекс-методу, що враховує особливості транспортної задачі. Метод потенціалів ґрунтується на теоремі про потенціали. Метод потенціалів може бути застосований також до розв'язування транспортних задач з додатковими умовами. Цими додатковими умовами, або ускладненнями в постановці, можуть бути наступні: заборона на перевезення продукту з конкретних пунктів зберігання в конкретні пункти споживання; перевезення продукту з конкретного пункту зберігання в конкретний пункт споживання лише строго визначеної кількості; завезення з конкретного пункту зберігання в конкретний пункт споживання не менше визначеної кількості продукту; завезення з конкретного пункту зберігання в конкретний пункт споживання не більше визначеної кількості продукту.

6.3. Метод потенціалів є не чим іншим, як варіантом симплекс – методу, що орієнтований на максимальне врахування особливостей транспортної задачі. Теорема про потенціали є не чим іншим, як видозміною теореми про доповнюючу нежорсткість (другої теореми двоїстості).

6.4. Для уникнення незручностей, які виникають в методі потенціалів і пов'язані з виродженістю, застосовується метод диференційних рент. Оскільки метод диференційних рент має простішу логічну структуру, ніж метод потенціалів, він і використовується частіше для програмних реалізацій.

6.5. Задача про призначення є не чим іншим, як специфічним випадком транспортної – роботи можна розглядати, як пункти зберігання виконавців – як пункти споживання, запаси в кожному з пунктів зберігання та потреби в пунктах споживання рівні одиниці, тариф на перевезення c_{ij}. Розв'язок задачі про призначення завжди буде виродженим, оскільки для невиродженого розв'язку ранг матриці коефіцієнтів С повинен становити $2 \times n - 1$, в той час як призначеними можуть бути n робіт. В той же час виявилося можливим з врахуванням специфічних особливостей задачі про призначення побудувати ефективні алгоритми її розв'язування, одним з яких є так званий угорський алгоритм.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 6.1. Метод потенціалів.

Розв'язати приклад 6.2 за допомогою методу потенціалів, використовуючи для побудови опорного плану а)метод мінімального елементу; б)евристичний метод Фойгеля. Порівняти кількість ітерацій при застосуванні цих методів та методу північно-західного кута.

Завдання 6.2. Відкрита транспортна задача.

Розв'язати наступну транспортну задачу за допомогою методу потенціалів та методу диференційних рент. Запаси продукту на 3-х складах та потреби в 4-х пунктах споживання, а також тарифи наведені в таблиці:

	B1	B2	B3	B4	Запаси
A1	7	9	4	8	180
A2	1	8	6	5	350
A3	6	4	8	7	20
Потреби	110	90	100	80	

Завдання 6.3. Транспортна задача з ускладненнями в постановці.

Розв'яжіть наступну транспортну задачу за умов, що з A1 в B2 повинно бути перевезено не менш 50 одиниць продукту, з A3 в B5 – не менш 60, з A2 в B4 не більш ніж 40.

	B1	B2	B3	B4	B5	Запаси
A1	5	3	2	4	8	160
A2	7	6	5	3	1	90
A3	8	9	4	5	2	140
Потреби	90	60	80	70	90	390

Завдання 6.4. Задача про призначення.

Деталі потрібно призначити на верстати таким чином, щоб сумарна вартість опрацювання була мінімальною. Вартості опрацювання деталей на верстатах задані матрицею, рядки матриці відповідають деталям.

5	12	6	4	8
1	7	2	6	3
16	14	9	10	2
13	18	21	24	30
15	20	23	25	31

Завдання 6.5. Задача про призначення.

Необхідно обрати претендентів на робочі місця, щоб сумарна ефективність їх була максимальною. Ефективності задані таблицею, стовпці - робочі місця, рядки - преденденти.

25	12	6	4	8
11	7	12	6	3
16	14	9	10	22
13	18	21	24	30
15	10	13	25	17

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Дайте змістовне формулювання транспортної задачі.
2. Наведіть формальну постановку транспортної задачі.
3. Чим відрізняється транспортна задача закритого типу від відкритої?
4. Наведіть основні властивості транспортної задачі та розкрийте їх значення.
5. Які методи використовуються для знаходження початкового опорного плану транспортної задачі та чим вони відрізняються один від одного?
6. Доведіть теорему про потенціали.
7. Які основні кроки включає алгоритм методу потенціалів?
8. За якими правилами будується цикл в таблиці транспортної задачі?
9. Яким чином розв'язуються транспортні задачі з ускладненнями в постановці?
10. Як інтерпретується метод потенціалів у термінах симплекс-методу?
11. Проаналізуйте пряму та двоїсту транспортну задачу.
12. Порівняйте між собою метод диференційних рент та метод потенціалів.
13. Яким чином усувається виродження при розв'язуванні транспортної задачі?
14. Поясніть основні кроки алгоритму диференційних рент.
15. Наведіть формальну та змістовну постановку задачі про призначення.
16. Поясніть основні кроки алгоритму угорського методу

ТЕМА 7. ЗАДАЧІ ОПТИМІЗАЦІЇ НА МЕРЕЖАХ

Задачі потокового типу дуже часто виникають на практиці – в багатьох випадках потрібно визначити множину найкоротших шляхів, мінімізувати мережу, розрахувати пропускну спроможність водогінної системи та „вузьке місце" в ній. Задачі потокового типу можуть бути сформульовані як задачі лінійного програмування та розв'язані за допомогою симплекс-методу, але ця можливість є лише принциповою, тому що розмірність відповідних задач лінійного програмування буде надзвичайно великою, і матриці коефіцієнтів обмежень дуже розрідженими. Водночас спеціальна структура окремих класів потокових задач дозволяє побудувати ефективні алгоритми їх розв'язування. Теоретичним фундаментом для потокових задач є теорема Форда-Фалкерсона, а також властивість унімодулярності матриці коефіцієнтів обмежень, що дозволяє, з одного боку, побудувати ефективний алгоритм пошуку максимального потоку та мінімального розрізу – власне того „вузького місця" в мережі, а також забезпечити цілочисельність розв'язку за умови цілочисельності значень параметрів задачі. Загальна постановка задачі потокового типу дозволяє краще зрозуміти основні типи потокових задач та взаємні зв'язки між ними. Знання теоретичних основ та володіння потоковими алгоритмами дозволяє ефективно розв'язувати різноманітні практичні задачі.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати: ⇨ *теорему Форда-Фалкерсона як теоретичну основу для розв'язування потокових задач;* ⇨ *загальну постановку задачі потокового типу та її окремі реалізації;* ⇨ *алгоритми розв'язання потокових задач та особливості їх побудови;* ⇨ *узагальнення задачі про максимальний потік;*

вміти: ⇨ *здійснювати формальну постановку практичних ситуацій у вигляді потокових задач;* ⇨ *вміти розв'язувати задачі пошуку найкоротших шляхів, мінімізації мережі, пошуку максимального потоку та мінімального розрізу в мережі;* ⇨ *приводити узагальнені потокові задачі до класичного вигляду та розв'язувати їх за допомогою алгоритмів розташування позначок і Гоморі-Ху.*

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ потік
- ☑ потокові задачі
- ☑ зв'язна мережа
- ☑ джерело
- ☑ витік
- ☑ мінімізація мережі
- ☑ пропускна здатність розрізу
- ☑ максимальний потік
- ☑ мінімальний розріз
- ☑ теорема Форда-Фалкерсона
- ☑ однопродуктовий потік
- ☑ унімодулярність
- ☑ позначка
- ☑ аугментальний шлях
- ☑ задача аналізу мережі
- ☑ задача синтезу мережі
- ☑ дерево розрізів
- ☑ найкоротший шлях

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

7.1. *Поняття потоку. Теорема Форда-Фалкерсона. Загальна постановка та часткові випадки потокових задач.*

7.2. *Задачі пошуку найкоротшого маршруту в мережі. Алгоритм Дійкстри. Задача мінімізації мережі.*

7.3. *Задача про багатополюсний найкоротший ланцюг. Алгоритм Флойда.*

7.4. *Задача пошуку максимального потоку. Алгоритм розташування позначок. Узагальнення задачі про максимальний потік.*

7.1. ПОНЯТТЯ ПОТОКУ. ТЕОРЕМА ФОРДА-ФАЛКЕРСОНА. ЗАГАЛЬНА ПОСТАНОВКА ТА ЧАСТКОВІ ВИПАДКИ ПОТОКОВИХ ЗАДАЧ

Задачі потокового типу можуть бути сформульовані як задачі лінійного програмування та розв'язані за допомогою симплекс-методу, але ця можливість є лише принциповою, тому що в цьому випадку розмірність відповідних задач лінійного програмування буде надзвичайно великою, і матриці коефіцієнтів обмежень дуже розрідженими – тобто кількість нульових елементів досягатиме 90-98%. Водночас спеціальна структура окремих класів потокових задач дозволяє побудувати ефективні алгоритми їх розв'язування.

Будемо розглядати мережу $G=(N,A)$ як двійку, де N - множина вузлів, що зв'язані між собою дугами з множини A. Якщо дуги є спрямованими, то мережа — орієнтована. Мережу називатимемо зв'язною, якщо для довільного розбиття множини вузлів мережі N на дві підмножини X та $\overline{X}$ ($N=X\bigcup\overline{X}$, $X\bigcap\overline{X}=\varnothing$) знайдеться щонайменше одна дуга $(i,j)\in N$, така що $i\in X$ та $j\in\overline{X}$ або $i\in\overline{X}$ та $j\in X$.

В потокових задачах розглядаються лише зв'язні мережі. Розв'язання потокової задачі полягає в знаходженні таких значень функції потоку для всіх дуг мережі, що протікає від джерела до витоку мережі і забезпечує екстремальне значення функції мети. Джерело t – це вузол мережі, в який може втікати потік із зовнішнього середовища. Витік s – це вузол мережі, з якого потік може витікати в зовнішнє середовище.

Потоком в орієнтованій мережі називається цілочисельна функція f_{ij}, визначена на множині дуг мережі, для якої виконуються наступні умови:

$$\forall((i,j)\in A): f_{ij}\geq 0 \tag{7.1}$$

$$\forall((i\in N)\wedge(i\neq s)\wedge(j\neq t)): \sum_{j\in\alpha_i} f_{ij}-\sum_{j\in\beta_i} f_{ji}=0 \tag{7.2}$$

$$\forall((i,j)\in A): f_{ij}\leq u_{ij} \tag{7.3}$$

де β_i – множина всіх вузлів, що зв'язані безпосередньо з вузлом i дугами, спрямованими до нього; α_i — множина всіх вузлів, що зв'язані безпосередньо з вузлом i дугами, що спрямовані від нього; u_{ij} – пропускна здатність дуги $(i,j)\in A$.

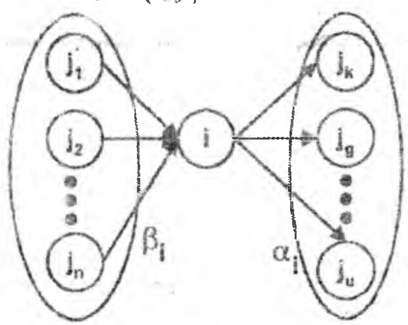


Рис. 7.1. Взаємозв'язки сусідніх вузлів

Таким чином, функція потоку, визначена на кожній дузі мережі, є невід'ємною, значення якої не перевищує пропускної здатності дуг (7.1, 7.3). Умова (7.2) є умовою збереження потоку – який потік втікає в будь-який проміжний вузол мережі, такий і витікає.

$s-t$ розрізом називається розбиття множини N вузлів мережі на дві підмножини X та $\overline{X}$ такі, що $s\in X$ та $t\in\overline{X}$. Пропускна здатність розрізу $U(X,\overline{X})$ дорівнює

$$U(X,\overline{X})=\sum_{(i,j)\in A\wedge i\in X\wedge j\in\overline{X}} u_{ij}, \tag{7.4}$$

тобто є сумою пропускних здатностей "прямих" дуг, що йдуть з вершин множини X до вершин множини $\overline{X}$.

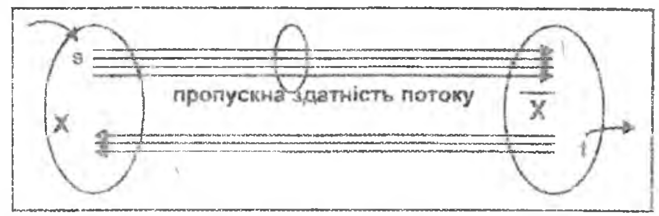


Рис. 7.2. Пропускна здатність потоку

Максимальний потік в мережі визначається за допомогою пропускних здатностей розрізів мережі. Звичайно, величина максимального потоку не повинна перевищувати величину пропускної здатності розрізу. Вимога орієнтованості мережі не є обмежуючою, тому що завжди неорієнтовану дугу з пропускною здатністю u_{ij} можна замінити парою протилежно спрямованих орієнтованих дуг, кожна з яких матиме пропускну здатність u_{ij}. Наступна теорема є теоремою про максимальний потік в мережі.

Теорема 7.1. (Теорема Форда-Фалкерсона).

Величина потоку з джерела s до витоку t в орієнтованій мережі не перевищує пропускної здатності $U(X,\overline{X})$ будь-якого $s-t$ розрізу. Величина максимального потоку дорівнює пропускній здатності мінімального розрізу, тобто і потік V, і розріз $U(X,\overline{X})$ одночасно оптимальні в тому і лише в тому випадку, якщо

$f_{ij}=0$ для всіх дуг $(i,j)\in A$, таких що $i\in\overline{X}$, $j\in X$ та

$f_{ij}=u_{ij}$ для всіх дуг $(i,j)\in A$, таких що $i\in X$, $j\in\overline{X}$.

Доведення.

Для довільного розрізу $(X,\overline{X})$ справедлива рівність

$$F = \sum_{i\in X}\left(\sum_{j\in N} f_{ij} - \sum_{j\in N} f_{ji}\right), (i,j)\subset A,\ (j,i)\subset A, \qquad (7.5)$$

де F – деякий припустимий потік з s в t. Так як $N = X\bigcup\overline{X}$ і вузол не може одночасно належати до двох підмножин X та $\overline{X}$, то

$$F = \sum_{(i\in X)\wedge(j\in X)} f_{ij} + \sum_{(i\in X)\wedge(j\in \overline{X})} f_{ij} - \sum_{(i\in X)\wedge(j\in X)} f_{ji} - \sum_{(i\in X)\wedge(j\in \overline{X})} f_{ji}, \qquad (7.6)$$

тобто

$$F = \left(\sum_{(i\in X)\wedge(j\in \overline{X})} f_{ij} - \sum_{(i\in X)\wedge(j\in \overline{X})} f_{ji}\right) + \left(\sum_{(i\in X)\wedge(j\in X)} f_{ij} - \sum_{(i\in X)\wedge(j\in X)} f_{ji}\right) \qquad (7.7)$$

Оскільки друга складова рівна нулю, $F = \left(\sum_{(i\in X)\wedge(j\in \overline{X})} f_{ij} - \sum_{(i\in X)\wedge(j\in \overline{X})} f_{ji}\right)$,

і як наслідок $F \le \sum_{(i\in X)\wedge(j\in \overline{X})} f_{ij}$, тому що $f_{ji} \ge 0$. Таким чином, значення F довільного потоку обмежене пропускною здатністю довільного розрізу $\left(X, \overline{X}\right)$, звідки значення максимального припустимого потоку обмежене згори пропускною здатністю мінімального розрізу, тобто, враховуючи $f_{ij} \le u_{ij}$ отримаємо:

$$F \le U(X,\overline{X}) = \sum_{(i,j)\in A\wedge i\in X\wedge j\in \overline{X}} u_{ij}. \qquad (7.8)$$

Нехай $\left(X, \overline{X}\right)$ – розріз, такий що $s\in X$, $t\in \overline{X}$ та $j\in X$, якщо знайдеться таке $i\in X$, для якого $f_{ij} < u_{ij}$ або $f_{ji} > 0$.

Мінімальний розріз будується за допомогою наступної процедури:

☑ визначаємо множину $X = \{s\}$;

☑ якщо для якогось $i\in X$ знайдеться таке $j\notin X$, $(i,j)\in A$, що $f_{ij} < u_{ij}$ або $f_{ji} > 0$, то j включаємо в X;

☑ включення повторюємо до моменту, поки X неможливо буде розширити далі.

В результаті $t\in \overline{X}$, так як в іншому випадку існував би збільшуючий (аугментальний) шлях з s в t, що суперечить припущенню про те, що потік максимальний.

Значення $F_{\max}$ можна розрахувати за отриманою вище формулою:

$$F = \left(\sum_{(i\in X)\wedge(j\in \overline{X})} f_{ij} - \sum_{(i\in X)\wedge(j\in \overline{X})} f_{ji}\right), (i,j)\in A.$$

З процедури побудови множини X (перерізу) витікає, що:

☑ якщо $i\in X$, $j\in \overline{X}$, $(i,j)\in A$, то $f_{ij} = u_{ij}$;

☑ якщо $i\in X$, $j\in \overline{X}$, $(j,i)\in A$, то $f_{ji} = 0$.

Внаслідок цього $\sum_{(i,j)\in A\wedge i\in X\wedge j\in \overline{X}} f_{ij} = \sum_{(i,j)\in A\wedge i\in X\wedge j\in \overline{X}} u_{ij}$ та

$$\sum_{(i,j)\in A\wedge i\in X\wedge j\in \overline{X}} f_{ji} = 0. \qquad (7.9)$$

Тому $F = \sum_{(i,j)\in A\wedge i\in X\wedge j\in \overline{X}} u_{ij} \ge U(X,\overline{X})$. Оскільки $U(X,\overline{X}) \ge F \ge U(X,\overline{X})$, то $F = U(X,\overline{X})$ ■

Для кращого розуміння типів потокових задач та взаємних зв'язків між ними побудуємо загальну постановку задачі потокового типу на розглянемо часткові її випадки.

Розглянемо мережу, в якій існує α джерел, β витоків та φ проміжних вузлів. Однопродуктовий потік повинен протікати з джерел у витоки через проміжні вузли. Характеристиками кожної дуги мережі $(i,j)\in A$ є пропускна здатність u_{ij} та вартість доставки нею одиниці потоку c_{ij}.

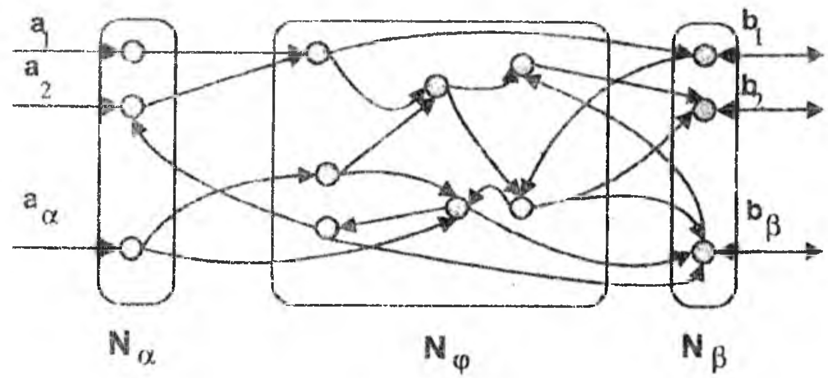


Рис. 7.3. Загальна потокова задача

Необхідно визначити потік мінімальної вартості з джерел у витоки за умови, що наявно не більш ніж $\sum_{i\in N_\alpha} a_i$ одиниць продукту, а в витоки повинно бути доставлено не менш ніж $\sum_{i\in N_\beta} b_i$ одиниць продукту, де $N_\alpha, N_\beta, N_\varphi$ множини джерел, витоків та проміжних вузлів відповідно, a_i – кількість продукту в джерелі α_i, b_j – потреби в продукті в витоку β_j.

Формальна постановка задачі про максимальний потік мінімальної вартості наступна:

$$\sum_i \sum_j c_{ij} f_{ij} \Rightarrow Min \quad , \tag{7.10}$$

$$\sum_j f_{ij} - \sum_j f_{ji} \le a_i, \ \ i \in N_\alpha \ , \tag{7.11}$$

$$\sum_j f_{ij} - \sum_j f_{ji} = 0, \ \ i \in N_\varphi \ , \tag{7.12}$$

$$\sum_j f_{ji} - \sum_j f_{ij} \ge b_i, \ \ i \in N_\beta \ , \tag{7.13}$$

$$0 \le f_{ij} \le u_{ij}, \qquad (i,j) \in A. \tag{7.14}$$

Нерівності групи (7.11) відображають той факт, що в мережу з джерела не може надходити більша кількість продукту, ніж наявна. (7.12) – умови збереження потоку для кожного з проміжних вузлів, (7.13) – попит на продукт у витоках повинен бути задоволений, (7.14) – потоки дугами є обмеженими пропускними здатностями дуг.

Окремими випадками цієї задачі є відомі задачі – транспортна, задача про призначення, задача про максимальний потік, задача про найкоротший ланцюг.

Випадок 1. Транспортна задача.

Потік тече безпосередньо з джерел до витоків без проміжних пунктів: $N_\varphi = \varnothing, \ \ u_{ij} = \infty$.

Випадок 2. Задача про призначення.

Потік тече безпосередньо з джерел до витоків без проміжних пунктів: $N_\varphi = \varnothing, \ \ u_{ij} = \infty$. Крім того, $\forall i \in N_\alpha : a_i = 1$ та $\forall i \in N_\beta : b_i = 1$.

Випадок 3. Задача про максимальний потік.

В мережі є лише одне джерело s та витік t, тобто $card(N_\alpha) = card(N_\beta) = 1$. Кількість продукту, що надходить в мережу через джерело, не обмежена,

а мінімальні потреби дорівнюють нулю – $a = \infty$, $b = 0$. Пропускні здатності дуг обмежені значеннями u_{ij} вартості пересилання одиниці потоку дугами становлять $\forall((i,j)\in A)\wedge(j \ne t): c_{ij} = 0$, $\forall((i,t)\in A): c_{it} = -1$. У результаті отримаємо задачу максимізації значення сумарного потоку, що надходить з мережі до витоку:

$$\sum_{(i,t)\in A} f_{it} \Rightarrow Max \ , \ \sum_j f_{ij} - \sum_j f_{ji} = 0, \ \ i \ne s, \ \ i \ne t \tag{7.15}$$

$$0 \le f_{ij} \le u_{ij}, \qquad (i,j) \in A.$$

Випадок 4. Задача про найкоротший ланцюг.

Вважатимемо, що c_{ij} — вартість передачі одиниці потоку з вузла i в вузол j. Нехай $card(N_\alpha) = card(N_\beta) = 1$, тобто в мережі є лише одне джерело та один витік, в мережу з джерела надходить $a = 1$ потоку, попит у витоку становить $b = 1$. Потік будь-якою дугою не може бути більшим за 1. Мережа не має контурів від'ємної ваги. Задача знаходження потоку мінімальної вартості в такому випадку означатиме не що інше, як знаходження найкоротшого шляху з джерела до витоку – c_{ij} в цьому випадку розглядатимемо як віддалі — довжини дуг.

$$\sum_i \sum_j c_{ij} f_{ij} \Rightarrow Min, \quad \sum_j f_{sj} - \sum_j f_{js} = 1, \quad \sum_j f_{tj} - \sum_j f_{jt} = -1, \tag{7.16}$$

$$\sum_j f_{ij} - \sum_j f_{ji} = 0, \ \ i \ne s, \ i \ne t, \ \ (i,j) \in A \ .$$

Ці задачі мають одну дуже важливу особливість: якщо значення a_i, b_j, u_{ij} – цілі, то будь-який базовий розв'язок загальної задачі про максимальний потік мінімальної вартості є цілочисельним. Розглянемо наступні визначення.

Визначення 1. Цілочисельна квадратна матриця, детермінант якої дорівнює +1, -1 або 0, називається унімодулярною.

Визначення 2. Цілочисельна матриця називається абсолютно унімодулярною, якщо всі її квадратні підматриці унімодулярні.

Цілочисельність базових розв'язків визначається наступними теоремами, які ми наводимо без доведення.

Теорема 1. Якщо матриця D абсолютно унімодулярна, то будь-який базовий розв'язок системи $Df = B$, $f \ge 0$ є цілочисельним.

Теорема 2. Якщо цілочисельна матриця D є абсолютно унімодулярною, то всі базові розв'язки задачі лінійного програмування з цілими коефіцієнтами $Df \le B$, $f \ge 0$ є цілочисельними, і навпаки, якщо всі базові розв'язки задачі лінійного програмування з цілими коефіцієнтами

$Df \le B, f \ge 0$ є цілочисельними, то матриця D є абсолютно унімодулярною.

7.2. ЗАДАЧА ПОШУКУ НАЙКОРОТШОГО МАРШРУТУ В МЕРЕЖІ. АЛГОРИТМ ДІЙКСТРИ. ЗАДАЧА МІНІМІЗАЦІЇ МЕРЕЖІ

Розглянемо задачу пошуку найкоротшого маршруту в мережі з невід'ємними дугами. Задана мережа з одним джерелом s та одним витоком t, а також вартості доставки одиниці продукту $c_{ij} \ge 0$ для кожної з дуг мережі $(i, j) \in A$. Необхідно передати з джерела до витоку одиничний потік мінімальної вартості. Ця задача еквівалентна задачі пошуку найкоротшого шляху з джерела до витоку. Алгоритм пошуку найкоротшого шляху, що запропонований Дійкстрою, використовує наступну ідею.

Замість пошуку найкоротшого маршруту з s в t шукаються найкоротші маршрути з s у всі інші вершини мережі, так як будь-яка вершина i може виявитися деякою проміжною вершиною найкоротшого ланцюга з s в t. Якщо вершина i належить найкоротшому ланцюгу з s в t, то частина цього ланцюга з s в i є найкоротшим ланцюгом з s в i. В іншому випадку вказану частину ланцюга з s в i можна було б замінити на найкоротшу, при цьому можна було б отримати коротший ланцюг з s в t, що суперечило б первісному припущенню. Таким чином, по суті, будується дерево найкоротших шляхів з s у всі вершини мережі. При пошуку найкоротшого шляху з s в t роботу алгоритму можна припинити, коли буде досягнута вершина t.

Розглянемо алгоритм Дійкстри. Перед виконанням алгоритму всі вершини та дуги мережі не переглянуті. Кожній вершині при роботі алгоритму присвоюється значення $d(x)$, що рівне найкоротшій відстані з s в t, що включає лише переглянуті вершини.

Алгоритм Дійкстри.

Крок 1. Початкові присвоювання. Присвоїмо $d(s)=0$ та $d(x)=\infty$ для всіх $x \ne s$. Позначимо вершину s та присвоїмо $y=s$, де y остання з переглянутих вершин.

Крок 2. Для кожної з непереглянутих вершин x перераховуємо значення $d(x)$ за формулою: $d(x)= \underset{(y,x)\in A}{Min}\{d(x); d(y)+c(y,x)\}$, де $c(y,x)$ – довжина дуги (y,x). Якщо $d(x)=\infty$ для всіх непереглянутих вершин x, то стоп: в графі відсутні шляхи з s в непереглянуті вершини, тобто шлях з s в t відсутній.

Крок 3. Позначаємо як переглянуту ту з непереглянутих вершин x, для якої $d(x)=Min$. Позначаємо дугу, що веде до обраної вершини, та присвоюємо $y=x$.

Крок 4. Якщо $y=t$, то стоп: найкоротший шлях з s в t знайдений. Перехід до кроку 2.

Для побудови дерева найкоротших шляхів з s у всі інші вершини робота алгоритму продовжується, поки всі вершини не будуть позначені.

Задача мінімізації мережі.

Задача мінімізації мережі є задачею побудови найкоротшого дерева-основи (остову). Дерево – це зв'язна множина неорієнтованих ребер, що не має циклів. Кожен з вузлів дерева пов'язаний з іншим єдиним шляхом. В дереві з m вершинами є (m-1)-е ребро. Розглянемо мережу, що має n вершин. Деревом-основою мережі називається зв'язна множина, що складається з n-1 ребер та n вершин. Якщо задані віддалі між вершинами, то мінімальне дерево-основа — це така основа, для якої сума віддалей між вузлами є мінімальною.

Задача про побудову мінімального дерева-основи має важливе практичне значення — яким чином організувати зв'язок, щоб прокласти найменше кабелю і зв'язати всі пункти, яким чином побудувати систему розподілу природного газу, щоб зв'язати всіх споживачів з мінімальними витратами труб.

Задача мінімізації мережі є однією з небагатьох, яка може бути ефективно розв'язана за допомогою алгоритму "пожираючого" типу.

Алгоритм мінімізації мережі.

Крок 1. Визначаємо розбиття множини вузлів мережі N на дві підмножини: S – множину з'єднаних вершин та $\bar{S}$ – множини не з'єднаних вершин дугами дерева-основи. Початково всі вершини мережі належать до множини $\bar{S}$. Обираємо довільну вершину з $\bar{S}$ та з'єднаємо її з найближчою сусідньою. Видалимо ці вершини з $\bar{S}$ та приєднаємо їх до множини S з найкоротшою дугою.

Крок 2. Знаходимо найкоротшу дугу $(i, j)\in A, i\in S, j\in \bar{S}$ серед всіх, що з'єднують дуги множини S з дугами множини $\bar{S}$. Видаляємо вузол j з множини $\bar{S}$ та приєднуємо його до S разом з найкоротшою дугою.

Крок 3. Якщо $\bar{S} \ne \varnothing$, переходимо до кроку 2. В іншому випадку **стоп**: дерево-основа побудоване.

7.3. ЗАДАЧА ПРО БАГАТОПОЛЮСНИЙ НАЙКОРОТШИЙ ЛАНЦЮГ. АЛГОРИТМ ФЛОЙДА

Алгоритм Дійкстри не дозволяє розв'язувати задачі з від'ємними довжинами дуг. Для розв'язування таких задач використовується алгоритм Флойда, який знаходить найкоротші ланцюги між всіма парами вузлів мережі. Неорієнтована дуга заміняється парою орієнтованих в протилежних напрямках дуг з такими ж довжинами, як у неорієнтованої. Довжини дуг можуть бути від'ємними, однак довжина кожного циклу повинна бути невід'ємною. Нехай $N = \{1,...,n\}$ – множина вузлів мережі, c_{ij} – довжина орієнтованої дуги $(i,j) \in A$, d^*_{ik} – довжина найкоротшого ланцюга з вузла i в вузол k, $D(n \times n)$ – матриця віддалей, $R(n \times n)$ матриця маршрутів.

Алгоритм Флойда будується, виходячи з наступних міркувань.

Нехай d_{ik} – найкраща біжуча оцінка довжини найкоротшого ланцюга з вузла i в вузол k. Розглянемо довжину найкоротшого ланцюга з i в k, який проходить через певний проміжний вузол j – $d_{ik}(j) = d_{ij} + d_{jk}$. Якщо $d_{ik}(j) < d_{ik}$, то біжуче значення $d_{ik} = d_{ik}(j)$. Виходячи з цього, алгоритм Флойда працює наступним чином:

☑ початкова оцінка довжини найкоротшого шляху становить значення c_{ij}, і далі послідовно перевіряються всі проміжні вузли, що розташовані між i та k;

☑ якщо довжина ланцюга, що проходить через деякий проміжний вузол менша, ніж біжуче значення оцінки, то це значення присвоюється оцінці.

Ця процедура повторюється для всіх пар вузлів до того часу, поки не будуть отримані значення найкоротших віддалей.

Алгоритм Флойда (версія, вдосконалена Ху).

Крок 1. Початкові присвоювання значень елементам матриць віддалей та маршрутів — для всіх $i = \overline{1,n}$, $j = \overline{1,n}$:

$d_{ij} = \min\{\infty; c_{ij}\}, i \neq j$; $d_{ii} = 0$; $r_{ij} = j$; $l = 0$ (параметр l використовується для позначення базового вузла на кожній ітерації).

Крок 2. $l = l + 1$. Якщо $l > n$, то **стоп**: значення найкоротших віддалей між всіма парами вершин знайдені та зберігаються в матриці D, а маршрути – в матриці R. В іншому випадку вузол l буде базовим. Викреслюємо базовий стовпчик та базовий рядок матриці D, а також ті її рядки та стовпчики, які мають значення ∞, що знаходиться в базовому рядку та стовпчику.

Крок 3. Перерахунок значень елементів матриць D та R. Для всіх $i = \overline{1,n}$, $k = \overline{1,n}$, $i \neq l \neq k$:

$$r_{ik} = \begin{cases} l, \text{ якщо } d_{ik} > d_{il} + d_{lk} \\ r_{ik} \quad \text{в іншому випадку;} \end{cases} \quad d_{ik} = Min\{d_{ik}; d_{il} + d_{lk}\}.$$

Розрахунки на цьому кроці доцільно виконувати з врахуванням викреслених рядків та стовпчиків. Викреслювання є мнемонічним прийомом, що полегшує розрахунки за вищенаведеними формулами, а також виключає непотрібні перерахунки значень.

Порівнюємо кожен невикреслений елемент d_{ik} з сумою елементів згідно з вищенаведеним та переписуємо перераховані значення в матриці наступної ітерації. Викреслені елементи переписуємо без змін

Перехід до кроку 2.

Задача про багатополюсний найкоротший ланцюг має ряд важливих практичних застосувань. Наприклад, таке.

Нехай існують шляхи передачі інформації між відділами фірми і відомий час передачі одиниці інформації між кожною парою безпосередньо пов'язаних між собою пунктів. Визначити оптимальні маршрути передачі інформації для кожного з відділів можна, розв'язавши задачу про багатополюсний ланцюг.

7.4. ЗАДАЧА ПОШУКУ МАКСИМАЛЬНОГО ПОТОКУ. АЛГОРИТМ РОЗТАШУВАННЯ ПОЗНАЧОК

Алгоритм знаходження максимального потоку в мережі ґрунтується на врахуванні особливостей задачі і полягає в знаходженні на кожному кроці аугментального (збільшуючого) потоку – тобто потоку, який збільшує існуючий і на шляху якого існує хоча б одна дуга, пропускна здатність якої буде використана повністю.

Алгоритм розташування позначок складається з наступних кроків.

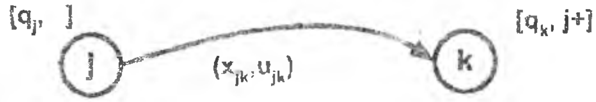
Крок 1. Побудова початкового потоку.

Як початковий потік за будь-яких умов може бути обраний нульовий – тобто потік з нульовими значеннями в кожній дузі мережі.

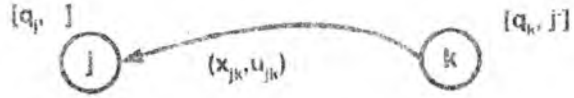
Крок 2. Розташування позначок.

На кожному кроці методу розташування позначок вузол може належати до однієї з наступних трьох категорій: "не позначений"; "позначений та непереглянутий"; "позначений та переглянутий".

На початку роботи алгоритму всі вузли "не позначені". Джерело s отримує позначку $[\infty,-]$. Надалі обираємо довільний позначений та непереглянутий вузол j. Два вузли є сусідніми, якщо вони з'єднані дугою. Сусідні з j непозначені вузли позначаємо. Позначка проставляється наступним чином в залежності від напрямку дуги:



$q_k = Min\left[q_j, u_{jk} - x_{jk}\right] > 0$.



$q_k = Min\left[q_j, x_{kj}\right] > 0$.

Рис. 7.4. Правила обчислення позначок

Якщо ж $q_k = 0$, то вузол k не може бути позначений, і позначка біля нього не ставиться.

Якщо всі вузли переглянуті, то вузол j буде позначеним та переглянутим (позначка у вигляді $\left[q_j, \quad\right]$.

Процес продовжуємо до того моменту, поки не досягнемо витоку у цьому випадку переходимо до кроку 3. Якщо ж неможливо позначити ні одного з вузлів, що залишилися, і витік t ще не досягнутий, то стоп побудований максимальний потік.

Крок 3. Зміна потоку.

Якщо витік має позначку $\left[q_t, k^+\right]$, то потік дугою (k,t) збільшується на значення q_t, якщо ж $\left[q_t, k^-\right]$, то зменшуємо на q_t. Таким чином, послідовно просуваємося дугами від витоку до джерела, і в залежності від знаку позначки додаємо або віднімаємо значення q_t (з позначки при витоку !), поки не досягнемо s. Витираємо старі позначки та переходимо до кроку 2.

Таким чином, принцип роботи алгоритму полягає в послідовному переході від одного до іншого базового розв'язку шляхом пошуку

збільшуючого (аугментального) потоку, причому те, що наступний розв'язок буде базовим, гарантується заповненням до значення пропускної здатності хоча б однієї дуги, що знаходиться на збільшуючому шляху.

Узагальнення задачі про максимальний потік.

Існує декілька узагальнених постановок задач про максимальний потік. Деякі з них приводяться до класичної задачі, інші розв'язуються за допомогою специфічних алгоритмів. Розглянемо деякі з них.

Задача про максимальний потік з декількома джерелами та витоками.

В мережі існує декілька джерел та декілька витоків, потік може текти з довільного джерела в довільний витік, в i-те джерело s_i з зовнішнього середовища втікає не більш ніж a_i одиниць потоку, потреби у витоку t_i становлять b_i одиниць. Таким чином, утворюються наступні групи обмежень:

$$\sum_{(\forall i\in I):(i,j)\in A} f_{ij} - \sum_{(\forall i\in I):(j,i)\in A} f_{ji} = b_j,\ \ j\in N_B, \tag{7.17}$$

$$\sum_{(\forall i\in I):(i,j)\in A} f_{ij} - \sum_{(\forall i\in I):(j,i)\in A} f_{ji} \le a_j,\ \ j\in N_a. \tag{7.18}$$

Ця задача приводиться до класичної задачі про максимальний потік шляхом утворення одного джерела з необмеженим вхідним потоком, яке пов'язане з джерелами первісної задачі за допомогою дуг пропускної здатності a_i та одного витоку, який пов'язаний з витоками первісної задачі дугами пропускної здатності b_j.

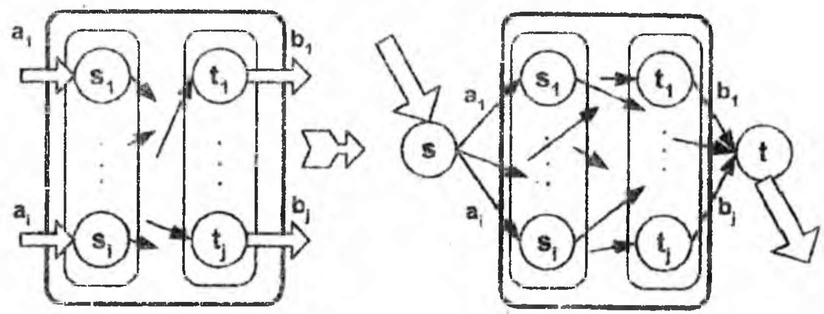


Рис.7.5. Зведення задачі до класичної

Якщо в мережі потік повинен текти з певних джерел до певних витоків, це означає, що ці потоки є різними, тобто їх можна розглядати як потоки різних продуктів, і таким чином виникає ***задача про багатопродуктовий потік.***

Задача про максимальний потік з обмеженнями знизу на пропускні здатності дуг.

Нехай в мережі є одна така дуга $(i, j) \in A$, $0 < l_{ij} \le f_{ij} \le u_{ij}$, тобто одна дуга з обмеженням на пропускну здатність "знизу". Розширимо мережу, вважаючи вузол j новим джерелом, а вузол i – новим витоком. Змінимо пропускну здатність дуги $(i, j) \in A$, $0 \le f_{ij} \le u_{ij} - l_{ij}$ та додамо орієнтовану дугу з безмежною пропускною здатністю від вузла t до вузла s.

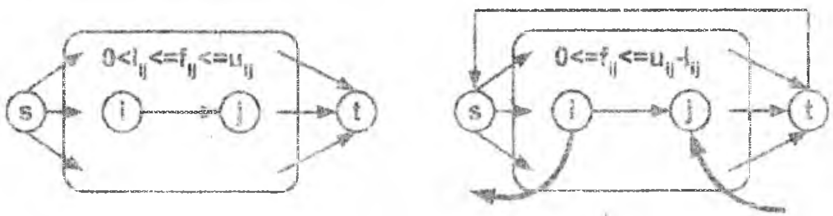


Рис. 7.6. Модифікація мережі

В розширеній мережі шукатимемо максимальний потік з джерела j до витоку i. Якщо величина цього потоку в розширеній мережі більша чи дорівнює l_{ij}, а потік по дузі (t, s) рівний f_{ts}, то в первісній мережі існує потік з джерела s до витоку t, що дорівнює $f_{ts} = f_{ij} \ge l_{ij}$.

Якщо в мережі наявні декілька дуг, що мають ненульові нижні границі для потоків, потрібно ввести декілька штучних джерел та витоків, а потім задачу з декількома джерелами та витоками привести до задачі з одним джерелом та витоком.

Задачі про багатополюсний максимальний потік. Алгоритм Гоморі-Ху.

В багатьох практичних задачах виникає проблема визначення максимальних потоків між усіма парами вузлів мережі (інформаційні мережі та мережі зв'язку, транспортні мережі та ін.). Задачу про багатополюсний максимальний потік можна розв'язати шляхом послідовного розв'язування задачі про максимальний потік між одним джерелом та одним витоком. В цьому випадку необхідно би було розв'язати $(n-1) \times n / 2$ разів цю задачу. В той же час існують ефективніші алгоритми, в яких максимальний потік визначається лише $(n-1)$ разів.

Задачі про багатополюсний потік є двох видів:

☑ **задача аналізу мережі**. Задана мережа з обмеженими пропускними здатностями дуг. Необхідно визначити максимальні потоки між всіма парами вузлів мережі;

☑ **задача синтезу мережі**. Необхідно побудувати мережу, в якій максимальні потоки між всіма парами вузлів задовольняють заданим обмеженням знизу і в яких загальна пропускна здатність всіх дуг мінімальна.

Розв'язок задачі про багатополюсний максимальний потік ґрунтується на ідеї побудови на основі первісної мережі дерева-основи або дерева розрізів.

Нехай $G = (N, A)$ неорієнтована мережа, і нехай пропускні здатності всіх дуг рівні в прямому та оберненому напрямі – $u_{ij} = u_{ji}$. Нехай $i, j, k \in N$. У цьому випадку, згідно з теоремою Форда-Фалкерсона, $F_{ij} = U_{ij}(X, \overline{X})$, де F_{ij} – максимальний потік з джерела i до витоку j, $(X, \overline{X})$ – мінімальний розріз, що відділяє i від j, $U_{ij}(X, \overline{X})$ – пропускна здатність мінімального перерізу з джерелом i та витоком j. Якщо $k \in \overline{X}$, то $F_{ik} \le U_{ij}(X, \overline{X})$, а якщо $k \in X$, то $F_{kj} \le U_{ij}(X, \overline{X})$, і таким чином, $F_{ij} \ge F_{ik}$ та $F_{ij} \ge F_{kj}$, тобто $F_{ik} \ge \min\{F_{ip}; F_{pk}\}$ $F_{kj} \ge \min\{F_{kq}; F_{qj}\}$, де $\{i; p; k; q\}$ зв'язна множина вузлів, що належать N. Внаслідок цього $F_{ij} \ge \min\{F_{ip}; F_{pk}; F_{kq}; F_{qj}\}$. В загальному випадку $F_{ij} \ge \min\{F_{ii_1}; F_{i_1 i_2}; F_{i_2 i_3}; \ldots; F_{i_r j}\}$, де $\{i; i_1; i_2; \ldots; j\}$ – зв'язна множина вузлів з N.

Максимальне дерево основа — дерево розрізів мережі має таку властивість: $w_{ij} \le \min\{w_{ii_1}; w_{i_1 i_2}; w_{i_2 i_3}; \ldots; w_{i_r j}\}$, де (i, j) – довільна дуга, що не належить даному дереву розрізів, $\{i; i_1; i_2; \ldots; j\}$ – єдина послідовність вузлів, що з'єднують гілки дерева, w_{ij} – вага гілки дерева.

Припустимо, що вищеописана властивість не виконується. В цьому випадку замість будь-якої дуги шляху з і в j можна взяти дугу (i, j) внаслідок чого буде побудоване дерево з більшою вагою. Це протиріччя й доводить справедливість нерівності.

Якщо ваги w_{ij} дерева основи покласти рівними F_{ij}, то для будь-якої дуги (i, j), що не належить дереву, справедливим буде співвідношення $F_{ij} \ge \min\{F_{ii_1}; F_{i_1 i_2}; F_{i_2 i_3}; \ldots; F_{i_r j}\}$, де $\{i; i_1; i_2; \ldots; j\}$ – зв'язна послідовність вузлів дерева, що належать шляху з і до j.

Враховуючи вищенаведені нерівності, отримуємо, що для будь-якої дуги, що не належить дереву, $F_{ij} = \min\{F_{ii_1}; F_{i_1 i_2}; F_{i_2 i_3}; \ldots; F_{i_r j}\}$.

Максимальне дерево-основа (остов), що задовільняє останній рівності, й називається деревом розрізів тому, що кожна його гілка відповідає розрізу, а вага гілки дорівнює пропускній здатності розрізу. Якщо в цьому дереві потрібно визначити величину максимального потоку між двома довільними вузлами, то в дереві необхідно знайти шлях, що з'єднує ці два вузли, та обрати на цьому шляху дугу з мінімальною вагою — вага цієї дуги й буде рівною пропускній здатності мінімального розрізу. Отримані результати використовуються в алгоритмі Гоморі-Ху для побудови дерева розрізів.

Алгоритм Гоморі-Ху.

Крок 1. Множина гілок дерева розрізів пуста. Всі вузли об'єднані в одну групу. Обираємо довільну пару вузлів s та t. Покладемо $l = 1$

Крок 2. Визначаємо максимальний потік з s в t за допомогою процедури розташування позначок.

Крок 3. Знаходимо мінімальний розріз, що відділяє s від t. Представляємо цей розріз гілкою в дереві розрізів. Вага гілки рівна пропускній здатності цього розрізу. Ця гілка повинна сполучати вузли чи групи вузлів, які розташовані по різні боки від знайденого мінімального розрізу.

Крок 4. Якщо $l = n - 1$, то стоп. Дерево розрізів побудоване. В іншому випадку переходимо до кроку 5.

Крок 5. Обираємо довільну пару вузлів i та j, які ще не відділені один від одного в дереві розрізів, що будується. Покласти $s = i$ та $t = j$.

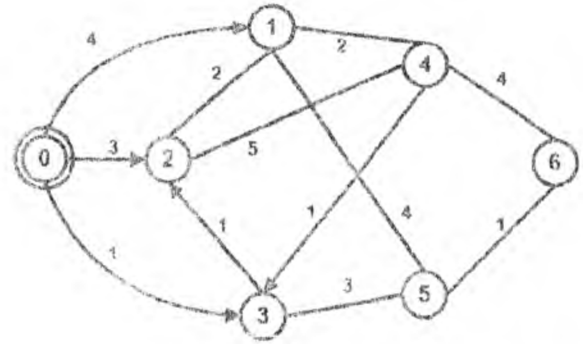
Крок 6. Сконденсувати в один вузол кожну зв'язну підмережу, що з'єднана з групою, в якій знаходяться вузли i та j. Покласти $l = l + 1$.

Перехід до кроку 2.

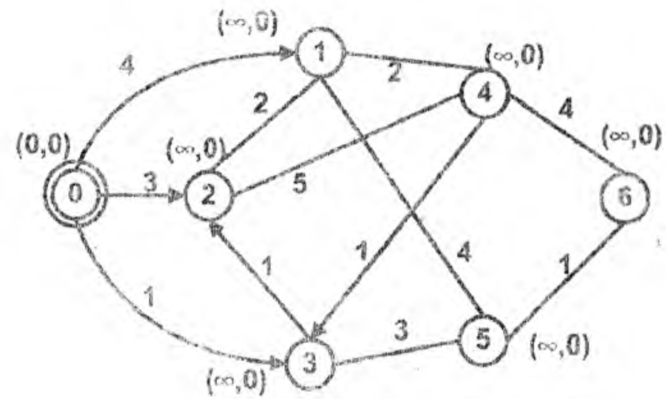
ПРИКЛАДИ

Приклад 7.1. ***Знаходження найкоротшого шляху в мережі.***

Для заданого графу знайти найкоротший шлях з вершини 0 до вершини 6 за допомогою алгоритму Дійкстри.



Розв'язання.

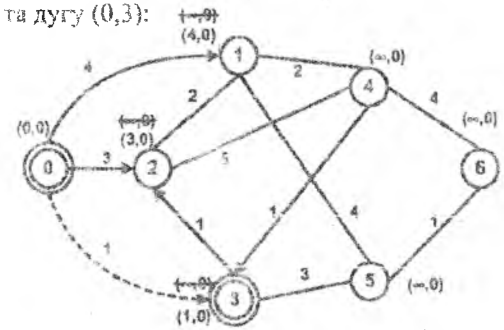


1. *Початкові присвоювання.* Позначимо джерело та витік: $s = 0, t = 6$. Присвоїмо $d(0) = 0$ та $d(x) = \infty$ для всіх $x \neq s$. Позначимо вершину 0 та присвоїмо $y = 0$, де y – остання з переглянутих вершин. Переглянуту вершину позначатимемо подвійним колом, а позначка біля кожної з вершин x – $(d(x), z)$ складатиметься з двох частин: $d(x)$ – біжуче значення віддалі від джерела s до вершини x, z – вершина, з якої отримано значення $d(x)$. При визначенні біжучої переглянутої вершини для наочності позначаємо також ребро, що входить до дерева найкоротших шляхів — ця інформація наявна в другій частині позначки.

2. Для кожної з непереглянутих вершин перераховуємо значення $d(y)$

d(1) = min{d(1), d(0)+c(0,1)}= min{ ∞; 0+4}=4,
d(2) = min{d(2), d(0)+c(0,2)}= min{ ∞; 0+3}=3,
d(3) = min{d(3), d(0)+c(0,3)}= min{ ∞; 0+1}=1,
d(4) = min{d(4), d(0)+c(0,4)}= min{ ∞; ∞} = ∞,
d(5) = min{d(5), d(0)+c(0,5)}= min{ ∞; ∞} = ∞,
d(6) = min{d(6), d(0)+c(0,6)}= min{ ∞; ∞} = ∞.

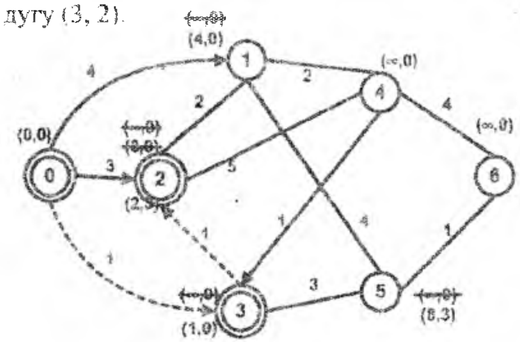
Визначаємо min{d(1), d(2),…, d(6)}=d(3)=1. Позначимо вершину 3, y=3, та дугу (0,3):



3. Оскільки умови завершення роботи алгоритму не виконані, переходимо до наступної ітерації:

d(1) = min{d(1), d(3)+c(3,1)}= min{ 4; 1+ ∞}=4,
d(2) = min{d(2), d(3)+c(3,2)}= min{ 3; 1+1}=2,
d(4) = min{d(4), d(4)+c(3,4)}= min{ ∞; ∞}=∞,
d(5) = min{d(5), d(3)+c(3,5)}= min{ ∞; 1+7}=8,
d(6) = min{d(6), d(3)+c(3,6)}= min{∞; ∞}=∞.

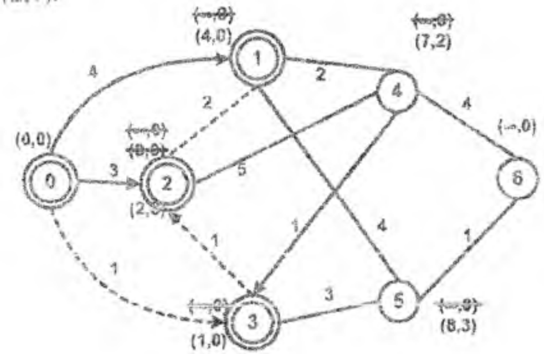
min{d(1), d(2), d(4), d(5), d(6)}=d(2)=2. Позначаємо вершину 2, y=2 та дугу (3, 2).



4. Виконуємо наступну ітерацію:

d(1) = min{d(1), d(2)+c(2,1)}= min{ 4; 2+2}=4 – залишається попереднє значення,
d(4) = min{d(4), d(2)+c(2,4)}= min{ ∞; 2+5}=7,
d(5) = min{d(5), d(2)+c(2,5)}= min{8; 2+∞}= 8,
d(6) = min{d(6), d(2)+c(2,6)}= min{∞; ∞}=∞.

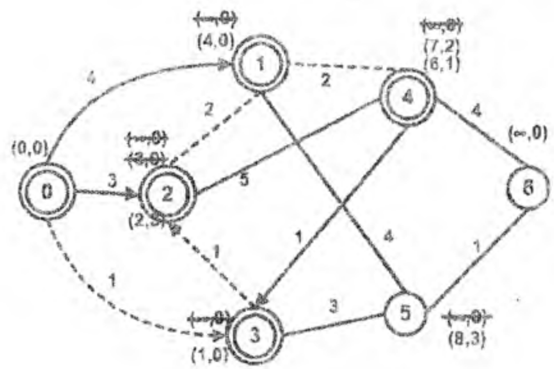
min{d(1), d(4), d(5), d(6)}=d(1)=4. Позначаємо вершину 1, y=1, та дугу (2,1).



5. В результаті наступної ітерації отримуємо:

d(4) = min{d(4), d(1)+c(1,4)}= min{7, 4+2}=6,
d(5) = min{d(5), d(1)+c(1,5)}= min{8; 4+4}= 8 – залишається попереднє,
d(6) = min{d(6), d(1)+c(1,6)}= min{∞; ∞}=∞.

min{d(4), d(5), d(6)}=d(4)=6. Позначаємо вершину 4, y=4, та дугу (1,4).



6. Результат наступної ітерації:

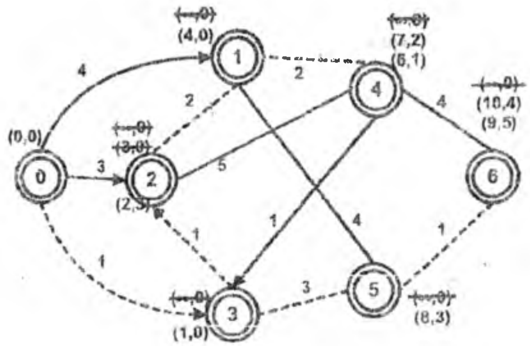
d(5) = min{d(5),d(4)+c(4,5)}= min{8; 6+ ∞}= 8 , d(6) = min{d(6), d(4)+c(4.6)}= min{∞;6+4}=10.

min{d(5), d(6)}=d(5)=8. Позначаємо вершину 5, y=5, та дугу (3,5).

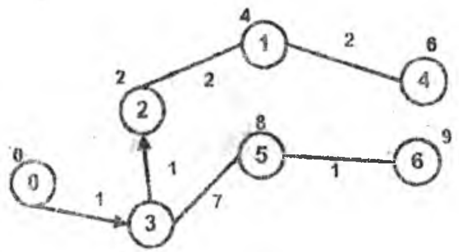
І, нарешті, d(6) = min{d(6), d(5)+c(5,6)}= min{10;8+1}=9.

min{d(6)}=d(6)=9. Позначаємо вершину 6, y=6, та дугу (5,6).

Так як ми позначили вершину 6, то **стоп**. Окрім цього, більше непозначених вершин в мережі немає, тобто одночасно побудоване дерево найкоротших шляхів з вершини 0 у всі інші.



Дерево найкоротших шляхів з вершини 0:



Приклад 7.2. ***Застосування задачі про найкоротший шлях.***

Кожен користувач комп'ютера повинен вирішити питання про купівлю нового комп'ютера через певний проміжок часу, що пов'язане з оновленням елементної бази, застосуванням нових процесорів та технічних рішень. Питання в тому, що рано чи пізно необхідно замінювати старіший комп'ютер на новий, і якщо це робити занадто часто, то це приведе до непродуктивних витрат. Аналогічна ситуація може виникнути й тоді, коли проміжок часу буде занадто довгим.

Вважатимемо, що в початковий момент користувач не має власного комп'ютера, і що він збирається купувати новий комп'ютер щонайменше раз на чотири роки, причому рішення про це приймається раз на рік на початку кожного року. Необхідно визначити, як часто протягом восьмирічного періоду необхідно міняти комп'ютер.

Розв'язання.

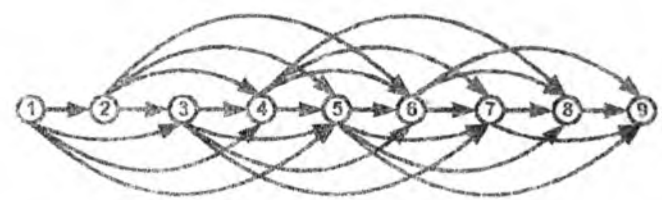
При розв'язанні цієї проблеми враховуватимемо початкову вартість комп'ютера, експлуатаційні витрати, які зростають з часом, залишкову вартість комп'ютера в момент заміни його на новий. Ця задача може бути представлена за допомогою мережі наступним чином:

☑ початок кожного року восьмирічного періоду зображатимемо вузлом мережі;

☑ якщо купівля комп'ютера відбувається на початку i-го року, а на початку j-го року він замінюється новим, то такому варіанту заміни відповідатиме дуга (i,j), довжина якої c_{ij} відображатиме загальні затрати;

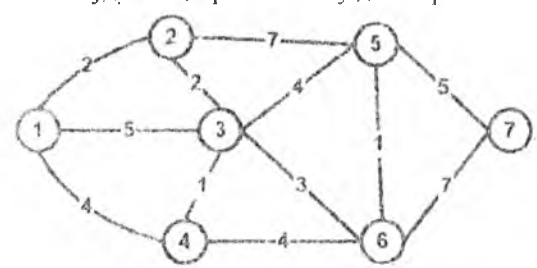
☑ загальні затрати розраховуються за допомогою наступного співвідношення $c_{ij} = P_i - O_j + \sum_{k=1}^{j-1} m_k$, де P_i – початкова вартість комп'ютера на початку i-го року; O_j – залишкова вартість комп'ютера на початку j-го року; m_k – експлуатаційні витрати протягом k-го року.

Відповідна мережа зображена нижче.



Приклад 7.3. ***Алгоритм мінімізації мережі.***

Побудувати дерево-основу для мережі:



Розв'язання.

Розв'язуємо задачу за допомогою алгоритму мінімізації мережі.

$S=\varnothing; \bar{S}=\{1,2,3,4,5,6,7\}$. Довільно обираємо вузол 6. Найближчий до нього є вузол 5 з дугою (6,5) довжиною 1. Таким чином, $S=\{5,6\}; \bar{S}=\{1,2,3,4,7\}$ і біжуча довжина =1. Наступні кроки:

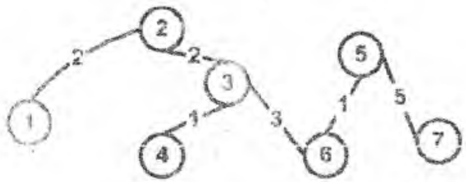
$S=\{6,5,3\}; \bar{S}=\{1,2,4,7\}, Q=1+3=4$, дуга (6,3).

$S=\{6,5,3,4\}; \bar{S}=\{1,2,7\}, Q=4+1=5$, дуга (3,4),

$S=\{6,5,3,4,2\}; \bar{S}=\{1,7\}, Q=5+2=7$, дуга (3,2),

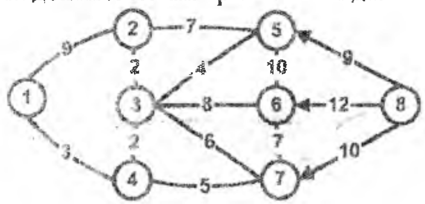
$S=\{6,5,3,4,2,1\}; \bar{S}=\{7\}, Q=7+2=9$, дуга (2,1),

$S=\{6,5,3,4,2,1,7\}; \bar{S}=\varnothing, Q=9+5=14$, дуга (5,7) – стоп.



Приклад 7.4. ***Задача про багатополюсний найкоротший ланцюг.***

Знайти найкоротші віддалі між всіма парами вершин заданої мережі за допомогою алгоритма Флойда:



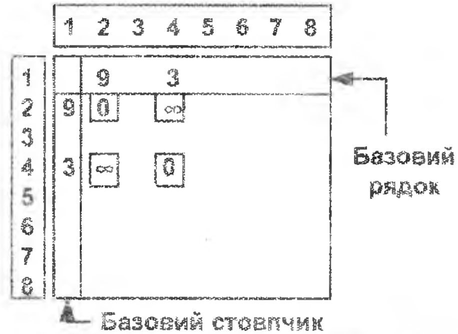
Розв'язання.

Початкові присвоювання:

$$D^0=\begin{bmatrix} 0 & 9 & \infty & 3 & \infty & \infty & \infty & \infty \\ 9 & 0 & 2 & \infty & 7 & \infty & \infty & \infty \\ \infty & 2 & 0 & 2 & 4 & 8 & 6 & \infty \\ 3 & \infty & 2 & 0 & \infty & \infty & 5 & \infty \\ \infty & 7 & 4 & \infty & 0 & 10 & \infty & \infty \\ \infty & \infty & 8 & \infty & 10 & 0 & 7 & \infty \\ \infty & \infty & 6 & 5 & \infty & 7 & 0 & \infty \\ \infty & \infty & \infty & \infty & 9 & 12 & 10 & 0 \end{bmatrix}, \quad R^0=\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{bmatrix}$$

Подальші обчислення виконуємо згідно з алгоритмом:

1-а ітерація. Вузол $l=1$ є базовим. В матриці D^0 викреслюється перший рядок та перший стовпець. Стовпці 3,5,6,7,8 мають у базовому рядку безмежність, а рядки 3,5,6,7,8 – безмежність у базовому стовпчику, і отже, теж викреслюються. Таким чином, перерахунок слід здійснити лише для наступних елементів матриці D^0 – d_{22}, d_{24}, d_{42}, d_{44}:



Так як діагональні елементи матриці не перераховуються, перераховуємо значення тих, що залишилися:

$$d^1_{24}=\min\{d^0_{24}; d^0_{21}+d^0_{14}\}=\min\{\infty; 9+3\}=12$$

$$d^1_{42}=\min\{d^0_{42}; d^0_{41}+d^0_{12}\}=\min\{\infty; 3+9\}=12.$$

Таким чином, в матриці D^1 перерахованими порівняно з матрицею D^0 будуть елементи, що розглядалися вище, а інші перейдуть без зміни з D^0. Відповідно зміняться лише 2 елементи матриці R^1: $r^1_{24}=1$, $r^1_{42}=1$. Таким чином:

$$D^1=\begin{bmatrix} 0 & 9 & \infty & 3 & \infty & \infty & \infty & \infty \\ 9 & 0 & 2 & 12 & 7 & \infty & \infty & \infty \\ \infty & 2 & 0 & 2 & 4 & 8 & 6 & \infty \\ 3 & 12 & 2 & 0 & \infty & \infty & 5 & \infty \\ \infty & 7 & 4 & \infty & 0 & 10 & \infty & \infty \\ \infty & \infty & 8 & \infty & 10 & 0 & 7 & \infty \\ \infty & \infty & 6 & 5 & \infty & 7 & 0 & \infty \\ \infty & \infty & \infty & \infty & 9 & 12 & 10 & 0 \end{bmatrix}, \quad R^1=\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 1 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 1 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{bmatrix}.$$

2-а ітерація. Вузол $l=2$ буде базовим. Викреслюємо в матриці D^1

другий рядок та другий стовпчик, а також 6-8-й стовпчик та 6-8-й рядок як такі, відповідні елементи яких в базовому рядку та базовому стовпчику рівні безмежності, і діагональні елементи, отримуючи множину елементів матриці D^1, значення яких повинні бути перераховані:

	1	2	3	4	5	6	7	8
1	0	9	∞	3	∞			
2	9		2	12	7			
3	∞	2	0	2	4			
4	3	12	2	0	∞			
5	∞	7	4	∞	0			
6								
7								
8								

Базовий рядок

Базовий стовпчик

В результаті перерахунку отримаємо:

$$d^2_{13} = \min\left\{d^1_{13}; d^1_{12} + d^1_{23}\right\} = \min\left\{\infty; 9+2\right\} = 11,$$

$$d^2_{15} = \min\left\{d^1_{15}; d^1_{12} + d^1_{25}\right\} = \min\left\{\infty; 9+7\right\} = 16,$$

$$d^2_{31} = \min\left\{d^1_{31}; d^1_{32} + d^1_{21}\right\} = \min\left\{\infty; 2+9\right\} = 11,$$

$$d^2_{45} = \min\left\{d^1_{45}; d^1_{42} + d^1_{25}\right\} = \min\left\{\infty; 12+7\right\} = 19.$$

$$d^2_{51} = \min\left\{d^1_{51}; d^1_{52} + d^1_{21}\right\} = \min\left\{\infty; 7+9\right\} = 16,$$

$$d^2_{54} = \min\left\{d^1_{54}; d^1_{52} + d^1_{24}\right\} = \min\left\{\infty; 7+12\right\} = 19.$$

Інші з елементів $d_{14}, d_{34}, d_{35}, d_{41}, d_{43}, d_{53}$ в результаті перерахунку збережуть попередні значення, а тому не зміняться й відповідні значення r_{ij}.

Таким чином, $r^2_{13} = r^2_{15} = r^2_{31} = r^2_{45} = r^2_{51} = r^2_{54} = 2$, і матриці будуть мати вигляд:

$$D^2 = \begin{bmatrix} 0 & 9 & 11 & 3 & 16 & \infty & \infty & \infty \\ 9 & 0 & 2 & 12 & 7 & \infty & \infty & \infty \\ 11 & 2 & 0 & 2 & 4 & 8 & 6 & \infty \\ 3 & 12 & 2 & 0 & 19 & \infty & 5 & \infty \\ 16 & 7 & 4 & 19 & 0 & 10 & \infty & \infty \\ \infty & \infty & 8 & \infty & 10 & 0 & 7 & \infty \\ \infty & \infty & 6 & 5 & \infty & 7 & 0 & \infty \\ \infty & \infty & \infty & \infty & 9 & 12 & 10 & 0 \end{bmatrix}, \quad R^2 = \begin{bmatrix} 1 & 2 & 2 & 4 & 2 & 6 & 7 & 8 \\ 1 & 2 & 3 & 1 & 5 & 6 & 7 & 8 \\ 2 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 1 & 3 & 4 & 2 & 6 & 7 & 8 \\ 2 & 2 & 3 & 2 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{bmatrix}.$$

Результати обчислень, отримані на наступних ітераціях, наведені нижче

$$D^3 = \begin{bmatrix} 0 & 9 & 11 & 3 & 15 & 19 & 17 & \infty \\ 9 & 0 & 2 & 4 & 6 & 10 & 8 & \infty \\ 11 & 2 & 0 & 2 & 4 & 8 & 6 & \infty \\ 3 & 4 & 2 & 0 & 6 & 10 & 5 & \infty \\ 15 & 6 & 4 & 6 & 0 & 10 & 10 & \infty \\ 19 & 10 & 8 & 10 & 10 & 0 & 7 & \infty \\ 17 & 8 & 6 & 5 & 10 & 7 & 0 & \infty \\ \infty & \infty & \infty & \infty & 9 & 12 & 10 & 0 \end{bmatrix}, \quad R^3 = \begin{bmatrix} 1 & 2 & 2 & 4 & 2 & 2 & 2 & 8 \\ 1 & 2 & 3 & 3 & 3 & 3 & 3 & 8 \\ 2 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 3 & 3 & 4 & 3 & 3 & 7 & 8 \\ 2 & 3 & 3 & 3 & 5 & 6 & 3 & 8 \\ 3 & 3 & 3 & 3 & 5 & 6 & 7 & 8 \\ 3 & 3 & 3 & 4 & 3 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{bmatrix},$$

$$D^4 = \begin{bmatrix} 0 & 7 & 5 & 3 & 9 & 13 & 8 & \infty \\ 7 & 0 & 2 & 4 & 6 & 10 & 8 & \infty \\ 5 & 2 & 0 & 2 & 4 & 8 & 6 & \infty \\ 3 & 4 & 2 & 0 & 6 & 10 & 5 & \infty \\ 9 & 6 & 4 & 6 & 0 & 10 & 10 & \infty \\ 13 & 10 & 8 & 10 & 10 & 0 & 7 & \infty \\ 8 & 8 & 6 & 5 & 10 & 7 & 0 & \infty \\ \infty & \infty & \infty & \infty & 9 & 12 & 10 & 0 \end{bmatrix}, \quad R^4 = \begin{bmatrix} 1 & 4 & 4 & 4 & 4 & 4 & 4 & 8 \\ 4 & 2 & 3 & 3 & 3 & 3 & 3 & 8 \\ 4 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 3 & 3 & 4 & 3 & 3 & 7 & 8 \\ 4 & 3 & 3 & 3 & 5 & 6 & 3 & 8 \\ 4 & 3 & 3 & 3 & 5 & 6 & 7 & 8 \\ 4 & 3 & 3 & 4 & 3 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{bmatrix},$$

$$D^5 = \begin{bmatrix} 0 & 7 & 5 & 3 & 9 & 13 & 8 & \infty \\ 7 & 0 & 2 & 4 & 6 & 10 & 8 & \infty \\ 5 & 2 & 0 & 2 & 4 & 8 & 6 & \infty \\ 3 & 4 & 2 & 0 & 6 & 10 & 5 & \infty \\ 9 & 6 & 4 & 6 & 0 & 10 & 10 & \infty \\ 13 & 10 & 8 & 10 & 10 & 0 & 7 & \infty \\ 8 & 8 & 6 & 5 & 10 & 7 & 0 & \infty \\ 18 & 15 & 13 & 15 & 9 & 12 & 10 & 0 \end{bmatrix}, \quad R^5 = \begin{bmatrix} 1 & 4 & 4 & 4 & 4 & 4 & 4 & 8 \\ 4 & 2 & 3 & 3 & 3 & 3 & 3 & 8 \\ 4 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 3 & 3 & 4 & 3 & 3 & 7 & 8 \\ 4 & 3 & 3 & 3 & 5 & 6 & 3 & 8 \\ 4 & 3 & 3 & 3 & 5 & 6 & 7 & 8 \\ 4 & 3 & 3 & 4 & 3 & 6 & 7 & 8 \\ 5 & 5 & 5 & 5 & 5 & 6 & 7 & 8 \end{bmatrix}.$$

На наступних ітераціях матриці D^6, R^6, D^7, R^7, D^8, R^8 залишаються незмінними.

Розглянемо найкоротший шлях з вузла 1 до вузла 5 :

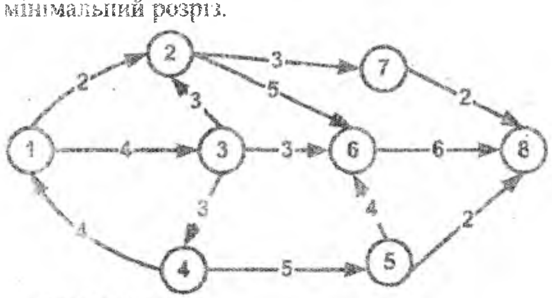
☒ довжина цього шляху становить $d^8_{15} = 9$;

☒ за допомогою матриці R^8 визначаємо проміжні вузли – $r^8_{15} = 4 \Rightarrow r^8_{45} = 3 \Rightarrow r^8_{35} = 5$; таким чином це шлях $1 \Rightarrow 4 \Rightarrow 3 \Rightarrow 5$.

Приклад 7.5. ***Задача про максимальний потік.***

Задана мережа з джерелом 1 і витоком 8 та пропускними здатностями дуг.

Необхідно знайти максимальний потік з джерела до витоку та мінімальний розріз.

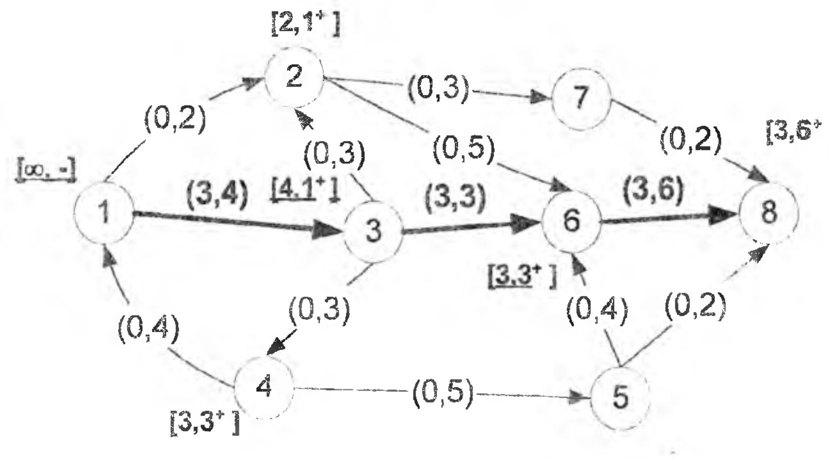


Розв'язання.

Вважаємо, як і в більшості випадків, що початковий потік в мережі є нульовим.

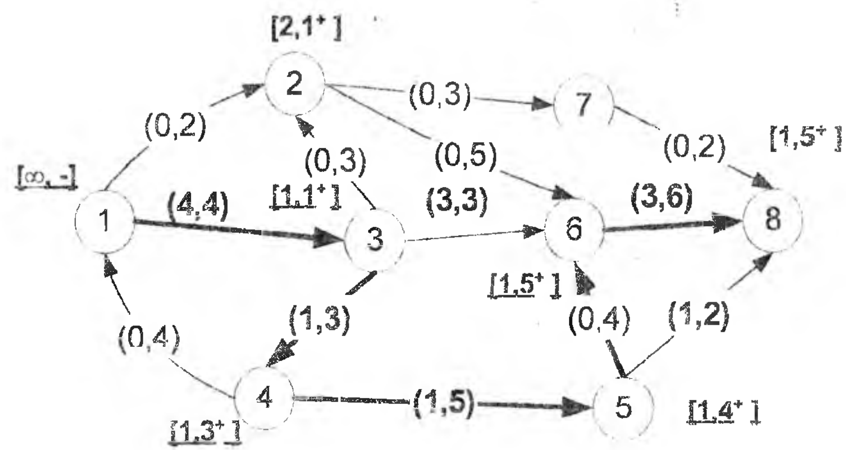
Ітерація 1. Позначаємо джерело 1 та переглядаємо сусідні вузли – 2 та 3 позначаємо, вузол 4 не може бути позначений. Відмічаємо вузол 1 як позначений та переглянутий. Далі обираємо будь-який позначений та непереглянутий вузол (у прикладі обираємо з 2 та 3). Обираємо вузол 3 і переглядаємо сусідні з ним непозначені та непереглянуті вузли (вузли 4

та 6) і позначаємо їх:

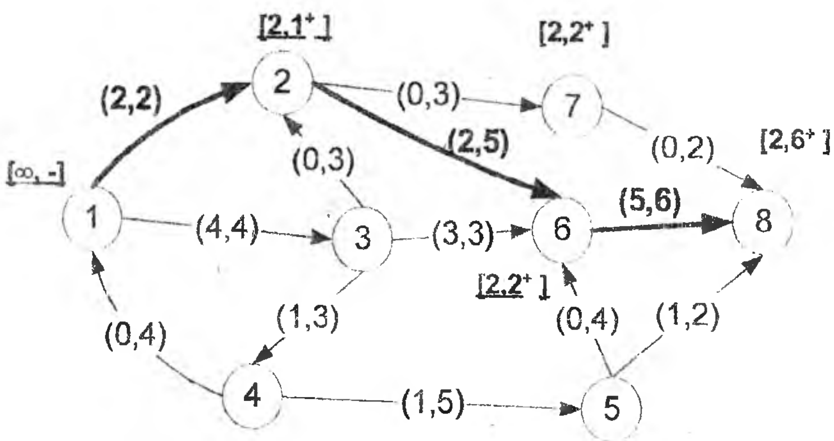


Відмічаємо вузол 3 як позначений та переглянутий. Обираємо один з позначених та непереглянутих вузлів (2, 4 або 6) – вузол 6 та переглядаємо сусідні з ним непозначені та непереглянуті вузли (вузли 5 та 8) і позначаємо їх (вузол 5 не може бути позначений). Оскільки ми позначили джерело, то збільшуємо потік по кожній з дуг аугментального шляху 1-3-6-8 (рухаючись за допомогою позначок від витоку до джерела) на величину 3

Ітерація 2. Знаходимо наступний збільшуючий шлях:



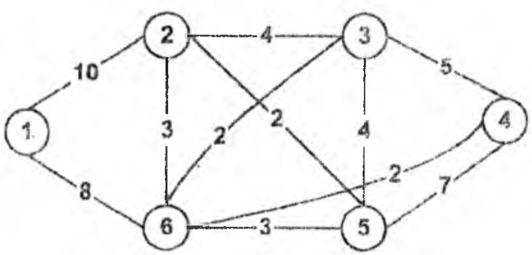
Ітерація 3. Знаходимо наступний збільшуючий потік:



Ітерація 4. Знаходимо наступний збільшуючий потік. З вузла 1 неможливо позначити ні одного вузла, тобто побудований максимальний потік. Мінімальний переріз утворюють прямі дуги (1,2), (1,3) та обернена (4,1). Максимальний потік становить 6.

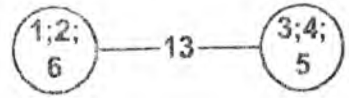
Приклад 7.6. Алгоритм Гоморі-Ху

Знайти максимальну інтенсивність інформаційних потоків між всіма вузлами мережі за допомогою алгоритму Гоморі-Ху, якщо задані інтенсивності потоків інформації між безпосередньо пов'язаними вузлами цієї мережі.

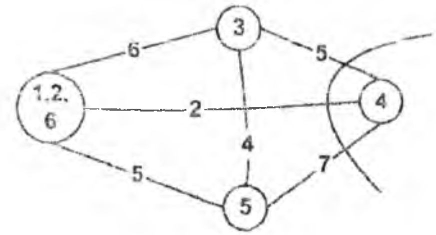


Розв'язання.

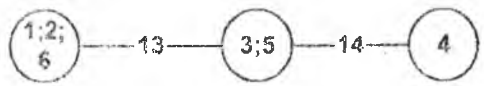
Ітерація 1. Спочатку обираємо два довільні вузли – 1 та 3 і шукаємо максимальний потік між ними за допомогою процедури розташування позначок. В результаті отримуємо мінімальний розріз X ={1;2;6}, $\bar{X}$ = {3;4;5} та максимальний потік через нього 4+2+2+2+3=13 і починаємо будувати дерево розрізів:



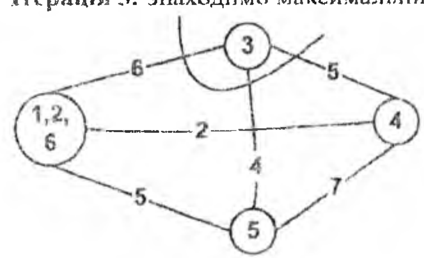
Ітерація 2. Знаходимо максимальний потік між вузлами 3 та 4. Вузли 1, 2, та 6 об'єднані в конденсований вузол. Вузол 3 з'єднаний з вузлами 2 та 6, тобто пропускна здатність відповідної конденсованої дуги становитиме 2+4=6 і аналогічно вузол 5:



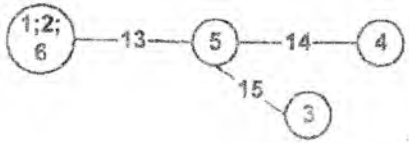
Максимальний потік дорівнює 5+2+7=14, мінімальний розріз X = {1;2;6;3;5} , $\bar{X}$ ={4}. За мінімальним розрізом визначаємо, що вузли 3, 5 та 1, 2, 6 знаходяться з одного боку в мінімальному розрізі, що ділить вузли 3 та 4:



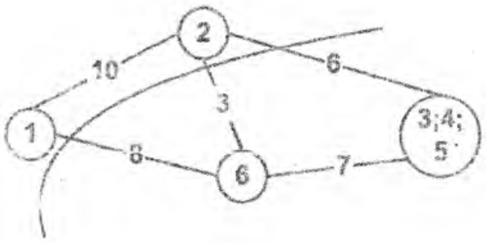
Ітерація 3. Знаходимо максимальний потік між вузлами 3 та 5:



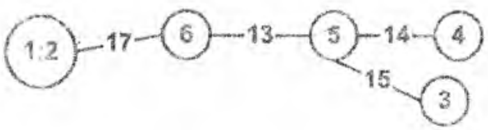
Отримуємо мінімальний розріз $X=\{1;2;6;5;4\}$, $\overline{X}=\{3\}$, пропускна здатність якого становить 6+4+5=15:



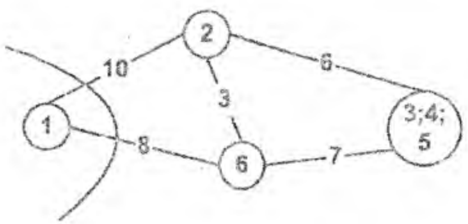
Ітерація 4. Знаходимо максимальний потік між вузлами 1 та 6 в мережі:



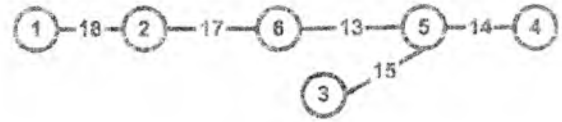
Мінімальний розріз $X=\{1;2\}$, $\overline{X}=\{6;3;4;5\}$, його пропускна здатність становить 6+3+8=17:



Ітерація 5. Знайдемо максимальний потік між вузлами 1 та 2:



Мінімальний розріз $X=\{1\}$, $\overline{X}=\{2;3;4;5;6\}$, його пропускна здатність становить 10+8=18:



Таким чином, задача розв'язана, і на основі дерева розрізів отримуємо матрицю максимальних потоків між вузлами мережі:

	1	2	3	4	5	6
1	∞	18	13	13	13	17
2	18	∞	13	13	13	17
3	13	13	∞	14	15	13
4	13	13	14	∞	14	13
5	13	13	15	14	∞	13
6	17	17	13	13	13	∞

РЕЗЮМЕ

7.1. Розв'язання потокової задачі полягає в знаходженні таких значень функції потоку для всіх дуг мережі, що протікає від джерела до витоку мережі і забезпечує екстремальне значення функції мети. Функція потоку визначена на кожній дузі мережі, є невід'ємною, значення якої не перевищує пропускної здатності дуг. Максимальний потік в мережі визначається за допомогою пропускних здатностей розрізів мережі. Звичайно, величина максимального потоку не повинна перевищувати величину пропускної здатності розрізу. Окремими випадками загальної постановки задачі про максимальний потік є відомі задачі – транспортна, задача про призначення, задача про максимальний потік, задача про найкоротший ланцюг. Ці задачі мають одну дуже важливу особливість: якщо значення коефіцієнтів в них – цілі, то будь-який базовий розв'язок загальної задачі про максимальний потік мінімальної вартості є цілочисельним.

7.2. В задачі пошуку найкоротшого маршруту замість пошуку найкоротшого маршруту з s в t шукаються найкоротші маршрути з s у всі інші вершини мережі, так як будь-яка вершина i може виявитися деякою проміжною вершиною найкоротшого ланцюга з s в t. Таким чином, по суті, будується дерево найкоротших шляхів з s у всі вершини *мережі. При пошуку найкоротшого шляху з s в t роботу алгоритму можна припинити, коли буде досягнута вершина t. Задача мінімізації мережі є задачею побудови найкоротшого дерева-основи. Задача мінімізації мережі є однією з небагатьох, яка може бути ефективно розв'язана за допомогою алгоритму "пожираючого" типу.*

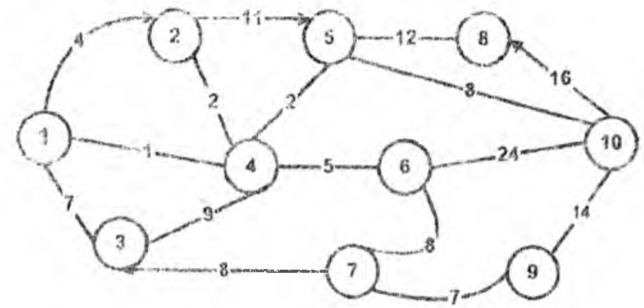
7.3. Алгоритм Дійкстри не дозволяє розв'язувати задачі з від'ємними довжинами дуг. Для розв'язування таких задач використовується алгоритм Флойда, який знаходить найкоротші ланцюги між всіма парами вузлів мережі. Неорієнтована дуга заміняється парою орієнтованих в протилежних напрямках дуг з такими ж довжинами, як у неорієнтованої. Довжини дуг можуть бути від'ємними, однак довжина кожного циклу повинна бути невід'ємною.

7.4. Алгоритм знаходження максимального потоку в мережі ґрунтується на врахуванні особливостей задачі і полягає в знаходженні на кожному кроці аугментального (тобто збільшуючого) потоку – тобто потоку, який збільшує існуючий і на шляху якого існує хоча б одна дуга, пропускна здатність якої буде використана повністю. Те, що наступний розв'язок буде базовим, гарантується заповненням до значення пропускної здатності хоча б однієї дуги, що знаходиться на збільшуючому шляху. Існує декілька узагальнених постановок задач про максимальний потік, деякі з них приводяться до класичної задачі, інші розв'язуються за допомогою специфічних алгоритмів: задача про максимальний потік з декількома джерелами та витоками; задача про максимальний потік з обмеженнями знизу на пропускні здатності дуг; задача про багатополюсний максимальний потік.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

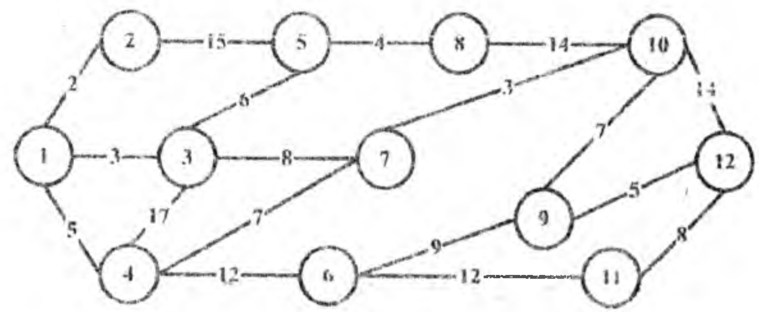
Завдання 7.1. ***Знаходження найкоротших шляхів в мережі.***

Побудувати дерево найкоротших шляхів в графі з вершини 1 за допомогою алгоритму Дійкстри. Знайти віддалі між всіма парами вершин підграфа, що складається з вершин 1-6 графа за допомогою алгоритму Флойда.



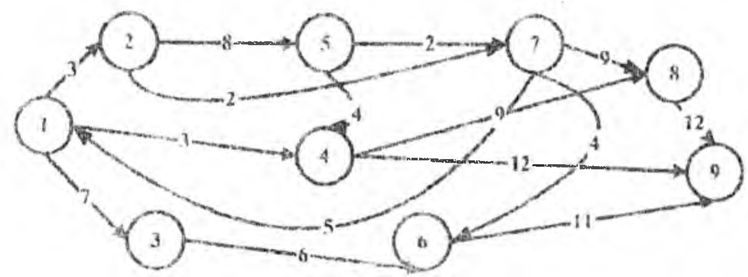
Завдання 7.2. ***Мінімізація мережі.***

Побудувати мінімальне дерево-базу для наступної мережі.



Завдання 7.3. ***Знаходження максимального потоку в мережі.***

Знайти максимальний потік та мінімальний розріз в мережі за допомогою алгоритму розташування позначок. Джерело - вузол 1, витік - вузол 9.



ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Якими властивостями характеризується потік?
2. Які особливості задач потокового типу?
3. Дайте визначення розрізу в мережі.
4. Яким чином визначається пропускна здатність розрізу?
5. Доведіть теорему Форда-Фалкерсона та поясніть її значення.
6. Яким чином будується мінімальний розріз згідно з доведенням теореми Форда-Фалкерсона?
8. Опишіть постановку потокової задачі загального типу та покажіть, які задачі і чому можна розглядати як часткові випадки загальної?
9. Дайте визначення унімодулярності та поясніть його значення для потокових задач.
10. На якій ідеї будується алгоритм Дійкстри та які задачі можна розв'язувати за його допомогою?
11. Опишіть послідовність кроків алгоритму Дійкстри.
12. Чим відрізняється задача мінімізації мережі від задачі побудови найкоротших шляхів від заданої вершини мережі?
13. Опишіть послідовність кроків алгоритму Флойда та назвіть його переваги стосовно алгоритму Дійкстри.
14. Що служить основою для побудови алгоритму пошуку максимального потоку в мережі?
15. Перерахуйте узагальнення задачі про максимальний потік та способи зведення їх до задачі про максимальний потік.
16. Які ідеї використовує алгоритм Гоморі-Ху?

РОЗДІЛ 3

ЗАДАЧІ З ЦІЛОЧИСЕЛЬНИМИ ЗМІННИМИ

ТЕМА 8. ЦІЛОЧИСЕЛЬНІ ЗАДАЧІ МАТЕМАТИЧНОГО ПРОГРАМУВАННЯ

Цілочисельні задачі є широко розповсюдженими, і водночас (за виключенням спеціальних випадків з унімодулярною матрицею коефіцієнтів обмежень) важкими до розв'язання. Якщо у випадку великих значень змінних розв'язків для розв'язання застосовують симплекс-метод з наступним округленням отриманих результатів, то для малих значень це неможливо, а для булевих задач взагалі втрачає сенс. Тому й були розроблені підходи до розв'язання цілочисельних задач, два з яких і розглядаються в цій темі. Варіант методу відтинаючих площин був запропонований Гоморі, і алгоритми, що ґрунтуються на ідеї відтинань, набули достатнього поширення для розв'язання мішаних задач лінійного програмування. Загальніший та гнучкий підхід пропонується в методі розгалужень та границь (Branch&Bound) – це скоріше не метод, а методологія, яка дозволяє конструювати при доцільному її використанні ефективні алгоритми розв'язання специфічних класів цілочисельних задач, як, наприклад, алгоритм для розв'язання задачі про комівояжера. Окрім того, алгоритми, побудовані на ідеях методу розгалужень та границь, можуть бути без проблем модифіковані для отримання наближених розв'язків (ε -оптимальних розв'язків), що дозволяє різко скоротити об'єм перебору з отриманням наближених розв'язків, що не гірші від оптимального більш ніж на ε без знання точного значення критерію оптимальності для оптимального розв'язку.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати ⇨*основні класи алгоритмів розв'язання цілочисельних задач;* ⇨*метод Гоморі для розв'язування мішаних задач лінійного програмування;* ⇨ *теоретичні основи та основні структурні елементи методу* *розгалужень та границь;*

вміти: ⇨ *розв'язувати мішані задачі лінійного програмування за допомогою методу Гоморі;* ⇨ *застосовувати ідеї методу розгалужень та границь до конструювання ефективних алгоритмів розв'язування цілочисельних задач.*

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ задачі булевого програмування
- ☑ методи відтинань
- ☑ задачі мішаного програмування
- ☑ границя
- ☑ задача про наплечник
- ☑ дерево розгалужень
- ☑ релаксація
- ☑ комбінаторні методи
- ☑ активний вузол
- ☑ евристичні методи
- ☑ критерій відтинання вузлів
- ☑ метод розгалужень та границь
- ☑ стратегія розгалуження
- ☑ задача про комівояжера

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

8.1. Постановка задачі цілочисельного лінійного програмування, її інтерпретація та основні підходи до розв'язування.

8.2. Розв'язування лінійних задач мішаного програмування методом Гоморі і методом розгалужень та границь.

8.3. Структура та основні складові методу розгалужень та границь.

8.1. ПОСТАНОВКА ЗАДАЧІ ЦІЛОЧИСЕЛЬНОГО ЛІНІЙНОГО ПРОГРАМУВАННЯ, ЇЇ ІНТЕРПРЕТАЦІЯ ТА ОСНОВНІ ПІДХОДИ ДО РОЗВ'ЯЗУВАННЯ

Структурні особливості задач цілочисельного лінійного програмування служать основою для розробки ефективних алгоритмів розв'язування деяких задач. В загальному випадку цілочисельні задачі є NP-повними, тобто ефективних алгоритмів їх розв'язування не існує.

Розглянемо ілюстративний приклад простої задачі цілочисельного програмування. Для саду необхідно придбати 107 кг добрив. Добрива

продаються в мішках: вагою по 35 кг – за 14 грошових одиниць мішок; по 24 кг – 12. Мета – купити не менш 107 кг, витративши якомога менше грошей. Формальна модель цієї задачі є наступною:

$$14x_1 + 12x_2 \Rightarrow Min; \; 35x_1 + 24x_2 \geq 107; \; x_1, x_2 \geq 0.$$

Якщо вважати, що змінні є неперервними, то оптимальним розв'язком буде купити $3\frac{2}{35}$ мішка вагою 35 кг. Однак за умови цілочисельності оптимальний розв'язок буде зовсім іншим – необхідно купити 1 мішок по 35 кг та 3 мішки по 24 кг. Таким чином, введення умови цілочисельності різко змінює характер розв'язку, тобто в більшості цілочисельних задач заокруглення неперервного розв'язку до цілих значень небажане – таким чином можна отримати або неоптимальний, або ж неприпустимий розв'язок. Це тим більше стосується задач з цілочисельними змінними, які можуть набувати лише двох значень – 0 або 1 (такі задачі називаються задачами булевого програмування). В задачах, що включають булеві змінні, за їх допомогою можуть моделюватися умови, що можуть або виконуватися, або ні – і зрозуміло, що дробове значення не має сенсу.

Якщо розглядати з цієї точки зору задачі виробничого планування, то у випадку масового виробництва цілком можливо розглядати задачу як неперервну, розв'язати її за допомогою звичайного симплекс-методу і розв'язок заокруглити до цілих. Такий підхід цілком виправданий внаслідок великих об'ємів випуску продукції та неточностей визначення значень параметрів задачі. В той же час такий підхід неприпустимий при визначенні плану одиничного виробництва – наприклад, авіабудівельного, коли об'єм випуску 2 чи максимально 3 основних моделей літаків рідко коли досягає десятків. У цьому випадку заокруглення отриманого за допомогою звичайного симплекс-методу розв'язку може привести або до неприпустимості задачі – порушаться обмеження на виробничі ресурси, або ж до їх неповного використання, тобто неоптимальності розв'язку.

Розглянемо деякі приклади цілочисельних задач.

Задача про наплечник. Мандрівник, що збирається в похід, повинен визначити набір предметів, які він бере з собою в наплечник обмеженого об'єму, що становить V. Кожен з предметів i займає певний об'єм v_i та характеризується цінністю c_i для мандрівника. Будемо вважати, що якщо мандрівник бере з собою i-й предмет, то $x_i = 1$, в іншому випадку – нуль.

Таким чином, отримуємо наступну задачу булевого програмування:

$$\sum_i c_i x_i \Rightarrow Max, \; \sum_i v_i x_i \leq V, \; \forall x_i \in \{0,1\}, \; \forall c_i > 0, \; \forall v_i > 0 \; .$$

На прикладі цієї задачі яскраво видно, що розв'язування за допомогою класичного симплекс методу з наступним заокругленням результатів не має сенсу. Ця задача є найпростішою характерною, на якій випробовувалися алгоритми, що застосовувалися згодом для розв'язування значно складніших цілочисельних задач.

Субоптимальний розв'язок задачі про наплечник можна отримати з використанням ідеї пріоритетів. Вважатимемо, що предмети впорядковані за цінністю – від найціннішого до найменш цінного – $c_1 \geq c_2 \geq \ldots \geq c_n$ і в цьому порядку заповнюватимемо наплечник, поки буде місце або поки не переглянемо всі предмети. Однак такий розв'язок не завжди буде прийнятним – особливо в тих випадках, коли цінніші предмети є великого об'єму. Тому можна впорядкувати предмети за спаданням об'єму $v_1 \leq v_2 \leq \ldots \leq v_n$ і заповнювати наплечник в такому порядку. Однак і в цьому випадку за умови, якщо предмету малого об'єму відповідатиме ще менша питома цінність, розв'язок може виявитися достатньо далеким від оптимального. Вдалішим видається впорядкування такого типу:

$\frac{c_1}{v_1} \geq \frac{c_2}{v_2} \geq \ldots \geq \frac{c_n}{v_n}$, яке враховує як цінність, так і об'єм предмету.

Методи такого типу ґрунтуються на правдоподібних міркуваннях і зводяться до побудови функції пріоритету або системи пріоритетів.

Задача про ефективну експедицію. В експедицію необхідно обрати мінімальну кількість учасників з числа кандидатів таким чином, щоб були наявні фахівці необхідних спеціальностей. Кожен з кандидатів володіє однією або декількома спеціальностями.

Позначивши S_i – множина осіб-кандидатів, що володіють i-ю спеціальністю, $x_j \in \{0,1\}$, 0 – якщо j-й кандидат не відбирається до складу експедиції, 1 – в іншому випадку, $j \in J$, де J – множина індексів-"номерів" кандидатів, представимо формальну модель задачі:

$$\sum_{j \in J} x_j \Rightarrow Min, \ \forall i: \sum_{k \in S_i} x_k \geq 1 \ .$$

В цій задачі булеві змінні використані для відображення логічних умов, а тому розв'язання її без врахування умов цілочисельності (з додатковими обмеженнями $0 \leq x_j \leq 1$) взагалі не має сенсу.

Для розв'язування цілочисельних задач застосовуються методи, що належать до однієї з наступних груп.

Методи відтинань. Ідея відтинань належить Ґоморі. Методи цієї групи

призначені для розв'язування лінійних змішаних (як з неперервними, так і з цілочисельними змінними) задач і будуються таким чином, щоб в процесі розв'язування відтинались частини області припустимих розв'язків, що не містять оптимуму до того моменту, поки не буде знайдений цілочисельний оптимальний розв'язок. Спочатку розв'язується задача з послабленими обмеженнями (без врахування цілочисельності), надалі на кожній ітерації вводиться додаткове обмеження, що враховує вимоги цілочисельності, і таким чином многогранник припустимих розв'язків тнеться до того часу, поки отриманий розв'язок не стане цілочисельним. Методи цієї групи різняться в основному способом формування додаткових обмежень.

Комбінаторні методи. Комбінаторні методи побудовані на використанні ідеї неповного перебору з відтинанням на кожному кроці підмножин області припустимих розв'язків, в яких гарантовано не знаходиться оптимальний розв'язок. В методах цієї групи в більшості випадків використовуються процедури пошуку по дереву розв'язків.

Евристичні методи. Методи цієї групи будуються на правдоподібних припущеннях або використовують ідеї алгоритмів розв'язування простіших задач і не дозволяють гарантовано отримати оптимальний розв'язок, хоча в середньому отримувані розв'язки будуть близькими до оптимальних. Широко розповсюдженим прийомом є синтез правил пріоритетів (як в задачі про наплечник), починаючи від лексикографічного впорядкування та простих правил пріоритетів до деревовидних структур гнучких ієрархічних пріоритетів.

8.2. РОЗВ'ЯЗУВАННЯ ЛІНІЙНИХ ЗАДАЧ МІШАНОГО ПРОГРАМУВАННЯ МЕТОДОМ ҐОМОРІ ТА МЕТОДОМ РОЗГАЛУЖЕНЬ ТА ГРАНИЦЬ

Метод Ґоморі належить до методів відтинань і застосовується до розв'язування лінійних задач наступного типу:

$$Q = \sum_{j=1}^{n} c_j x_j \Rightarrow Max, \ \sum_{j=1}^{n} a_{ij} x_j = b_i, \ i = \overline{1,m}, \ j = \overline{1,n}, \qquad (8.1)$$

на деякі (або на всі) змінні накладена умова цілочисельності. Алгоритм Ґоморі працює наступним чином.

Схема алгоритму методу Ґоморі.

Крок 1. Використовуючи будь-який з варіантів симплекс-методу, знаходимо оптимальний розв'язок задачі без врахування умов цілочисельності. Якщо задача не має розв'язку – **стоп**. Якщо для отриманого розв'язку виконуються умови цілочисельності, то **стоп** – знайдений оптимальний розв'язок задачі.

Крок 2. Укладаємо додаткове обмеження для змінної, яка в біжучому оптимальному розв'язку має максимальну дробову частину та повинна бути цілочисельною згідно з умовами задачі.

Крок 3. За допомогою двоїстого симплекс-методу знаходимо оптимальний розв'язок задачі з додатковими обмеженнями.

Крок 4. Якщо для отриманого розв'язку виконуються умови цілочисельності, то **стоп** – знайдений оптимальний розв'язок задачі. Якщо розв'язок задачі відсутній – **стоп**. В іншому випадку **переходимо до кроку 2.**

Розглянемо детальніше, яким чином формується додаткове обмеження задачі та чим викликане застосування на наступних кроках двоїстого симплекс-методу. За визначенням дробовою частиною числа a – $f(a)$ вважатимемо найменше невід'ємне число b таке, що різниця $a-b$ буде цілою: $f(2,2)=0,2 \Rightarrow (2,2-0,2=2)$;

$f(-2,2)=0,8 \Rightarrow (-2,2-0,8=-3)$.

Додаткове обмеження формується на основі рядка біжучої симплекс-таблиці, в якому знаходиться змінна з максимальною дробовою частиною з числа тих, які за умовою задачі повинні бути цілочисельними. Нехай цей рядок має вигляд:

$$\sum_{j=1}^{k} a_{lj}x_j = b_l,\quad l = \arg \underset{i\in I_N}{Max}\left(f\left(b_i\right)\right), \tag{8.2}$$

$I_N \subseteq \left\{\overline{1,m_g}\right\}$ – множина індексів змінних біжучої бази, які повинні бути цілими згідно з умовою задачі і які ними не є, $card\left(I_N\right)$ – кількість базових змінних на біжучому кроці методу Ґоморі (внаслідок формування додаткових обмежень під час розв'язування задачі кількість обмежень та змінних в задачі зростатимуть). Додаткове, $\left(card\left(I_N\right)+1\right)$-е обмеження матиме в початковій формі вигляд:

$$\sum_{j=1}^{k}\gamma_{lj}x_j \ge f\left(b_l\right), \tag{8.3}$$

де $f\left(b_l\right)$ - дробова частина b_l. Значення коефіцієнтів γ_{lj} розраховуються за наступною схемою залежно від того, чи відповідне значення x_j, згідно з умовами задачі, повинно бути цілим чи ні, а також від співвідношень

між значеннями коефіцієнтів обраного рядка симплекс-таблиці та значеннями їх дробових частин:

Схема обчислення значень γ_{lj}

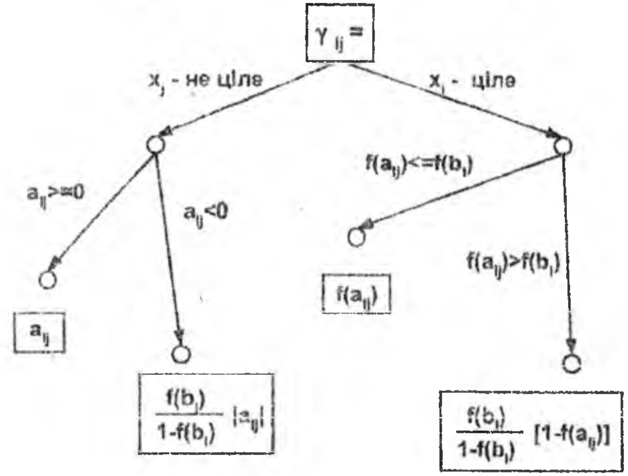


Рис. 8.1. Схема обчислення значень коефіцієнтів додаткового рівняння

Отримане обмеження $\sum_{j=1}^{k}\gamma_{lj}x_j \ge f\left(b_l\right)$ необхідно відобразити в наступній симплекс-таблиці. При приведенні до канонічної форми отримаємо обмеження:

$$\sum_{j=1}^{k}\gamma_{lj}x_j - x_{k+1} = f\left(b_l\right), \tag{8.4}$$

і, перемноживши ліву і праву частини на (-1), відразу введемо змінну x_{k+1} до бази, порушивши при цьому для нього умову припустимості. Внаслідок цього для розв'язування задачі лінійного програмування на наступних кроках застосовуємо двоїстий симплекс-метод.

Розв'язування задач мішаного та лінійного програмування за допомогою методу розгалужень та границь.

Розглянемо лінійну задачу мішаного програмування. Нехай на певному кроці x_r – змінна, на яку накладене обмеження цілочисельності, і яка має на даному кроці розв'язування неціле значення x_r^*. Таким чином, дійсне значення цієї змінної повинне на даному етапі знаходитися поза межами інтервалу $[x_r^*] < x_r^* < [x_r^* + 1]$,, тобто в цьому інтервалі оптимальних розв'язків немає. Тому значення xr повинне задовільняти одну з двох нерівностей: $x_r \le [x_r^*]$ або $x_r \ge [x_r^*]+1$. Введення цих двох взаємовиключаючих умов породжує дві не пов'язані між собою задачі – тобто первісна задача розщеплюється на дві підзадачі. Таким чином, здійснюється врахування необхідних умов цілочисельності, а саме: послідовно виключаються частини многогранника припустимих розв'язків, які явно не мають точок з цілочисельними координатами.

Після цього кожна з підзадач розв'язується як класична задача ЛП. Якщо отриманий оптимальний план є цілочисельним, то подальше розгалуження не проводиться і отримуємо оптимальний розв'язок підзадачі.

Ефективність методу підвищується шляхом введення границь оцінок згори (для задачі максимізації) для підзадач, якими будуть розв'язки задач лінійного програмування для кожної з підзадач. Це дозволяє при наявності біжучого розв'язку задачі відкинути ті підзадачі, для яких значення границі менше ніж значення функції мети для біжучого розв'язку. Таким чином реалізується відтинання неперспективних підмножин розв'язків (підзадач) і зменшується обсяг роботи порівняно з методом повного перебору.

8.3. СТРУКТУРА ТА ОСНОВНІ СКЛАДОВІ МЕТОДУ РОЗГАЛУЖЕНЬ ТА ГРАНИЦЬ

Метод розгалужень та границь ґрунтується на ідеї розбиття первісної задачі на підзадачі та скорочення об'єму перебору за рахунок відсіювання підмножин неперспективних розв'язків первісної задачі, що реалізується шляхом обчислення оцінок значень функції мети (границь) або безпосереднім чином, або за допомогою розв'язування оціночних (релаксованих) задач. З одного боку, чим ближче до оптимального значення функції мети для підзадачі значення границі, тим більше вірогідність того, що об'єм перебору буде меншим, а з іншого – чим складнішими є оціночні задачі, тим більшою буде трудомісткість методу. Таким чином, в процесі синтезу алгоритму, що ґрунтується на методі розгалужень та границь, необхідно знайти оптимальний компроміс між точністю обчислення границь та складністю оціночних задач.

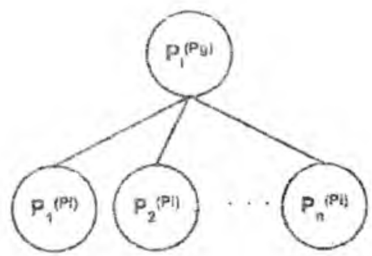
Метод розгалужень та границь є значно ширшим, ніж поняття методу розв'язування певної задачі, оскільки залишає свободу вибору в багатьох складових структурних елементах.

До основних елементів методу розгалужень та границь належать:

- ☑ дерево розгалужень;
- ☑ послаблена (релаксована) задача;
- ☑ границя;
- ☑ активні вузли дерева розгалужень;
- ☑ критерії відтинання вузлів;
- ☑ стратегія розгалуження.

Дерево розгалужень.

Процес розв'язування задачі за допомогою методу розгалужень та границь наочно представляється у вигляді динамічного дерева розгалужень $B = (D; E)$ з коренем P_0, який відповідає первісній задачі, D – множина вузлів дерева, що являють собою підзадачі, E – множина дуг дерева розгалужень. В процесі розв'язування задачі деякі вузли відтинаються як неперспективні, замість них з'являються інші, тобто дерево розгалужень є динамічним. Розгалуження є розбиттям певної підзадачі на підзадачі наступного рівня, що пов'язані з задачею-предком дугами:



$$\bigcup_{j=1}^{n} P_j^{(P_i)} = P_i^{(P_0)}.$$

$$P_k^{(P_i)} \bigcap_{k \ne l} P_l^{(P_i)} = \varnothing.$$

Послаблена (релаксована) задача.

Для кожного вузла дерева розв'язується послаблена (релаксована) задача – підзадача "занурюється" в ширший клас задач, для яких існують ефективні алгоритми їх розв'язування. Внаслідок розв'язання похідної послабленої задачі отримується "надоптимальний" розв'язок для основної підзадачі (верхня – у випадку максимізації границя значення критерію якості основної задачі). Так, наприклад, зняття умови цілочисельності в задачі лінійного програмування приводить до релаксованої – послабленої – задачі, для розв'язання якої можна застосувати симплекс-метод, тобто задача цілочисельного чи мішаного лінійного програмування „занурюється" в ширший клас задач лінійного програмування. Звичайно, отриманий внаслідок розв'язування такої задачі розв'язок буде не гіршим, ніж оптимальний розв'язок первинної задачі і може бути використаний для її оцінювання, тобто, таким чином, ми приходимо до поняття границі.

Границя.

Границя є значенням критерію якості для оптимального розв'язку послабленої задачі.

Границя вузла (підзадачі) P_i - $B(P_i)$ має наступні властивості (вважаємо, що розв'язується задача максимізації):

$\forall(i\in D):B(P_i)\geq Q^*(P_i)$, де $B(P_i)$ - значення границі, $Q^*(P_i)$ - значення критерію якості для оптимального розв'язку задачі p;

$B(P_j)\leq B(P_i)$, якщо P_j - безпосередній нащадок P_i;

$B(P_i)=Q(P_i)$, якщо P_i – лист дерева розгалужень (лист відповідає побудові повного розв'язку первісної задачі P_0).

Таким чином: границя є „надоптимальним" розв'язком задачі; значення границі уточнюється при просуванні вглиб дерева рішень – для підзадачі (вузол-нащадок) значення границі не більше (для задачі максимізації), ніж її значення для задачі, що піддавалась декомпозиції (безпосередній вузол-предок); при досягненні повного розв'язку первісної задачі значення границі є рівним значенню критерію якості для цього повного розв'язку.

Активний вузол дерева розгалужень.

Вузли дерева розгалужень (підзадачі) належать до однієї з наступних категорій:

☑ проміжні вузли, які вже розбиті на підзадачі (тобто вузли, що мають безпосередніх нащадків);

☑ листя дерева, в яких досягнутий розв'язок первісної задачі; якщо шукається один оптимальний розв'язок первісної задачі, то в дереві є один

лист, що відповідає біжучому найкращому розв'язку, в іншому випадку кількість листя дерева може бути більшою;

☑ активні вузли дерева, які можуть бути піддані декомпозиції (розбиттю на підзадачі).

Вузол P належить до множини активних A, якщо виконуються наступні умови:

☑ P не визначає повного припустимого розв'язку первісної задачі P_0;

☑ в даний момент (на біжучому етапі розв'язування задачі) не може бути встановлена справедливість умови $O(P)\cap O(P_0)=\varnothing$, де $O(P)$ – множина оптимальних розв'язків P, $O(P_0)$ – відповідно P_0, тобто не встановлено, чи належить підзадача P до таких, в яких гарантовано немає оптимального розв'язку;

☑ вузол P не розбивався на підзадачі.

Критерії відтинання вузлів.

Для забезпечення ефективності методу розгалужень та границь необхідно відтинати якомога більш вузлів, якнайближчих до кореня. Це дозволяє відразу виключити з розгляду значні підмножини розв'язків та зменшити об'єм перебору. Для відтинання неперспективних вузлів використовується істинність умови $O(P)\cap O(P_0)=\varnothing$. Ця умова завжди виконується, якщо виконується хоча б одна з наступних умов:

☑ перетин множини припустимих розв'язків для первісної задачі $X(P_0)$ та підзадачі $X(P)$ – пуста множина: $X(P_0)\cap X(P)=\varnothing$ (це свідчить про те, що в підзадачі P немає ні одного припустимого розв'язку первісної задачі P_0);

☑ виконується умова відтинання вузла $B(P)\leq Q^*$, де Q^* значення критерію для найкращого біжучого (повного) розв'язку задачі (в цьому випадку серед припустимих розв'язків підзадачі P гарантовано не знайдеться кращого, ніж вже знайдений біжучий найкращий розв'язок);

☑ існує вузол $P_k\in D$, що не є предком P, який домінує над ним, тобто встановлено, що $Q^*(P_k)\geq Q^*(P)$, де $Q^*(P_k)$ – оптимальний розв'язок задачі P_k, а $Q^*(P)$ – відповідно P (для встановлення цього факту зовсім не обов'язково знати відповідні значення критеріїв);

☑ окрім того, відтинається вузол, який піддавався декомпозиції і у якого відтяті всі безпосередні нащадки.

☑ Якщо для деякого вузла встановлено, що $B(P_i)=Q(P_i)$, то P_i далі не розбивається, тому що знайдений один з повних розв'язків задачі. Коли ж $Q^*(P_i)\geq Q^*$, то найкращим біжучим розв'язком стає знайдений.

$Q^* = Q^*(P_i)$, а вузол, що відповідав попередньому біжучому найкращому розв'язку, відтинається.

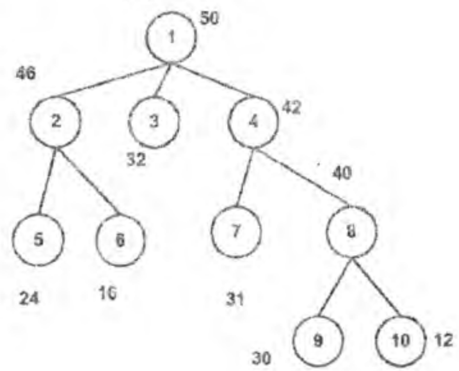
Стратегія розгалуження.

Стратегія розгалуження визначає спосіб побудови динамічного дерева розгалужень та вибору наступного вузла з множини активних для декомпозиції. Структуру стратегії розгалуження представимо у вигляді двох відображень: $G: A \Rightarrow C$, $C \subseteq A$, та $h: C \Rightarrow R^1$, де R^1 – множина числових значень. За допомогою відображення G виділяється множина вузлів – кандидатів для декомпозиції з числа активних вузлів, а за допомогою h – вибір одного з них за допомогою кількісного критерію $P^*_i = \arg \underset{P_i \in C}{Max}\, h(P_i)$. Розглянемо наступні стратегії розгалуження: за найкращим значенням границі; вглиб дерева розгалужень; вшир дерева розгалужень; розгалуження нагонами.

В стратегії розгалуження за найкращим значенням границі на кожному кроці з числа активних обирається вузол з найбільшим (задача на пошук максимуму) значенням границі.

В стратегії розгалуження за найкращим значенням границі на кожному кроці з числа активних обирається вузол з найбільшим (задача на пошук максимуму) значенням границі,

$$C \equiv A,\ P^*_i = \arg \underset{P_i \in C}{Max}\, B(P_i). \tag{8.5}$$



В прикладі біля вузлів проставлені значення границь. Номери вузлів відображають порядок породження нащадків згідно зі стратегією розгалуження за кращим значенням границі. У багатьох випадках ця стратегія приводить до найкращого розв'язку за найкоротший час, але внаслідок тенденції до росту дерева вшир необхідні значні ресурси оперативної

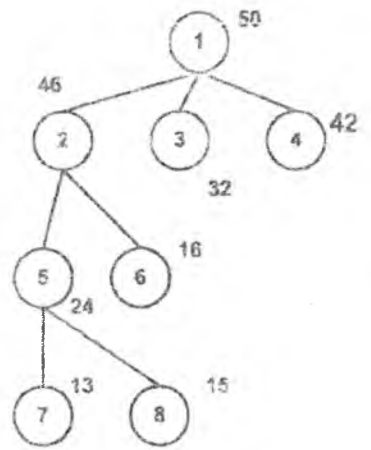
пам'яті для ефективної роботи відповідних алгоритмів. Крім того, необхідний досить значний час для знаходження першого біжучого найкращого розв'язку, тому в системах реального часу застосовувати її не варто.

В стратегії розгалуження вглиб дерева розгалужень на кожному кроці визначаються вузли, які знаходяться найдальше від кореня (віддаль від кореня дерева вимірюється як число ребер, що зв'язують вузол з коренем), і серед них обирається вузол з найбільшим значенням границі,

$$C = \left\{ P_i \in A \middle| d(P_i) = Max \left\{ d(P_j) \middle| P_j \in A \right\} \right\}, P^*_i = \arg \underset{P_i \in C}{Max}\, B(P_i), \tag{8.6}$$

$$P^*_i = \arg \underset{P_i \in C}{Max}\, B(P_i).$$

Після того, як досягнутий розв'язок, відбувається повернення на рівень догори, і процедура повторюється.



Ця стратегія дозволяє за найкоротший час отримати найкращий біжучий розв'язок, але витрачає багато часу на його покращення внаслідок пошуку на сусідніх рівнях та гілках дерева розгалужень.

В стратегії розгалуження вшир дерева розгалужень на кожному кроці визначається множина вузлів, що розташовані найближче до кореня, і серед них здійснюється вибір вузла з найбільшим значенням границі:

$$C = \left\{ P_i \in A \middle| d(P_i) = Min \left\{ d(P_j) \middle| P_j \in A \right\} \right\}, P^*_i = \arg \underset{P_i \in C}{Max}\, B(P_i). \tag{8.7}$$

Ця стратегія приводить до переважного росту дерева розгалужень вшир і на практиці не застосовується.

Стратегія розгалуження нагонами є комбінацією двох найпоширеніших стратегій – розгалуження за найкращим значенням границі та вглиб дерева. Ця стратегія спрямована на синтез позитивних особливостей вищенаведених стратегій – по-перше, якомога швидше досягнути повного розв'язку первісної задачі, і, по-друге, витрачати на пошук наступного біжучого оптимального розв'язку, що достатньо віддалений від біжучого, якомога менше часу. Згідно з цією стратегією, декомпозиція підзадач відбувається за стратегією вглиб дерева розгалужень до моменту, поки не буде досягнутий повний розв'язок. Після цього вибір наступного вузла для розгалуження відбувається згідно зі стратегією розгалуження за найкращим значенням границі, і знову застосовується стратегія розгалуження вглиб дерева.

Слід зазначити, що переваги чи недоліки тієї чи іншої стратегії не є абсолютними, і в значній мірі залежать від особливостей конкретних типів задач.

ПРИКЛАДИ

Приклад 8.1. Метод Ґоморі

Розв'язати задачу цілочисельного програмування за допомогою методу Ґоморі: $x_1 + 4x_2 \Rightarrow Max$: $2x_1 + x_2 \le 19/3$; $x_1 + 3x_2 \le 4$; x_1, x_2 – цілі.

Розв'язання.

Приведемо задачу до канонічної форми:

$$x_1 + 4x_2 \Rightarrow Max;\ 2x_1 + x_2 + x_3 = 19/3;\ x_1 + 3x_2 + x_4 = 4.$$

У результаті задача, що здавалася чисто цілочисельною, виявляється задачею змішаного програмування, оскільки на x_3, x_4 не накладаються обмеження цілочисельності.

На кроці 1 розв'язуємо задачу за допомогою звичайного симплекс-методу.

В останній симплекс-таблиці обираємо останній рядок, що відповідає x2 і розраховуємо відповідні значення коефіцієнтів додаткового обмеження.

xb	cb	P0	1	4	0	0	
			P1	P2	P3	P4	
x3	0	19/3	2	1	1	0	19/3
x4	0	4	1	3	0	1	4/3
Q	=	0	-1	-4	0	0	
x3	0	5	5/3	0	1	-1/3	
x2	4	4/3	1/3	1	0	1/3	←
Q	=	16/3	1/3	0	0	4/3	

		b2	x1	x2	x3	x4	
x2		4/3	1/3	1	0	1/3	←
		f(b2)=1/3	ціле f(a21)<=f(b2) 1/3<=1/3 f(a21)=1/3	ціле f(a22)<=f(b2) 0<=1/3 f(a22)=0	неціле a23>=0 0>=0 a23=0	неціле a24>=0 1/3>=0 a24=1/3	

Формуємо додаткове обмеження:

$1/3x_1$	$+0x_2$	$+0x_3$	$+1/3x_4$		$>=1/3$
x_1	$+0x_2$	$+0x_3$	$+x_4$		$>=1$
x_1	$+0x_2$	$+0x_3$	$+x_4$	$-x_5$	$=1$
$-x_1$	$-0x_2$	$-0x_3$	$-x_4$	$+x_5$	$=-1$

Переходимо до наступного кроку – доповнюємо симплекс-таблицю новим обмеженням – рядком та базовою змінною – стовпчиком і розв'язуємо задачу за допомогою двоїстого симплекс-методу.

xb	cb	P0	1	4	0	0	0
			P1	P2	P3	P4	P5
x3	0	5	5/3	0	1	-1/3	0
x2	4	4/3	1/3	1	0	1/3	0
x5	0	-1	-1	0	0	-1	1
Q	=	16/3	1/3	0	0	4/3	0
x3	0	10/3	0	0	1	-2	5/3
x2	4	1	0	1	0	0	1/3
x1	1	1	1	0	0	1	-1
Q	=	5	0	0	0	1	1/3

Отримано оптимальний розв'язок задачі. Стоп.

Геометрично процес розв'язування представлено на рис.8.2.

Приклад 8.2. Метод розгалужень та границь для розв'язування цілочисельних задач.

Необхідно розв'язати наступну задачу, використовуючи метод розгалужень та границь.

Рис. 8.2. Геометричний процес розв'язування прикладу 8.1

$$2x_1+3x_2 \Rightarrow Max,\ 5x_1+7x_2 \le 35,\ 4x_1+9x_2 \le 36,\ x_1,x_2 \ge 0\,.$$

Розв'язування.

Застосуємо до цієї задачі метод розгалужень та границь. Оскільки в задачі є дві змінні, то розв'язуватимемо виникаючі задачі лінійного програмування, що надають значення верхніх оцінок, графічно.

Спочатку (корінь дерева) розв'язуємо задану задачу 1, знявши умови цілочисельності. В результаті отримаємо розв'язок релаксованої задачі Q=14 8/17, x_1=3 12/17, x_2=2 6/17. Розбиваємо задачу на 2 підзадачі за змінною x_2, що може бути або більшою за 3, або ж меншою за 2. Відповідно до цього отримаємо задачі 2 та 3. Для подальшої декомпозиції обираємо задачу 3, оскільки вона має більше значення границі, ніж задача 2. Розбиваючи задачу 3 на підзадачі 4 та 5, отримуємо біжучий оптимальний розв'язок для задачі 5, Q=14, x_1=4, x_2=2. Активними вузлами на даний момент є вузли 2 та 4. Вузол 2 відтинаємо, оскільки для нього значення границі Q=13 1/2, що менше ніж 14.

Далі розбиваємо задачу 4 на підзадачі 6 та 7. В підзадачі 6 припустимі розв'язки відсутні, тому продовжуємо і розбиваємо задачу 7 на підзадачі 8 та 9. Вузол 8 відтинаємо, оскільки для нього значення границі 13, що менше ніж для біжучого найкращого розв'язку 14 1/7. І, піддаючи декомпозиції задачу 9, отримуємо, що для задачі 10 припустимих розв'язків немає, а для задачі 11 значення критерію оптимальності не більше ніж для біжучого найкращого розв'язку.

Відтинаємо вузли 10 та 11, оскільки від вузла 9 задачі відтяті, то і його відтинаємо, надалі вузол 7 – також не має нащадків (вузол 8 був відтятий раніше), аналогічно далі відтинаємо 4, 3 і врешті-решт зупиняємося – первісна задача – корінь дерева – не має більше нащадків, тобто відсутні активні вузли, а це означає, що біжучий оптимальний розв'язок є оптимальним.

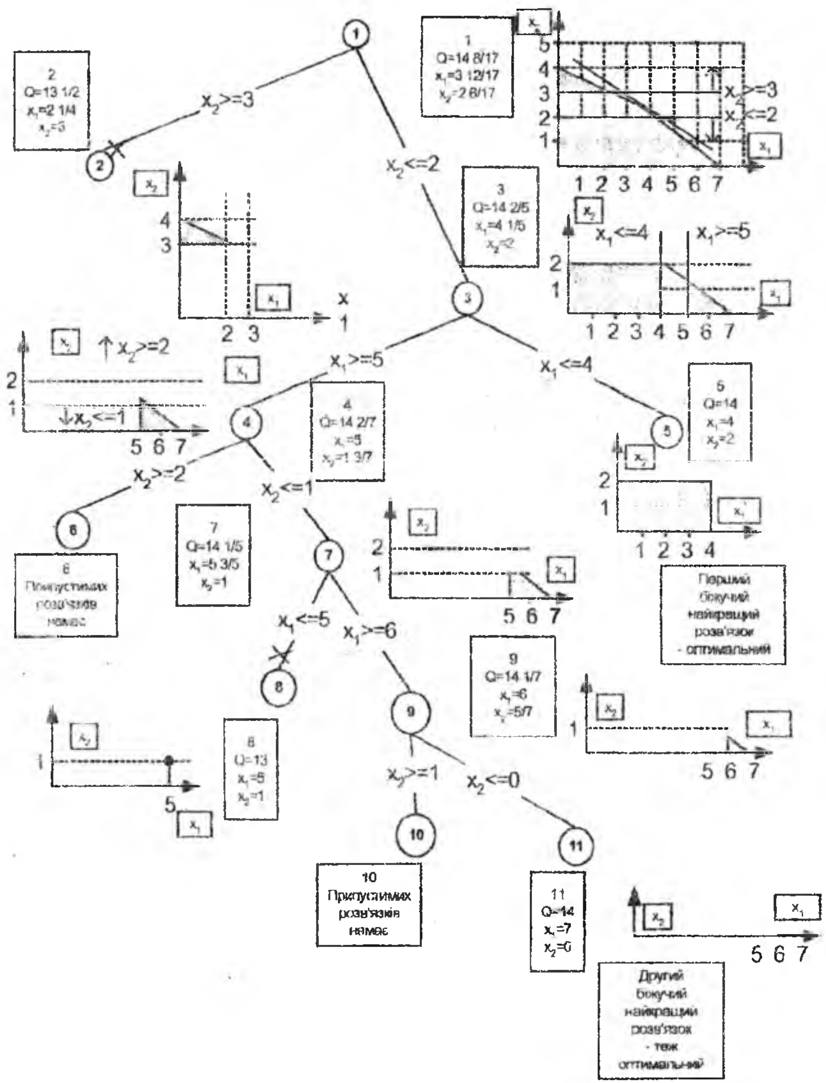
Перебіг процесу розв'язування зображений нижче.

РЕЗЮМЕ

8.1. Структурні особливості задач цілочисельного лінійного програмування служать основою для розробки ефективних алгоритмів розв'язування деяких задач. В загальному випадку цілочисельні задачі є NP-повними, тобто ефективних алгоритмів їх розв'язування не існує. Для розв'язування цілочисельних задач застосовуються методи, що належать до однієї з наступних груп: методи відтинань; комбінаторні методи, евристичні методи.

8.2. Метод Гоморі належить до методів відтинань і застосовується до розв'язування лінійних задач. Введення двох взаємновиключаючих умов в методі розгалужень та границь породжує дві не пов'язані між собою задачі – тобто первісна задача розщеплюється на дві підзадачі. Таким чином, здійснюється врахування необхідних умов цілочисельності, а саме: послідовно виключаються частини многогранника припустимих розв'язків, які завідомо не мають точок з цілочисельними координатами.

8.3. Метод розгалужень та границь ґрунтується на ідеї розбиття первісної задачі на підзадачі та скорочення об'єму перебору за рахунок відсіювання підмножин неперспективних розв'язків первісної задачі, що реалізується шляхом обчислення оцінок значень функції мети (границь)



або безпосереднім чином, або за допомогою розв'язування оціночних (релаксованих) задач. З одного боку, чим ближче до оптимального значення функції мети для підзадачі значення границі, тим більше вірогідність того, що об'єм перебору буде меншим, а з іншого — чим складнішими є оціночні задачі, тим більшою буде процехмісткість методу. Таким чином, в процесі синтезу алгоритму, що ґрунтується на методі розгалужень та границь, необхідно знайти оптимальний компроміс між точністю обчислення границь та складністю оціночних задач. До основних елементів методу розгалужень та границь належать: дерево розгалужень, послаблена (релаксована) задача; границя; активні вузли дерева розгалужень; критерії відтинання вузлів; стратегія розгалуження

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 8.1. Метод Гоморі та метод розгалужень та границь.

Розв'язати задачу цілочисельного програмування за допомогою методу Гоморі та гілок і границь.

$$4x_1 + 5x_2 \Rightarrow Max$$
$$3x_1 + 2x_2 \le 10$$
$$x_1 + 4x_2 \le 11$$
$$3x_1 + 3x_2 \le 14,$$
$$x_1,\ x_2 \ge 0,\ \text{цілі.}$$

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Наведіть формальні постановки задач про наплічник та про ефективну експедицію.

2. Які існують основні групи методів для розв'язування цілочисельних задач?

3. До яких задач застосовується метод Гоморі та в чому його суть?

4. Чому як один з кроків в методі Гоморі застосовується двоїстий симплекс-метод?

5. В чому полягає ідея розв'язання мішаної задачі лінійного програмування за допомогою методу розгалужень та границь?

6. Які основні структурні елементи методу розгалужень та границь?

7. Навіщо в методі розгалужень та границь використовується релаксація?

8. Яким чином інтерпретується дерево розгалужень?

9. Які властивості границі в методі розгалужень та границь?

10. До яких категорій належать вузли дерева розгалужень?

11. Які умови виконуються для активних вузлів дерева розгалужень?

12. Перерахуйте критерії відтинання вузлів в методі розгалужень та границь.

ТЕМА 9. ПРАКТИЧНІ РЕАЛІЗАЦІЇ МЕТОДУ РОЗГАЛУЖЕНЬ ТА ГРАНИЦЬ

Розв'язати практично важливі задачі булевого програмування можна, лише засвоївши відповідні алгоритми. В залежності від специфіки задачі можуть бути розроблені і застосовані різні варіанти методу розгалужень та границь. Так, для багатовимірної задачі про наплечник, яка має чітку економічну інтерпретацію, розвинутий значно простіший алгоритм, ніж до розв'язування загальної задачі булевого програмування. Окрім того, слід вміти приводити задачі цілочисельного програмування до булевих, оскільки це в деяких випадках дозволяє ефективно розв'язувати задачі середньої розмірності. Однією з класичних задач, яка тим не менше знайшла широке застосування як в економічних задачах, так і в задачах планування роботи мікропроцесорів, є задача про комівояжера, яка за формальною постановкою надзвичайно подібна до задачі про призначення за винятком однієї не зовсім приємної особливості – наявності системи додаткових обмежень, що забезпечують побудову глобального циклу замість множини локальних – а це, своєю чергою, відразу позиціонує цю задачу, як NP-повну, для розв'язання якої проте розроблені достатньо ефективні варіанти методу розгалужень та границь.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ:

знати:⇨алгоритм розв`язування багатовимірної задачі про наплечник та його економічну інтерпретацію;⇨ основні ідеї скорочення перебору при розв`язуванні загальної задачі булевого програмування; ⇨способи приведення цілочисельних задач до булевих;⇨обґрунтування реалізації методу розгалужень та границь до алгоритму розв`язування задачі про комівояжера;

вміти:⇨розв'язувати багатовимірні задачі про наплечник за допомогою методу розгалужень та границь;⇨розв'язувати загальні задачі булевого програмування за допомогою адитивного алгоритму;⇨розв`язувати задачу про комівояжера за допомогою методу розгалужень та границь;⇨зводити цілочисельні задачі до булевих.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ задачі булевого програмування
- ☑ задачі мішаного програмування
- ☑ задача про наплечник
- ☑ задача про комівояжера
- ☑ комбінаторні методи
- ☑ евристичні методи
- ☑ методи відтинань
- ☑ границя
- ☑ дерево розгалужень
- ☑ релаксація
- ☑ активний вузол
- ☑ критерій відтинання вузлів
- ☑ стратегія розгалуження
- ☑ метод розгалужень та границь

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

9.1.Розв'язання багатовимірної задачі про наплечник за допомогою методу гілок та границь.

9.2.Загальна постановка задачі булевого програмування. Алгоритм Балаша.

9.3. Методи приведення цілочисельних задач до булевих.

9.4. Задача про комівояжера.

9.1. РОЗВ'ЯЗАННЯ БАГАТОВИМІРНОЇ ЗАДАЧІ ПРО НАПЛЕЧНИК ЗА ДОПОМОГОЮ МЕТОДУ ГІЛОК ТА ГРАНИЦЬ

Багатовимірна задача про наплечник має наступний вигляд:

$$\sum_{j=1}^{n} c_j x_j \Rightarrow Max,\ \sum_{j=1}^{n} a_{ij} x_j \le b_i,\ i=\overline{1,m},\ j=\overline{1,n},\ \forall x_j \in \{0;1\},$$

$$\forall (i,j):\ a_{ij} \ge 0, c_j \ge 0. \qquad (9.1)$$

Таким чином, у багатовимірній задачі про наплечник значення всіх коефіцієнтів є невід'ємними. Ця задача є не чим іншим, як варіантом задачі виробничого планування за умови, що вироби випускаються дискретно. У цьому випадку c_j - прибуток, що його корпорація отримує за одиницю j-го виробу, a_{ij} кількість i-го ресурсу, що використовується для

виробництва одиниці j-го виробу, b_i - величина запасу i-го ресурсу, x_j кількість виробів j-го типу (ціла). Така задача виникає в тому випадку, коли випускаються унікальні вироби в невеликих кількостях (колекційні автомобілі, літаки), і заокруглення результатів, отриманих за допомогою симплекс-методу, до цілих приведе до значних похибок (пере- чи недовикористання певних ресурсів).

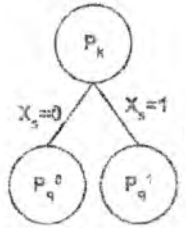
До задач з цілочисельними булевими змінними схема методу розгалужень та границь може бути успішно застосована, причому як до загальних задач, так і до розширень задачі про наплечник з декількома обмеженнями.

Розглянемо застосування методу розгалужень та границь до розширеної задачі про наплечник (тобто задачі, для якої $\forall (i,j):\ a_{ij} \ge 0, c_j \ge 0$). Для розв'язання задачі насамперед впорядкуємо змінні в порядку спадання значень коефіцієнтів функції мети при них (наприклад, якщо в задачі є 5 змінних і значення коефіцієнтів функції мети становлять $c_1=5, c_2=3, c_3=6, c_4=1, c_5=9$, то впорядкування буде $x_5 \succ x_3 \succ x_1 \succ x_2 \succ x_4$. В цьому порядку реалізуватимемо розгалуження з метою якнайшвидшого відтинання підмножин неперспективних розв'язків якомога раніше. Таким чином, дерево розгалужень матиме 5 рівнів, що відповідатимуть кожен окремій змінній задачі у відповідності до побудованого впорядкування. В якості стратегії розгалуження використовуватимемо стратегію вглиб дерева. Окрім того, для цієї задачі існує декілька можливостей для обчислення верхньої границі.

З одного боку, верхню границю нащадка можна обчислювати як верхню границю предка, зменшену на значення коефіцієнта функції мети при змінній, яка фіксується при даному розгалуженні в нулі, з іншої розв'язуючи підзадачу як задачу лінійного програмування без врахування умов цілочисельності (не слід при цьому забувати, що значення кожної зі змінних може бути лише між нулем та одиницею). Значення верхніх границь, визначених за останнім методом, будуть ближчими до дійсних оптимальних значень функції мети, ніж ті, які визначені за першим способом. Однак у цьому випадку для отримання значення границі необхідно розв'язати задачу лінійного програмування, що порівняно з операцією віднімання вимагає значно більшої кількості обчислень.

Розглянемо, яким чином розраховуватимуться значення верхньої границі та значення обмежень безпосередньо без розв'язування допоміжної задачі лінійного програмування для багатовимірної задачі про наплечник.

Для кореня дерева розгалужень значення верхньої границі $B=\sum_{j=1}^{n} c_j$, біжучі значення коефіцієнтів правих частин обмежень (залишки ресурсів, якщо розглядати багатовимірну задачу про наплечник як задачу планування виробництва з цілочисельними змінними) рівні їх початковим значенням $b=(b_1,...,b_m)^T$.



Далі обчислення продовжуються для безпосередніх нащадків P_q^0, P_q^1 предка P_k (задача розбивається на дві підзадачі шляхом присвоєння значення 0 чи 1 відповідній цьому рівневі дерева булевій змінній x_s) за наступними формулами:

$$B\left(P_q^0\right)=B\left(P_k\right)-c_s, \ \forall(j=\overline{1,m}): b_j\left(P_q^0\right)=b_j\left(P_k\right), \tag{9.2}$$

$$B\left(P_q^1\right)=B\left(P_k\right), \ \forall(j=\overline{1,m}): b_j\left(P_q^1\right)=b_j\left(P_k\right)-a_{ij}, \tag{9.3}$$

де P_q^0, P_q^1 - безпосередні нащадки P_k, отримані у результаті фіксації x_s в нулі чи одиниці відповідно, $B\left(P_q^0\right), B\left(P_k\right)$ – значення верхніх границь для вузлів P_q^0, P_k відповідно, $b_j\left(P_q^0\right), b_j\left(P_k\right)$ – біжучі значення правих частин обмежень для вузлів P_q^0, P_k відповідно. Критерієм відтинання вузла може бути порушення обмежень – якщо для вузла виконується умова $\exists\left(u\in\overline{1,m}\right): b_u(P)<0$, тобто знайдеться хоча б одне негативне значення залишку ресурсів, то вузол P відтинається (виконання цієї умови означає порушення хоча б одного обмеження). Також відтинання вузла P реалізується, коли виконується умова $B(P)\le Q^*$, де $B(P)$ – значення верхньої границі для цього вузла, Q^* - значення критерія якості для біжучого найкращого розв'язку.

В прикладі 3 детально розглянуто роботу алгоритму, побудованого на цих міркуваннях.

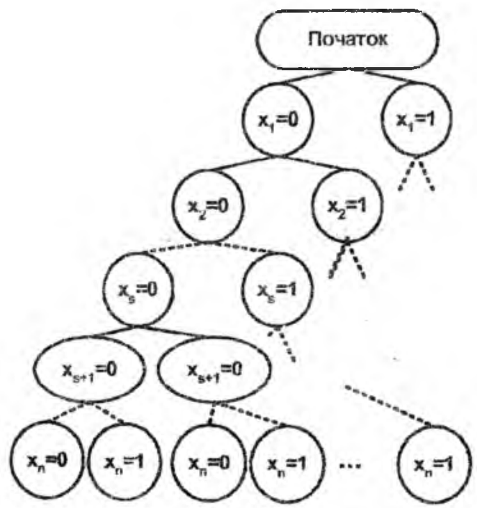
9.2. ЗАГАЛЬНА ПОСТАНОВКА ЗАДАЧІ БУЛЕВОГО ПРОГРАМУВАННЯ. АЛГОРИТМ БАЛАША

Задача *булевого* лінійного програмування має наступний вигляд:

$$\sum_{j=1}^{n} c_j x_j \Rightarrow Max, \ \sum_{j=1}^{n} a_{ij}x_j \le b_i, \ i=\overline{1,m}, \ j=\overline{1,n}, \ \forall x_j \in \{0;1\}. \tag{9.4}$$

У випадку, коли значення коефіцієнтів в булевій задачі можуть мати будь-який знак, тобто і додатними, і від'ємними, для розв'язання можна застосувати алгоритм Балаша.

Розглянемо деяку підмножину x_j, у якому кожної змінній x_j, поставлено у відповідність визначене числове значення 0 чи 1. Така підмножина називається ***частковим розв'язком***. Змінні x_j, що не входять у частковий розв'язок, називаються ***вільними змінними***. Будь-який вибір числових значень вільних змінних називається доповненням відповідного часткового розв'язку. Якщо частковий розв'язок містить s змінних, то існує 2^{n-s} доповнень.



Формально процес пошуку оптимального розв'язку може бути представлений у вигляді конструювання деякого дерева варіантів, де кожна вершина (не остаточна) відповідає певному частковому розв'язку, а можливі його доповнення породжують гілки дерева.

Крім того, кожному частковому розв'язку відповідає деяка підзадача, що належить до основного списку.

Припустимо, що відомо нижню досягнуту оцінку оптимального значення цільової функції і зафіксоване припустиме рішення, що дає цю оцінку (найкращий біжучий припустимий розв'язок задачі). Тоді, якщо для деякого часткового розв'язку якимось чином можна довести, що не існує припустимого доповнення, що має значення цільової функції, яке перевищує нижню поточну оцінку, то немає необхідності його продовження (тобто воно відкидається). У цьому випадку говорять, що часткове рішення прозондовано. При зондуванні часткового рішення, що містить s змінних, неявним чином перебирається 2^{n-s} можливих значень.

Для розв'язання задач такого типу застосовується адитивний алгоритм Балаша, що використовує ідеї методу розгалужень та границь з врахуванням специфіки задачі булевого програмування.

Стосовно даного алгоритму принцип оптимальності формулюється наступним чином.

Для заданого часткового розв'язку значення інших змінних повинні обиратися таким чином, щоб доповнення цього розв'язку було оптимальним. За відсутності таких значень інших змінних, котрі дають припустимий розв'язок за умовами виконання обмежень, чи у випадку, коли отримані оптимальні значення цих змінних приводять до гіршого розв'язку (за значенням цільової функції), ніж знайдений раніше, не існує оптимального розв'язку, що містить задане часткове рішення.

На будь-якій ітерації n відома нижня оцінка $Q_0^{(n)}$ оптимального значення цільової функції (значення Q_0 можна обирати так само, як і в методі розгалужень та границь). На кожній ітерації формується основний список задач, у якому кожній задачі відповідає певний частковий розв'язок. На ітерації 1 основний список містить 2 задачі, отримані в результаті вибору x_k, причому приймається, що частковий розв'язок однієї задачі $x_k = 0$, а іншої $x_k = 1$.

Існує кілька способів, які можна застосувати, щоб установити, чи існує яке-небудь доповнення, що забезпечує значення функції мети, яке буде перевищувати поточну нижню оцінку.

Наприклад, якщо задача містить обмеження

$-1x_1 + 2x_2 - 5x_3 + 4x_4 - 3x_5 \le 0$,

то не існує доповнення часткового рішення $x_1 = 1$, $x_3 = 0$, $x_4 = 1$, що задовольняє цьому обмеженню.

Припустимо, що критерій якості описується виразом:

$Q = 1x_1 + 2x_2 + 4x_3 + 3x_4 - 1x_5$ і що $Q_0^{(n)} = 8$.

Тоді, щоб поліпшити значення Q, доповнення повинне задовольняти обмеженню:

$1x_1 + 2x_2 + 4x_3 + 3x_4 - 1x_5 \ge Q_0^{(n)} + 1 = 9$.

Жодне доповнення часткового розв'язку $x_3 = 0$, $x_5 = 0$ не є припустимим за умовою справдження цього обмеження.

Формалізуємо наші міркування в наступний спосіб.

При заданому частковому розв'язку $x_j = \delta_j$, $j \in I$ представимо обмеження задачі у наступному вигляді:

$$\sum_{j \notin I} a_{ij} x_j \le b_i - \sum_{j \in I} a_{ij} x_j \,, \quad i = 1, 2, \ldots, n \,, \qquad (9.5)$$

де I – множина індексів змінних часткового розв'язку; $N \setminus I$ – індекси вільних змінних, і кожній змінній x_j часткового розв'язку відповідає певне значення, що входить у знак $\sum$ у правій частині цього обмеження, а коефіцієнти в обмеженні при $i = 0$ (функція мети) рівні $a_{0j} = -c_j$ та $b_0 = -Q_0^{(n)} - 1$.

Тоді не існує припустимого доповнення, якому відповідає значення цільової функції, що перевершує нижню оцінку, якщо

$$\sum_{j \notin I} \min\{a_{ij}, 0\} > b_i - \sum_{j \in I} a_{ij} x_j \,. \qquad (9.6)$$

Як бачимо, у лівій частині є сума всіх негативних коефіцієнтів при вільних змінних. Якщо ця сума більше правої частини нерівності, то навіть, покладаючи $x_j = 1$ для всіх тих вільних змінних, для яких $a_{ij} < 0$ неможливо виконати i-те обмеження.

Іншим важливим моментом є те, що при заданому частковому розв'язку іноді вдається визначити, яке значення повинне мати вільна змінна при будь-якому припустимому доповненні та у випадку, коли значення цільової функції перевершує поточну нижню оцінку.

Зокрема, для будь-якої вільної змінної x_k у випадку, коли:

$$\sum_{j \notin I} \min\{a_{ij}, 0\} + |a_{ik}| > b_i - \sum_{j \in I} a_{ij} x_j \qquad (9.7)$$

при будь-якому i, $x_k = 0$, якщо $a_{ij} > 0$ і $x_k = 1$, якщо $a_{ij} < 0$.

Застосування алгоритму Балаша потребує представлення булевої задачі у наступній формі, що задовольняє наступним вимогам:

☑ У виразі функції мети всі коефіцієнти повинні бути невід'ємними, а сама функція повинна підлягати мінімізації.

☑ Всі обмеження повинні бути типу $\leq$, можливо з від'ємними значеннями у правій частині. Надалі ці обмеження перетворюються у рівності шляхом введення додаткових змінних у ліві частини обмежень.

Зрозуміло, що будь-яка булева задача може бути подана у такому вигляді.

Якщо критерій якості підлягає максимізації, тоді треба його помножити на -1. Так само, якщо обмеження типу $\geq$, помножити на -1.

Щоб позбутися від'ємних коефіцієнтів функції мети, необхідно здійснити перехід до нових змінних:

$$x_i = \begin{cases} y_i, x_i \geq 0 \\ 1 - y_i, x_i < 0, \end{cases} \tag{9.8}$$

$i = \overline{1, n}$, де n – кількість змінних.

Описання адитивного алгоритму.

Нехай вже проведено k ітерацій і знайдені поточна нижня оцінка оптимального розв'язку $Q_0^{(k)}$ та список задач. Опишемо $(k+1)-y$ ітерацію:

Крок 1. Припинити обчислення, якщо основний список задач порожній. У протилежному випадку вибрати задачу з основного списку і викреслити її з нього.

Крок 2. Якщо можна знайти вільні змінні, котрі повинні мати певні значення при будь-якому доповненні, коли значення функції мети перевершує $Q_0^{(k)}$, то відповідним чином розширити обраний частковий розв'язок. Якщо можна установити, що не існує припустимого доповнення, у якого значення функції мети перевершує $Q_0^{(k)}$, тоді покласти $Q_0^{(k+1)} = Q_0^{(k)}$ і повернутися до першого кроку. У протилежному випадку перейти до третього кроку.

Крок 3. Якщо розширений частковий розв'язок є повним (тобто містить усі n змінних), зафіксувати його, прийнявши $Q_0^{(k+1)}$ рівним відповідному значенню функції мети і повернутися до першого кроку. У протилежному випадку перейти до четвертого кроку.

Крок 4. Обрати будь-яку вільну змінну x_s, що не входить у (розширений) частковий розв'язок. Внести дві нові задачі в основний список. В одній з них покласти $x_s = 0$ у розширеному частковому розв'язку, а в іншому $x_s = 1$. Покласти $Q_0^{(k+1)} = Q_0^{(k)}$ і повернутися до першого кроку.

Зауваження.

При припиненні виконання операцій алгоритму у випадку, коли зафіксовано припустимий розв'язок, що дає значення $Q_0^{(k)}$, цей розв'язок

є оптимальним. У протилежному випадку припустимого розв'язку не існує. Алгоритм забезпечує збіжність за скінчене число ітерацій.

9.3. МЕТОДИ ПРИВЕДЕННЯ ЦІЛОЧИСЕЛЬНИХ ЗАДАЧ ДО БУЛЕВИХ

Будь-яку задачу цілочисельного програмування можна привести до лінійної задачі булевого програмування.

Для цього задача спочатку приводиться до поліноміальної булевої, а потім до лінійної булевої.

Приведення до поліноміальної задачі можна здійснити двома способами:

☑ замінити кожну з цілочисельних змінних x_j сумою бінарних $x_j = \sum_{i=1}^{R_j} y_{ij}$; недоліком такого представлення є значна кількість булевих змінних;

☑ замінити x_j двійковим представленням $x_j = \sum_{i=1}^{P} 2^i y_{ij}$, де P найбільше ціле, для якого $R_j \leq 2^{P+1} - 1$ за умови $x_j \leq R_j$.

Своєю чергою булева поліноміальна задача завжди може бути приведена до лінійної:

☑ здійснюємо заміну $(x_j)^n \Rightarrow x_j$;

☑ якщо наявні добутки двох різних змінних, то заміняємо їх додатковою змінною та парою обмежень – $x_j x_k \Rightarrow x_{jk}$; $x_j + x_k - x_{jk} \leq 1$; $-x_j - x_k + 2x_{jk} \leq 0$,

x_j	x_k	x_{jk}
0	0	0
0	1	0
1	0	0
1	1	1

правильність такої заміни перевіряється безпосередньо за допомогою складання таблички та підстановки результатів в обмеження;

☑ в загальному випадку спочатку добуток багатьох змінних заміниться однією змінною, потім з добутком виконується перша заміна та додаються два обмеження:

$$x_Q = \prod_{j\in Q}(x_j)^{n_j};\ \prod_{j\in Q}(x_j)^{n_j} \Rightarrow \prod_{j\in Q} x_j,$$

$$\begin{cases} \sum\limits_{j\in Q} x_j - x_Q \le card(Q) - 1 \\ -\sum\limits_{j\in Q} x_j + x_Q card(Q) \le 0 \end{cases} \qquad (9.9)$$

Звичайно, можливість приведення цілочисельних задач до булевого типу використовується далеко не завжди, оскільки при цьому суттєво зростає розмірність, а тому для задач певних спеціалізованих типів розроблені спеціалізовані ж алгоритми їх розв'язання.

9.4. ЗАДАЧА ПРО КОМІВОЯЖЕРА

Задача про комівояжера (мандрівного торговця) формулюється наступним чином. Комівояжер з запасом деякого товару повинен скласти маршрут подорожі деякою множиною міст, віддалі між якими відомі, таким чином, щоб побувати в кожному місті лише один раз, відвідати кожне місто і забезпечити мінімальну довжину маршруту. Ця задача знайшла багато практичних застосувань, і, незважаючи на простоту формулювання, не розв'язується за допомогою простих алгоритмів, як подібні на неї зовні задача про найкоротший шлях та задача про багатополюсний ланцюг. Таким чином, задача комівояжера є тісно пов'язаною з задачею пошуку гамільтонового контуру (за іменем ірландського математика Вільяма Гамільтона, що у 1859 р. першим почав вивчення цих задач) найменшої загальної довжини в графі.

Цей зв'язок з'ясовується за допомогою наступного твердження. Якщо для кожної пари вершин (x, y) і будь якої іншої вершини z графа G виконується умова $\forall(z \ne x, z \ne y): d(x, y) \le d(x, z) + d(z, y)$, де $d(x, y)$ міра віддалі, то гамільтонів контур є розв'язком загальної задачі комівояжера на графі G (якщо розв'язок взагалі існує).

Ця умова стверджує лише те, що безпосередня віддаль від x до y ніколи не перевищує віддалі від x до y через будь-яку іншу вершину (нерівність трикутника).

Сформулюємо задачу комівояжера у вигляді задачі цілочисельного лінійного програмування.

$$\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij} x_{ij} \to Min,\ c_{ii} = \infty,\ i = 1, 2, \ldots, n.$$

$$\sum_{j=1}^{n} x_{ij} = 1,\ i = \overline{1,n}, \qquad (9.10)$$

$$\sum_{i=1}^{n} x_{ij} = 1,\ j = \overline{1,n}.$$

Значення змінних $x_{ij} = 1$, якщо цикл включає переїзд з міста i в місто j, $x_{ij} = 0$, якщо ні. Умова $c_{ii} = \infty$ приймається для того, щоб виключити можливість появи в оптимальному розв'язкові значень $x_{ii} = 1$, що не мають сенсу. Перша і друга умови вимагають того, щоб цикл містив точно один в'їзд і один виїзд з кожного міста.

Зауважимо, що така постановка задача є повністю еквівалентною задачі про призначення. Але вона є неповною, тому що в ній бракує обмежень, що виключали б можливість утворення підциклів замість одного повного циклу. Існує багато способів задання лінійних обмежень на цілочисельні змінні, що виключали б можливість утворення підциклів. Наприклад, для конкретної задачі обмеження $x_{23} + x_{34} + x_{42} \le 2$ виключає можливість утворення підциклу між містами 2, 3, 4. Однак потрібно застосувати такий спосіб формування лінійних обмежень, що виключали б усі підцикли.

Введемо змінні u_i, $i = \overline{2, n}$ і накладемо на них такі $(n-1)^2 - (n-1)$ наступних умов:

$$u_i - u_j + n x_{ij} \le n - 1,\ i = \overline{2,n},\ j = \overline{2,n} \quad (i \ne j). \qquad (9.11)$$

Пояснимо, чому ці обмеження: 1) виключають всі цикли і 2) не виключають жодного повного циклу.

1) Розглянемо довільний цикл між k містами, який визначається значенням $x_{ij} = 1$ для k змінних. Додамо до системи k обмежень, які відповідають змінним, що утворюють підцикл. Для кожного з k міст в отриманій сумі фігурують величини u_i і $-u_i$. Отже, в загальному обмеженні не буде жодної величини u_i. Але це неприпустимо для всіх x_{ij} в цьому обмеженні, бо в такому випадку сума в лівій частині нерівності, що дорівнює nk, буде більше суми в правій частині нерівності, що дорівнює $(n-1)k$. Отже, обмеження виключає можливість появи будь-якого підциклу. Наприклад, для підциклу 2-3-4-2:

$u_2 - u_3 + nx_{23} \le n-1,$

$u_3 - u_4 + nx_{34} \le n-1,$

$u_4 - u_2 + nx_{42} \le n-1.$

Додавши, отримаємо:

$(u_2 - u_3 + u_3 - u_4 + u_4 - u_2) + n(x_{23} + x_{34} + x_{42}) \le 3(n-1).$

Очевидно, нерівність не виконується.

2) Розглянемо цикл, який за означенням полягає у відвіданні кожного міста 2,3,...,n. Нехай t_i – це положення в такому циклі, коли комівояжер потрапляє в місто i, причому для міста 1 приймаємо $t_1 = 1$. Тоді в циклі місто 1 – місто 3 – місто 5 ... значення будуть рівні: $t_1 = 1$, $t_3 = 2$, $t_5 = 3$, ... При цій умові $u_i = t_i$, i=2, 3, ..., n, є допустима множина значень t. Щоб переконатись в цьому, припустимо, що в даному циклі $x_{ij} = 1$, таким чином, $t_j = t_i + 1$. Тоді обмеження для цього x_{ij} в (9.11) виконується, оскільки

$$t_i - (t_i + 1) + n(1) \le n-1. \tag{9.12}$$

Припустимо тепер, що $x_{ij} = 0$. Тоді відповідна нерівність має вигляд $(u_i - u_j \le n-1)$. Ця нерівність повинна виконуватися, оскільки $u_i \le n$ і $u_j > 1$.

Отже, тепер наша постановка задачі у вигляді задачі цілочисельного ЛП виглядає досить компактно, але додаткові обмеження перетворюють задачу про призначення на складну NP-повну комбінаторну задачу.

Обчислювальні експерименти показують, що одним з найефективніших алгоритмів розв'язування цієї задачі є алгоритм, що побудований на схемі методу розгалужень та границь, який розглядається нижче.

Алгоритм методу розгалужень та границь для розв'язування задачі про комівояжера.

Крок 1. Побудова деякого довільного припустимого розв'язку задачі та розрахунок значення критерію для нього.

Крок 2. Послідовне розбиття біжучої множини маршрутів, що залишилися, на все менші підмножини. Розбиття здійснюється шляхом вибору дуги $(i;j)$ та розбиття біжучої множини маршрутів на дві підмножини:

- ☑ таку, в яку включається ланка $(i;j)$;
- ☑ таку, в яку ланка $(i;j)$ не включається.

Для розбиттів обчислюється нижня границя довжини біжучого найкращого маршруту та відтинаються неперспективні підмножини. В залежності від включення $(i;j)$ в маршрут забороняються всі маршрути, які утворюють локальні контури. Нижня границя розраховується як приріст до нижньої границі вузла-предка у випадку включення $(i;j)$ до маршруту, або як штраф за невключення у випадку $(i;j)$.

Крок 3. В результаті послідовних розбиттів будуємо повний маршрут (тобто отримуємо біжучий розв'язок). Якщо отриманий розв'язок є кращим, ніж біжучий найкращий розв'язок, то отриманий розв'язок стає біжучим найкращим, і неперспективні вузли, які можуть з'явитися внаслідок цього, відтинаються. В іншому випадку відтинається отриманий біжучий розв'язок.

Крок 4. Якщо активних вузлів в дереві немає – **стоп.** Біжучий найкращий розв'язок є оптимальним. В іншому випадку обираємо вузол з найменшим значенням нижньої границі та **переходимо до кроку 2.**

Розглянемо деталі реалізації окремих кроків алгоритму.

Обчислення нижніх границь.

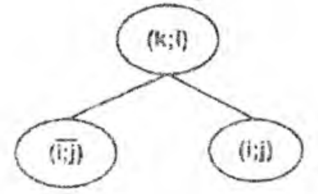
Для початкового дерева розгалужень (яке на момент початку обчислень матиме лише корінь) нижню границю обчислюємо шляхом редукції матриці віддалей $D = \{d_{ij}\}$, $\forall i$: $d_{ii} = \infty$, H – сума констант редукції.

Якщо $z(T)$ – довжина довільного маршруту T, що визначений з матриці $D = \{d_{ij}\}$ до редукції, то $z(T) = H + z_1(T)$, де $z_1(T)$ – те ж після редукції. Таким чином, H є нижньою границею довжини маршруту T, оскільки редукована матриця D є невід'ємною.

В ідеальному випадку, коли б у кожному стовпчику та рядку було по одному нулеві – маршрут був би нульової довжини.

Реалізація розгалуження (розбиття біжучої підмножини маршрутів).

Нехай підмножина маршрутів, що має в своєму складі ланку $(k;l)$ розбивається на дві підмножини: $\overline{Y}$, до якої ланка $(i;j)$ не входить $(i;j) \notin \overline{Y}$ (на дереві розгалужень їй відповідає вузол з позначенням $(\overline{i;j})$) і Y – до якої входить ланка $(i;j)$, $(i;j) \in Y$. В тому випадку, коли обрана ланка $(i;j)$, забороняється вибір ланки $(j;i)$, тобто відповідна віддаль в матриці віддалей $d_{ji} = \infty$.



Здійснюючи таким чином розбиття, ми врешті-решт побудуємо повний розв'язок. Встановити побудований маршрут можна, рухаючись від листа дерева до кореня.

Вибір ланки маршруту, по якій здійснюється розбиття.

Метою є розбиття на підмножини є реалізація розбиття таким чином, щоб найкращий варіант більш очікуваним був у множині Y, аніж у множині $\overline{Y}$. Тому в першу чергу до Y включаються ланки з $d_{ij} = 0$, однак у кожному рядку та стовпчикові матриці може бути більш ніж один 0. Остаточний вибір робиться за результатами розрахунку границь.

Розрахунок нижніх границь для маршрутів з $(i;j)$.

Пункт i повинен бути зв'язаним з деяким іншим, причому довжина ланки повинна бути не меншою, аніж значення мінімального елементу i-го рядка біжучої матриці віддалей, не враховуючи d_{ij}, $A_i = \underset{(k \ne l) \wedge d_{ij} \ne 0}{Min} d_{ij}$. Аналогічно, якась ланка повинна входити в i-тий пункт, відповідно довжина цієї ланки буде не меншою, ніж $B_j = \underset{(k \ne l) \wedge d_{ij} \ne 0}{Min} d_{ij}$. Розрахуємо для всіх нульових елементів суми $\forall\{(i;j)|d_{ij} = 0\}: F_{ij} = A_i + B_j$. Кожне зі значень F_{ij} розглядатимемо як штраф за невключення $(i;j)$ в маршрут. Виходячи з цього, для включення в маршрут обираємо ту ланку, штраф за невключення якої до маршруту буде найбільшим

$(i;j) = \arg \underset{\{(u;g)|d_{ug}=0\}}{Max} F_{ug}$.

Нова нижня границя не повинна перевищувати довжину ні одного маршруту, до яких ланка $(i;j)$ не входить, тобто для вузла з позначкою $\overline{(i;j)}$ значення нижньої границі $B_{ij} = B_{kl} + F_{ij}$. Для визначення максимального значення F_{ij} лише елементи $d_{ij} = 0$, тому що для $d_{ij} \ne 0 \Rightarrow F_{ij} = 0$ (якщо замінити $d_{ij} = \infty$ і зробити редукцію i-го рядка та стовпчика, то їх сума становитиме F_{ij}. Таким чином, F_{ij} – це додаткова віддаль, яку проїжджатиме комівояжер у випадку, коли в маршрут не включено ланку $(i;j)$, що відповідає $d_{ij} = 0$ редукованої біжучої матриці віддалей.

Розрахунок нижніх границь для маршрутів з $(i;j)$.

Якщо ми включаємо до маршруту ланку $(u;g)$, то u-й рядок та g-й стовпчик матриці віддалей у цій підмножині маршрутів можемо більше не розглядати, тобто викреслити. Крім того, $d_{gu} = \infty$, оскільки ланка $(u;g)$ належить всім маршрутам цієї підмножини. Також для всіх інших ланок $(m;n)$, за допомогою яких можуть бути побудовані неповні маршрути (підмаршрути), що не проходять через всі пункти, $d_{mn} = \infty$.

Таким чином, врахування конкретних особливостей задачі про комівояжера дозволило сконструювати ефективний алгоритм її розв'язування, що базується на методі розгалужень та границь.

ПРИКЛАДИ

Приклад 9.1. Застосування методу розгалужень та границь до розв'язання булевих задач.

Знайти оптимальний розв'язок наступної задачі за допомогою методу розгалужень та границь:

$$3x_1 + 2x_2 + 4x_3 + x_4 + 5x_5 \Rightarrow Max$$

$$x_1 + x_2 + 5x_3 + 2x_4 + 4x_5 \le 9$$

$$2x_1 + 3x_2 + 3x_3 + 5x_4 + 3x_5 \le 9$$

значення кожного x – нуль або одиниця.

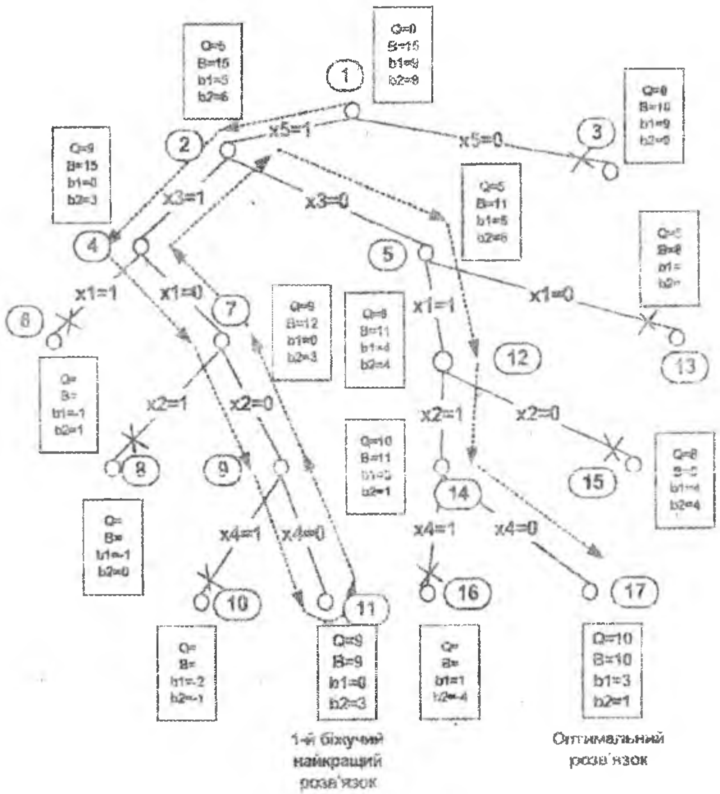
Розв'язання.

Аналіз задачі дозволяє впевнитися в тому, що значення всіх коефіцієнтів задачі позитивні, тобто це є багатовимірна задача про наплечник.

Для розв'язання цієї задачі насамперед впорядкуємо змінні в порядку спадання значень коефіцієнтів функції мети при них: $x_5 \succ x_3 \succ x_1 \succ x_2 \succ x_4$. В цьому порядку реалізуватимемо розгалуження з метою якнайшвидшого відтинання підмножин неперспективних розв'язків якомога раніше. В якості стратегії розгалуження використовуватимемо стратегію вглиб дерева. Окрім того, для цієї задачі існує декілька можливостей для обчислення верхньої границі.

З одного боку, верхню границю нащадка можна обчислювати як верхню границю предка, зменшену на значення коефіцієнта функції мети при змінній, яка фіксується при даному розгалуженні в нулі, з іншої — розв'язуючи підзадачу як задачу лінійного програмування без врахування умов цілочисельності (не слід при цьому забувати, що значення кожної зі змінних може бути лише між нулем та одиницею). Значення верхніх границь, визначених за останнім методом, будуть ближчими до дійсних оптимальних значень функції мети, ніж ті, які визначені за першим способом. Однак у цьому випадку для отримання значення границі необхідно розв'язати задачу лінійного програмування, що порівняно з операцією віднімання вимагає значно більшої кількості обчислень.

Хід розв'язання задачі зображений нижче у вигляді дерева розгалужень:



Згідно зі стратегією розгалуження просуваємося вглиб дерева, обираючи кожен раз вузол з найбільшим значенням границі, серед тих, які найвіддаленіші (віддаль вимірюється кількістю ребер) від кореня.

Таким чином, рухаємося вузлами 1-2-4-7-9-11, по ходу відтявши вузли 6, 8, 10, що відповідають неприпустимим розв'язкам і отримуємо 1-й біжучий найкращий розв'язок, читаємо який, рухаючись до кореня: $Q = 9, x_4 = 0, x_2 = 0, x_1 = 0, x_3 = 1, x_5 = 1$.

Надалі просуваємося догори деревом, поки не зустрінемо вузол, що ще має безпосередніми нащадками активні вузли (вузол), а саме вузол 2 і,

опускаючись вниз, розбиваємо вузли 5-12-14, поки не досягнемо наступного повного розв'язку – вузол 17. В ході просування після декомпозиції вузла 5 його безпосередній нащадок відтинаємо з тієї причини, що оптимальних розв'язків там не може бути – значення верхньої границі для цього вузла є меншим, аніж значення критерію якості для біжучого найкращого повного розв'язку. З аналогічної причини відтинаємо вузол 15, а для вузла 16 порушуються обмеження. Таким чином, ми отримуємо новий біжучий повний найкращий розв'язок, що відповідає вузлу 17, а саме: $Q = 10, x_4 = 0, x_2 = 1, x_1 = 1, x_3 = 0, x_5 = 1$.

Активним залишається лише вузол 3; але для нього значення верхньої границі рівне значенню критерію якості для біжучого найкращого розв'язку, а тому кращих розв'язків, ніж вже знайдений, ми не отримаємо, відтинаємо вузол 3 і, таким чином, біжучий найкращий розв'язок і буде оптимальним.

Приклад 9.2. Булеві задачі цілочисельного програмування.

Розв'язати наступну задачу за допомогою алгоритму Балаша.

$$Q = 3y_1 + 2y_2 - 5y_3 - 2y_4 + 3y_5 => Max$$

$$y_1 + y_2 + y_3 + 2y_4 + y_5 \le 4,$$
$$7y_1 + 3y_3 - 4y_4 + 3y_5 \le 8,$$
$$11y_1 - 6y_2 + 3y_4 - 3y_5 \ge 3$$
$$y_i \in \{0;1\}, i = \overline{1,5}$$

Розв'язання.

Цю задачу неважко представити у вигляді, що задовольняє умови адитивного алгоритму. Для цього виконаємо наступне.

Функцію мети помножимо на –1. Третє обмеження помножимо на –1. Введемо додаткові змінні s_1, s_2, s_3 для перетворення обмежень у рівності.

Щоб коефіцієнти цільової функції були додатними, проведемо підстановку: $y_1 = 1 - x_1, y_2 = 1 - x_2, y_3 = x_3, y_4 = x_4, y_5 = 1 - x_5$.

Наступні перетворення призводять до наступної цільової функції:

$$Q' = 3x_1 + 2x_2 + 5x_3 + 2x_4 + 3x_5 - 8 => Min.$$

Для зручності будемо ігнорувати константу –8 та замінимо $Q' + 8 = Q$, отже, перетворена задача буде мати наступний вигляд:

$$Q = 3x_1 + 2x_2 + 5x_3 + 2x_4 + 3x_5 \Rightarrow Min$$

$$-x_1 - x_2 + x_3 + 2x_4 - y_5 + s_1 = 1,$$

$$-7x_1 + 3x_3 - 4x_4 - 3x_5 + s_2 = -2,$$

$$11y_1 - 6x_2 - 3x_4 - 3x_5 + s_3 = -1$$

$$x_i = [0,1], i = \overline{1,5}$$

Так як у перетвореній задачі ведеться пошук мінімуму функції мети з додатними коефіцієнтами, логічно, що у початковому рішенні усі змінні повинні бути рівними 0. У цьому випадку додаткові змінні будуть рівні правим частинам обмежень.

Внаслідок того, що всі булеві змінні дорівнюють нулю, додаткові змінні приймають наступні значення: $(s_1,s_2,s_3)=(1,-2,-1)$, при цьому $Q=0$. Якби всі додаткові змінні були невід'ємними, ми зробили б висновок, що розглянутий розв'язок, у якому всі булеві змінні дорівнюють нулеві, є оптимальним. Оскільки деякі з додаткових змінних є неприпустимими (тому що негативні), необхідно збільшити значення однієї чи декількох булевих змінних до 1, щоб одержати припустимий розв'язок (чи прийти до висновку, що задача припустимого розв'язку не має).

Збільшення значень булевих змінних до 1 в адитивному алгоритмі відбувається *окремо*. Обрана змінна називається змінною розгалуження. Її вибір ґрунтується на використанні спеціальних тестів.

Змінна розгалуження повинна зменшити (за абсолютною величиною) негативні значення додаткових змінних. Як випливає з наведених міркувань, змінна x_3, не може бути обрана як змінна розгалуження, тому що її коефіцієнти в другому і третьому обмеженнях ненегативні. Отже, поклавши $x_3 = 1$, ми лише збільшимо (за абсолютною величиною) негативні значення додаткових змінних s_2 і s_3. Кожна зі змінних (булевих), що залишились, має принаймні один негативний коефіцієнт у другому і третьому обмеженнях. Отже, комбінація цих змінних може привести до позитивних значень додаткових змінних. Таким чином, лише змінні x_1, x_2, x_4 та x_5 можна розглядати як можливих кандидатів на змінну розгалуження.

Вибір змінної розгалуження з можливих претендентів ґрунтується на використанні *міри неприпустимості додаткової змінної*. Ця міра ґрунтується на припущенні, що значення булевої змінної x_j буде збільшене до 1, визначається співвідношенням:

$$I_j = \sum_{по\_всіх\_i} \min(0, s_i - a_{ij}),$$

де s_i — поточне значення додаткової змінної, a_{ij} — коефіцієнт при змінній x_j у i-му обмеженні.

У дійсності I_j є не що інше, як сума значень негативних додаткових змінних, що є результатом збільшення значення змінної x_j до 1. Складна на вид формула може бути спрощена в такий спосіб:

$$I_j = \sum_{по\_всіх\_i} (від'ємне\_значення\_s_i\_при\_заданому\_x_j = 1).$$

Наприклад, якщо ми покладемо $x_1 = 1$, то одержимо $s_1 = 1 - (-1) = 2$, $s_2 = -2 - (-7) = 5$ і, $s_3 = -1 - 11 = -12$. Отже, $I_1 = -12$. Аналогічно, $I_2 = -2$, $I_4 = -1$ і $I_5 = 0$ (нагадаємо, що змінна x_3 була виключена з претендентів на змінну розгалуження як безперспективна).

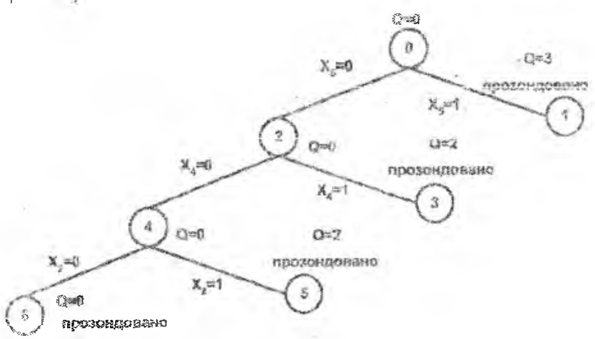
Через те, що I_5 має найменшу міру неприпустимості, змінна x_5 обирається як змінна розгалуження. На рис. нижче зображені дві гілки, що відповідають $x_5 = 1$ і $x_5 = 0$, і утворені при цьому вузли 1 і 2. Вершина 1 дає припустиме значення додаткових змінних $(s_1,s_2,s_3)=(2,1,2)$ і $Q = 3$, значення змінних $X = \{0,0,0,0,1\}$. Так як присвоєння значення 1 будь-якій із змінних, що залишилися, тільки збільшить значення Q (так як всі коефіцієнти функції мети додатні), а шукаємо мінімум, тоді, вузол 1 прозондовано, і значення $Q = 3$ визначає поточну *верхню границю* оптимального значення цільової функції.

Прозондувавши вузол 1, переходимо до вузла 2, для якого $x_5 = 0$. Тут маємо $(s_1,s_2,s_3) = (1,-2,-1)$ і $Q = 0$, тобто розв'язок неприпустимий. Змінні x_1, x_2, x_3 та x_4 є можливими кандидатами на змінну розгалуження. (Примітка: хоча розв'язки у вузлах 0 і 2 ідентичні, вузол 2 відрізняється тим, що змінна x_5 не є більше претендентом на розгалуження.). Як і у вершині 0, тут змінна x_3 безперспективна, тому що не зменшує (за абсолютною величиною) негативних значень додаткових змінних s_2 і s_3. Крім того, значення $x_3 = 1$ приводить до значення критерію якості, рівному 5, що гірше за поточне значення верхньої границі $Q = 3$. Змінна x_1 також безперспективна, тому що відповідний коефіцієнт у критерієві якості дорівнює 3, тому значення $x_1 = 1$ не приводить до поліпшення наявного значення цільової функції. Для змінних, що залишились, x_2 і x_4 обчислюємо міру неприпустимості $I_2 = -2$, $I_4 = -1$. Отже, у вузлі 2 змінною розгалуження буде x_4

На рис. нижче показані гілки $x_4 = 1$ і $x_4 = 0$, що ведуть до вузлів 3 і 4 відповідно. У вузлі 3 (визначеному обмеженнями $x_5 = 0$ і $x_4 = 1$) маємо розв'язок $(s_1,s_2,s_3) = (1,-2,-1)$ і $Q = 2$, що не є припустимим. Кандидатами на розгалуження є змінні x_1, x_2 та x_3. Однак, збільшуючи значення кожної з них до 1, ми погіршимо значення цільової функції Q у порівнянні з поточною верхньою границею (Q =3). Отже, усі змінні виключені з кандидатів на розгалуження, і вузол 3 прозондовано.

Далі у вузлі 4, визначеному обмеженнями $x_5 = x_4 = 0$, маємо $(s_1,s_2,s_3) = (1,-2,-1)$, Q=2. Змінні x_1 та x_3 виключаються з претендентів на розгалуження через тест на верхню границю. Змінну, що залишилася – x_2, не можна виключити з розгляду на підставі тестів. Отже, x_2 є змінною розгалуження.

На рис. показані вузли 5 і 6, що виходять з вузла 4. У вузлі 5 маємо $(s_1,s_2,s_3) = (1,-2,-1)$, Q=2, і змінні x_1 та x_3 є кандидатами на розгалуження. Змінна x_1 виключається тестом на верхню границю, а x_3 — як тестом на верхню границю, так і тестом на припустимість додаткових змінних. Це означає, що вузол 5 прозондовано. Вузол 6 також прозондовано, тому що ні x_1, ні x_3 не можуть привести до поліпшення припустимого розв'язку.



Остаточне "дерево пошуку".

Тепер усі вузли прозондовано, тому метод закінчує роботу. Оптимальний розв'язок знайдений у вузлі 1, тобто $x_1 = 1$, Q = 3, всі інші змінні дорівнюють нулеві. Звідси одержуємо розв'язок вихідної задачі:

$$y_1 = y_2 = 1, y_3 = y_4 = y_5 = 0, Q = 5$$

З рисунка видно, що чим менше гілок, які ведуть до прозондованого вузла, тим ефективнішим є алгоритм. Наприклад, вершина 1 визначена за допомогою фіксування однієї змінної ($x_5 = 1$), і її зондування автоматично відповідає за $2^{5-1} = 16$ булевих рішень (усі розв'язки, для яких $x_5 = 1$). Вузол 3 визначений шляхом фіксування двох булевих змінних, і його зондування відповідає лише за $2^{5-2} = 8$ булевих розв'язків.

Приклад 9.3. Задача про комівояжера.

Знайти маршрут комівояжера мінімальної довжини, якщо задана

наступна матриця віддалей:

	1	2	3	4	5	6
1	∞	27	43	16	30	26
2	7	∞	16	1	30	30
3	20	13	∞	35	5	0
4	21	16	25	∞	18	18
5	12	46	27	48	∞	5
6	23	5	5	9	5	∞

Оберемо довільний маршрут (1-4-5-3-6-2-1). Довжину маршруту розрахуємо, виходячи з матриці віддалей. Це значення В=73 й буде верхньою границею для множини можливих маршрутів.

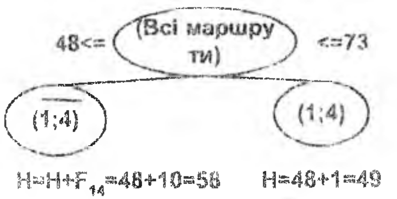
Редукуємо матрицю віддалей

	1	2	3	4	5	6	Δ	$Min\ A_i$
1	∞	11	27	$0^{[10]}$	14	10	16	10
2	1	∞	15	$0^{[1]}$	29	29	1	1
3	15	13	∞	35	5	$0^{[5]}$	0	5
4	$0^{[1]}$	$0^{[0]}$	9	∞	2	2	16	0
5	2	41	22	43	∞	$0^{[2]}$	5	2
6	13	$0^{[0]}$	$0^{[9]}$	4	$0^{[2]}$	∞	5	0
Δ	5	0	0	0	0	0	$H = 48$	
$Min\ B_j$	1	0	9	0	2	0		

Значення нижньої границі для всіх маршрутів становитиме 48. Розраховуємо значення $Min\ A_i$ та $Min\ B_j$. Розраховуємо штраф за невключення до маршруту ланок, яким відповідають нульові елементи.

(i;j)	1;4	2;4	3;6	4:1	4;2	5;6	6;2	6;3	6;5
F_{ij}	10	1	5	1	0	2	0	9	2

Таким чином, розгалуження здійснюватиметься шляхом розбиття множини можливих маршрутів на дві підмножини: підмножину, до якої не включається ланка $(1,4)$, та підмножини, яка включатиме цю ланку. Таким чином, отримуємо початок дерева розгалужень у вигляді:



Значення нижньої границі для вузла $(1;4)$ отримаємо шляхом викреслювання з матриці віддалей вузла-предка 1-го рядка та 4-го стовпчика, заміною значення $d_{41} = \infty$ та розрахунком суми констант редукції для нової матриці.

Ланку (1,4) включаємо:

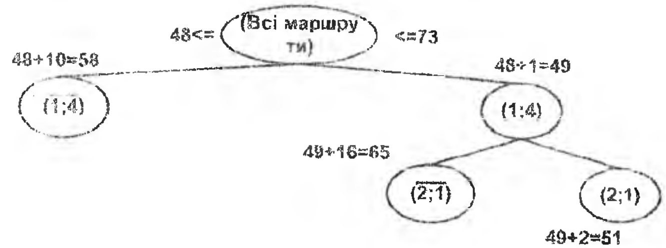
$F_{21} = 16$

	1	2	3	5	6
2	1	∞	15	29	29
3	15	13	∞	5	0
4	0	0	9	2	2
5	2	41	22	∞	0
6	13	0	0	0	∞

	1	2	3	5	6	q_i	A_i
2	$0^{[16]}$	∞	14	28	28	1	14
3	15	13	∞	5	0[illegible]	0	5
4	∞	$0^{[2]}$	9	2	2	0	2
5	2	41	22	∞	$0^{[2]}$	0	2
6	13	$0^{[0]}$	$0^{[9]}$	$0^{[2]}$	∞	0	0
q_j	0	0	0	0	0	$H = 1$	
B_j	2	0	9	2	0		

Матриця віддалей для $(\overline{1,4})$, коли ланку (1,4) не включаємо, має наступний вигляд.

	1	2	3	4	5	6
1	∞	11	27	∞	14	10
2	1	∞	15	0	29	29
3	15	13	∞	35	5	0
4	0	0	9	∞	2	2
5	2	41	22	43	∞	0
6	13	0	0	4	0	∞



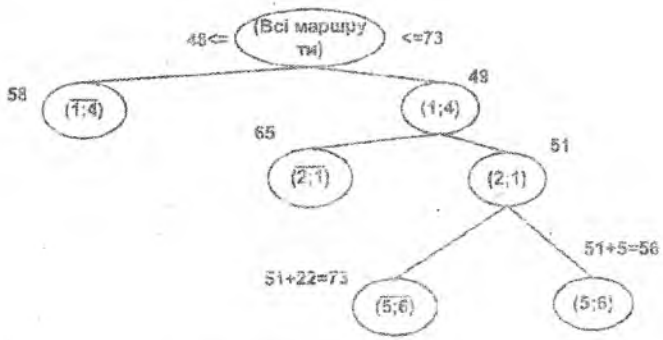
	2	3	5	6
3	13	∞	5	0
4	0	9	2	2
5	41	22	∞	0
6	0	0	0	∞

	2	3	5	6		A_i
3	13	∞	5	$0^{[5]}$	0	5
4	∞	7	$0^{[0]}$	$0^{[0]}$	2	0
5	41	22	∞	$0^{[22]}$	0	22
6	0	0	0	∞	0	0
B_j	0	0	0	0	$H = 2$	
	13	7	0	0		

$F_{56} = 22$

Матриця віддалей для $(\overline{2,1})$, коли ланку (2.1) не включаємо, має наступний вигляд:

	1	2	3	5	6
2	∞	∞	14	28	28
3	15	13	∞	5	0
4	∞	0	9	2	2
5	2	41	22	∞	0
6	13	0	0	0	∞

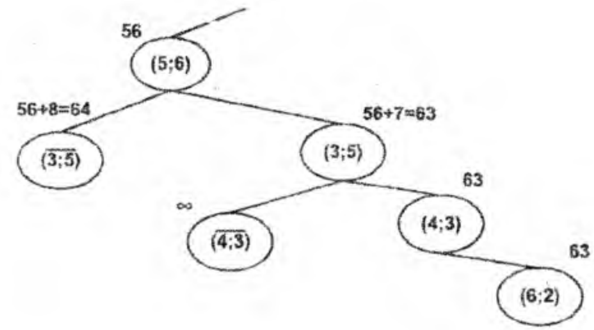


$(5;6)$ 2-1-4 5-6 $d_{65}=\infty$, $d_{42}=\infty$, оскільки в цих випадках утворюються локальні цикли.

$\overline{(5;6)}$

	2	3	5	6
3	13	∞	5	0
4	∞	7	0	0
5	41	22	∞	∞
6	0	0	0	∞

	2	3	5		A_i
3	8	∞	$0^{[8]}$	5	8
4	∞	7	$0^{[7]}$	0	7
6	$0^{[8]}$	$0^{[7]}$	∞	0	0
	0	0	0	$H=5$	
B_j	8	7	0		



$\overline{(3,5)}$ $(3,5)$ 2-1-4 3-5-6

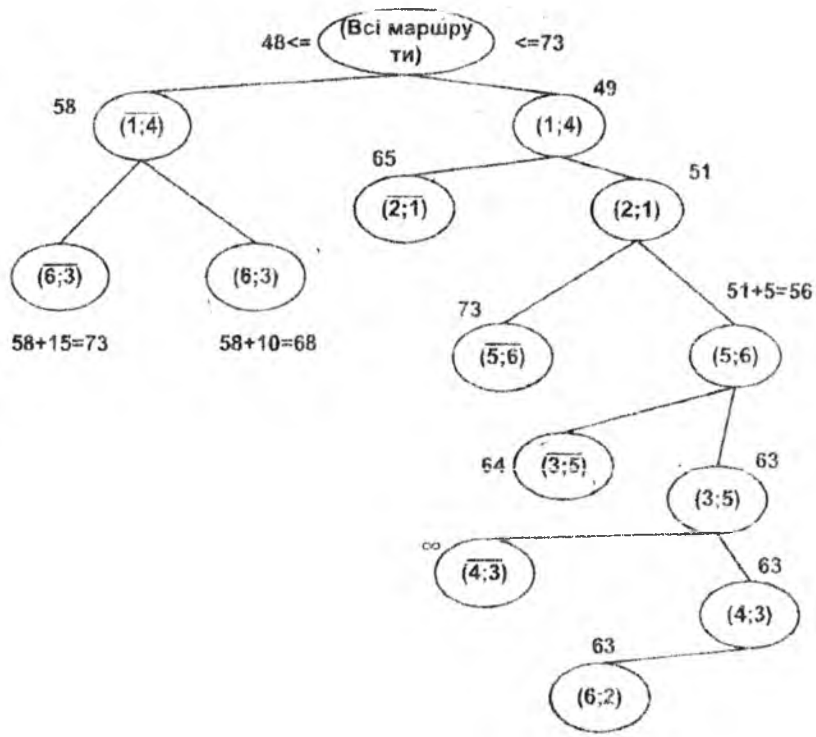
	2	3	5
3	8	∞	∞
4	∞	7	0
6	0	∞	∞

	2	3
4	∞	7
6	0	∞

	2	3		
4	∞	$0^{[\infty]}$	7	∞
6	$0^{[\infty]}$	∞	0	∞
	0	0	7	
	∞	∞		

Таким чином, досягнутий перший припустимий розв'язок задачі. проходячи до кореня дерева, читаємо його: (6,2), (4,3), (3,5), (5,6), (2,1), (1,4), тобто маршрут (1-4-3-5-6-2-1) довжиною 63.

Відтинаємо всі неперспективні вузли, і активним залишиться лише вузол $(1,4)$. Розгалужуємо його, і відтинаємо нащадки як неперспективні вузли. Таким чином, знайдений розв'язок, тобто маршрут (1-4-3-5-6-2-1) довжиною 63 є оптимальним. Процес розв'язування у вигляді дерева наведений нижче.



РЕЗЮМЕ

9.1. У багатовимірній задачі про наплечник значення всіх коефіцієнтів є невід'ємними. Ця задача є не чим іншим, як варіантом задачі виробничого планування за умови, що вироби випускаються дискретно. До задач з цілочисельними булевими змінними схема методу розгалужень та границь може бути успішно застосована, причому як до загальних задач, так і до розширень задачі про наплечник з декількома обмеженнями.

9.2. У випадку, коли значення коефіцієнтів в булевій задачі можуть мати будь-який знак, тобто бути і додатними, і від'ємними, для розв'язання можна застосувати алгоритм Балаша. Формально процес пошуку

оптимального розв'язку може бути представлений у вигляді конструювання деякого дерева варіантів, де кожна вершина (не остаточна) відповідає певному частковому розв'язку, а можливі його доповнення породжують гілки дерева.

9.3. Будь яку задачу цілочисельного програмування можна привести до лінійної задачі булевого програмування. Для цього задача спочатку приводиться до поліноміальної булевої, а потім до лінійної булевої. Можливість приведення цілочисельних задач до булевого типу використовується далеко не завжди, оскільки при цьому суттєво зростає розмірність, а тому для задач певних спеціалізованих типів розроблені спеціалізовані ж алгоритми їх розв'язання

9.4. Задача про комівояжера знайшла багато практичних застосувань, і, незважаючи на простоту формулювання, не розв'язується за допомогою простих алгоритмів, як подібні на неї зовні задача про найкоротший шлях та задача про багатополюсний ланцюг. Додаткові обмеження перетворюють задачу про призначення на складну NP-повну комбінаторну задачу. Обчислювальні експерименти показують, що одним із найефективніших алгоритмів розв'язування цієї задачі є алгоритм, що побудований на схемі методу розгалужень та границь.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 9.1. Багатовимірна задача про наплечник.

Розв'язати задачу булевого програмування за допомогою методу гілок та границь:

$$4x_1 + 5x_2 + 2x_3 + 2x_4 + x_5 \Longrightarrow Max$$
$$3x_1 + 2x_2 + 2x_3 + x_4 + 3x_5 \le 8$$
$$x_1 + 6x_2 + 3x_3 + x_4 + x_5 \le 12$$
$$\forall x_i \in \{0;1\}$$

Завдання 9.2. Задача булевого програмування.

Розв'язати наступну задачу за допомогою алгоритму Балаша.

$$Q = 3y_1 + 2y_2 - 3y_3 - 1y_4 + 4y_5 \Rightarrow Max$$

$$\begin{aligned} & y_1 + 2y_2 + y_3 + y_4 + y_5 \le 4, \\ & 5y_1 + 3y_3 - 4y_4 + 4y_5 \le 8, \\ & 10y_1 - 4y_2 + 5y_4 - 3y_5 \ge 3 \\ & y_i \in \{0;1\}, i = \overline{1,5} \end{aligned}$$

Завдання 9.3. Задача про комівояжера.

Задана наступна матриця віддалей між містами:

∞	11	21	6	8
13	∞	17	8	11
19	18	∞	7	21
22	15	11	∞	17
32	4	12	6	∞

Необхідно знайти найкоротший маршрут комівояжера за допомогою методу гілок та границь.

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Яким чином розв'язуються задачі булевого програмування за допомогою методу розгалужень та границь?
2. Наведіть основні кроки алгоритму Балаша.
3. Яким чином задачі цілочисельного програмування приводяться до булевих?
4. Наведіть формальну постановку задачі про комівояжера.
5. Яким чином обчислюються нижні границі в задачі комівояжера?
6. Яким чином піддається декомпозиції на підзадачі задача комівояжера?

РОЗДІЛ 4

ПЛАНУВАННЯ НА МЕРЕЖАХ

ТЕМА 10. МЕТОДИ РОЗВ'ЯЗАННЯ ЗАДАЧ ПЛАНУВАННЯ НА МЕРЕЖАХ

Планування та керування на мережах - це комплекс графічних і розрахункових методів, організаційних заходів, що забезпечують моделювання, аналіз і динамічну перебудову плану виконання складних проектів і розробок. Характерною рисою таких проектів є те, що вони складаються з ряду окремих, елементарних робіт, які є певним чином залежні одна від іншої, так що виконання деяких робіт не може бути початє раніш, ніж завершені деякі інші. Методи планування на мережах включають наступні основні етапи: структурне планування; календарне планування; оперативне керування. Для розв'язування задач планування на мережах застосовуються такі методи, як CPM та PERT. CPM (метод критичного шляху) орієнтований на розв'язування детермінованих задач планування на мережах. Метод PERT орієнтований на врахування того факту, що тривалості робіт є випадковими величинами, і дозволяє оцінити основні параметри, а саме: вірогідність завершення проекту в заданий директивний термін; очікувану тривалість виконання проекту. Основною перевагою цих методів є те, що вони дозволяють виявити "вузькі" місця проекту та здійснити необхідний перерозподіл наявних ресурсів.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ:

знати:⇨основні етапи планування на мережах;⇨структуру, правила та послідовність побудови мережі проекту;⇨основні параметри мереж типу CPM та PERT та способи їх визначення;⇨проблематику оптимізації планування на мережах;

вміти:⇨розраховувати детерміновані мережі за методом критичного шляху;⇨розробляти, розраховувати та імітувати плани дій на моделях планування на мережах;⇨ будувати мережу за методом PERT та здійснювати інтервальне оцінювання її параметрів; ⇨ оптимізувати мережну модель за критеріями вартості та тривалості проекту з отриманням відповідної множини Парето-оптимальних розв'язків.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ проект
- ☑ ранній термін початку та завершення роботи
- ☑ центральна гранична теорема
- ☑ планування на мережах
- ☑ пізній термін початку та завершення роботи
- ☑ оптимізація мережі
- ☑ операція проекту
- ☑ резерв часу роботи (повний, вільний, незалежний, гарантований)
- ☑ песимістична тривалість роботи
- ☑ пізній термін звершення події
- ☑ розподіл ресурсів
- ☑ критичний шлях
- ☑ ранній термін звершення події
- ☑ тривалість проекту
- ☑ метод критичного шляху
- ☑ найімовірніша тривалість роботи
- ☑ вартість проекту
- ☑ метод PERT
- ☑ оптимістична тривалість роботи
- ☑ Парето-оптимальні розв'язки
- ☑ подія
- ☑ графік Ганта
- ☑ структура мережі
- ☑ резерв часу події

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

10.1. Основні поняття та визначення.
10.2. Структура та правила побудови мережі.
10.3. Основні параметри мережі типу СРМ та їх визначення.
10.4. Метод PERT.
10.5. Оптимізація мережі.

10.1. ОСНОВНІ ПОНЯТТЯ ТА ВИЗНАЧЕННЯ

Планування та керування на мережах - це комплекс графічних і розрахункових методів, організаційних заходів, що забезпечують

моделювання, аналіз і динамічну перебудову плану виконання складних проектів і розробок, наприклад, таких як: будівництво і реконструкція якихнебудь об'єктів; виконання науково-дослідних і конструкторських робіт; підготовка виробництва до випуску продукції; переозброєння армії; розгортання системи медичних чи профілактичних заходів.

Характерною рисою таких проектів є те, що вони складаються з ряду окремих, елементарних ***робіт***. Роботи є певним чином залежні одна від іншої, так що виконання деяких робіт не може бути почате раніш, ніж завершені деякі інші. Наприклад, укладання фундаменту не може бути почате раніше, ніж будуть доставлені необхідні матеріали; ці матеріали не можуть бути доставлені раніш, ніж будуть побудовані під'їзні колії; будь-який етап будівництва не може бути початий без складання відповідної технічної документації і т.д.

Розвиток сучасних методів керування проектами почався наприкінці 50-х років з появою перших робіт з мережевого моделювання, які ввели у користування так звані традиційні (або класичні) мережеві моделі. В основі перших систем керування проектами з використанням мережних моделей (PERT, CPM) лежав „метод критичного шляху" (CPM – Critical-Path Method) - дієвий, але досить простий метод аналізу планування і календарного розподілу робіт при виконанні складних проектів. Цей метод дає можливість визначити, по-перше, які роботи з числа багатьох, що складають проект, є „критичними" за своїм впливом на загальну календарну тривалість виконання проекту, і, по-друге, яким чином побудувати найкращий календарний план проведення всіх робіт з даного проекту для того, щоб дотриматись заданих термінів при мінімальних витратах.

Метод PERT є спрощенням процедури RISC-аналізу. Цей метод був розроблений консультантами фірм Booz, Allen & Hamilton у співробітництві з ВМС США та застосований вперше при плануванні розробки ракетної системи Polaris, для координування діяльності більш ніж 11000 контрактних працівників, що працювали над нею. Основний підхід ґрунтувався на поділі проекту на окремі задачі. Вперше було введено вірогідності часу для виконання конкретних задач, визначено зв'язки між ними, проведено простий мережевий аналіз і розраховано найімовірніший час завершення проекту.

Методи планування на мережах включають наступні основні етапи:

1.Структурне планування.

2.Календарне планування.

3.Оперативне керування.

Структурне планування починається з розбиття проекту на чітко визначені операції, для яких визначається тривалість. Потім визначаються відношення передування, що вказують, які роботи повинні бути обов'язково виконані, щоб могли початися ті, що виконуються безпосередньо за ними, на ґрунті чого будується мережа, що відображає взаємозв'язки робіт проекту. Це дозволяє детально проаналізувати всі роботи і поліпшити структуру проекту ще до початку його реалізації.

Календарне планування передбачає побудову календарного графіка, що визначає моменти початку і завершення кожної роботи й інші часові характеристики мережевого графіка. Це дозволяє, зокрема, виявляти критичні операції, яким необхідно присвятити особливу увагу, щоб закінчити проект у директивний термін. Під час календарного планування визначаються часові характеристики всіх робіт з метою оптимізації мережевої моделі, що поліпшує ефективність використання певного ресурсу.

Під час **оперативного керування** використовуються мережевий і календарний графіки для складання періодичних звітів про хід виконання проекту. При цьому мережева модель може оперативно корегуватися, внаслідок чого буде розроблятися новий календарний план для частини проекту, що ще не виконана.

Задачі планування на мережах широко застосовуються в управлінні проектами, а тому виникає необхідність познайомитися з основними термінами та їх визначеннями, тим більш, що в багатьох програмних засобах з управління проектами використовується ця ж термінологія.

Проект – це сукупність взаємнопов'язаних операцій (робіт), які необхідно виконати в певному порядку, щоб реалізувати задану мету проекту.

Операція проекту – робота, виконання якої потребує певного часу і витрат ресурсів. Операція характеризується набором параметрів, а саме: час виконання, трудомісткість виконання, вид (види) ресурсу, необхідного для виконання та ін.

Графік Ганта (Gantt) – графічне представлення на осі часу порядку виконання робіт.

Для розв'язування задач планування на мережах застосовуються такі методи, як CPM (Critical Path Methode) та PERT (Programm Evaluation & Review Technique). Окрім того, для детермінованих систем існують методи оптимізації за двома основними критеріями – часом виконання та вартістю

проекту.

CPM (МКШ – метод критичного шляху) орієнтований на розв'язування детермінованих задач планування на мережах. В більшості задач такого типу вважається, що тривалість кожної з робіт є сталою, або ж пропорційна до кількості виділеного на її виконання ресурсу. Розраховуються часові та ресурсні параметри мережі. Сучасні програмні пакети орієнтовані в основному на роботу з детермінованими мережами.

Метод PERT орієнтований на врахування того факту, що тривалості робіт є випадковими величинами, і дозволяє оцінити основні параметри: вірогідність завершення проекту в заданий директивний термін; очікувана тривалість виконання проекту.

Основною перевагою цих методів є те, що вони дозволяють виявити "вузькі" місця проекту та здійснити необхідний перерозподіл наявних ресурсів.

За допомогою CPM та PERT керівник проекту має можливість:

☑ завчасно планувати роботу над проектом та передбачати можливі джерела ускладнень і затримання виконання його в заданий директивний термін;

☑ планувати завершення робіт проекту в необхідні терміни у відповідності до послідовності виконання завдань з метою якнайшвидшої реалізації проекту загалом;

☑ координувати та контролювати перебіг робіт для виконання календарного графіку та завершення проекту в термін.

З принципами, що лежать в основі методів планування на мережах, тісно пов'язані задачі розподілу та використання ресурсів, скорочення термінів виконання окремих робіт з метою зменшення тривалості проекту та аналізу припустимих затримок (резервів часу) робіт.

Це дозволяє керівникові проекту здійснювати координацію та розв'язати наступні задачі управління проектом:

☑ встановлювати послідовність та терміни виконання обмежених ресурсів протягом усього періоду реалізації проекту;

☑ здійснювати оптимальний розподіл засобів, що виділені на проект, з метою скорочення тривалості проекту загалом;

☑ реалізувати динамічне регулювання термінів початку кожної роботи;

☑ виконати аналіз компромісних співвідношень між витратами та термінами виконання різних робіт з врахуванням наявних резервів часу.

Для того, щоб мати можливість застосувати методи CPM та (або)

PERT, проекти повинні мати такі спільні риси:

☑ проект повинен складатися з чітко визначеної множини робіт (операцій), виконання яких означатиме завершення проекту;

☑ завдання (роботи) є напіввпорядкованими, тобто деякі з робіт повинні виконуватись чітко в певному порядку, а деякі можуть виконуватися паралельно;

☑ тривалість виконання кожної з робіт та відповідний об'єм необхідних ресурсів можуть бути достатньо точно визначені чи оцінені;

☑ почата робота виконується без переривання;

☑ виконання наступної роботи не обов'язково повинне розпочинатися після завершення попередніх, але робота може початися лише тоді, коли всі попередні будуть виконані;

☑ мінімальна тривалість проекту визначається послідовністю операцій, які визначають найдовший шлях у мережі (критичний шлях).

10.2. СТРУКТУРА ТА ПРАВИЛА ПОБУДОВИ МЕРЕЖІ

Розглянемо основні завдання та особливості етапу структурного планування.

Побудова мережевої моделі (**структурне планування**) починається з розбиття проекту на чітко визначені роботи, для яких визначається тривалість. **Робота** – це деякий процес, що приводить до досягнення певного результату, який вимагає витрат певних ресурсів і триває в часі.

За фізичною природою роботи можна розглядати як:

☑ **дія**: заливання фундаменту бетоном, складання заявки на матеріали, вивчення кон'юнктури ринку;

☑ **процес**: старіння виливків, витримування вина, травлення плат;

☑ **очікування**: очікування постачання комплектуючих, пролежування деталі в черзі до верстата.

За кількістю витраченого часу робота може бути:

☑ **реальною (дійсною)**, тобто такою, що потребує витрат часу;

☑ **фіктивною**, тобто такою, що формально не потребує витрат часу.

Фіктивна робота може реально існувати, наприклад, „передавання документів від одного відділу до іншого". Якщо тривалість такої роботи є неспівмірно малою порівняно з тривалістю інших робіт проекту, то формально її вважають рівною 0. Існують також фіктивні роботи, яким насправді не відповідають ніякі дії. Такі фіктивні роботи лише узгоджують реальні взаємні зв'язки з формою мережевої моделі. Роботи пов'язані між

собою таким чином, що виконання певних робіт може бути почато ***лише після*** завершення деяких інших.

Подія – це момент часу, в який завершуються одні роботи і починаються інші. Подія є результатом виконаних робіт і, на відміну від робіт, не має тривалості в часі.

Взаємозв'язок між роботами та подіями, що необхідний для досягнення остаточної мети проекту, зображається за допомогою **мережевого графіка** (мережевої моделі). Роботи зображуються **орієнтованими дугами**, що з'єднують **вершини**, які відображають події. Початок і закінчення будь-якої роботи описуються парою подій i, j, що називаються **початковою** і **завершаючою** подіями роботи (i, j) відповідно. Тому для вказання конкретної роботи використовують позначення дуги у вигляді пари подій (i, j).

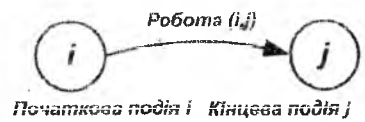


Рис. 10.1. Позначення роботи

Будь-яка подія може вважатися такою, що трапилася (звершилася) тільки тоді, коли закінчаться ***усі*** роботи, що входять до цієї події. Тому роботи, що виходять з деякої події, не можуть початися, поки не будуть завершені ***всі*** роботи, що входять у цю подію.

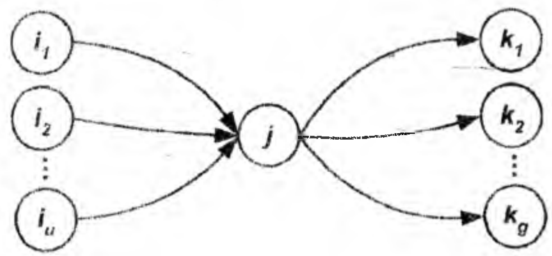


Рис. 10.2. Звершення події

Будь-яка з робіт (j, k_r) може початися лише тоді, коли завершаться **всі** роботи, що входять в цю подію. Роботи, перед якими немає ніяких інших, називаються **початковими**, і відповідно в подію, з якої вони виходять, не входить ні одна робота. Роботи, після яких немає ніяких інших, називаються **завершаючими**, і відповідно з події, до якої вони входять, не виходить ні однієї роботи. Відповідно ці події називаються **початковою** та **завершаючою подіями в мережі. В мережі може бути лише одна**

початкова та одна завершаюча подія.

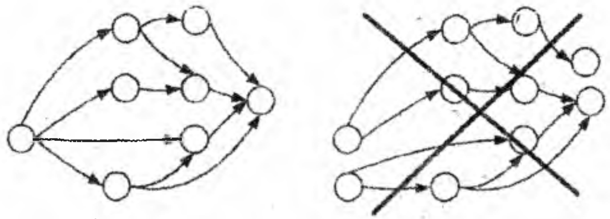


Рис. 10.3. Початкові та завершаючі роботи і події мережі

Подія звершується, коли закінчується виконання всіх робіт, що входять до неї. Звершення завершаючої події проекту означає його повне виконання.

При побудові мережевого графа необхідно дотримуватись наступних правил:

☑ довжина дуги не залежить від тривалості роботи;

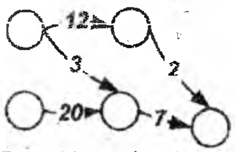


Рис. 10.4. Відображення робіт

☑ дуга не обов'язково повинна бути прямолінійним відрізком;

☑ для дійсних робіт використовуються суцільні, а для фіктивних - пунктирні дуги;

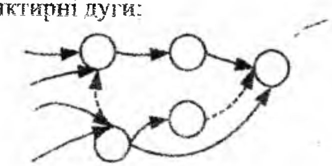


Рис. 10.5. Фіктивні роботи

☑ кожна операція повинна бути представлена тільки однією дугою;

☑ не повинно бути паралельних робіт між тими самими подіями, для запобігання такій ситуації використовують фіктивні роботи;

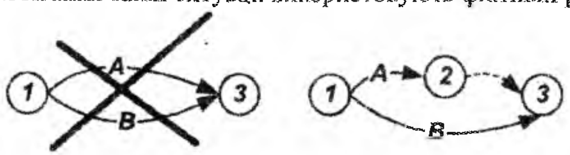


Рис. 10.6. Відображення „паралельних" робіт

☑ бажано уникати перетинання дуг (якщо це можливо);

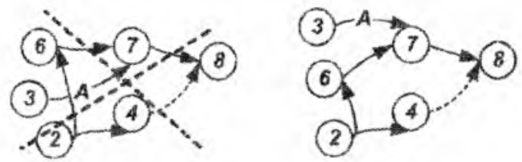


Рис. 10.7. Уникання перетинань робіт

☑ номер початкової події для кожної з робіт повинен бути меншим, ніж кінцевої номера кінцевої події;

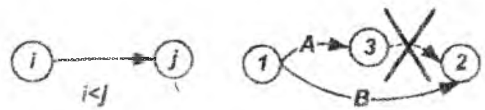


Рис. 10.8. Нумерація робіт

☑ не повинно бути циклів.

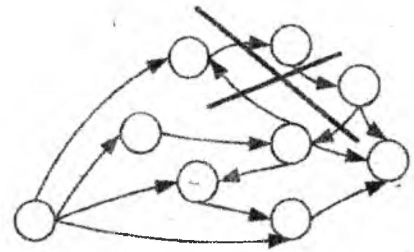


Рис. 10.9. Відсутність циклів

Оскільки роботи, що входять у проект, логічно пов'язані між собою, то перед побудовою мережі необхідно дати відповіді на наступні запитання:

☑ Які роботи необхідно завершити безпосередньо перед початком розглянутої роботи?

☑ Які роботи повинні безпосередньо виконуватись після завершення даної роботи?

☑ Які операції можуть виконуватися одночасно з розглянутою роботою?

Первинні дані для побудови мережевої моделі можуть задаватися різними способами, наприклад:

☑ описанням передбачуваного проекту; у цьому випадку необхідно самостійно розбити його на окремі роботи й установити їхні взаємні зв'язки;

☑ списком робіт проекту; у цьому випадку необхідно проаналізувати зміст робіт і установити існуючі між ними зв'язки;

☑ списком робіт проекту з указанням їхнього упорядкування; у цьому випадку необхідно лише відобразити роботи на мережевому графіку.

Таким чином, підсумуємо вимоги та порядок реалізації структурного планування.

Мережа* CPU/PERT *має наступну структуру:

☑ кожна дуга мережі представляє операцію та обов'язково має певну орієнтацію;

☑ мережа проекту є безконтурною;

☑ не припускається ймовірнісне розгалуження (як у стохастичних лотерей);

☑ робота не може початися, поки всі попередні не будуть виконані;

☑ робота позначається парою подій, що її однозначно ідентифікують.

Побудова мережі відбувається на основі заданого відношення слідування (передування) на множині операцій (робіт) проекту.

Це відношення є транзитивним. Операціям проекту відповідають дуги мережі, вузлам – події. Подія вважається такою, що звершилася, якщо всі роботи, що входять в неї, виконані.

Мережа будується за наступними правилами:

☑ кожній роботі проекту відповідає одна і лише одна дуга;

☑ кожна робота однозначно ідентифікується парою подій;

☑ при побудові мережі необхідно знати: які роботи повинні бути завершені безпосередньо перед початком наступної, або які роботи безпосередньо слідують за кожною з робіт проекту; які роботи можуть виконуватися одночасно.

Послідовність побудови мережі включає наступні кроки:

☑ структуризація проекту до множини взаємопов'язаних робіт, виконання яких означатиме реалізацію проекту;

☑ отримання відношення безпосереднього передування (слідування) на множині робіт задачі шляхом аналізу її змістовної постановки;

☑ визначення множин початкових та завершальних робіт проекту;

☑ побудова мережі проекту з включенням у необхідних випадках фіктивних робіт. Фіктивні роботи мають нульову тривалість та не потребують для свого виконання ніяких ресурсів і вводяться з метою забезпечення логічних умов слідування робіт проекту та для їх однозначної

ідентифікації парою подій;

☑ нумерація вузлів (подій) мережі таким чином, щоб будь-яка робота виходила з події з меншим номером і входила в подію з більшим номером $(\forall (i,j) \in A : i < j)$.

10.3. ОСНОВНІ ПАРАМЕТРИ МЕРЕЖІ ТИПУ CPM ТА ЇХ ВИЗНАЧЕННЯ

Календарне планування на мережі здійснюється шляхом розрахунку параметрів робіт та подій в часі.

Основні параметри детермінованої мережі CPM поділяються на параметри подій мережі та параметри робіт мережі, і включають ранні та пізні терміни звершення подій і початку та завершення робіт й резерви часу.

Для подій розрізняють такі параметри, як: ранні та пізні терміни звершення подій і резерви часу подій.

Для робіт мережі існують такі параметри: ранні та пізні терміни початку та завершення робіт (4 параметри) та резерви часу робіт, яких є декілька — повний, вільний, незалежний та гарантований.

Ранній термін звершення події – це найраніший термін, в який трапиться подія. Подія вважається такою, що звершилася, якщо виконані всі роботи, що входять в цю подію. Якщо подія звершилася, можуть почати виконання (але не зобов'язані) всі роботи, що виходять з неї. Виходячи з цих визначень, ранній термін звершення події j – t_j обчислюється рекурентно на основі значень тривалостей робіт, що входять у подію j та ранніх термінів звершення подій, з яких виходять ці роботи:

$$t_j = \underset{\forall (i \in N)(i,j) \in A}{Max} \left(t_i + t_{ij} \right). \qquad (10.1)$$

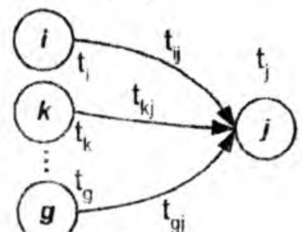


Рис. 10.10. Ранній термін звершення події

Для початкової події мережі $t_0 = 0$. Таким чином найраніше подія може трапитися (звершитися), коли завершаться всі роботи, що входять в неї, або іншими словами, коли завершить виконання остання з робіт, що входять в цю подію. Ранній термін звершення останньої події мережі й визначатиме тривалість виконання проекту загалом. Окрім того, для керівництва проектом важливим є послідовність робіт, яка визначає тривалість виконання проекту – критичний шлях. Для визначення критичного шляху необхідно розрахувати пізні терміни звершення подій та резерви часу подій.

Пізній термін звершення події – це найпізніший можливий термін, коли може трапитися подія за умови не збільшення загального часу виконання проекту (який визначений у процесі розрахунку ранніх термінів звершення подій). Пізній термін звершення події j – T_j визначається також рекурентним шляхом на основі значень тривалостей робіт, що виходять з даної події та пізніх термінів звершення подій, в які вони входять. Оскільки відомим є загальний термін робіт над проектом, то розрахунок реалізується оберненою ходою – починаючи з останньої події проекту – його завершення в напрямку до початкової, причому для останньої події n $T_n = t_n$, а для інших

$$T_j = \min_{\forall(i \in N):(j,i) \in A} \left(T_i - t_{ji}\right). \qquad (10.2)$$

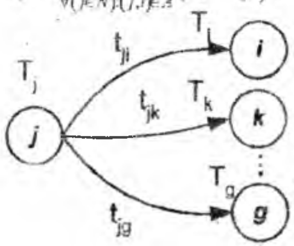


Рис. 10.11. Пізній термін звершення події

Резерв часу події визначає значення, на яке може бути відсунутий в часі момент, коли подія трапиться, порівняно з раннім терміном її звершення: $R_j = T_j - t_j$. Якщо резерв часу події рівний нулю, то вона обов'язково повинна звершитися у вказаний ранній час, тобто є критичною.

Результати розрахунку параметрів подій мережі типу CPM зручно відображати наступним чином:

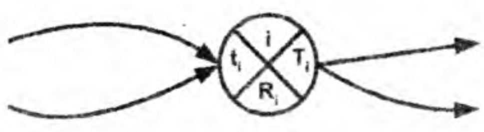


Рис. 10.12. Параметри подій мережі

Критичний шлях в мережі проекту (найдовший шлях) проходить через ті події, резерв часу яких є нуль. Ніяка робота з числа тих, що знаходяться на критичному шляху, не може бути подовжена, тому що це приведе до відповідного зростання тривалості виконання проекту загалом. В деяких випадках критичний шлях може бути визначений через події неоднозначно – однозначність забезпечує такий параметр, як резерв часу роботи. І навпаки – роботи та події, що не знаходяться на критичному шляху, можуть бути затримані на певний ненульовий проміжок часу без впливу на термін завершення проекту загалом.

З точки зору керівництва проектом, при управлінні основна увага повинна бути звернута на роботи критичного шляху. Скорочення критичного шляху приводить до скорочення термінів виконання проекту, але при цьому може з'явитися інший критичний шлях, і надалі скорочення терміну завершення проекту буде можливим за умови зменшення тривалості кожного з критичних шляхів.

Ранній термін початку роботи дорівнює ранньому терміну звершення тієї події, з якої виходить робота – $t_{ij}^{rn} = t_i$. (10.3)

Ранній термін завершення роботи визначається як $t_{ij}^{rc} = t_i + t_{ij}$ (10.4)

Пізній термін завершення роботи рівний пізньому терміну звершення події, до якої входить робота – $t_{ij}^{pe} = T_j$. (10.5)

Пізній термін початку роботи визначається через пізній термін завершення роботи – $t_{ij}^{pa} = T_j - t_{ij}$. (10.6)

З метою планування термінів виконання різних робіт важливо мати деякі інші параметри, які б відображали ступінь свободи для задання термінів за наявності різноманітних умов. Резерв часу є показником гнучкості планування термінів в мережній моделі проекту. Резерви служать не лише для визначення критичного шляху – важливішим їх призначенням є планування фактичних термінів виконання робіт, що не знаходяться на критичному шляху. При плануванні термінів за допомогою CPU вважається, що обмеження на ресурси , необхідні для виконання проекту, відсутні. Насправді потрібно розподілити роботи в часі так, щоб наявні ресурси споживалися рівномірно, але в будь-якому випадку відправною точкою є розрахунок часових параметрів мережі.

Повний резерв часу роботи визначається як $r_{ij}^{a} = T_j - t_i - t_{ij}$. (10.7)

Повний резерв часу визначає, на скільки можна зсунути на осі часу початок виконання роботи за умови, що всі інші роботи мережі будуть виконуватися якнайраніше, тобто є максимально можливою затримкою у виконанні роботи, яка ще не приведе до збільшення терміну завершення робіт над проектом. Робота, що має значення повного резерву часу рівне нулю, знаходиться на критичному шляху (таким чином, критичний шлях однозначно виявляється як послідовність робіт з нульовими значеннями повного резерву часу). Однак, хоча повний резерв є важливим для визначення критичного шляху, при плануванні термінів початку для окремих робіт він є недостатнім.

Вільний резерв часу роботи розраховується так:

$$r_{ij}^{f} = t_j - t_i - t_{ij} \tag{10.8}$$

і показує, на скільки можна зсунути на осі часу початок виконання роботи за умови, що проміжок між раннім та пізнім терміном звершення подій j та i буде постійним і рівним $\Delta = t_j - t_i$. Вільний резерв часу – це показник максимальної затримки роботи (i, j), яка не впливатиме на початок наступних робіт. В цьому випадку теж вважається, що попередні роботи завершаться якнайраніше.

Незалежний резерв часу роботи є величиною затримки у виконанні роботи за найнесприятливіших умов – $r_{ij}^{н} = \max\left\{0; t_j - T_i - t_{ij}\right\}$. (10.9)

Нуль введений для того, щоб виключити від'ємні значення. Від'ємні значення є також показником можливого ступеня порушення зв'язку між роботами проекту.

Гарантований резерв часу роботи є величиною максимально можливої затримки роботи, яка не впливає на остаточний термін завершення проекту загалом за умови, що передуючі роботи виконуються з запізненням –

$$r_{ij}^{g} = T_j - T_i - t_{ij} \tag{10.10}$$

Цей показник є одним з найзручніших, так як допускає затримку лише наступних робіт, але не проекту загалом.

Часові параметри мережі розраховуються наступним чином.

Крок 1. Розраховуємо ранні терміни звершення подій.

Для початкової події мережі $t_0 = 0$. Для інших подій ранні терміни розраховуємо рекурентно, просуваючись у напрямку останньої події згідно зі співвідношенням $t_j = \underset{\forall(i\in N):(i,j)\in A}{Max}\left(t_i + t_{ij}\right)$. Останнім розраховуємо ранній термін звершення останньої події мережі n - t_n, що й визначає тривалість робіт над проектом.

Крок 2. Розраховуємо пізні терміни звершення подій .

Для останньої події мережі $T_n = t_n$. Для інших подій ранні терміни розраховуємо рекурентно, просуваючись у напрямку початкової події згідно

зі співвідношення $T_j = \underset{\forall(j\in N):(j,i)\in A}{\min}\left(T_i - t_{ji}\right)$. Останнім розраховуємо пізній термін звершення початкової події мережі $T_0 = 0$, значення якого рівне нулю.

Крок 3. Визначаємо резерви часу подій та критичний шлях.

Резерви часу подій розраховуємо на ґрунті пізніх та ранніх термінів їх звершення за формулою $R_j = T_j - t_j$. Критичний шлях проходить через події, резерви часу яких нульові.

Крок 4.Визначаємо ранні та пізні терміни початку та завершення робіт.

Ці характеристики розраховуємо на основі t_i, T_j - ранніх та пізніх термінів звершення подій та тривалостей відповідних робіт t_{ij} за наступними співвідношеннями - $t_{ij}^{rn} = t_i$, $t_{ij}^{rz} = t_i + t_{ij}$, $t_{ij}^{pn} = T_j - t_{ij}$, $t_{ij}^{pz} = T_j$.

Крок 5.Розраховуємо резерви часу робіт:

- ☑ повний резерв часу роботи визначається як $r_{ij}^{п} = T_j - t_i - t_{ij}$;
- ☑ вільний резерв часу роботи розраховується як $r_{ij}^{f} = t_j - t_i - t_{ij}$;
- ☑ незалежний резерв часу роботи дорівнює $r_{ij}^{н} = \max\left\{0; t_j - T_i - t_{ij}\right\}$;
- ☑ гарантований резерв часу роботи визначається як $r_{ij}^{g} = T_j - T_i - t_{ij}$.

Крок 6. Оптимізація розподілу ресурсів, що витрачаються на виконання робіт.

Після розрахунків отримані результати аналізуються і служать для остаточного встановлення термінів початку та завершення робіт (для робіт критичного шляху ці терміни зберігаються, тому що їх зміна викличе збільшення часу виконання проекту загалом) в межах їх резервів таким чином, щоб ресурси використовувалися якомога рівномірніше. Ця процедура загалом має евристичний характер.

10.4. МЕТОД PERT

Метод PERT орієнтований на врахування невизначеностей у тривалостях виконання робіт мережі, які описуються стохастичними характеристиками.

Кожна робота проекту характеризується трьома оцінками її тривалості, які отримуються зазвичай шляхом опитування експертів:

- ☑ найвірогіднішою тривалістю виконання m ;

☑ найменшою очікуваною тривалістю виконання — оптимістична тривалість a;

☑ найбільша очікувана тривалість b – песимістична оцінка.

Найвірогідніший час виконання роботи – це оцінка часу її виконання за нормальних умов. Оптимістична та песимістична оцінки визначають розмах коливань тривалості під дією стохастичних факторів. Песимістична оцінка не враховує незвичні тривалі, затримки чи катастрофи, а тому фактична тривалість виконання роботи може знаходитися й за межами визначеного інтервалу тривалостей.

Для описання розподілу вірогідності виконання роботи залежно від часу використовується β-розподіл. Нам необхідно, використовуючи цю інформацію, отримати такі параметри закону, як математичне сподівання та дисперсію у вигляді функцій від значень a, b, m. Для цього використаємо деякі евристичні прийоми. Форма β-розподілу в загальному випадку відображена нижче.

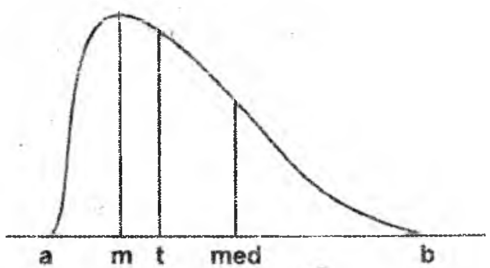


Рис 10.13. Розподіл тривалості роботи.

Припустимо, що "вага" середньої тривалості (медіани розподілу med), $med=(a+b)/2$ в два рази менша, ніж "вага" найімовірнішої тривалості m (моди розподілу). За цього припущення значення математичного сподівання буде середнім арифметичним між med та зваженим значенням m, а саме $t=(med+2m)/3=(a+b+4m)/6$. (10.11)

Розмах (a,b) вважатимемо рівним $\approx 6\sigma$ (за цієї умови для β-розподілу біля 90% площі під функцією густини розподілу буде знаходитися в межах розмаху). Виходячи з цього, $\sigma=(b-a)/6$, тобто дисперсія $D=(b-a)^2/36$. (10.12)

Припустимо також, що всі операції проекту статистично незалежні. Якщо подія i зв'язана з подією 0 (початковою) одним шляхом, то $M[t_i]$ — матсподівання часу звершення події, буде рівним сумі матсподівань операцій, що знаходяться на цьому шляху, а дисперсія $D[t_i]$ – відповідно

сумі дисперсій. Якщо ж існує більш ніж один шлях, то вважатимемо, що досить отримати статистичний розподіл тривалості лише цього шляху і за ним розрахувати $M[t_i]$ та $D[t_i]$.

Однак і за цього спрощуючого припущення задача все одно залишається достатньо складною в загальному випадку, а тому вважатимемо, що з достатнім ступенем точності ці характеристики отримаємо для шляху, сума очікуваної тривалості операцій для якого найбільша. При рівності цих значень для декількох шляхів обиратимемо той, значення дисперсії для якого максимальне, як такий, що даватиме надійніший результат.

Оскільки $M[t_i]$ є сумою декількох незалежних випадкових величин, то згідно до центральної граничної теореми теорії ймовірностей цей розподіл зі зростанням числа складових в сумі та приблизної рівноцінності їх випадкових дій асимптотично наближатиметься до нормального.

Таким чином, з достатнім ступенем впевненості вважатимемо, що вірогідність звершення події i з раннім терміном звершення t_i в директивний термін d_i становитиме:

$$P(t_i \le d_i)=P\left(z\le\frac{d_i-M[t_i]}{\sqrt{D_i}}\right)=\Phi\left(\frac{d_i-M[t_i]}{\sqrt{D_i}}\right)-\Phi(-\infty), \qquad (10.13)$$

де $z=\dfrac{t_i-M[t_i]}{\sqrt{D}}$ – випадкова величина, розподілена за нормальним законом з середнім 0 та дисперсією 1 (нормована нормальна величина, значення інтегральної функції для якої знаходимо з таблиць).

Таким чином, розрахунок мережі PERT здійснюється в наступній послідовності.

Крок 1. Визначаємо значення параметрів робіт мережі.

Шляхом оцінювання або експертного опитування визначаємо найімовірнішу тривалість виконання m, оптимістичну тривалість a та песимістичну тривалість b для кожної з робіт мережі. Для всіх робіт розраховуємо їх середні тривалості $t=(a+b+4m)/6$ та значення дисперсій $D=(b-a)^2/36$, вважаючи, що тривалість виконання кожної роботи є випадковою величиною з β-розподілом.

Крок 2. Розрахунок критичного шляху.

На основі розрахованих середніх тривалостей робіт розраховуємо ранні та пізні терміни звершення подій та їх резерви, як у методі критичного шляху, і отримуємо події та роботи, через які проходить критичний шлях, а також його тривалість.

Крок 3. Статистичне оцінювання.

Розрахована в попередньому пункті тривалість критичного шляху вважається середньою - $M[t_i]$, і розраховується дисперсія тривалості $D[t_i]$, як сума дисперсій робіт, що входять до критичного шляху. Тривалість критичного шляху вважається випадковою величиною, що розподілена за нормальним законом та має розраховані значення параметрів – середнього та дисперсії. Ґрунтуючись на цих характеристиках, за допомогою співвідношення

$$P(t_i \le d_i) = P\left(z \le \frac{d_i - M[t_i]}{\sqrt{D_i}}\right) = \Phi\left(\frac{d_i - M[t_i]}{\sqrt{D_i}}\right) - \Phi(-\infty),$$

$z = \dfrac{t_i - M[t_i]}{\sqrt{D}}$ – випадкова величина, розподілена за нормальним законом з середнім 0 та дисперсією 1, розраховуємо вірогідність завершення робіт над проектом в заданий час, можливо також здійснити обернений розрахунок – при заданому рівні довірчої вірогідності отримати інтервальну оцінку тривалості проекту.

В деяких випадках доцільно порівняти інтервальні оцінки для критичного та одного або декількох підкритичних шляхів, тому що внаслідок різних значень дисперсії для кожного шляху ці оцінки будуть різними. Окрім того, таку оцінку можна реалізувати для всіх подій мережі зазвичай оцінювання реалізується лише для найважливіших подій.

10.5. ОПТИМІЗАЦІЯ МЕРЕЖІ

Широке розповсюдження методів PERT та CPM в 60-х роках викликало зростання числа їх застосувань для планування проектами та довело їх корисність. Однак ці методи доволі наближено відображають реальну ситуацію, оскільки розглядаються лише терміни, але не оцінюється потреба в необхідних ресурсах та динаміка їх використання.

Внаслідок цього в подальшому основна увага зверталася на розробку методів, які дозволяли б оцінити й потребу в ресурсах на виконання проекту – тобто задача управління проектом розглядається як двокритерійна – за часом та за грошовим чи іншим інтегрованим виразом ресурсів.

Проблематика розподілу ресурсів полягає в тому, що за нормальних умов вони повинні споживатися рівномірно, і навіть у випадку детермінованих тривалостей операцій після розрахунку параметрів мережі

за допомогою методу CPM результати можуть бути приведені до календарного плану виконання робіт багатьма варіантами. Це пояснюється наявністю резервів часу у робіт, що не є критичними, тобто ці роботи в певних межах можна пересувати вздовж осі часу, що змінює інтенсивність споживання ресурсу під час виконання проекту.

Для ілюстрації цих положень розглянемо приклад.

Нехай задана мережа з тривалостями робіт та кількостями споживаного ресурсу, причому інтенсивність споживання ресурсу, необхідного для виконання роботи, є постійною, тобто ресурс використовується рівномірно. Розрахуємо параметри подій мережі.

Відобразимо два з можливих варіанти розміщення робіт та розподілу ресурсів.

Зрозуміло, що ці два варіанти нерівноцінні щодо розподілу ресурсів – варіант а за певних умов може бути більш привабливим, ніж b, внаслідок порівняно значної інтенсивності використання ресурсів на початку виконання проекту, та зменшення її в кінці, що у випадку непередбачених причин створює кращі умови для завершення проекту в термін.

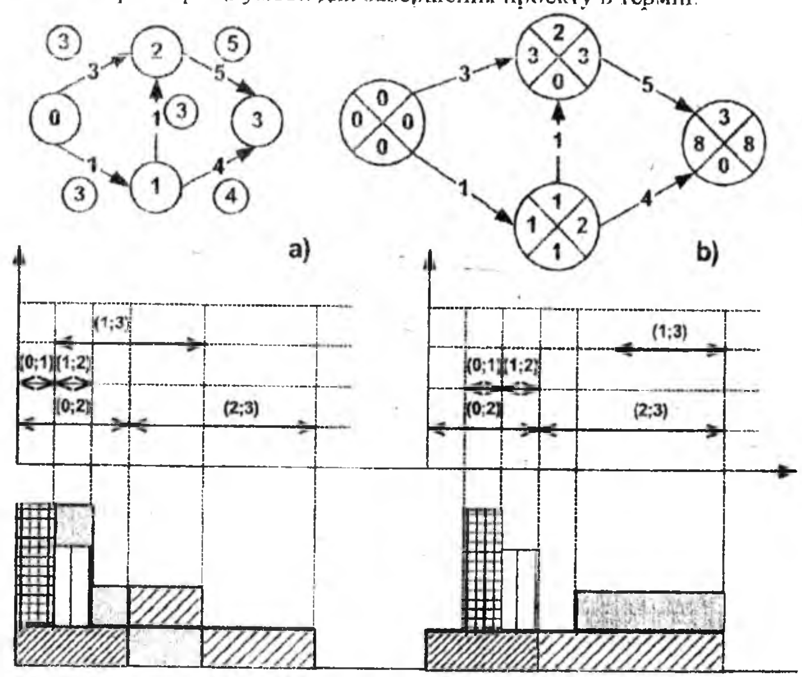


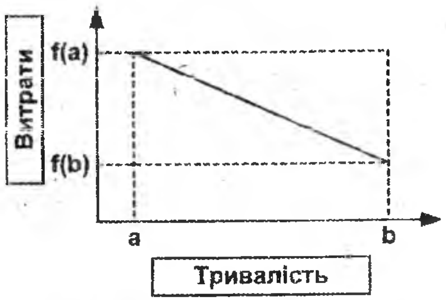
Рис. 10.14. Варіанти розподілу ресурсів на проект

У більшості випадків прагнуть до рівномірного розподілу ресурсів протягом виконання проекту.

Взагалі для кожної з робіт проекту існує певний інтервал зміни можливої тривалості виконання роботи в залежності від кількості використаного ресурсу – чим більша кількість ресурсу використовується, тим за менший час можна виконати роботу. У більшості випадків виявляється можливим апроксимувати цю залежність у вигляді лінійної. Для однозначного розрахунку параметрів цієї залежності достатньо знання необхідної кількості ресурсу у двох крайніх точках – при мінімальній та максимальній тривалості роботи.

Таким чином, вважатимемо, що вартість виконання роботи лінійно залежить від її тривалості – $f(x) = -kx + d$. (10.14)

Знаючи координати двох точок цієї прямої, розрахуємо значення k та d.



$$k = \frac{f(a) - f(b)}{b - a}, \quad d = \frac{af(b) - bf(a)}{a - b}, \quad a \le t \le b. \qquad (10.15)$$

Коефіцієнт k при розв'язуванні задачі є одним з найважливіших параметрів і є вартістю, яку необхідно вкласти, щоб скоротити виконання роботи на одиницю часу. Іншими словами, чим меншим є значення k, тим меншу кількість ресурсу необхідно витратити для скорочення тривалості роботи.

В цій задачі необхідно для кожного значення тривалості проекту визначити мінімальний об'єм ресурсів (в грошових одиницях чи іншій однорідній одиниці вимірювання), який необхідно витратити для виконання проекту в цей проміжок часу. Для розв'язування задачі застосуємо наступний алгоритм.

Алгоритм мінімізації витрат на виконання робіт мережі у встановлені терміни.

Крок 1. Будуємо мережу за заданими відношеннями передування. Розраховуємо значення k_{ij} для кожної роботи мережі $(i, j) \in A$. Вважаємо, що всі тривалості виконання операцій є максимальними, $\forall (i, j) \in A : t_{ij} = b_{ij}$, та рахуємо сумарну вартість проекту

$$C = \sum_{(i,j) \in A} f(b_{ij}).$$

Крок 2. Розраховуємо наступні характеристики мережі: ранні та пізні терміни звершення подій, і на їх основі вільні резерви часу всіх операцій мережі: $t_0 = 0$; $t_j = \max\limits_{\forall (i \in N)(i,j) \in A} (t_i + t_{ij})$; $T_n = t_n$; $T_j = \min\limits_{\forall (j \in N)(j,i) \in A} (T_i - t_{ji})$; $r_{ij} = t_j - t_i - t_{ij}$. Таким чином, в просторі "тривалість – вартість проекту" отримуємо координати однієї з граничних точок множини Парето-оптимальних розв'язків.

Крок 3. Визначаємо операції – кандидати на скорочення. Для цього спочатку визначаємо множину критичних шляхів мережі $L = \{K_1, \dots, K_n\}$. На кожному з шляхів $l = 1, n$ визначаємо операцію-кандидат на скорочення: $(i, j)_l = \arg \min\limits_{((i,j) \in K_l) \wedge (t_{ij} \ne a_{ij})} k_{ij}$.

Якщо хоча б для одного шляху такої операції не існує (тобто всі операції такого критичного шляху зменшені до мінімальних значень a_{ij}), то **стоп.** Досягнута інша гранична точка області Парето-оптимальних рішень, тобто побудована область Парето-оптимальних рішень задачі. В іншому випадку виконуємо наступний крок.

Крок 4. Визначаємо величину, на яку можуть бути зменшені одночасно ці операції – $\Delta_0 = \min\limits_{\forall (i,j)_l} (t_{ij} - a_{ij})$.

В результаті скорочення всіх критичних шляхів на величину Δ_0 може з'явитися новий критичний шлях з числа підкритичних, тому необхідно визначити, на яку величину можна скорочувати критичні шляхи з огляду на те, щоб не з'явився новий єдиний критичний шлях.

З цією метою використовуємо розраховані значення незалежних резервів часу робіт, що не належать до критичних шляхів. Досліджуємо, чи впливає зменшення всіх операцій $(i, j)_l$ на одиницю на зменшення незалежних резервів некритичних робіт. Всі некритичні роботи, які зменшуватимуть значення незалежного резерву при зменшенні $(i, j)_l$, утворюють множину $Q = \left\{ (i, j) \in A \setminus L \,\middle|\, t'_{(i,j)_l} = t_{(i,j)_l} - 1 \Rightarrow r'_{ij} < r_{ij} \right\}$.

Визначаємо межу скорочення, зумовлену зміною незалежних резервів часу робіт, як $\Delta_r = \min\limits_{(i,j)\in Q} r_{ij}$. Якщо $Q = \varnothing$, то $\Delta = \Delta_0'$, інакше $\Delta = Min\{\Delta_0, \Delta_r\}$.

Крок 5. Перераховуємо тривалості операцій та вартість комплексу при скороченні: $t_{(i,j)_l} = t_{(i,j)_l} - \Delta$; $T = T - \Delta$; $C = C + \Delta \sum\limits_{(i,j)_l} k_{(i,j)}$.

Переходимо до кроку 2.

Побудована таким чином множина Парето-оптимальних розв'язків може бути використана для аналізу наявних можливостей розподілу ресурсу як в інтегрованому вигляді, так і з визначенням конкретних значень потреби в ресурсах та тривалості кожної з робіт мережі. У цьому випадку для однозначної ідентифікації цих характеристик для кожної роботи необхідно зберігати характеристики варіантів мережі в скінченій множині точок злому ламаної, що відображає множину Парето-оптимальних розв'язків в просторі двох критеріїв – часу та ресурсів.

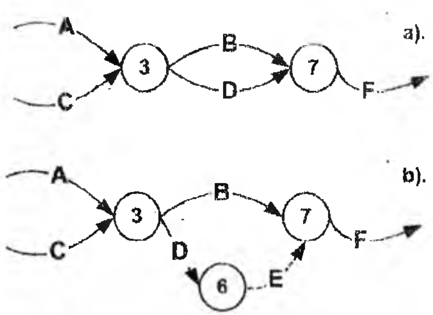
ПРИКЛАДИ

Приклад 10.1. Основні випадки введення фіктивних робіт.

Заданий наступний фрагмент відношення передування для мережі: $A \prec B, D; C \prec B, D; B \prec F; D \prec F$ (запис $D \prec F$ означає, що робота D безпосередньо передує роботі F). Необхідно побудувати відповідний фрагмент мережі проекту.

Розв'язання.

Якщо спробувати безпосередньо побудувати відповідний фрагмент мережі, то ми отримаємо а):



У випадку а) роботи B та D ідентифікуються однією парою подій $(3, 7)$, що неприпустимо. Тому вводимо фіктивну роботу E – позначена на b) пунктиром, в результаті чого кожна з робіт фрагменту однозначно ідентифікується різними парами подій.

Приклад 10.2. Основні випадки введення фіктивних робіт.

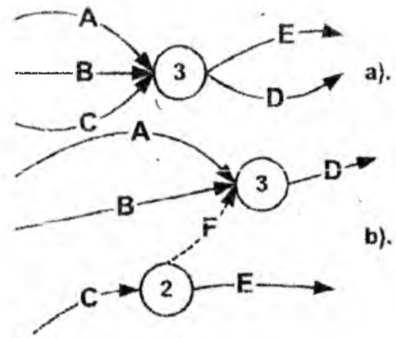
Заданий наступний фрагмент відношення передування:

$A \prec D; B \prec D; C \prec D, E$.

Необхідно побудувати фрагмент мережі графічно.

Розв'язання.

Побудуємо відповідний фрагмент мережі:



Варіант а) є неприпустимим, тому що він реалізує інше, ніж задане, відношення передування, а саме $A \prec D, E; B \prec D, E; C \prec D, E$. У варіанті b) проблема розв'язана за рахунок введення фіктивної роботи F .

Приклад 10.3. Структурне планування.

Необхідно побудувати мережу проєкту за наступними відношеннями передування:

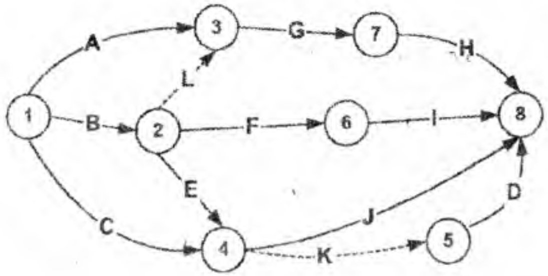
$A \prec G; B \prec G, F, E; C \prec J, D; E \prec J, D; F \prec I; G \prec H.$

Розв'язання.

Визначимо множину початкових робіт мережі. Перед початковими роботами немає ніяких інших, які б їм передували, і тому вони можуть бути лише в лівій частині відношення передування (надкреслені роботи). Аналогічно завершальними є роботи, після яких ніяких інших немає, тобто знаходяться лише в правій частині відношення передування (підкреслені роботи).

$\overset{=}{A} \prec G; \overset{=}{B} \prec G, F, E; \overset{=}{C} \prec \underline{\underline{J}}, \underline{\underline{D}}; E \prec \underline{\underline{J}}, \underline{\underline{D}}; F \prec \underline{\underline{I}}; G \prec \underline{\underline{H}}.$

Будуємо мережу з початкової події, з якої виходять всі початкові роботи, та з кінцевої події, в яку входять всі завершальні роботи. Для уникнення неоднозначності вводимо фіктивну роботу K, а для забезпечення виконання логіки виконання робіт – фіктивну роботу L.



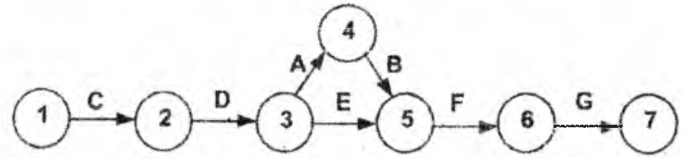
Події нумеруємо таким чином, щоб робота завжди виходила з події з меншим номером, і входила в подію з більшим номером. Оскільки таких способів існує декілька (і відповідні алгоритми нумерації), то обирається будь-який з них.

Приклад 10.4. Структурне проектування.

Необхідно побудувати мережну модель програми опитування суспільної думки, що включає розробку анкет (A; 1 день) і роздрук анкет (B; 0,5 дня), приймання на роботу (C; 2 дні) і навчання (D; 2 дні) персоналу, вибір множини опитуваних (E; 2 дні), розсилання їм анкет (F; 1 день) і аналіз отриманих даних (G; 5 днів).

Розв'язання.

З умови задачі нам відомий зміст робіт, але безпосередньо не зазначені взаємозв'язки між роботами. Тому для їхнього встановлення необхідно проаналізувати зміст кожної конкретної роботи і з'ясувати, які з інших робіт повинні їй безпосередньо передувати. Вихідною роботою, що починає мережу, у даному випадку є „приймання на роботу" (C), оскільки всі інші роботи повинні виконуватися вже прийнятими на роботу співробітниками. Перед виконанням усіх робіт з опитування суспільної думки співробітників необхідно навчити персонал (D). Перед тим як розіслати анкети (F), їх треба розробити (A), роздрукувати (B) і обрати множину опитуваних (E), причому роботу з анкетами і вибір опитуваних можна виконувати одночасно. Завершальною роботою проекту є аналіз отриманих даних (G), яку не можна виконати без попереднього розсилання анкет (F). У результаті цих міркувань побудуємо мережну модель і перенумеруємо події моделі наступним чином:



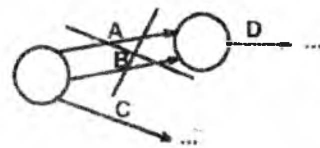
Приклад 10.5. Структурне проектування.

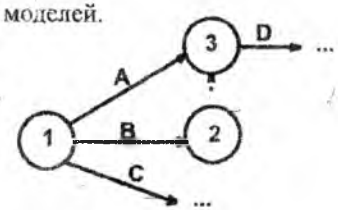
Побудуйте мережну модель, що включає роботи A, B, C, ..., L, що відображає наступне упорядкування робіт:

1).A, B і C – роботи, що починають проект; 2).A і B передують D; 3).B передує E, F і H; 4).F і C передує G; 5).E і H передують I і J; 6).C, D, F і J передують K; 7).K передує L.

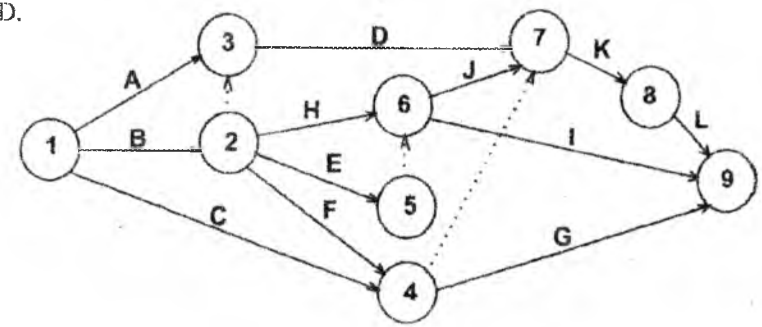
Розв'язання.

У пункті 1) умови безпосередньо зазначено, що A, B і C є вихідними роботами, тому зобразимо їхніми трьома дугами, що виходять з вихідної події 1. Пункт 2) умови означає, що дуги робіт A і B повинні закінчитися в одній події, з якої вийде дуга роботи D. Але оскільки дуги робіт A та B також і починаються в одній події, то є паралельність робіт, що суперечить правилам побудови мережних моделей.





Для її усунення введемо додаткову подію 2, у яку ввійде робота B, після чого з'єднаємо події 2 і 3, у які входять роботи A і B пунктирною дугою фіктивної роботи. У цьому випадку фіктивна робота (2,3) не відповідає ніякій реальній роботі, а лише відображає логічний зв'язок між роботами B і D.



Відповідно до пункту 3) умови задачі з події 2, виходять три дуги робіт E, F і H. Відповідно до пункту 4) умови задачі дуги робіт C і F повинні ввійти в спільну подію, з якої вийде дуга роботи G. Проблема з паралельністю робіт E і H (пункт 5 умови задачі) вирішується шляхом уведення додаткової події 5 і фіктивної роботи (5,6). Для відображення в мережній моделі пункту 6) умови задачі заведемо дуги робіт D і J у подію 7, а зв'язок робіт F і C з роботою K відобразимо за допомогою фіктивної роботи (4,7). Дуги робіт F і C не можна було безпосередньо вводити в подію 7, тому що наступною після них повинна бути робота G, що з роботами D і J ніяк не пов'язана. Стрілка роботи L виходить з події 8, тобто після закінчення роботи K відповідно до пункту 7) умови задачі.

Оскільки в умові не зазначено, що роботи L, I і G передують якимнебудь іншим роботам, то саме ці роботи є завершальними і ввійдуть у завершальну подію 9. Нумерацію подій проводимо після побудови сіткового графіка, стежачи за тим, щоб номер початкової події кожної роботи був меншим за номер її кінцевої події.

Приклад 10.6. Розрахунок параметрів мережі CPM.

Розрахувати параметри мережі , для якої задане відношення передування:

$A \prec I; B \prec D, J, H; C \prec E; I \prec J, H;$

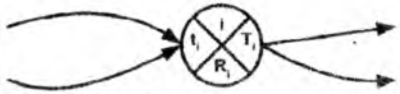
$D \prec G, K; E \prec F, G, K; H \prec K; F \prec L$.

Тривалості робіт задані таблицею.

A	B	C	D	E	F	G	H	I	J	K	L
3	6	2	12	8	15	4	7	3	4	5	1

Розв'язання.

Результати розрахунку параметрів подій мережі типу CPM відображатимемо наступним чином:



Спочатку будуємо мережу (структурне планування).

Оскільки проект вже структуризований до множини взаємопов'язаних робіт і задані відношення безпосереднього передування (слідування) на множині робіт задачі, то приступаємо до побудови структури мережі.

Для цього визначаємо визначення множину початкових *(A,B,C)* та завершальних робіт *(J,K,L,G)* мережі, і будуємо мережу. Для того, щоб розв'язати суперечність у відношеннях передування $B \prec D, J, H; I \prec J, H$, вводимо фіктивну роботу *M*, і з тої ж причини для відношень $D \prec G, K; E \prec F, G, K; H \prec K$ вводимо фіктивні роботи *O* та *N*.

Нумеруємо вузли (події) мережі таким чином, щоб будь-яка робота виходила з події з меншим номером і входила в подію з більшим номером. Слід зауважити, що таких нумерацій існує, як правило, декілька.

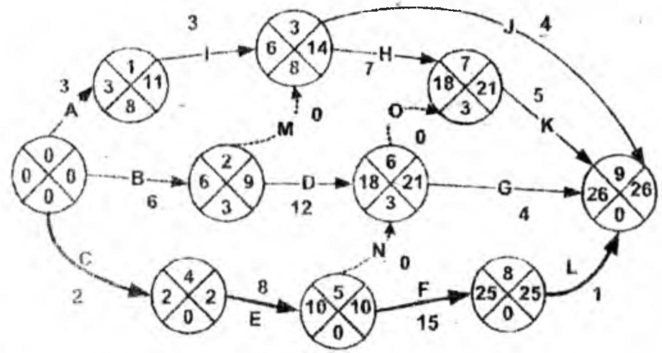
Після цього розраховуємо часові характеристики мережі (календарне планування).

Розраховуємо ранні строки звершення подій у наступній послідовності: подія 0, потім події 1,2, та 4; далі 3,5; потім 6 та 8; і нарешті 9. Отримана тривалість робіт над проектом становить 26 одиниць часу.

Розраховуємо пізні строки звершення подій. Починаємо з останньої події 9, далі 7 та 8; наступні 6, 3; після цього 1, 2, 5; наступна 4; і нарешті 0 – для цієї події пізній строк становить завжди 0.

Після цього розраховуємо резерви часу подій і визначаємо критичний шлях, що проходить через події 0-4-5-8-9. Далі за формулами розраховуємо часові характеристики робіт.

Отримані результати відображені нижче:



Характеристики подій мережі CPM

	0	1	2	3	4	5	6	7	8	9
t_i	0	3	6	6	2	10	18	18	25	26
T_i	0	11	9	14	2	10	21	21	25	26
R_i	0	8	3	8	0	0	3	3	0	0

Характеристики робіт мережі СРМ:

	t_{ij}	(i,j)	t^{ra}_{ij}	t^{re}_{ij}	t^{pa}_{ij}	t^{pe}_{ij}	r^{a}_{ij}	r^{f}_{ij}	r^{u}_{ij}	r^{g}_{ij}
A	3	0,1	0	3	8	11	8	0	0	8
B	6	0,2	0	6	3	9	3	0	0	3
C	2	0,4	0	2	0	2	0	0	0	0
D	12	2,6	6	18	9	21	3	0	0	0
E	8	4,5	2	10	2	10	0	0	0	0
F	15	5,8	10	25	10	25	0	0	0	0
G	4	6,9	18	22	22	26	4	4	1	1
H	7	3,7	6	13	14	21	8	5	0	0
I	3	1,3	3	6	11	14	8	0	0	0
J	4	3,9	6	10	22	26	16	16	8	8
K	5	7,9	18	23	21	26	3	3	0	0
L	1	8,9	25	26	25	26	0	0	0	0
M	0	2,3	6	6	14	14	8	0	0	5
N	0	5,6	10	10	21	21	11	8	8	11
O	0	6,7	18	18	21	21	3	0	0	0

Приклад 10.7. Метод PERT.

Проект заданий відношенням передування на множині робіт: $A \prec E, D; B \prec C; C \prec F, J, H, G; D \prec F, J, H, G;$. $E \prec G, H; F \prec I; G \prec I$.

Оцінки тривалостей кожної з робіт, отримані експертним шляхом, наведені в таблиці:

Робота	Оцінки: (a;b;m)	Робота	Оцінки: (a;b;m)
B	(1;3;2)	F	(1;7;2.5)
A	(2;8;2)	J	(1;3;2)
C	(1;3;2)	G	(6;8;7)
D	(1;11;1.5)	H	(3;11;4)
E	(0.5;7.5;1)	I	(4;8;6)

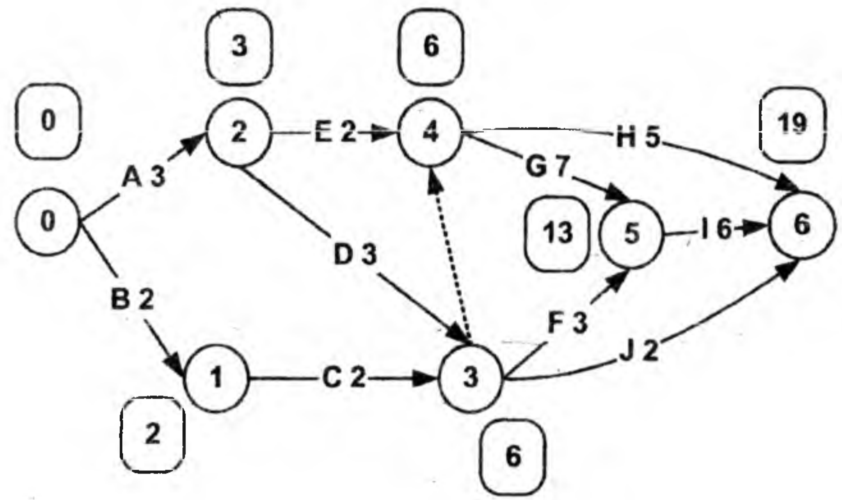
Необхідно визначити вірогідність виконання робіт проекту в заданий директивний строк – 20 одиниць часу .

Розв'язання.

Побудуємо мережу за відношеннями передування і розрахуємо середні тривалості і значення дисперсії для кожної з робіт мережі за формулами: $t = (a + b + 4m)/6$ та $D = (b - a)^2 / 36$.

Роб.	Події	M[tij]	D[tij]	Роб.	Події	M[tij]	D[tij]
B	0;1	2	0.11	F	3;5	3	1.00
A	0;2	3	1.00	J	3;6	2	0.11
C	1;3	2	0.11	G	4;5	7	0.11
D	2;3	3	2.78	H	4;6	5	1.78
E	2;4	2	1.36	I	5;6	6	0.44

Розрахуємо ранні строки звершення подій мережі:



Результати розрахунку з шляхами зводимо в таблицю, розраховуємо значення нормованої нормальної величини $z = \frac{t_i - M[t_i]}{\sqrt{D}}$ та з таблиць нормованого нормального розподілу визначаємо вірогідність виконання проекту в директивний строк, використовуючи формулу

$$P(t_i \le d_i) = P\left(z \le \frac{d_i - M[t_i]}{\sqrt{D_i}}\right) = \Phi\left(\frac{d_i - M[t_i]}{\sqrt{D_i}}\right) - \Phi(-\infty).$$

Подія	Шлях	$M[t_i]$	$D[t_i]$	d_i	u_i	$P(t_i<=d_i)$
6	0-2-3-4-5-6	19	4.35	20	0.480	0.684

Таким чином, вірогідність виконання проекту загалом в заданий директивний строк 20 становить 0.684.

Приклад 10.8. Оптимізація мережі.

Необхідно оптимізувати мережу за критеріями "вартість – тривалість", якщо задане відношення передування

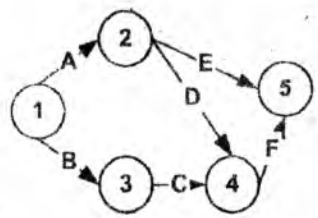
$A \prec D, E; B \prec C; C \prec F; D \prec F$.

Тривалості операцій та витрати ресурсу задані таблицею:

Роб.	Події	Норм. Режим		Напр. режим		k_{id}
		Трив.	Витр.	Трив.	Витр.	
A	1;2	8	100	6	200	50
B	1;3	4	150	2	350	100
D	2;4	2	50	1	90	40
E	2;5	10	100	5	400	60
C	3;4	5	100	1	200	25
F	4;5	3	80	1	100	10

Розв'язання.

Будуємо мережу проекту:

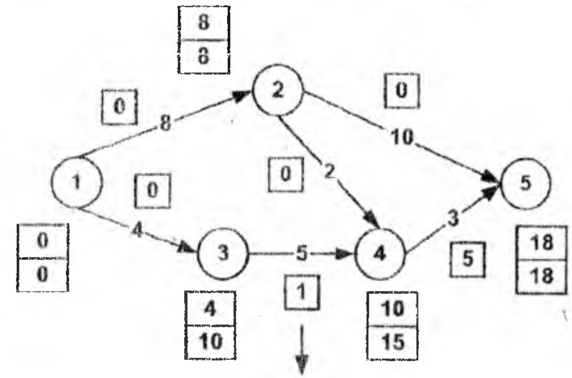


У цю ж таблицю зведені розрахунки k_{id} та пари подій, що відповідають роботам згідно побудованої мережі.

Після побудови мережі розраховуємо ранні та пізні строки звершення подій (на малюнку в прямокутнику згори – ранній строк, знизу – пізній строк звершення події, в квадраті – резерв часу операції.

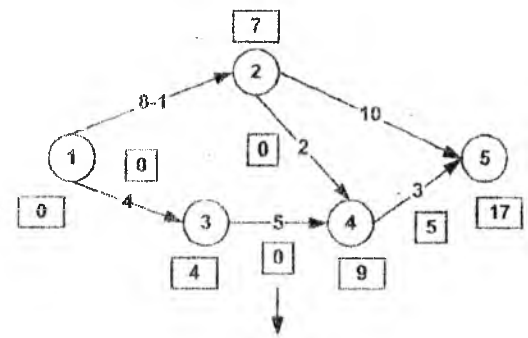
Ітерація 1.

Початкова вартість проекту становить С=100+150+50+100+100+80=580. Множина критичних шляхів включає один критичний шлях, L={K_1},



K_1={1-2-5}. Визначаємо на критичному шляху роботу кандидат на скорочення: $(i,j)_1 = \arg\ Min\left\{50_{(1;2)}, 60_{(2;5)}\right\} = (1;2)$. Розраховуємо, на яку величину можна скоротити виконання роботи: $\Delta_0 = 8-6 = 2$, імітуючи скорочення роботи (1,2) на одиницю (див. мал. нижче) $\Delta_r = Min\{1\}$ Величина скорочення $\Delta = Min\{2;1\} = 1$.

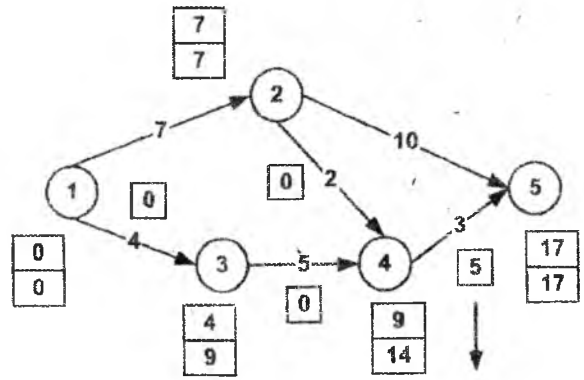
Скорочуємо роботу (1;2) на одиницю: $t_{12}=t_{12}-1=7$. Тривалість проекту теж зменшиться на одиницю: Т=Т-1=18-1=17. Вартість проекту становитиме $C = C + \Delta k_{12} = 580 + 1 \times 50 = 630$



Ітерація 2.

Вартість проекту становить С=630. Множина критичних шляхів включає один критичний шлях, L={K_1}, K_1={1-2-5}. Визначаємо на критичному шляху роботу – кандидат на скорочення:

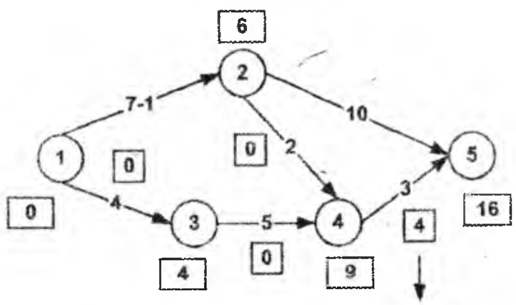
$(i,j)_1 = \arg\ Min\left\{50_{(1;2)}, 60_{(2;5)}\right\} = (1;2)$. Розраховуємо, на яку величину



можна скоротити виконання роботи: $\Delta_0 = 7 - 6 = 1$, імітуючи скорочення роботи (1;2) на одиницю (див. мал. нижче) $\Delta_r = 5$. Величина скорочення $\Delta = Min\{1;5\} = 1$.

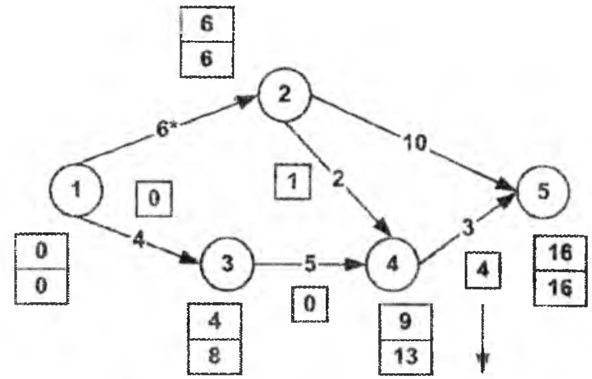
Скорочуємо роботу (1;2) на одиницю: $t_{12}=t_{12}-1=7-1=6^*$. Роботу (1;2) більш скорочувати не можна – досягнута мінімальна її тривалість, і тому відзначаємо її зірочкою (*) .Тривалість проекту теж зменшиться на одиницю: T=T-1=17-1=16.

Вартість проекту $C = C + \Delta k_{12} = 630 + 1\times 50 = 680$.



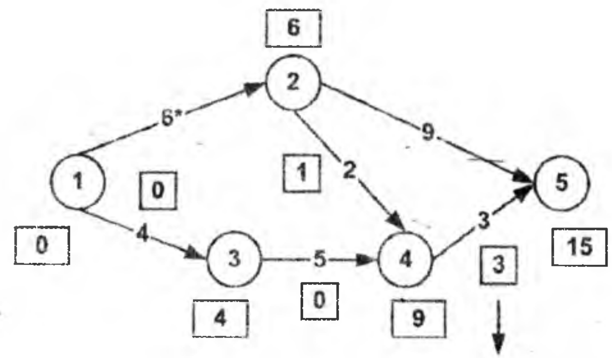
Ітерація 3.

Вартість проекту становить C=680. Множина критичних шляхів включає один критичний шлях, L={K_1}, K_1={1-2-5}. Визначаємо на критичному шляху роботу – кандидат на скорочення: $(i,j)_1 = \arg\ Min\{60_{(2;5)}\} = (2;5)$. Розраховуємо, на яку величину можна скоротити виконання роботи: $\Delta_0 = 10 - 5 = 5$, імітуючи скорочення роботи



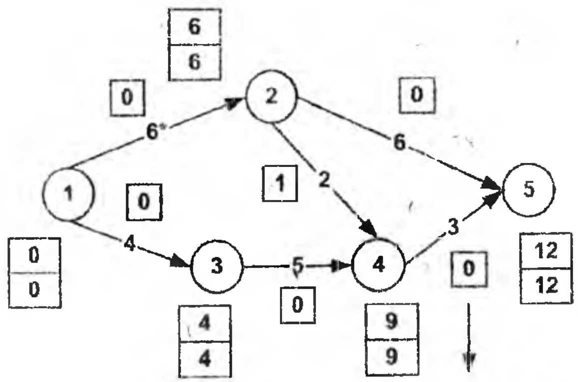
(1;2) на одиницю (див. мал. нижче) $\Delta_r = 4$. Величина скорочення $\Delta = Min\{5;4\} = 4$.

Скорочуємо роботу (2,5) на чотири. $t_{25}=t_{25}-4=10-4=6$. Тривалість проекту теж зменшиться на чотири: T=T-4=16-4=12. Вартість проекту $C = C + \Delta k_{25} = 680 + 4 \times 60 = 920$.



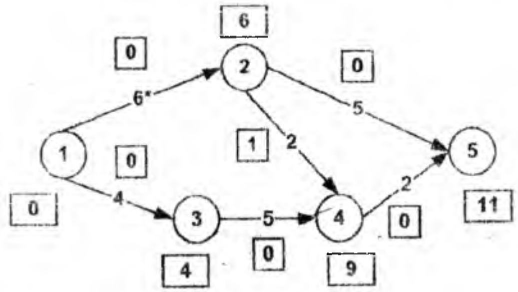
Ітерація 4.

Вартість проекту становить C=920. Множина критичних шляхів включає два критичні шляхи, L={K_1,K_2}, K_1={1-2-5}, K_2={1-3-4-5}. Визначаємо на критичних шляхах роботи – кандидати на скорочення: $(i,j)_1 = \arg\ Min\{60_{(2;5)}\} = (2;5)$, $(i,j)_2 = \arg\ Min\{100;25;10\} = (4;5)$ Розраховуємо, на яку величину можна скоротити виконання робіт: $\Delta_0 = Min\{6-5;3-1\} = 1$, імітуючи скорочення роботи (4;5) на одиницю



(див. мал. нижче) $\Delta_r = 1$. Величина скорочення $\Delta = Min\{1;2\} = 1$.

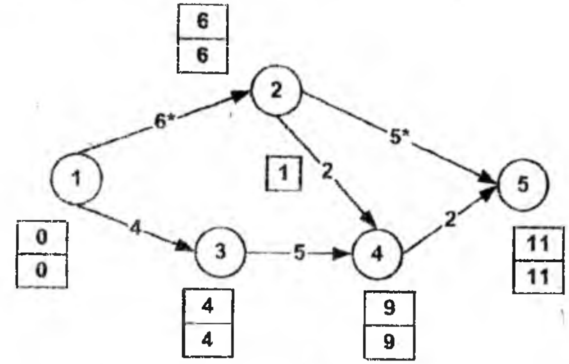
Скорочуємо роботи (2;5) та (4;5) на одиницю: $t_{25}=t_{25}-1=5^*$, $t_{45}=t_{45}-1=2$. Тривалість проекту теж зменшиться на одиницю: T=T-1=12-1=11. Вартість проекту $C = 920+1\times(60+10)=990$.



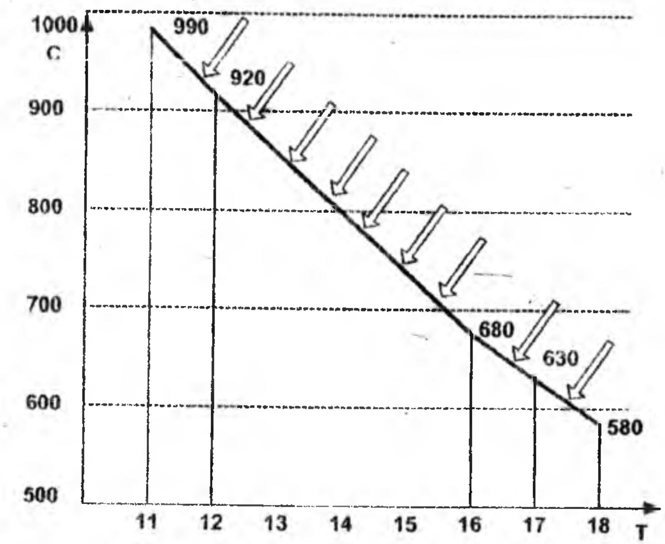
Ітерація 5.

Вартість проекту становить C=990. Множина критичних шляхів включає два критичні шляхи, $L=\{K_1,K_2\}$, $K_1=\{1\text{-}2\text{-}5\}$, $K_2=\{1\text{-}3\text{-}4\text{-}5\}$. Визначаємо на критичних шляхах роботи – кандидати на скорочення. Оскільки на критичному шляху K_1 не можна скоротити ні однієї роботи – то **стоп.** Множина Парето-оптимальних розв'язків побудована. Графічне відображення множини Парето-оптимальних розв'язків (ламана лінія) наведене нижче.

Таким чином, отриманий образ множини Парето-оптимальних розв'язків в області двох критеріїв – часу виконання проекту та його вартості. Зі зростанням часу виконання проекту вартість його для оптимальних розв'язків зменшується. Якщо збережені проміжні результати



розрахунків, що відповідають вершинам ламаної – множини Парето, то можна задати значення будь-якого з критеріїв, за ним не лише визначити значення іншого, але й відновити тривалості робіт та значення споживаного ресурсу для оптимального розв'язку.



РЕЗЮМЕ

10.1. Планування та керування на мережах – це комплекс графічних і розрахункових методів, організаційних заходів, що забезпечують моделювання, аналіз і динамічну перебудову плану виконання складних *проектів і розробок. Характерною рисою проектів є те, що вони складаються з ряду окремих, елементарних робіт, що є певним чином залежні одна від іншої. Метод СРМ дає можливість визначити, по-перше, які роботи з числа багатьох, що складають проект, є критичними за своїм впливом на загальну календарну тривалість виконання проекту, і, по-друге, яким чином побудувати найкращий календарний план реалізації всіх робіт проекту для того, щоб дотримати задані терміни при мінімальних витратах. В методі PERT вперше було введено вірогідності часу для виконання конкретних задач, визначено зв'язки між ними, проведено простий мережевий аналіз і розраховано найімовірніший час завершення проекту. Методи планування на мережах включають наступні основні етапи: структурне планування; календарне планування; оперативне керування.*

10.2. Послідовність побудови мережі включає наступні кроки: структуризація проекту до множини взаємопов'язаних робіт, виконання яких означатиме реалізацію проекту; отримання відношення безпосереднього передування (слідування) на множині робіт задачі шляхом аналізу її змістовної постановки; визначення множин початкових та завершальних робіт проекту; побудова мережі проекту з включенням у необхідних випадках фіктивних робіт.

10.3. Календарне планування на мережі здійснюється шляхом розрахунку параметрів робіт та подій в часі. Основні параметри детермінованої мережі СРМ поділяються на параметри подій мережі та параметри робіт мережі і включають ранні та пізні строки звершення подій, початку та завершення робіт і резерви часу. З точки зору керівництва проектом, при управлінні основна увага повинна бути звернута на роботи критичного шляху.

10.4. Метод PERT орієнтований на врахування невизначеностей у тривалостях виконання робіт мережі, які описуються стохастичними характеристиками. Кожна робота проекту характеризується трьома оцінками її тривалості, які отримуються зазвичай шляхом опитування експертів: найвірогіднішою тривалістю виконання; оптимістичною тривалістю: песимістичною оцінкою тривалості. Розрахунок мережі PERT здійснюється в наступній послідовності: розрахунок значень параметрів робіт мережі; визначення критичного шляху; статистичне оцінювання.

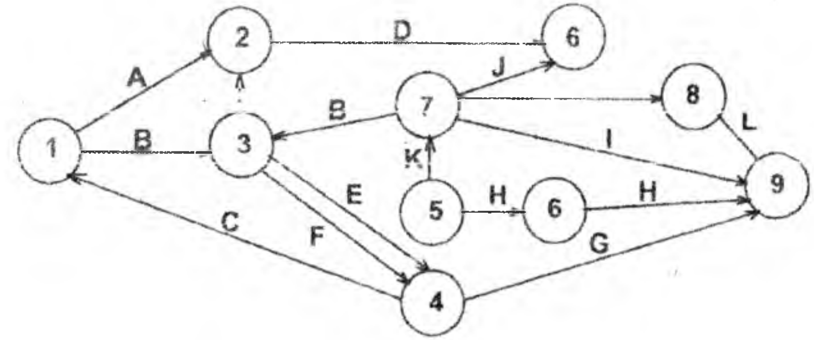
10.5. Побудована в результаті оптимізації мережі за критеріями вартості та тривалості множина Парето-оптимальних розв'язків може

бути використана для аналізу наявних можливостей розподілу ресурсу як в інтегрованому вигляді, так і з визначенням конкретних значень потреби в ресурсах та тривалості кожної з робіт мережі. У цьому випадку для однозначної ідентифікації цих характеристик для кожної роботи необхідно зберігати характеристики варіантів мережі в скінченій множині точок зламу ламаної, що відображає множину Парето-оптимальних розв'язків в просторі двох критеріїв – часу та ресурсів.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 10.1.

Знайдіть порушення правил побудови мережі на наступному рисунку



Назва	Попередні роботи	Тривалість
A	–	9
B	D	6
C	B, F, G	5
D	–	8
E	B, F, G	8
F	A, N	4
G	–	5
H	C, L	7
I	B, G	1
J	I,M	12
K	H,I,M	6
L	I,M	4
M	D	2
N	–	6

Використовуючи дані нижче у таблиці про роботи, що безпосередньо передують наведеним, перерахуйте роботи, що неправильно відображені на мережі та усуньте знайдені помилки.

Завдання 10.2.

Побудуйте мережу проекту, розрахуйте параметри подій (ранні та пізні строки звершення, резерви часу) та робіт (ранні та пізні строки початку та закінчення робіт, повні, вільні, незалежні та гарантовані резерви часу), визначте критичний шлях для заданого відношення передування та детермінованих тривалостей робіт за методом CPM.

Відношення передування:

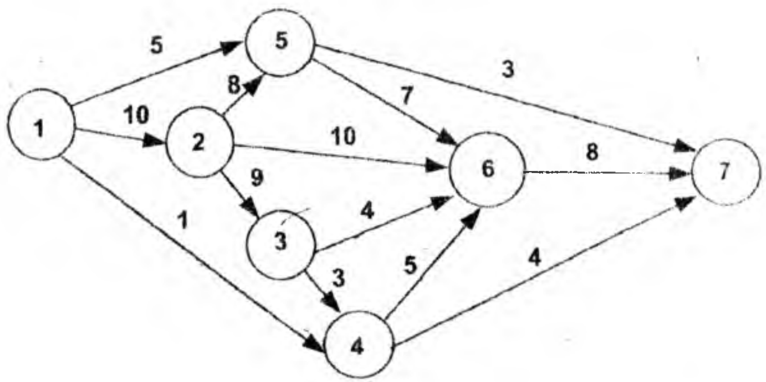
$F \prec K;\ G \prec I;\ B \prec D,G,E,F;\ D \prec J;\ A \prec G;\ C \prec E,F;\ G \prec H;\ E \prec J$.

Тривалості робіт зведені в таблицю.

A	B	C	D	E	F	G	H	I	J	K
14	8	5	16	17	9	6	11	17	12	4

Завдання 10.3.

Проаналізуйте, як вплине на хід виконання представленого проекту одночасна затримка наступних робіт: (1,5) – на 19 днів, (3,6) – на 3 дні. Аргументуйте свою відповідь.



Завдання 10.4.

Необхідно побудувати мережу проекту та визначити вірогідність того, що дійсна тривалість проекту буде на 10% меншою, ніж середнє її значення, використовуючи метод PERT. Задані відношення передування, песимістична – b, найімовірніша – m та оптимістична – a тривалості для кожної з робіт.

Відношення передування:

$B \prec F,K;\ G \prec H;\ I \prec H;\ D \prec K;\ A \prec G,F,K;\ C \prec E,D;\ F \prec I,J;\ E \prec K$.

Тривалості робіт зведені в таблицю:

Трив.	A	B	C	D	E	F	G	H	I	J	K
a	4	6	3	8	4	1	6	3	4	2	7
b	8	12	9	14	10	4	10	5	9	8	7
m	6	8	4	10	6	2	8	4	6	4	7

Завдання 10.5.

Необхідно побудувати множину Парето-оптимальних розв'язків для задачі оптимізації проекту за критеріями тривалості та вартості. Для заданої вартості виконання проекту C=60 визначити оптимальні тривалості всіх робіт.

Відношення передування: $B \prec C,E,F;\ F \prec G;\ A \prec D;\ C \prec D$.

	A	B	C	D	E	F	G
Мінімальна тривалість	4	3	3	6	2	1	3
Вартість при мінімальній тривалості	16	17	9	22	15	21	10
Максимальна тривалість	10	10	4	10	6	8	6
Вартість при максимальній тривалості	4	10	4	14	3	7	4

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Назвіть особливості планування та керування на мережах.
2. Розкрийте зміст основних етапів методів планування на мережах.
3. Що може здійснити керівник проекту за допомогою CPM та PERT?
4. Що потрібно для того, щоб мати можливість застосувати методи CPM та PERT?
5. Яку структуру має мережа проекту?
6. Згідно з якими правилами повинна будуватись мережа проекту?
7. Якою повинна бути послідовність побудови мережі?
8. Перерахуйте основні параметри подій мережі CPM та поясніть, яким чином вони обчислюються.
9. Назвіть основні параметри робіт мережі PERT та поясніть, яким чином вони обчислюються.
10. Якою є послідовність розрахунку параметрів мережі CPM?
11. Виходячи з яких міркувань обчислюються часові характеристики

робіт в методі PERT?

12. В якій послідовності здійснюється розрахунок мережі PERT?
13. Розкрийте проблематику оптимізації мережі.
14. Які основні кроки алгоритму оптимізації мережі?

РОЗДІЛ 5

ІГРОВІ ЗАДАЧІ ДОСЛІДЖЕННЯ ОПЕРАЦІЙ

ТЕМА 11. ІГРИ ДВОХ ОСІБ З НУЛЬОВОЮ СУМОЮ

Теорія ігор - це розділ дослідження операцій, що займається теорією математичних моделей прийняття оптимальних рішень в умовах конфлікту. Адекватна математична модель соціально-економічного явища повинна відображати властиві йому риси конфлікту: відмінність інтересів сторін – учасників конфлікту, а також різноманітність відповідних дій, які ці сторони можуть здійснювати для досягнення своїх цілей. Гра характеризується за допомогою системи правил, які описують сутність конфліктної ситуації: кількість гравців; вибір способу дій гравців на кожному з етапів гри; інформацію, якою володіє кожен з гравців при здійсненні таких виборів; виплату для кожного з гравців після завершення довільного етапу гри. Основними задачами теорії ігор є: синтез принципів оптимальності; встановлення факту існування оптимальних у цьому сенсі ситуацій; знаходження їх реалізацій. Найпростішими та детально дослідженими є антагоністичні ігри двох осіб, які й розглядаються нижче. Ці ігри завжди можуть бути приведені до пари двоїстих задач лінійного програмування та розв'язані у змішаних стратегіях. Джон фон Нойман довів справедливість основної теореми про мінімакс для загальних умов і, крім того, створив багату ідеями теорію ігор з числом гравців більше двох. Наступному швидкому розвитку теорії ігор значною мірою сприяла Друга світова війна. Під час війни було розгорнуто широку діяльність у напрямку наукового чи, щонайменше, систематичного підходу до таких задач, які раніше знаходилися виключно в компетенції "практиків", а саме: організація тилу, пошук підводних човнів, протиповітряна оборона і т.і. Теорія ігор є одним із найскладніших теоретичних обґрунтувань з тих, що з'явилися у цій галузі.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати:⇨основні завдання та поняття теорії ігор, класифікацію ігор; ⇨теоретичні результати теорії ігор двох осіб з нульовою сумою;⇨загальний алгоритм розв'язання антагоністичної гри двох осіб з нульовою сумою;

вміти: ⇨спрощувати та геометрично розв'язувати ігри двох осіб з нульовою сумою розміру 2*n та m*2; ⇨знаходити розв'язки гри двох осіб з нульовою сумою шляхом розв'язування відповідної задачі лінійного програмування.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ гра двох осіб з нульовою сумою
- ☑ рівновага в грі
- ☑ мішана стратегія
- ☑ конфліктна ситуація
- ☑ антагоністична гра
- ☑ середній виграш
- ☑ правила гри
- ☑ матрична гра
- ☑ матриця виплат
- ☑ стратегія гри
- ☑ спрощення гри
- ☑ виграш
- ☑ нижня ціна гри
- ☑ коаліційна гра
- ☑ оптимальні стратегії
- ☑ сідлова точка
- ☑ кооперативна гра
- ☑ графічне розв'язування гри
- ☑ чиста стратегія
- ☑ домінована стратегія
- ☑ верхня ціна гри

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

11.1. Основні поняття теорії ігор.

11.2. Класифікація ігор.

11.3. Матричні ігри двох осіб з нульовою сумою. Матриця гри. Верхня та нижня ціна гри. Теорема про мінімакс.

11.4. Мішані стратегії в іграх двох осіб з нульовою сумою.

11.5. Представлення гри у вигляді задач лінійного програмування.

11.6. Ігри порядку 2×2, $2\times n$ та $m\times 2$. Графічне розв'язування ігор.

11.1. ОСНОВНІ ПОНЯТТЯ ТЕОРІЇ ІГОР

Теорія ігор – це розділ дослідження операцій, що займається теорією математичних моделей прийняття оптимальних рішень в умовах конфлікту. При цьому математичні моделі, що застосовуються, є достатньо спрощеними та ідеалізованими схемами реальних явищ.

Таким чином, **теорія ігор досліджує питання поведінки і виробляє оптимальні правила (стратегії) поведінки для кожного з учасників конфліктної ситуації.** Розв'язання суперечностей за допомогою теорії ігор можливе лише після проведення математичного моделювання ситуацій у вигляді гри.

Сучасний математичний підхід до різних інтересів – теорії ігор, звичайно, приписують фон Нойману, що виклав його у своїх статтях 1928 і 1937 рр., хоча в 1953 році Фреше вказав, що основи теорії ігор було намічено в деяких статтях Бореля на початку 20-х років. Хоча Борель дав ясне формулювання важливого класу теоретико-ігрових задач і ввів поняття чистих та мішаних стратегій, він, як зауважив фон Нойман, не отримав основного висновку – теореми про мінімакс, без якої не може бути жодної теорії ігор.

Фон Нойман довів справедливість цієї теореми для загальних умов і, крім того, створив багату ідеями теорію ігор з числом гравців більше двох. На жаль, жодна з цих двох груп робіт не привернули до себе великої уваги при публікації, очевидно тому, що статті були написані для математиків, а не для соціологів. Справжній інтерес до теорії ігор пробудила робота фон Ноймана і Моргенштерна "Теорія ігор та економічна поведінка", що вийшла в світ у 1944 році.

Наступному швидкому розвитку теорії ігор значною мірою сприяла Друга світова війна. Під час війни було розгорнуто широку діяльність у напрямку наукового чи, щонайменше, систематичного підходу до таких задач, які раніше знаходилися виключно в компетенції "практиків". Маються на увазі такі питання, як: організація тилу, пошук підводних човнів, противоповітряна оборона і т.і. Теорія ігор є одним із найскладніших теоретичних обґрунтувань з тих, що з'явилися у цій галузі. Своїм подальшим розвитком теорія ігор завдячує таким відомим вченим, як Р. Данкан Льюс, Ховард Райфа, Ерве Мулен, Н. Н. Воробйов та ін.

Слід зазначити, що теорія ігор є, насамперед, математичною дисципліною. В основному це пояснюється тим, що початок їй поклав математик і викладена вона була як достатньо формальна побудова, що зробило її доступною в якості знаряддя дослідження лише для математиків. Тим не менше, в наш час теорія ігор застосовується в найрізноманітніших галузях науки і суспільного життя, її результатами користуються політики, економісти, військові, букмекери.

Одна з характерних рис будь-якого суспільного, соціально-економічного явища полягає у множинності, різнобічності інтересів, в наявності сторін, що мають відмінні інтереси та нетотожні цілі, або хоча б у наявності кількох різних активних точок зору стосовно явища та його результату. **У цьому сенсі можна сказати, що будь-якому соціально-економічному явищу властиві риси конфлікту.**

Слід зауважити, що кожна зацікавлена сторона повинна мати різні можливості діяти, задовольняти свої інтереси (вперше це твердження було сформульоване Вільямом Ешбі і отримало назву "закону про необхідну різноманітність"). В іншому випадку, коли сторона має лише одну таку можливість, вона перестає відігравати роль сторони у процесі, що розглядається, і перетворюється на обставину, яка однозначно впливає на цей процес.

Отже, адекватна математична модель соціально-економічного явища повинна відображати властиві йому риси конфлікту: відмінність інтересів сторін – учасників конфлікту, а також різноманітність відповідних дій, які ці сторони можуть здійснювати для досягнення своїх цілей. Це означає, що соціально-економічне явище при його математичному моделюванні повинно поряд з іншими можливими представленнями припускати ще й представлення у вигляді конфлікту, тобто таке, в якому відображені наступні його компоненти: зацікавлені сторони; можливі дії кожної зі сторін; інтереси сторін.

Розглянемо основні визначення теорії ігор.

Ситуація називається конфліктною, якщо в ній беруть участь сторони, інтереси яких повністю або частково протилежні.

Гра – це конфлікт, в якому наявні щонайменше 2 учасники (гравці), кожний з яких прагне досягнення власних цілей. Будь-яка гра складається з партій, які починаються і закінчуються, після чого гравцям виплачуються їх виграші. Своєю чергою, кожна партія складається з ходів, які одночасно або послідовно роблять гравці. Опис гри як послідовності ходів носить назву позиційної форми гри.

Правила гри – це припустимі дії кожного з гравців, спрямовані на досягнення певної мети.

Кількісна оцінка результатів гри називається **виплатою**.

Описання вибору гравця в кожній з можливих ситуацій, при яких він повинен зробити хід, називається **стратегією гри**.

Стратегія гри називається оптимальною, якщо при багаторазовому повторенні гри вона забезпечує гравцеві максимально можливий середній виграш.

Таким чином, гра характеризується за допомогою системи правил, які описують сутність конфліктної ситуації:

☑ кількість гравців;

☑ вибір способу дій гравців на кожному з етапів гри;

☑ інформацію, якою володіє кожен з гравців при здійсненні таких виборів;

☑ виплату для кожного з гравців після завершення довільного етапу гри.

Виграш у грі є засобом ефекту для гравця, в теорії ігор виграш оцінюється кількісно. Виграш гравця залежить як від його стратегії, так і від стратегії інших осіб. В іграх з нульовою сумою програш дорівнює сумі виграшів, тому гра є антагоністичною.

Завдання дослідника конфліктної ситуації полягає в приведенні її з мінімальними втратами до формальної гри.

Основними принципами, що використовуються при пошуку розв'язків ігор, є принципи оптимальності та рівноваги.

Оптимальність.

Дослідження конфліктів, а у відповідності до цього – ігор, можна проводити з різних точок зору:

☑ дескриптивної, яка полягає у визначенні того, які ситуації фактично складаються (або можуть складатися) в тих чи інших конфліктах;

☑ нормативної, що визначає, яку поведінку гравців слід вважати оптимальною (розумною, адекватною);

☑ конструктивної, яка вказує, як реалізовувати потрібні (наприклад, оптимальні) стратегії або ситуації;

☑ прогностичної, що займається передбаченням фактичного результату конфлікту.

Теорія ігор як математична дисципліна в її сучасному стані займається нормативним вивченням ігор.

Основними задачами теорії ігор можна вважати наступні:

☑ синтез принципів оптимальності;

☑ встановлення можливостей реалізації принципів оптимальності (тобто встановлення факту існування оптимальних у цьому сенсі ситуацій);

☑ знаходження їх реалізацій.

Основними змістовними рисами конфлікту стосовно результатів або множини результатів конфлікту вважаються інтуїтивні уявлення про ***вигідність, стійкість*** та ***справедливість***.

У найпростішому випадку, коли у грі бере участь лише єдиний гравець, вигідність можна розуміти єдиним чином, а саме: як максимізацію значень функції виграшу на всій множині стратегій-ситуацій. Такі задачі, по суті, є задачами оптимізації і в теорії ігор не розглядаються.

Будь-яка можлива для гравця $i \in I$ дія – це його стратегія; множину всіх стратегій гравця *i* позначимо через X_i. В умовах конфлікту кожний гравець *i* обирає свою деяку стратегію x_i з множини X_i, в результаті чого складається набір стратегій $x = (x_1, ..., x_n)$, який називається *ситуацією*. Множина всіх ситуацій є, очевидно, декартовим добутком $\prod_{i \in I} X_i$ і позначається через *X*.

Зацікавленість гравців у ситуаціях виявляється в тому, що кожному гравцю $i \in I$ в кожній ситуації $x \in X$ приписується число, що відображає ступінь задоволення його інтересів у цій ситуації. Це число називатимемо ***виграшем*** гравця *i* в ситуації *x* і позначатимемо через $H_i(x)$. Відображення $H_i : x \to R$ називається *функцією виграшу (функцією корисності) гравця i*. В цих умовах перебіг конфлікту полягає у виборі кожним гравцем $i \in I$ його стратегії $x_i \in X_i$ та в отриманні ним у новій ситуації $x = (x_1, ..., x_n)$ виграшу $H_i(x)$ з деякого джерела.

Таким чином, будь-який конфлікт може бути представлений у вигляді системи $\left\langle I, \{X_i\}_{i \in I}, \{H_i\}_{i \in I} \right\rangle$. (11.1)

Така система називається ***безкоаліційною грою*** або просто ***грою***.

Серед усіх безкоаліційних ігор виділяється клас антагоністичних ігор, в яких число гравців дорівнює двом, а значення їх функцій виграшу в кожній ситуації рівні за величиною і протилежні за знаком: $H_1(x_1, x_2) = -H_2(x_1, x_2)$. (11.2)

Для скорочення вживання індексів, множини стратегій гравців 1 і 2 в антагоністичній грі, звичайно, позначимо через *A* та *B*, функція виграшу H_1 через *H*, а сама гра записується у вигляді $\langle A, B, H \rangle$. (11.3)

Нехай гра нетривіальна, тобто в ній бере участь декілька гравців. В цьому випадку змістовні уявлення про вигідність і стійкість, не кажучи вже про справедливість, можуть бути формалізовані по-різному.

Можна, наприклад, оптимальною ситуацією вважати таку, за якої одночасно досягають своїх максимумів функції виграшу кожного з гравців.

Умову оптимальності в цьому сенсі для ситуації x* у грі Г формально можна записати так:

$\forall i \in I \wedge x \in X: \quad H_i(x) \le H_i(x^*)$. (11.4)

Вигідність такої ситуації очевидна. Так само, як і її стійкість: будь-яке відхилення від неї гравців або групи гравців може призвести хіба що до зменшення виграшів усіх учасників гри (в тому числі – тих, що відхилилися). Справедливість цієї ситуації випливає з симетричності входження всіх гравців у вищенаведену умову. Однак існування таких ситуацій є винятком.

Рівновага.

Однією з плідних форм реалізації уявлень про оптимальність можна вважати поняття *рівноваги*, яке полягає в наступному. Ситуація називається *рівноважною*, якщо жоден з гравців не зацікавлений у відхиленні від неї. Формально це записується наступним чином. Нехай $\Gamma = \left\langle I, \{X_i\}_{i \in I}, \{H_i\}_{i \in I} \right\rangle$ - безкоаліційна гра, а $x = (x_1, ..., x_n)$ - деяка ситуація в ній. Якщо x_i' - довільна стратегія гравця *i*, то введемо наступне позначення $x \| x_i' = (x_1, ..., x_{i-1}, x_i', x_{i+1}, ..., x_n)$.

Таким чином, $x \| x_i'$ є результатом заміни в ситуації *x* стратегії x_i гравця *i* на його стратегію x_i'. Ситуація *x** називається ***рівноважною*** (або ***ситуацією рівноваги***), якщо $\forall i \in I \wedge x_i \in X_i: \quad H_i(x^* \| x_i) \le H_i(x^*)$. (11.5)

11.2. КЛАСИФІКАЦІЯ ІГОР

Класифікація ігор реалізується за певною множиною класифікаційних ознак, а саме: кількість гравців, кількість стратегій, характер взаємин між гравцями, характер виграшів, вигляд функції виграшів, момент вибору ходу, кількість ходів, стан інформації.

Кількість гравців. 0 – це описова модель ситуації. 1 – модель з необхідністю визначення найбільш доцільного способу дій за умови відсутності активної протидії. 2 – ці моделі належать власне до найбільш досліджених моделей теорії ігор. 3 та більше - досліджені лише спеціальні випадки внаслідок принципових труднощів визначення поняття положення рівноваги в такій грі. Зі зростанням кількості гравців труднощі розв'язання таких ігор зростають.

Кількість стратегій: скінчені та нескінчені. Якщо в грі кожен з гравців має скінчену множину можливих стратегій, то гра є скінченою, якщо ж хоча б один з гравців має безмежну множину стратегій, то гра є нескінченою. Зі зростанням кількості стратегій зростає складність розв'язання ігор.

Характер взаємин. За цією ознакою ігри поділяються на безкоаліційні, коаліційні та кооперативні. Безкоаліційними є ігри, в яких заборонені угоди між гравцями та утворення коаліцій (наприклад, чемпіонат світу з футболу). Коаліційними називаються ігри, в яких гравці можуть утворювати коаліції (військові ігри, економічні ситуації, пов'язані з оволодінням певними ринками збуту). Кооперативними є ігри, в яких коаліції відомі наперед та залишаються незмінними протягом гри.

Характер виграшів. За цією ознакою ігри поділяються на ігри з нульовою сумою та ігри з ненульовою сумою. В грі з нульовою сумою сума виграшів всіх гравців в кожній партії рівна нулю, тобто в цій грі загальний капітал всіх гравців не змінюється, а перерозподіляється між гравцями в залежності від результатів гри.

Гра двох гравців з нульовою сумою називається **антагоністичною,** оскільки цілі гравців в ній прямо протилежні: виграш одного гравця досягається за рахунок програшу іншого.

Прикладом гри з ненульовою сумою є торговельні взаємовідносини між країнами – в результаті застосування своїх стратегій всі країни можуть бути в виграші. Будь-яка гра, в якій необхідно виплачувати вступний внесок за право участі в ній, є грою з ненульовою сумою; у виграші завжди особа, що отримала внесок. В лотереї організатор завжди має виграш, а учасники гри отримують сумарний виграш менший, ніж внесли.

Вигляд функції виграшів. За цією ознакою ігри поділяються на матричні, біматричні, неперервні, опуклі, сепарабельні, типу дуелей та ін. Матрична гра – це скінчена гра двох осіб, в якій виграші першого гравця задаються елементами матриці і дорівнюють програшам другого гравця. Матричні ігри розв'язуються за допомогою методів лінійного програмування. В біматричних іграх виграш кожного з гравців задається окремою матрицею, і ці ігри є складнішими для розв'язування. Якщо функція виграшу в грі може бути представлена у вигляді суми функцій одного аргумента, то така гра є сепарабельною (може бути розділеною).

Дуель – це гра, що характеризується моментом вибору ходу та ймовірностями отримання виграшу в залежності від часу, що пройшов від моменту початку гри до моменту вибору. Приклад: кожна фірма вкладає капітал в певний момент часу; чим раніше буде здійснене вкладення, тим менша вірогідність оволодіти ринком, однак і при занадто пізньому вкладенні ринок збуту буде втрачено.

Кількість ходів. За цією ознакою ігри поділяються на однокрокові та багатокрокові. Однокрокові ігри завершуються після того, як кожен з гравців зробить по одному крокові. Матрична гра є однокроковою.

Багатокрокові ігри своєю чергою поділяються на **позиційні, стохастичні та диференційні.** Позиційні ігри полягають у тому, що кожен з гравців робить декілька ходів послідовно в часі, і виграші визначаються в залежності від результату гри. Якщо ж у грі робляться ходи, що приводять до вибору певних позицій, причому існує певна вірогідність повертання на попередню позицію, то така гра називається стохастичною. Якщо ходи робляться неперервно і умови їх проведення описуються диференційними рівняннями (гра типу "хижак-жертва"), то така гра називається диференційною.

Стан інформації. За цією ознакою розглядаються ігри з повною та неповною інформацією. Якщо на кожному ході гри кожному з гравців відомо, які вибори були зроблені гравцями раніше, то гра є з повною інформацією (шахи, шашки). Якщо ж у грі не все відомо про попередні вибори, то гра буде з неповною інформацією.

11.3. МАТРИЧНІ ІГРИ ДВОХ ОСІБ З НУЛЬОВОЮ СУМОЮ. МАТРИЦЯ ГРИ. ВЕРХНЯ ТА НИЖНЯ ЦІНА ГРИ. ТЕОРЕМА ПРО МІНІМАКС

Матрична гра двох осіб з нульовою сумою визначається наступним чином. Перший гравець (гравець A) має в своєму розпорядженні m стратегій A_i, $i=\overline{1,m}$, а другий (гравець B) – n стратегій B_j, $j=\overline{1,n}$.

$$\begin{array}{ccccc} & B_1 & B_2 & \dots & B_n \\ A_1 & a_{11} & a_{12} & & a_{1n} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ A_m & a_{m1} & a_{m2} & \dots & a_{mn} \end{array} \qquad (11.6)$$

Кожній парі стратегій ставиться у відповідність число a_{ij}, що є виграшем гравця A за умови застосування ним своєї стратегії A_i та гравцем B своєї стратегії B_j, і одночасно програшем гравця B. Кожна з стратегій A_i та B_j називається чистою стратегією. Матрична гра є антагоністичною.

Гра полягає в наступному. Гравець A та гравець B обирають кожен одну із своїх стратегій A_i та B_j (тобто одночасно роблять хід). Після цього гра закінчується і гравець A отримує виграш a_{ij}, а гравець B, відповідно, програє a_{ij} (тобто виграє $-a_{ij}$).

Наступним кроком після описання гри є визначення оптимальних стратегій та виграшів гравців. Для вибору оптимальної стратегії використовується принцип гарантованого результату (принцип "максиміну"), тобто обирається така стратегія, яка за найгірших умов забезпечить максимальний виграш. При цьому вважається, що суперник буде робити найкращі свої ходи, тобто оптимальним чином (зі своєї точки зору) протидіяти. Надалі ми вважатимемо, що всі значення матриці виграшів є невід'ємними – це припущення не зменшує загальності, оскільки будь-яку матричну гру можна привести до цього вигляду, не змінивши положення оптимуму (зміниться лише ціна гри).

Число α, визначене як $\alpha = \underset{i}{Max}\ \underset{j}{Min}\ a_{ij}$, (11.7)

називається нижньою ціною гри і показує, який виграш може гарантувати собі гравець А за умови застосування своїх чистих стратегій та всіх можливих дій гравця В.

Гравець В, своєю чергою, прагне максимально зменшити значення свого програшу шляхом вибору своїх чистих стратегій. Число β, визначене як

$\beta = \underset{j}{Min}\ \underset{i}{Max}\ a_{ij}$, (11.8)

є верхньою ціною гри і показує, який мінімальний програш може гарантувати собі гравець В.

Якщо в грі з матрицею виплат А нижня та верхня чисті ціни гри рівні між собою, то гра має сідлову точку в чистих стратегіях і чисту ціну гри $v = \alpha = \beta$. Поняття сідлової точки відображає наступну ситуацію: якщо один із гравців дотримується стратегії, що відповідає сідловій точці, то інший гравець не може діяти краще, як дотримуватися своєї стратегії, яка також відповідає сідловій точці.

Таким чином, для сідлової точки справедливе співвідношення:

$a_{ij_0} \le a_{i_0 j_0} \le a_{i_0 j}$, (11.9)

де A_i та B_j – довільні чисті стратегії гравців А та В відповідно, A_{i0}, B_{j0} – стратегії, що утворюють сідлову точку.

Пара чистих стратегій, A_{i0}, B_{j0}, що утворюють сідлову точку, та сідловий елемент матриці $a_{i_0 j_0}$ називаються розв'язком гри в чистих стратегіях. Основні теоретичні результати даються наступними теоремами про сідлову точку.

Теорема 11.1. Теорема про верхню та нижню ціну гри.

Нехай $f(x, y)$ – дійсна функція двох змінних $x \in A$ та $y \in B$ і існують $\alpha = \underset{x\in A}{Max}\ \underset{y\in B}{Min} f(x,y)$, $\beta = \underset{y\in B}{Min}\ \underset{x\in A}{Max} f(x,y)$. У цьому випадку $\alpha \le \beta$.

Доведення.

За визначенням максимуму та мінімуму

$\underset{y\in B}{Min} f(x,y) \le f(x,y) \le \underset{x\in A}{Max} f(x,y)$, тобто

$\underset{y\in B}{Min} f(x,y) \le \underset{x\in A}{Max} f(x,y)$.

Оскільки в лівій частині останньої нерівності значення x довільне, то $\underset{x\in A}{Max}\ \underset{y\in B}{Min}\ f(x,y) \le \underset{x\in A}{Max} f(x,y)$. Аналогічно, в правій частині цієї нерівності значення y довільне, і тому

$\underset{x\in A}{Max}\ \underset{y\in B}{Min}\ f(x,y) \le \underset{y\in B}{Min}\ \underset{x\in A}{Max} f(x,y)$, тобто $\alpha \le \beta$ ■

Матриця антагоністичної гри двох осіб $A = \{a_{ij}\}$ є частковим випадком $f(x,y)$, де $x = i$, $y = j$, $f(x,y) = a_{ij}$.

Визначення 11.1. Нехай $f(x,y)$ – дійсна функція двох змінних $x \in A$, $y \in B$. Точка (x^*, y^*) називається сідловою для функції $f(x,y)$, якщо для довільних $x \in A$, $y \in B$ виконується нерівність

$f(x, y^*) \le f(x^*, y^*) \le f(x^*, y)$. (11.10)

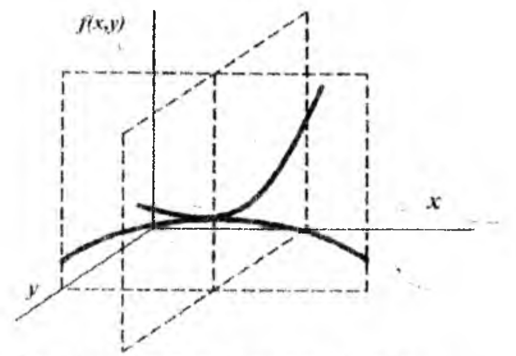


Рис 11.1. Ілюстрація сідлової точки

Теорема 11.2. Теорема про сідлову точку.

Нехай для дійсної функції $f(x,y)$, $x \in A$, $y \in B$ існують $\alpha = \underset{x\in A}{Max}\ \underset{y\in B}{Min} f(x,y)$ та $\beta = \underset{y\in B}{Min}\ \underset{x\in A}{Max} f(x,y)$. Якщо $\alpha = \beta$, то необхідною і достатньою умовою цього є існування сідлової точки (x^*, y^*) функції $f(x,y)$, тобто такої, що

$f(x, y^*) \le f(x^*, y^*) \le f(x^*, y)$.

Якщо (x^*, y^*) – сідлова точка функції $f(x,y)$, то

$f(x^*, y^*) = \underset{x\in A}{Max}\ \underset{y\in B}{Min}\ f(x,y) = \underset{y\in B}{Min}\ \underset{x\in A}{Max}\ f(x,y)$.

Доведення.

Достатність.

Нехай існує сідлова точка (x^*, y^*), тоді за визначенням справедливе співвідношення $f(x, y^*) \le f(x^*, y^*) \le f(x^*, y)$, тобто для довільних $x \in A$, $y \in B$ справедливі нерівності

$\underset{x\in A}{Max}\ f(x, y^*) \le f(x^*, y^*) \le \underset{y\in B}{Min}\ f(x^*, y)$.

За визначенням максимуму та мінімуму функції

$\underset{y\in B}{Min}\ \underset{x\in A}{Max}\ f(x,y) \le \underset{x\in A}{Max}\ f(x, y^*)$ та

$\underset{y\in B}{Min}\ f(x^*, y) \le \underset{x\in A}{Max}\ \underset{y\in B}{Min}\ f(x,y)$.

Порівнюючи наведені нерівності, отримуємо:

$\underset{y\in B}{Min}\ \underset{x\in A}{Max}\ f(x,y) \le f(x^*, y^*) \le \underset{x\in A}{Max}\ \underset{y\in B}{Min}\ f(x,y)$.

З іншого боку, за попередньою теоремою справедливе співвідношення $\underset{y\in B}{Min}\ \underset{x\in A}{Max}\ f(x,y) \ge f(x^*, y^*) \ge \underset{x\in A}{Max}\ \underset{y\in B}{Min}\ f(x,y)$, що сумісно можливо лише у випадку, коли $\underset{x\in A}{Max}\ \underset{y\in B}{Min}\ f(x,y) = \underset{y\in B}{Min}\ \underset{x\in A}{Max}\ f(x,y)$. Таким чином, достатність доведена.

Необхідність.

Нехай справедлива рівність $\underset{x\in A}{Max}\ \underset{y\in B}{Min}\ f(x,y) = \underset{y\in B}{Min}\ \underset{x\in A}{Max}\ f(x,y)$. У цьому випадку існують такі $x^* \in A$, $y^* \in B$, що

$\underset{x\in A}{Max}\left(\underset{y\in B}{Min}\ f(x,y)\right) = \underset{y\in B}{Min}\ f(x^*, y)$,

$\underset{y\in B}{Min}\left(\underset{x\in A}{Max}\ f(x,y)\right) = \underset{x\in A}{Max}\ f(x, y^*)$.

Покажемо, що (x^*, y^*) — сідлова точка.

Зі співвідношення $\underset{x\in A}{Max}\ \underset{y\in B}{Min}\ f(x,y) = \underset{y\in B}{Min}\ \underset{x\in A}{Max}\ f(x,y)$ наслідком є те, що $\underset{y\in B}{Min}\ f(x^*, y) = \underset{x\in A}{Max}\ f(x, y^*)$.

За визначенням мінімуму $\underset{y\in B}{Min}\ f(x^*, y) \le f(x^*, y^*)$, і враховуючи вищенаведене отримаємо $\underset{x\in A}{Max}\ f(x, y^*) \le f(x^*, y^*)$. За визначенням максимуму з останнього отримаємо $f(x, y^*) \le f(x^*, y^*)$. Аналогічно доводиться, що $f(x^*, y^*) \le f(x^*, y)$. З того, що

$\underset{y\in B}{Min}\ f(x^*, y) = \underset{x\in A}{Max}\ f(x, y^*)$ випливає справедливість рівності

$$f(x^*, y^*) = \underset{x\in A}{Max}\ \underset{y\in B}{Min}\ f(x,y) = \underset{y\in B}{Min}\ \underset{x\in A}{Max}\ f(x,y) \quad ■ \tag{11.11}$$

Якщо покласти $x = i,\ y = j,\ f(x,y) = a_{ij}$, то цим доведено справедливість співвідношення $a_{i j_0} \le a_{i_0 j_0} \le a_{i_0 j}$ для визначення сідлової точки антагоністичної матричної гри.

11.4. МІШАНІ СТРАТЕГІЇ В ІГРАХ ДВОХ ОСІБ З НУЛЬОВОЮ СУМОЮ

Не кожна матрична гра має розв'язок в чистих стратегіях. Дослідження гри починається зі знаходження сідлової точки. Якщо сідлова точка існує, то розв'язок гри знайдений – **стоп.** Якщо ж ні, то можна визначити верхню та нижню ціну гри, однак така інформація недостатня.

Покращення розв'язку слід шукати у використанні секретності застосування чистих стратегій за умови багаторазового повторення гри в вигляді партій. В середньому будуть досягнуті деякі виграші, які будуть більші, ніж нижня ціна гри, та менші, ніж верхня. Чим більше це середнє значення, тим краще застосовує стратегії гравець А. Тому виникла ідея застосовувати стратегії випадковим чином з певною вірогідністю, що повністю забезпечує секретність їх застосування. Кожен гравець може змінювати вірогідності застосування своїх чистих стратегій, щоб забезпечити собі максимальний середній виграш.

Визначення 11.2. Мішаною стратегією гравця називається повний набір вірогідностей застосування ним його чистих стратегій. Таким чином, якщо гравець А застосовує m чистих стратегій, то мішана стратегія

$$x = (x_1, \dots, x_m) \text{ задовільняє умови: } x_i \ge 0,\ \sum_{i=1}^{m} x_i = 1,\ \overline{i = 1, m}. \tag{11.12}$$

Аналогічно мішана стратегія гравця В

$$y = (y_1, \dots, y_n):\ y_j \ge 0,\ \sum_{j=1}^{n} y_j = 1,\ \overline{j = 1, n}. \tag{11.13}$$

Визначення 11.3. Середній виграш гравця А в матричній грі відображається як математичне сподівання його виграшу при застосуванні ним стратегії $x = (x_1, \dots, x_m)$, а гравцем В — стратегії $y = (y_1, \dots, y_n)$, і аналогічно для гравця В:

$$M[A, x, y] = \sum_{j=1}^{n}\left(\sum_{i=1}^{m} a_{ij} x_i\right) y_j = \sum_{i=1}^{m}\left(\sum_{j=1}^{n} a_{ij} y_j\right) x_i. \tag{11.14}$$

У цьому випадку верхня ціна гри

$$\beta = \underset{y}{Min}\ \underset{x}{Max}\ M[A,x,y], \qquad (11.15)$$

а нижня $\alpha = \underset{x}{Max}\ \underset{y}{Min}\ M[A,x,y]$. (11.16)

За аналогією з іграми, що мають сідлові точки в чистих стратегіях, введемо поняття оптимальних мішаних стратегій.

Визначення 11.4. Оптимальними мішаними стратегіями гравців А та В відповідно називаються такі стратегії $x^* = (x_1^*,\dots,x_m^*)$ та $y^* = (y_1^*,\dots,y_n^*)$, для яких виконується рівність

$$\underset{y}{Min}\ \underset{x}{Max}\ M[A,x,y] = \underset{x}{Max}\ \underset{y}{Min}\ M[A,x,y] = M[A,x^*,y^*]. \qquad (11.17)$$

Еквівалентним визначенням є наступне. Мішані стратегії $x^* = (x_1^*,\dots,x_m^*)$ та $y^* = (y_1^*,\dots,y_n^*)$ є оптимальними, якщо вони утворюють сідлову точку функції $M[A,x,y]$, тобто

$$M[A,x,y^*] \le M[A,x^*,y^*] \le M[A,x^*,y].$$

Визначення 11.5. Величина $M[A,x^*,y^*]$ називається ціною гри та позначається v.

Основна теорема теорії матричних антагоністичних ігор двох осіб вперше була доведена Дж. фон Нойманом і відповідає на запитання : Чи мають матричні ігри розв'язки і які?

Теорема 11.3. Основна теорема теорії матричних ігор.

Для матричної гри з довільною матрицею А величини $\alpha = \underset{x}{Max}\ \underset{y}{Min}\ M[A,x,y]$ та $\beta = \underset{y}{Min}\ \underset{x}{Max}\ M[A,x,y]$ існують завжди та є рівними між собою. Таким чином, для матричної гри завжди існує розв'язок в мішаних стратегіях.

Наступні теореми та наслідки з них дають відповідь на запитання: Яким чином знайти розв'язки матричних ігор?

Теорема 11.4. Для того, щоб у матричній грі з ціною v мішана стратегія гравця А x^* була оптимальною, необхідно та достатньо, щоб для довільної стратегії гравця В виконувалась нерівність $v \le M[A,x^*,y]$. (11.18)

Аналогічно для гравця В: для того, щоб у матричній грі з ціною v мішана стратегія гравця В y^* була оптимальною, необхідно та достатньо, щоб для довільної стратегії гравця В виконувалась нерівність $M[A,x,y^*] \le v$. (11.19)

Наслідок. Для того, щоб $x^* = (x_1^*,\dots,x_m^*)$, була оптимальною мішаною стратегією матричної гри з матрицею A та ціною гри v, необхідно та достатньо, щоб виконувалися наступні нерівності:

$$\sum_{i=1}^{m} a_{ij}x_i^* \ge v,\ j = \overline{1,n}. \qquad (11.20)$$

Аналогічно для гравця В: Для того, щоб $y^* = (y_1^*,\dots,y_m^*)$ була оптимальною мішаною стратегією матричної гри з матрицею A та ціною гри v, необхідно та достатньо, щоб виконувалися наступні нерівності:

$$\sum_{j=1}^{n} a_{ij}y_j^* \le v,\ i = \overline{1,m}. \qquad (11.21)$$

Таким чином, для розв'язування гри необхідно визначити стратегії, що задовільняють вищенаведені системи обмежень та умови нормування

$$x_i^* \ge 0,\ \sum_{i=1}^{m} x_i^* = 1,\ i = \overline{1,m},\ y_j^* \ge 0,\ \sum_{j=1}^{n} y_j^* = 1,\ j = \overline{1,n} \qquad (11.22)$$

Цей наслідок дозволяє сформулювати для розв'язування гри пару задач лінійного програмування.

Таким чином, розв'язання матричної гри зводиться до знаходження невід'ємних розв'язків системи лінійних нерівностей та рівнянь (тобто, може бути приведена до задачі лінійного програмування).

Теорема 11.5. Нехай є матрична гра з матрицею A, ціною гри v, оптимальними мішаними стратегіями $x^* = (x_1^*,\dots,x_m^*)$ та $y^* = (y_1^*,\dots,y_m^*)$ гравців А та В відповідно. Тоді, якщо для деякого i, буде $\sum_{j=1}^{n} a_{ij}y_j < v$, то $x_i = 0$; якщо для деякого j буде $\sum_{i=1}^{m} a_{ij}x_i > v$, то $y_j = 0$.

Ця теорема ґрунтується безпосередньо на співвідношеннях між прямою та двоїстою задачами лінійного програмування, що відповідають грі.

11.5. ПРЕДСТАВЛЕННЯ ГРИ У ВИГЛЯДІ ЗАДАЧ ЛІНІЙНОГО ПРОГРАМУВАННЯ

Надалі будемо вважати, що всі виграші не є негативними – з цією метою додамо до всіх елементів матриці гри значення найбільшого за абсолютним значенням негативного елементу. В результаті такого перетворення оптимальні стратегії не зміняться, лише ціна гри збільшиться на відповідне значення.

Якщо один з гравців застосовує свою оптимальну стратегію $x^* = (x_1^*,\dots,x_m^*)$, то інший не може покращити своє становище, тобто для оптимальної стратегії справедливі співвідношення

$$\sum_{i=1}^{m} a_{ij}x_i \ge v,\ j = \overline{1,n},\ x_i \ge 0,\ \sum_{i=1}^{m} x_i = 1,\ i = \overline{1,m} \text{ за умови } v \Rightarrow Max. \qquad (11.23)$$

Перетворимо цю задачу, здійснивши підстановку $p_i = \frac{x_i}{v}$, і отримаємо

$\sum_{i=1}^{m} a_{ij} p_i \ge 1,\ j = \overline{1,n},\ \sum_{i=1}^{m} \frac{x_i}{v} = \frac{1}{v} = \sum_{i=1}^{m} p_i \Rightarrow Min$, тому що $v \Rightarrow Max$.(11.24)

Таким чином, маємо задачу лінійного програмування, розв'язуючи яку, отримаємо значення p_i, за допомогою яких шляхом оберненої підстановки визначимо оптимальні значення вірогідностей, що складають оптимальну мішану стратегію.

Таким чином, підсумовуючи, оптимальні мішані стратегії гравців А та В задовільняють другий наслідок з основної теореми теорії антагоністичних ігор двох осіб, а саме:

$$\sum_{i=1}^{m} a_{ij} x_i \ge v,\ j = \overline{1,n},\ x_i \ge 0,\ \sum_{i=1}^{m} x_i = 1,\ \overline{i = 1,m}\ . \tag{11.25}$$

$$\sum_{j=1}^{n} a_{ij} y_j \le v,\ i = \overline{1,m},\ y_j \ge 0,\ \sum_{j=1}^{n} y_j = 1,\ \overline{j = 1,n}\ . \tag{11.26}$$

Здійснивши підстановки $p_i = \frac{x_i}{v}$ та $q_j = \frac{y_j}{v}$ і враховуючи, що гравець А прагне максимізувати свій середній виграш, а гравець В – мінімізувати програш, отримаємо пару двоїстих задач лінійного програмування, розв'язання яких дозволить визначити оптимальні стратегії гравців А та В:

$$\frac{1}{v} = \sum_{i=1}^{m} p_i \Rightarrow Min,\ \sum_{i=1}^{m} a_{ij} p_i \ge 1,\ \ p_i \ge 0,\ j = \overline{1,n}, \tag{11.27}$$

$$\frac{1}{v} = \sum_{j=1}^{n} q_j \Rightarrow Max,\ \sum_{j=1}^{n} a_{ij} q_j \le 1,\ \ q_j \ge 0,\ i = \overline{1,m}. \tag{11.28}$$

Таким чином, процедура розв'язування антагоністичної гри двох осіб є наступною:

Крок 1. Розраховуємо нижню та верхню ціну гри: якщо вони рівні між собою, то гра розв'язана – **стоп** (розв'язок гри є розв'язком в чистих стратегіях).

Крок 2. Спрощуємо гру шляхом виключення домінованих стратегій.

Крок 3. Формулюємо пару задач лінійного програмування, розв'язуючи одну з яких, встановлюємо оптимальну мішану стратегію одного з гравців (зручніше гравця В).

Крок 4. За розв'язком прямої задачі знаходимо розв'язок двоїстої.

Крок 5.Шляхом оберненої підстановки визначаємо оптимальні мішані стратегії для спрощеної гри та доповнюємо їх домінованими чистими стратегіями з вірогідностями використання, що рівні нулю.

11.6. ІГРИ ПОРЯДКУ 2×2, $2\times N$ ТА $M\times2$. ГРАФІЧНЕ РОЗВ'ЯЗУВАННЯ ІГОР

Матриця гри розміру 2×2 має вигляд $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$. (11.29)

Якщо є сідлова точка, то гра розв'язана. В іншому випадку шукаємо точку в мішаних стратегіях $x = (x_1, x_2)$, $y = (y_1, y_2)$. Для оптимальності мішаних стратегій необхідно виконання наступних співвідношень:

$$\begin{aligned} &a_{11}x_1 + a_{21}x_2 \ge v \quad && a_{11}y_1 + a_{12}y_2 \le v \\ &a_{12}x_1 + a_{22}x_2 \ge v \quad && a_{21}y_1 + a_{22}y_2 \le v \\ &x_1 + x_2 = 1 && y_1 + y_2 = 1 \end{aligned} \tag{11.30}$$

Якщо матрична гра не має сідлової точки в чистих стратегіях (тобто $0 < x_1 < 1$, $0 < x_2 < 1$, $0 < y_1 < 1$, $0 < y_2 < 1$), то, згідно з наслідкамиз теорем, вищенаведені нерівності перетворюються в рівності – якщо припустити, що яка-небудь з нерівностей виконується строго, то це б означало, що відповідна їй змінна двоїстої задачі повинна бути рівною нулю – що суперечить припущенню про відсутність сідлової точки, а саме цей випадок ми розглядаємо. Розв'язуючи системи рівнянь аналітично

$$\begin{aligned} &a_{11}x_1 + a_{21}x_2 = v \quad && a_{11}y_1 + a_{12}y_2 = v \\ &a_{12}x_1 + a_{22}x_2 = v \quad && a_{21}y_1 + a_{22}y_2 = v, \\ &x_1 + x_2 = 1 && y_1 + y_2 = 1 \end{aligned} \tag{11.31}$$

отримаємо

$$x_1 = \frac{a_{22} - a_{21}}{a_{11} - a_{12} - a_{21} + a_{22}},\ y_1 = \frac{a_{22} - a_{12}}{a_{11} - a_{12} - a_{21} + a_{22}},$$

$$v = \frac{a_{11}a_{22} - a_{21}a_{12}}{a_{11} - a_{12} - a_{21} + a_{22}}\ . \tag{11.32}$$

В іграх порядку $2\times n$ гравець А має лише 2 чисті стратегії, а В – n.

Матриця гри має вигляд: $A = \begin{array}{c|cccc} & y_1 & y_2 & \dots & y_n \\ \hline x_1 & a_{11} & a_{12} & \dots & a_{1n} \\ x_2 & a_{21} & a_{22} & \dots & a_{2n} \end{array}$. (11.33)

Якщо гра не має сідлової точки, то необхідно знайти такі мішані стратегії, для яких, згідно з наслідку теорем теорії ігор, виконувалися б наступні умови:

$$1)\ \begin{cases} a_{11}x_1 + a_{21}x_2 \ge v \\ \dots\dots\dots\dots \\ a_{1n}x_1 + a_{2n}x_2 \ge v \end{cases} \quad 2).\ \begin{cases} a_{11}y_1 + a_{12}y_2 + \dots + a_{1n}y_n \le v \\ a_{21}y_1 + a_{22}y_2 + \dots + a_{2n}y_n \le v \end{cases}$$

$$3)\ \ x_2 = 1 - x_1,\ x_1 \ge 0,\ x_2 \ge 0\ \ 4).\ \ y_1 + y_2 + \dots + y_n = 1,\ \forall y_j \ge 0. \qquad (11.34)$$

Оскільки гра не має сідлової точки, нерівності 2-ї групи перетворяться в рівності (якщо припустити, що хоча б одна з цих нерівностей строга, то їй мало б відповідати нульове значення x, що суперечить твердженню про те, що гра не має сідлової точки).

Таким чином, необхідно розв'язати систему (11.34) за умови, що група 2 – рівності. Для розв'язування використаємо графічний метод. Введемо позначення для лівої частини нерівностей 1-ї групи:

$$M_j(x_1) = a_{1j}x_1 + a_{2j}x_2 = (a_{1j} - a_{2j})x_1 + a_{2j}, \qquad (11.35)$$

де $M_j(x_1)$ – середній виграш гравця А, коли він застосовує змішану стратегію $x = (x_1, x_2)$, а гравець B – чисту стратегію B_j. Гравець B прагне мінімізувати програш – тобто обрати $M(x_1) = \underset{j}{Min}\ M_j(x_1)$, а гравець А – збільшити свій виграш $M(x_1{*}) = \underset{x_1}{Max}\ M(x_1)$.

Таким чином і визначається оптимальна мішана стратегія гравця А $x^* = (x_1{*}, x_2{*})$ та пара чистих стратегій гравця B, які в перетині дають $M^* = M(x_1{*})$. Далі розв'язуємо систему з тими чистими стратегіями гравця B, які знаходяться на перетині M^*.

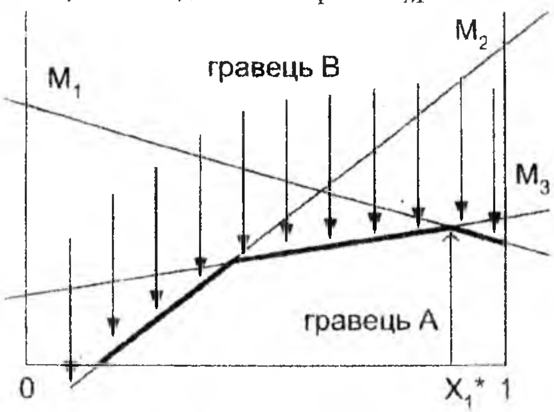


Рис. 11.2. Гра $2 \times n$

В іграх порядку $m \times 2$ гравець А має m чистих стратегій, а B – 2.

Матриця гри має вигляд:

$$A = \begin{array}{c|cc} & y_1 & y_2 \\ \hline x_1 & a_{11} & a_{12} \\ x_2 & a_{21} & a_{22} \\ \dots & \dots & \dots \\ x_m & a_{m1} & a_{m2} \end{array} \qquad (11.36)$$

Якщо гра не має сідлової точки, то необхідно знайти такі мішані стратегії, для яких, згідно з наслідку з теорем теорії ігор, виконувалися б наступні умови:

$$1).\ \begin{cases} a_{11}x_1 + a_{21}x_2 + \dots + a_{m1}x_m \ge v \\ a_{12}x_1 + a_{22}y_2 + \dots + a_{m2}x_m \ge v \end{cases} \quad 2).\ \begin{cases} a_{11}y_1 + a_{12}x_2 \le v \\ \dots\dots\dots\dots \\ a_{m1}y_1 + a_{m2}y_2 \le v \end{cases}$$

$$3).\ x_1 + x_2 + \dots + x_m = 1,\ \forall x_i \ge 0\ \ .4)\ y_2 = 1 - y_1,\ y_1 \ge 0,\ y_2 \ge 0. \qquad (11.37)$$

Оскільки гра не має сідлової точки, нерівності 1-ї групи перетворяться в рівності (якщо припустити, що хоча б одна з цих нерівностей строга, то їй мало б відповідати нульове значення у, що суперечить твердженню про те, що гра не має сідлової точки.

Таким чином, необхідно розв'язати систему (1-4) за умови, що група 1 – рівності. Для розв'язування використаємо графічний метод. Введемо позначення для лівої частини нерівностей 2-ї групи:

$$M_i(y_1) = a_{i1}y_1 + a_{i2}y_2 = (a_{i1} - a_{i2})y_1 + a_{i2}, \qquad (11.38)$$

де $M_i(y_1)$ – середній програш гравця А, коли він застосовує мішану стратегію $y = (y_1, y_2)$, а гравець А чисту стратегію A_i. Гравець А прагне максимізувати виграш – тобто обрати $M(y_1) = \underset{i}{Max}\ M_i(y_1)$, а гравець B – зменшити свій програш $M(y_1{*}) = \underset{y_1}{Min}\ M(y_1)$.

Таким чином і визначається оптимальна мішана стратегія гравця B $y^* = (y_1{*}, y_2{*})$ та пара чистих стратегій гравця А, які в перетині дають $M^* = M(y_1{*})$. Далі розв'язуємо систему з тими чистими стратегіями гравця А, які знаходяться на перетині M^*.

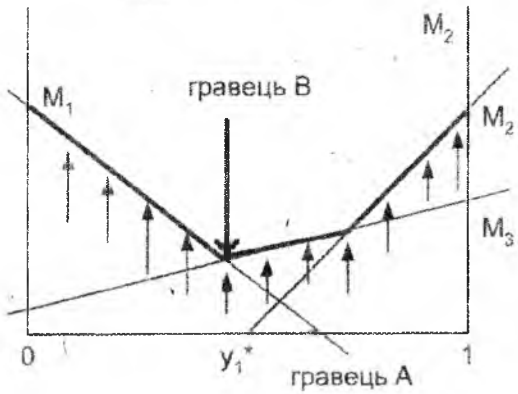


Рис. 11.3. Гра m×2

ПРИКЛАДИ

Приклад 11.1. Побудова матриці виплат.

Розглянемо одну з найпримітивніших ігор – “орлянку”. Гравець 1 кладе монету на стіл, а гравець 2 вгадує, якою стороною – гербом чи цифрою – догори вона лежить. У випадку вгадування він отримує від гравця 1 одну одиницю виграшу, в іншому випадку сам виплачує йому одиницю.

Ця гра – антагоністична. В ній множина припустимих стратегій кожного з гравців $A = B = \{Г, Ц\}$, а виграші одного (одночасно програші іншого Н(Г,Г)=Н(Ц,Ц)=-1 і Н(Ц,Г)=Н(Г,Ц)=1, або в матричній формі:

$$\begin{array}{c|cc|} & Г & Ц \\ Г & -1 & 1 \\ Ц & 1 & -1 \end{array}$$

Приклад 11.2. Приклади ігор.

Родинна суперечка.

Два економічні партнери (гравці 1 і 2) домовляються про сумісне проведення одної з дій, D_1 або D_2, кожна з яких вимагає спільної участі двох партнерів.

У випадку сумісного проведення дії D_1 гравець 1 отримує одну одиницю корисності, а гравець 2 – дві одиниці. Навпаки, у випадку сумісного проведення D_2 гравець 1 отримує дві одиниці, а гравець 2 – лише одну. Нарешті, якщо гравці виконують різні дії, то виграш кожного з них

дорівнює нулю.

В цьому випадку ми маємо справу з біматричною грою з матрицями виграшів:

Матриця виплат гравця 1 Матриця виплат гравця 2

$$\begin{array}{c|cc|} & D_1 & D_2 \\ D_1 & 1 & 0 \\ D_2 & 0 & 2 \end{array} \qquad \begin{array}{c|cc|} & D_1 & D_2 \\ D_1 & 2 & 0 \\ D_2 & 0 & 1 \end{array}$$

У багатьох роботах з теорії ігор цю гру інтерпретують як одночасний вибір подружжям вечірньої розваги: відвідування змагань з боксу або ж балету, причому у відвідуванні боксу чоловік зацікавлений більшою мірою, ніж дружина, при відвідуванні балету спостерігається зворотна картина, а у випадку невизначеної розбіжності вечір взагалі виявляється зіпсутим. Внаслідок такої інтерпретації гра часто називається “родинною суперечкою”.

Військова справа. Американський вчений Хейвуд дослідив зв'язок між доктриною військових рішень і теорією ігор двох осіб з нульовою сумою. Як ілюстрацію своїх досліджень, Хейвуд наводить приклад, взятий з Другої світової війни – битву в Ново-Гвінейському морі.

В критичні дні боротьби за Нову Гвінею розвідка повідомила, що японці збираються послати конвой з військами та провіантом з порту Ребаул (східне узбережжя Нової Британії) в Лае (Нова Гвінея, на захід від Нової Британії). Він міг пройти або північніше від Нової Британії, де очікувалася погана погода, або південніше від Нової Британії, де погода очікувалась ясна; в обох випадках подорож тривала би три дні. Генерал Кенней міг обрати зосередження основних сил своєї розвідувальної авіації або на одному, або на іншому шляху. Після виявлення конвою його можна було бомбити до прибуття в Лае. Штаб Кеннея визначив результати для різних варіантів вибору в днях бомбардувального часу наступним чином:

Стратегії японців:

Стратегії Кеннея:

$$\begin{array}{r|cc|} & \textit{Півн.\_шлях} & \textit{Півд.\_шлях} \\ \textit{Північний\_шлях:} & 2 & 2 \\ \textit{Південний\_шлях:} & 1 & 3 \end{array}$$

Не заглиблюючись на даному етапі в пояснення щодо розв'язання цієї задачі, відзначимо лише, що японський конвой було виявлено приблизно через день після його виходу в море; і японці зазнали важких втрат.

Приклад 11.3. Приклади ігор.

Теніс – це гра двох осіб, в якій програш одного з гравців є виграшем іншого, а ціною гри є кількість очок, що її отримує гравець, або умови проходження в наступний тур.

Футбол – це гра, в якій одна проти одної виступають коаліції – дві команди по 11 гравців в кожній – в принципі, за певних умов цю гру можна розглядати як гру двох осіб.

Шахи – можливі три результати – 0.5, 1 або 0 очок.

Три фірми діють на ринку, причому ні одна з них не має повного контролю над ним. В цьому випадку, якщо кожен з гравців діє самостійно, це є гра трьох осіб, якщо ж двоє з гравців вирішують об'єднатися, то це є гра двох осіб.

Приклад 11.4. Сідлові точки.

Чи може у матриці бути декілька сідлових точок?

$$A=\begin{pmatrix}12 & 13 & 12\\ 10 & 31 & 9\end{pmatrix}$$

Так, у цієї матриці виплат сідловими точками є a_{11}, a_{13}.

Приклад 11.5. Вибір чистої стратегії в грі двох осіб з нульовою сумою.

Нехай задана наступна матриця гри.

	B_1	B_2	B_3	B_4	B_5
A_1	3	4	5	2	3
A_2	1	8	4	3	4
A_3	10	3	1	7	6
A_4	4	5	3	4	8

Виникає запитання: якою стратегією скористатися гравцям?

Розв'язання.

Якщо гравець А обере стратегію A_3, сподіваючись на виграш в 10 одиниць, то гравець В може обрати стратегію B_3, і виграш гравця А становитиме лише 1 одиницю. Відповідь на це запитання, за умови повної відкритості гри, дає принцип максиміну: необхідно діяти таким чином, щоб мінімальний гарантований виграш був максимальним, тобто дорівнював $\alpha = \underset{i}{Max}\ \underset{j}{Min}\ a_{ij}$ (для прикладу, $\alpha = 3$, і гравцеві А потрібно би було обрати стратегію A_4).

	B_1	B_2	B_3	B_4	B_5	$\alpha_i = \underset{j}{Min}\ a_{ij}$
A_1	3	4	5	2	3	$\alpha_1 = 2$
A_2	1	8	4	3	4	$\alpha_2 = 1$
A_3	10	3	1	7	6	$\alpha_3 = 1$
A_4	4	5	3	4	8	$\alpha_4 = 3$
$\beta_j = \underset{i}{Max}\ a_{ij}$	$\beta_1 = 1$	$\beta_2 = 8$	$\beta_3 = 5$	$\beta_4 = 7$	$\beta_5 = 8$	

При цьому гравець В прагнутиме не дати гравцеві А отримати максимальний виграш, оскільки це для нього програш, і, своєю чергою, у відповідь він обиратиме свою стратегію, якій відповідатиме $\beta = \underset{j}{Min}\ \underset{i}{Max}\ a_{ij}$ (для прикладу, $\beta = 5$). Таким чином, за умови максимально обережної гри з обох боків та застосування чистих стратегій гравець А виграє 3 одиниці, а гравець В програє не більше 5-ти.

Однак, якщо гравець А знатиме, що В обрав стратегію B_3, то в нього виникне бажання збільшити свій виграш, обравши стратегію A_1. У відповідь гравець В прагнутиме обирати B_4, і т. д.

$(A_4 \rightarrow B_3 \rightarrow A_1 \rightarrow B_4 \rightarrow A_3 \rightarrow B_3)$.

Таким чином, гра в чистих стратегіях виявляється нестійкою.

Приклад 11.6. Розв'язання гри в чистих стратегіях.

Задана матриця гри. **Визначити верхню та нижню ціну гри.**

	B_1	B_2	B_3	B_4
A_1	2	4	7	5
A_2	7	6	8	7
A_3	5	3	4	1

Розв'язання.

Значення верхньої та нижньої ціни гри рівні між собою – $\alpha = \beta = 6$. У цьому випадку ні одному з гравців невигідно відхилятися від своїх стратегій, і отримуємо таким чином точку рівноваги:

	B_1	B_2	B_3	B_4	$\alpha_i = \underset{j}{Min}\ a_{ij}$
A_1	2	4	7	5	$\alpha_1 = 2$
A_2	7	6	8	7	$\alpha_2 = 6$
A_3	5	3	4	1	$\alpha_3 = 1$
$\beta_j = \underset{i}{Max}\ a_{ij}$	$\beta_1 = 7$	$\beta_2 = 6$	$\beta_3 = 8$	$\beta_4 = 7$	

Таким чином, у цій грі з матрицею A нижня та верхня чисті ціни гри рівні між собою, і гра має сідлову точку в чистих стратегіях а чисту ціну гри $\nu = \alpha = \beta = 6$. Поняття сідлової точки відображає наступну ситуацію: якщо один з гравців дотримується стратегії, що відповідає сідловій точці, то інший гравець не може діяти краще, як дотримуватися своєї стратегії, яка також відповідає сідловій точці.

Приклад 11.7. Розв'язання антагоністичної гри двох осіб.

Знайти розв'язок гри, заданої наступною матрицею:

	B_1	B_2	B_3	B_4	B_5
A_1	4	7	2	3	4
A_2	3	5	6	8	9
A_3	4	4	2	2	8
A_4	3	6	1	2	4
A_5	3	5	6	8	9

Розв'язання.

Гра не має сідлової точки – спрощуємо гру, виключаючи доміновані стратегії.

		3 $B_1>B_2$		5 $B_3>B_4$	4 $B_1>B_5$		
	B_1		B_3				
A_1	4		2			$\alpha_1 = 2$	
A_2	3		6			$\alpha_2 = 3$	
						$\alpha_3 = 2$	6 $A_1>A_3$
						$\alpha_4 = 1$	1 $A_1>A_4$
						$\alpha_5 = 3$	2 $A_2>A_5$
	$\beta_1=4$	$\beta_2=7$	$\beta_3=6$	$\beta_4=8$	$\beta_5=9$		

Стратегія A_i домінує стратегію A_j, якщо в A_i виграші не менші, ніж в A_j, і хоча би один строго більший. Стратегії еквівалентні, якщо всі виграші цих двох стратегій однакові. З числа еквівалентних стратегій також залишаємо лише одну. Стратегія B_i домінує стратегію B_j, якщо в B_i програші не більші, ніж в B_j, і хоча би один строго менше. Таким чином, оптимальні мішані стратегії гравців A та B будуть мати відповідно вигляд $x^* = (x_1, x_2, 0, 0, 0)$, $y^* = (y_1, 0, y_2, 0, 0)$, де ненульові значення отримуються після розв'язування спрощеної гри розміром 2×2.

Приклад 11.8. Представлення гри у вигляді задач лінійного програмування.

Знайти розв'язок гри, заданої наступною матрицею:

$$A = \begin{bmatrix} 3 & 6 & 1 & 4 \\ 5 & 2 & 4 & 2 \\ 1 & 4 & 3 & 5 \end{bmatrix}.$$

Хід розв'язання.

Так як доміновані стратегії відсутні, для розв'язування гри застосовуємо представлення у вигляді пари двоїстих задач лінійного програмування:

$$\begin{array}{ll} Q = q_1 + q_2 + q_3 + q_4 \Rightarrow Max & p_1 + p_2 + p_3 \Rightarrow Min \\ 3q_1 + 6q_2 + q_3 + 4q_4 <= 1 & 3p_1 + 5p_2 + p_3 >= 1 \\ 5q_1 + 2q_2 + 4q_3 + 2q_4 <= 1 \quad (1) & 6p_1 + 2p_2 + 4p_3 >= 1 \qquad (2) \\ q_1 + 4q_2 + 3q_3 + 5q_4 <= 1 & p_1 + 4p_2 + 3p_3 >= 1 \\ \forall\, q_j >= 0 & 4p_1 + 2p_2 + 5p_3 >= 1, \ \forall p_i >= 0. \end{array}$$

Розв'язуємо задачу (1) за допомогою звичайного симплекс-методу, і переходимо шляхом обернених підстановок $v=1/Q^*$, $y_j=q_j v$ до визначення оптимальної мішаної стратегії гравця B. З останньої симплекс-таблиці цієї задачі визначаємо оптимальні значення p_i, за допомогою яких, використовуючи підстановку $x_i=p_i v$, розраховуємо значення оптимальної мішаної стратегії гравця A.

Приклад 11.9. Графічне розв'язання гри.

Розв'язати графічно гру, задану наступною матрицею:

$$A = \begin{array}{c|cccccc} & y_1 & y_2 & y_3 & y_4 & y_5 & y_6 \\ \hline x_1 & 10 & 8 & 6 & 4 & 2 & 3 \\ x_2 & 1 & 2 & 4 & 3 & 12 & 6 \end{array}.$$

Розв'язання.

Випишемо середні виграші гравця A при застосуванні ним мішаної стратегії, а гравцем B – чистої для кожної з чистих стратегій гравця B:

$M_1(x_1) = (10-1)x_1 + 1 = 9x_1 + 1$

$M_2(x_1) = (8-2)x_1 + 2 = 6x_1 + 2$

$M_3(x_1) = (6-4)x_1 + 4 = 2x_1 + 4$

$M_4(x_1) = (4-3)x_1 + 3 = x_1 + 3$

$M_5(x_1) = (2-12)x_1 + 12 = -10x_1 + 12$

$M_6(x_1) = (3-6)x_1 + 6 = -3x_1 + 6.$

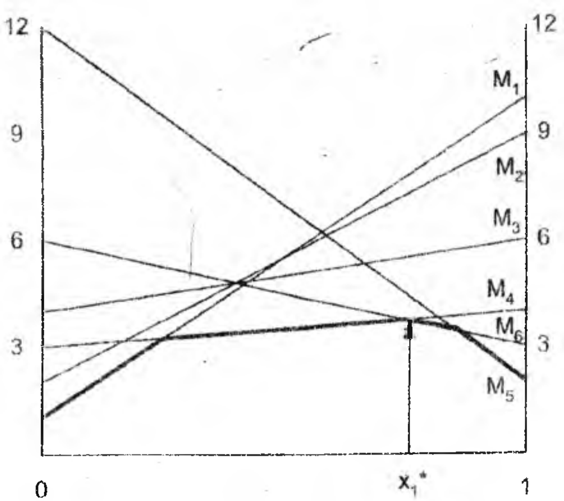
Зобразимо середні виграші графічно.

Найбільший виграш матиме гравець А, якщо він обере мішану стратегію, що відповідатиме застосуванню гравцем В суміші чистих стратегій B_4 та B_6. Звідси записуємо системи рівнянь для визначення мішаних стратегій гравців А та В:

$4x_1+3x_2=v$	$4y_4+3y_6=v$
$3x_1+6x_2=v$	$3y_4+6y_6=v$
$x_1+x_2=1$	$y_4+y_6=1.$

Розв'язуючи, отримаємо $x_1^*=3/4$, $x_2^*=1/4$, $y_4^*=3/4$, $y_6^*=1/4$, $v=15/4$.

Таким чином, оптимальні стратегії гравців будуть $x^*=(3/4,1/4)$, $y^*=(0,0,0,3/4,0,1/4)$, а ціна гри $v=15/4$.



Приклад 11.10. Графічне розв'язання гри.

Розв'язати графічно гру, задану наступною матрицею:

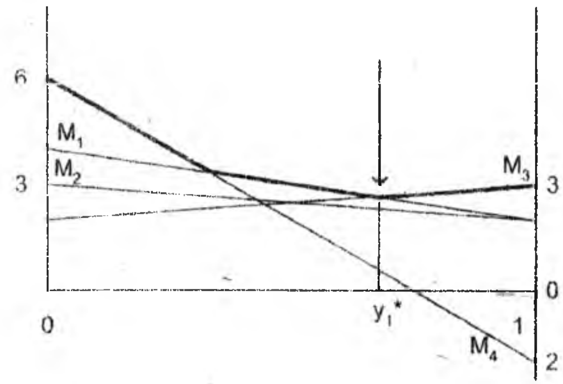
$$
\begin{array}{ccc}
 & y_1 & y_2 \\
x_1 & 2 & 4 \\
A = x_2 & 2 & 3 \\
x_3 & 3 & 2 \\
x_4 & -2 & 6
\end{array}.
$$

Випишемо середні програші гравця В при застосуванні ним мішаної стратегії, а гравцем А — чистої для кожної з чистих стратегій гравця А:

$M_1(y_1)=-2y_1+4;$ $M_2(y_1)=-y_1+3;$

$M_3(y_1)=y_1+2;$ $M_4(y_1)=-8y_1+6.$

Зобразимо середні виграші графічно.



Найменший програш матиме гравець В, якщо він обере змішану стратегію, що відповідатиме застосуванню гравцем А суміші чистих стратегій A_1 та A_3. Звідси записуємо системи рівнянь для визначення мішаних стратегій гравців А та В:

$2x_1+3x_3=v$	$2y_1+3y_2=v$
$4x_1+2x_3=v$	$3y_1+2y_2=v$
$x_1+x_3=1$	$y_1+y_2=1.$

Розв'язуючи, отримаємо $x_1^*=1/3$, $x_3^*=2/3$, $y_1^*=2/3$, $y_2^*=1/3$, $v=8/3$.

Таким чином, оптимальні стратегії гравців будуть $x^*=(1/3,0,2/3,0)$, $y^*=(2/3,1/3)$, а ціна гри $v=8/3$.

РЕЗЮМЕ

11.1. Теорія ігор досліджує питання поведінки і виробляє оптимальні правила (стратегії) поведінки для кожного з учасників конфліктної ситуації. Розв'язання суперечностей за допомогою теорії ігор можливе лише після проведення математичного моделювання ситуацій у вигляді гри. Таким чином, гра характеризується за допомогою системи правил, які описують сутність конфліктної ситуації: кількість гравців, вибір способу дій гравців на кожному з етапів гри; інформацію, якою володіє кожен з гравців при здійсненні таких виборів; виплату для кожного з гравців після завершення довільного етапу гри.

11.2. Класифікація ігор реалізується за певною множиною класифікаційних ознак, а саме: кількість гравців, кількість стратегій, характер взаємин між гравцями, характер виграшів, вигляд функції виграшів, момент вибору ходу, кількість ходів, стан інформації.

11.3. Матрична гра є антагоністичною. Для вибору оптимальної стратегії використовується принцип гарантованого результату (принцип "максиміну"), тобто обирається така стратегія, яка за найгірших умов забезпечить максимальний виграш. При цьому вважається, що суперник буде робити найкращі свої ходи, тобто оптимальним чином (зі своєї точки зору) протидіяти. Якщо в грі з матрицею виплат А нижня та верхня ціни гри рівні між собою, то гра має сідлову точку, тобто розв'язок у чистих стратегіях.

11.4. Мішаною стратегією гравця називається повний набір ймовірностей застосування ним його чистих стратегій. Середній виграш гравця А в матричній грі відображається як математичне сподівання його виграшу при застосуванні ним мішаної стратегії. Для матричної гри завжди існує розв'язок в мішаних стратегіях.

11.5. Процедура розв'язування гри є наступною: розраховуємо нижню та верхню ціну гри; якщо вони рівні між собою, то гра розв'язана; спрощуємо гру шляхом виключення домінованих стратегій; формулюємо пару задач лінійного програмування, розв'язуючи одну з яких, встановлюємо оптимальну мішану стратегію одного з гравців; за розв'язком прямої задачі знаходимо розв'язок двоїстої; шляхом оберненої підстановки визначаємо оптимальні мішані стратегії для спрощеної гри та доповнюємо їх домінованими чистими стратегіями з ймовірностями використання, що рівні нулю.

11.6. Гра розміром 2×2 розв'язується аналітично, а ігри порядку $2 \times n$ та $m \times 2$ - графічно.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 11.1.

Розв'язати антагоністичну гру з наступною матрицею виплат.

$$A = \begin{pmatrix} 18 & 3 & 0 & 2 \\ 0 & 3 & 8 & 20 \\ 5 & 4 & 5 & 5 \\ 16 & 4 & 2 & 25 \\ 9 & 3 & 0 & 20 \end{pmatrix}$$

Завдання 11.2.

Знайти розв'язок гри, заданої матрицею $\begin{pmatrix} 2 & 5 \\ 6 & 4 \end{pmatrix}$ і дати геометричну інтерпретацію цього рішення.

Завдання 11.3.

Знайти розв'язок гри, заданої матрицею $A = \begin{pmatrix} 7 & 9 & 8 \\ 10 & 9 & 6 \end{pmatrix}$.

Завдання 11.4.

Знайти розв'язок гри, заданої матрицею $A = \begin{pmatrix} 6 & 5 \\ 4 & 6 \\ 2 & 7 \\ 1 & 8 \end{pmatrix}$.

Завдання 11.5.

Знайти розв'язок гри, заданої матрицею $A = \begin{pmatrix} 2 & 3 & 4 & 1 \\ 5 & 2 & 2 & 3 \\ 4 & 1 & 3 & 2 \end{pmatrix}$.

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Дайте визначення теорії ігор та її основних понять.
2. В чому суть принципу оптимальності стосовно теорії ігор?
3. Яким чином визначається поняття рівноваги в грі?
4. Яким чином класифікуються ігри за кількістю гравців, кількістю стратегій та за характером взаємин між гравцями?
5. В чому суть гри типу дуелі?
6. Які особливості багатокрокових ігор?
7. Дайте визначення матричної гри та її основних характеристик.
8. Сформулюйте та доведіть теорему про верхню та нижню ціну гри.
9. Розкрийте значення та доведіть теорему про сідлову точку.
10. Дайте визначення мішаної стратегії, середнього виграшу та сформулюйте основну теорему теорії матричних ігор з нульовою сумою.
11. Обгрунтуйте приведення гри до пари задач лінійного програмування.
12. Перелічіть основні кроки алгоритму розв'язування антагоністичної матричної гри 2-х осіб.
13. Обгрунтуйте отримання формул для розв'язування антагоністичної гри 2×2.
14. Поясніть, яким чином розв'язуються ігри графічно.

ТЕМА 12. ПОЗИЦІЙНІ ІГРИ ТА ІГРИ ДЕКІЛЬКОХ ОСІБ

Позиційна гра – це природне розширення матричної гри двох гравців з нульовою сумою, в якій може брати участь скінчена кількість гравців, кожен з яких може робити послідовно скінчену кількість ходів, причому деякі з них можуть бути випадковими, а інформація про них може змінюватися від одного до іншого ходу. Процес приведення позиційної гри до матричної називається нормалізацією, а отримана матрична гра – грою в нормальній формі. Аналіз позиційних ігор за допомогою інформаційних множин доводить тезу про те, що в загальному випадку втрата інформації зменшує ціну гри. Один з найважливіших висновків теорії ігор – це те, що певні форми кооперування гравців за умови зовнішньо різних їх прагнень дійсно мають сенс. Біматричні ігри адекватніше відображають реальні конфлікти двох осіб порівняно з матричними, що використовуються для описання конфліктів з повною суперечністю і несумісністю інтересів, з відсутністю можливості компромісів; в таких конфліктах зростання прибутку одного гравця завжди означає збільшення втрат іншого. Біматрична гра – це найпростіша ігрова модель, що припускає можливість співробітництва. В іграх зі строгим суперництвом гравці не можуть досягти обопільної вигоди посередництвом якого-небудь співробітництва; в іграх з нестрогим суперництвом, навпаки, такий обопільний виграш завжди можливий шляхом укладання певних угод. На основі аналізу ігор можна зробити наступні висновки: існування ситуацій рівноваги в безкоаліційних іграх не визначає, взагалі кажучи, їх розв'язків, і однозначні рекомендації для оперуючих сторін відсутні; в багатьох випадках корисні (і навіть необхідні) контакти та угоди між гравцями, а тому моделі, що дозволяють кооперування, мають переваги; часткові постановки не виключають можливості використання теорії безкоаліційних ігор, і питання про пошук ситуацій рівноваги повинно досліджуватися окремо в кожному окремому випадку. Перехід конфліктуючих сторін до різних форм співробітництва (кооперування) створює якісно нові ситуації. В іграх з n учасниками розглядається три рівні взаємодії: обмін інформацією про перебіг гри та ситуації, що складаються; сумісний вибір стратегій на грунті загальної домовленості та взаємної інформованості; об'єднання активних засобів (ресурсів) з відповідною координацією дій. Важливим фактором, який визначає характер кооперування, є побічні виплати – "вступний внесок", "податок на кооперацію", "штраф за вихід *з кооперації". У цих випадках йдеться про зміну виграшів в той чи інший бік порівняно з початковими умовами гри, а тому побічні виплати перетворюються в частину стратегій, що застосовуються, та впливають на результат конфлікту.*

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ:

знати: ⇨послідовність кроків, необхідних для перетворення позиційної гри до матричної; ⇨проблематику ігор декількох осіб та основні рівні взаємодії гравців в цих іграх; ⇨методи дослідження кооперативних ігор;

вміти: ⇨приводити позиційні ігри до матричних та розв'язувати їх; ⇨ оцінювати втрати інформації в біматричних іграх за неповної інформації гравців; ⇨приймати рішення в умовах невизначеності та обґрунтовувати їх кількісно.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ позиційна гра
- ☑ безкоаліційна гра
- ☑ точка **status quo**
- ☑ граф позиційної гри
- ☑ коаліційна гра
- ☑ гра з природою
- ☑ інформаційна множина
- ☑ рівновага за Нешем
- ☑ матриця ризиків
- ☑ максимінна стратегія
- ☑ критерій Вальда
- ☑ кооперативна гра
- ☑ стратегія загрози
- ☑ критерій Севіджа
- ☑ біматрична гра
- ☑ переговорна множина
- ☑ критерій Гурвіца
- ☑ конфліктна ситуація

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

12.1. Поняття про позиційні ігри.

12.2. Кооперативні ігри та методи їх дослідження.

12.3. Прийняття рішень в умовах невизначеності.

12.1. ПОНЯТТЯ ПРО ПОЗИЦІЙНІ ІГРИ

Позиційна гра – це природне розширення матричної гри двох гравців з нульовою сумою, в якій може брати участь скінчена кількість гравців, кожен з яких може робити послідовно скінчену кількість ходів, причому деякі з них можуть бути випадковими, а інформація про них може змінюватися від одного до іншого ходу. Такі ігри можуть бути формалізовані, певним чином перетворені до гри, що еквівалентна деякій матричній грі двох осіб з нульовою сумою. Процес приведення позиційної гри до матричної називається нормалізацією, а отримана матрична гра — грою в нормальній формі.

Розглянемо наступний **приклад**. Дві корпорації мають бажання встановити між собою ділові зв'язки і вирішити питання про побудову на території другої корпорації виробництва. Гра складається з трьох ходів. Перша корпорація обирає число з множини $\{1,2\}$. Після цього друга корпорація обирає число з множини двох можливих $\{1,2\}$, знаючи, який вибір здійснила перша корпорація на першому ході. Третій хід робить перша корпорація: знаючи, який хід зробила друга корпорація, та пам'ятаючи про свій вибір на першому кроці, обирає число з множини $\{1,2\}$. На цьому гра завершується і розподіляються виграші: перший гравець виплачує другому певну суму, визначену функцією $M(x,y,z)$, яка визначена наступним чином в залежності від вибору гравцями 1-го – 3-го ходів x,y, та z відповідно:

$M(1,1,1)=-2$; $M(1,1,2)=-1$; $M(1,2,1)=3$; $M(1,2,2)=-4$;

$M(2,1,1)=5$; $M(2,1,2)=2$; $M(2,2,1)=2$; $M(2,2,2)=6$.

Змістовна інтерпретація цієї гри є наступною:

Хід 1. 1-а корпорація здійснює вибір з двох альтернатив: $x=1$ – запропонувати 2-й побудувати на її території складальне виробництво комп'ютерів, $x=2$ – побудувати виробництво мікропроцесорів.

Хід 2. 2-а корпорація, знаючи, яку альтернативу обрала 1-а на першому ході, здійснює вибір з двох альтернатив: $y=1$ – будувати складальне виробництво та запропонувати це 1-й корпорації; $y=2$ – будувати виробництво мікропроцесорів та запропонувати це 1-й.

Хід 3. 1-а корпорація, знаючи вибір другої на другому ході та пам'ятаючи свій вибір на першому ході, здійснює вибір з двох альтернатив: $z=1$ – погодитися з пропозицією 2-ї, $z=2$ – не погодитися з пропозицією 2-ї.

Після того, як зроблені ходи, партія зіграна, і 1-а корпорація отримує суму $M(x,y,z)$.

Для нормалізації цієї гри необхідно відтворити стратегії 1-го та 2-го гравця.

Стратегії 2-го гравця:

B_1 – обрати y=1 не зважаючи на вибір 1-го гравця на першому ході;

B_2 – обрати y=2 не зважаючи на вибір 1-го гравця на першому ході;

B_3 – погодитися з вибором 1-го гравця на першому ході, тобто обрати y=1, якщо x=1, і y=2, якщо x=2;

B_4 – не погодитися з вибором 1-го гравця на першому ході, тобто обрати y=2, якщо x=1, і y=1, якщо x=2.

Таким чином, 2-й гравець має 4 стратегії.

Стратегії 1-го гравця будуються аналогічно з врахуванням раніше зроблених виборів: вибір на першому кроці дає дві можливості, після вибору другого гравця з'являється чотири варіанти, і реалізація на третьому ході – 8 стратегій дії для 1-го гравця. Таким чином, стратегію 1-го гравця зображатимемо за допомогою трійки (i,i_1,i_2) – де i – вибір 1-го гравця на 1-му ході; i_1 – вибір 1-го гравця на третьому ході за умови вибору на 2-му ході 2-м гравцем y=1; i_2 – вибір 1-го гравця на 3-му ході за умови вибору 2-м на 2-му ході y=2.

Враховуючи відтворені стратегії, будуємо матрицю цієї гри:

	B_1	B_2	B_3	B_4
A_1 (1,1,1)	M(1,1,1)=-2	M(1,2,1)=3	M(1,1,1)=-2	M(1,2,1)=3
A_2 (1,1,2)	M(1,1,1)=-2	M(1,2,2)=-4	M(1,1,1)=-2	M(1,2,2)=-4
A_3 (1,2,1)	M(1,1,2)=-1	M(1,2,1)=3	M(1,1,2)=-1	M(1,2,1)=3
A_4 (1,2,2)	M(1,1,2)=-1	M(1,2,2)=-4	M(1,1,2)=-1	M(1,2,2)=-4
A_5 (2,1,1)	M(2,1,1)=5	M(2,2,1)=2	M(2,2,1)=2	M(2,1,1)=5
A_6 (2,1,2)	M(2,1,1)=5	M(2,2,2)=6	M(2,2,2)=6	M(2,1,1)=5
A_7 (2,2,1)	M(2,1,2)=2	M(2,2,1)=2	M(2,2,1)=2	M(2,1,2)=2
A_8 (2,2,2)	M(2,1,2)=2	M(2,2,2)=6	M(2,2,1)=6	M(2,1,2)=2

Розв'язком гри є дві сідлові точки, ціна гри – 5.

Побудуємо дерево позиційної гри. В цьому дереві вузол позначатиме номер гравця, що робить хід, а дуга – його хід. Листя дерева відображатимуть виграші, а кожний шлях від кореня до листка – партію.

Для відображення необхідних даних про зроблені вибори при певних ходах гравців за умов їх різної інформованості на різних ходах на дереві позиційної гри пунктиром позначатимемо інформаційні множини вузлів.

Оскільки 1-й хід робить 1-й гравець, то корінь відповідатиме ходу 1-го гравця та позначається 1. 2-й гравець робить 2-й хід, а тому вузли наступного рівня позначені 2, і так як йому відомий вибір 1-го гравця на

першому ході, то він, здійснюючи свій хід, в момент здійснення ходу знає точно, де він (на якій гілці дерева) знаходиться а тому кожен вузол нижнього рівня утворює окрему інформаційну множину (внаслідок повного знання ходів кожен з вузлів дерева цієї гри є окремою інформаційною множиною).

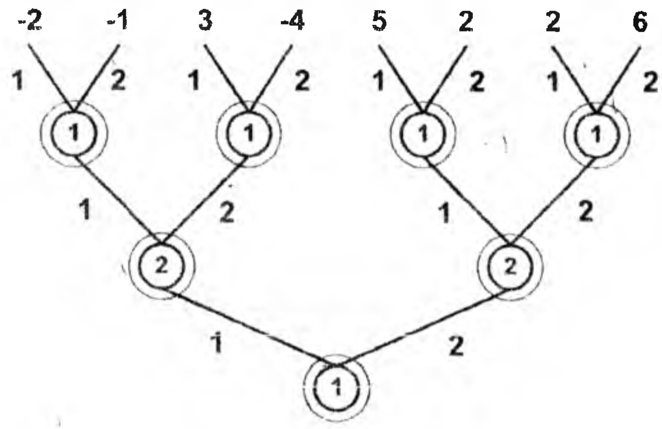


Рис.12.1. Інформаційні множини вузлів при знанні всіх ходів

Приклад 2. Порівняно з попереднім прикладом третій хід робить 1-й гравець, але вже не пам'ятаючи про те, який хід він зробив першим та не знаючи, який другий хід зробив другий гравець (1-го гравця можна уявити, як 2 особи, що знаходяться в окремих кімнатах та які не мають змоги обмінюватися інформацією). Відповідне дерево гри з інформаційними множинами наведене нижче (рис. 12.2).

Приведемо гру до нормальної форми. У 2-го гравця є 4 таких же стратегії, як і в попередньому випадку. У 1-го гравця можливості зменшуються за рахунок браку інформації: оскільки він на 3-му ході не знає попередніх виборів, то його стратегія складається з пари чисел (x,z), тобто обрати 1 або 2 на 1-му ході та 1 або 2 на 3-му ході. Відповідна матриця гри буде мати наступний вигляд:

	B_1	B_2	B_3	B_4
A_1 (1,1)	M(1,1,1)=-2	M(1,2,1)=3	M(1,1,1)=-2	M(1,2,1)=3
A_2 (1,2)	M(1,1,2)=-1	M(1,2,2)=-4	M(1,1,2)=-1	M(1,2,2)=-4
A_3 (2,1)	M(2,1,1)=5	M(2,2,1)=2	M(2,2,1)=2	M(2,1,1)=5
A_4 (2,2)	M(2,1,2)=2	M(2,2,2)=6	M(2,2,2)=6	M(2,1,2)=2

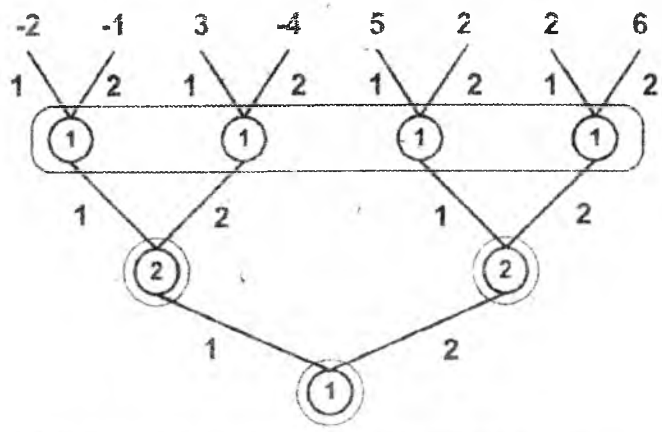


Рис 12.2. Інформаційні множини при частковому знанні

Отримана гра не має сідлової точки. Розв'язуючи гру в мішаних стратегіях, отримаємо мішані стратегії 1-го гравця – (0,0,4/7,3/7), 2-го гравця – (4/7,3/7,0,0), та ціна гри становитиме v=26/7. Таким чином, в загальному випадку втрата інформації зменшує ціну гри.

12.2. КООПЕРАТИВНІ ІГРИ ТА МЕТОДИ ЇХ ДОСЛІДЖЕННЯ

Конфліктні ситуації не завжди мають антагоністичний характер. Дуже часто учасники конфлікту, переслідуючи свої цілі, готові вступити в переговори один з одним, укладаючи деякі угоди чи навіть об'єднати свої зусилля в надії отримати з цього переваги.

Один з найважливіших висновків теорії ігор – це те, що певні форми кооперування гравців за умови зовнішньо різних їх прагнень дійсно мають сенс. Це частково пояснюється великою цінністю інформації, яка може бути передана від одного до іншого учасника гри, зростаючим значенням рішень, що приймаються спільно, синергічним ефектом від хоча б часткового об'єднання ресурсів.

Ігри двох осіб посідають центральне місце у всій теорії ігор. Серед таких ігор виділяються біматричні ігри. Причини цього наступні. Біматричні ігри адекватніше відображають реальні конфлікти двох осіб порівняно з матричними, що використовуються для описання конфліктів з повною суперечністю і несумісністю інтересів, з відсутністю можливості

компромісів: в таких конфліктах зростання прибутку одного гравця завжди означає збільшення втрат іншого. В житті такі конфлікти зустрічаються порівняно нечасто. Можна було б піддатися спокусі розглядати війну як крайній приклад зіткнення інтересів, але, взагалі кажучи, війна не є строгим суперництвом, оскільки обидві сторони вважають нічию кращим результатом, аніж взаємне знищення. Хоча локальне зіткнення або повітряний бій, очевидно, доцільно розглядати як гру зі строгим суперництвом. Але взагалі для описання військових конфліктів оперативно-тактичного плану, для яких притаманна наявність не менш ніж двох ієрархічно пов'язаних ланок "напад-оборона", звичайно як адекватні використовуються біматричні ігри

Біматричні ігри – це скінчені ігри двох осіб з довільною сумою, тобто такі, для яких не обов'язково виконується умова $a_{ij} = -b_{ij}$ (тобто виграш одного з гравців – це програш іншого. Ці ігри описуються або за допомогою двох матриць A та B – виграшів кожного з гравців, або ж за допомогою складеної матриці, елементами якої є пари (a_{ij}, b_{ij}).

$$\begin{pmatrix} (a_{11}, b_{11}) & (a_{12}, b_{12}) & \dots & (a_{1n}, b_{1n}) \\ (a_{21}, b_{21}) & (a_{22}, b_{22}) & \dots & (a_{2n}, b_{2n}) \\ \dots & \dots & \dots & \dots \\ (a_{m1}, b_{m1}) & (a_{m2}, b_{m2}) & \dots & (a_{mn}, b_{mn}) \end{pmatrix} \qquad (12.1)$$

Біматрична гра – це найпростіша ігрова модель, що припускає можливість співробітництва.

Гра з нестрогим суперництвом, як видно з її назви, є грою, в якій немає однозначного суперництва, тому що існує принаймні одна пара ситуацій x та x^*, така, що один гравець віддає перевагу ситуації x, а не x^*, а інший гравець не віддає переваги ситуації x^* перед ситуацією x. Для таких ігор неможливо обрати функції корисності гравців так, щоб сума їх дорівнювала нулю; тому терміни "ігри з нестрогим суперництвом" та "ігри з ненульовою сумою" є еквівалентними. Більшість економічних, політичних і військових конфліктів інтересів можна представити у формі ігор лише в тому випадку, якщо визнати властиве їм нестроге суперництво.

В іграх зі строгим суперництвом гравці не можуть досягти обопільної вигоди посередництвом якого-небудь співробітництва; в іграх з нестрогим суперництвом, навпаки, такий обопільний виграш завжди можливий.

Існують дві різновидності біматричних ігор – безкоаліційні, що забороняють будь-яке співробітництво, та кооперативні, що дозволяють співробітництво.

Результат біматричної безкоаліційної гри рідко можна передбачити, оскільки зв'язки між виграшами сторін відсутні, і з'являється можливість діяти самостійніше.

Значення середніх виграшів гравців А та В в біматричній грі рівні відповідно:

$$M_A[A,x,y]=\sum_{i=1}^{m}\sum_{j=1}^{n}a_{ij}x_iy_j \text{ та } M_B[B,x,y]=\sum_{i=1}^{m}\sum_{j=1}^{n}b_{ij}x_iy_j \qquad (12.2)$$

Ситуація рівноваги для біматричної гри – це така пара $\langle x,y\rangle$ мішаних стратегій гравців, для якої виконуються співвідношення

$$\sum_{j=1}^{n}a_{ij}y_j=\sum_{i=1}^{m}\sum_{j=1}^{n}a_{ij}x_iy_j, i=\overline{1,m}$$
$$\sum_{i=1}^{m}b_{ij}x_i=\sum_{i=1}^{m}\sum_{j=1}^{n}b_{ij}x_iy_j, j=\overline{1,n} \qquad (12.3)$$

та природні обмеження:

$$x_i\ge 0,\ \sum_{i=1}^{m}x_i=1,\ \overline{i=1,m},\ y_j\ge 0,\ \sum_{j=1}^{n}y_j=1,\ \overline{j=1,n}. \qquad (12.4)$$

Такі ситуації рівноваги в біматричній грі існують завжди – відповідна теорема про існування ситуацій рівноваги в мішаних стратегіях для скінчених неантагоністичних ігор двох осіб доведена Нешем (Nash J. F.).

Однак ця теорема не дає інформації про те, яким чином знайти ці ситуації рівноваги. Різноманітні алгоритми для знаходження всіх ситуацій рівноваги запропоновані багатьма авторами (Воробйов, Кун – Kuhn H. W. , Лемке та Хоусон – Lemke C. E., Howson J.J.) . Таким чином, виграші гравців у біматричних іграх задаються звичайними матричними добутками:

$$H_1(X,Y)=XAY^T, H_2(X,Y)=XBY^T. \qquad (12.5)$$

Розглянемо тепер основні ідеї, що стосуються ігор двох осіб з ненульовою сумою.

Некооперативна гра двох осіб.

Нехай задана гра двох осіб з матрицею

$$\begin{pmatrix} (a_{11},b_{11}) & (a_{12},b_{12}) & \dots & (a_{1n},b_{1n}) \\ (a_{21},b_{21}) & (a_{22},b_{22}) & \dots & (a_{2n},b_{2n}) \\ \dots & \dots & \dots & \dots \\ (a_{m1},b_{m1}) & (a_{m2},b_{m2}) & \dots & (a_{mn},b_{mn}) \end{pmatrix} \qquad (12.6)$$

В теорії розглядаються в основному дві стратегії поведінки гравців - це **максимінна стратегія** і так звана **стратегія загрози**.

Максимінна стратегія - це стратегія украй обережної людини, яка, розраховуючи на якнайгіршу ситуацію, хотіла би мати в цьому випадку максимум можливого.

Якщо один з гравців застосовує свою оптимальну максимінну стратегію $x^*=(x_1^*,\dots,x_m^*)$, то інший не може покращити своє становище, тобто для оптимальної стратегії справедливі співвідношення

$$\sum_{i=1}^{m}a_{ij}x_i\ge v,\ j=\overline{1,n},\ x_i\ge 0,\ \sum_{i=1}^{m}x_i=1,\ \overline{i=1,m} \text{ за умови } v\Rightarrow Max.$$

Перетворимо цю задачу, здійснивши підстановку $p_i=\frac{x_i}{v}$, і отримаємо:

$$\sum_{i=1}^{m}a_{ij}p_i\ge 1,\ j=\overline{1,n},\quad \sum_{i=1}^{m}\frac{x_i}{v}=\frac{1}{v}=\sum_{i=1}^{m}p_i\Rightarrow Min, \text{ тому що } v\Rightarrow Max.$$

Здійснивши підстановки $p_i=\frac{x_i}{v}$ та враховуючи, що гравець А прагне максимізувати свій середній виграш, отримаємо задачу лінійного програмування, розв'язання якої дозволить визначити оптимальну стратегію гравця А:

$$\frac{1}{v}=\sum_{i=1}^{m}p_i\Rightarrow Min,\ \sum_{i=1}^{m}a_{ij}p_i\ge 1,\ \ p_i\ge 0,\ j=\overline{1,n}, \qquad (12.7)$$

Друга можлива стратегія – це **стратегія загрози**, за якої гравець ставить за мету не виграти самому, а дати можливість виграти другому гравцеві якнайменше (при цьому в біматричній грі він і сам може виграти найменше!).

Станемо знову на позицію першого гравця. Хай він знову застосовує мішану мінімаксну стратегію (прагне мінімізувати виграш 2-го гравця) $x^*=(x_1^*,\dots,x_m^*)$. Таким чином, застосовуючи її, він рахує **не свій виграш, а виграш другого гравця**. Якщо другий гравець робить хід j, то його середній виграш становитиме $\sum_{i=1}^{m} b_{ij}x_i$. Перший гравець діє за принципом $\max_j \sum_{i=1}^{m} b_{ij}x_i \Rightarrow \min_x$, тобто він **мінімізує максимальний виграш другого гравця**. Якщо позначити максимальний виграш другого гравця через w, то ми матимемо:

$$\sum_{i=1}^{m} b_{ij}x_i \le w,\ j=\overline{1,n},\ x_i \ge 0,\ \sum_{i=1}^{m} x_i = 1,\ \overline{i=1,m} \text{ за умови } w \Rightarrow Min.$$

Перетворимо цю задачу, здійснивши підстановку $p_i = \frac{x_i}{w}$, і отримаємо:

$$\sum_{i=1}^{m} b_{ij}p_i \le 1,\ j=\overline{1,n},\ \sum_{i=1}^{m}\frac{x_i}{w} = \frac{1}{w} = \sum_{i=1}^{m} p_i \Rightarrow Max, \text{ тому що } w \Rightarrow Min.$$

Здійснивши підстановки $p_i = \frac{x_i}{w}$ та враховуючи, що гравець А прагне мінімізувати середній виграш гравця В, отримаємо задачу лінійного програмування, розв'язання якої дозволить визначити оптимальну стратегію гравця А:

$$\frac{1}{w} = \sum_{i=1}^{m} p_i \Rightarrow Max,\ \sum_{i=1}^{m} b_{ij}p_i \le 1,\ p_i \ge 0,\ j=\overline{1,n}. \qquad (12.8)$$

Розглянемо докладніше випадок $n=m=2$. Тоді матриця виплат гри має вигляд:

$$\begin{array}{cccl} & B_1 & B_2 & \\ A_1 & (a_{11}, b_{11}) & (a_{12}, b_{12}) & x_1 \\ A_2 & (a_{21}, b_{21}) & (a_{22}, b_{22}) & x_2 = 1 - x_1 \\ & y_1 & y_2 & \end{array} \qquad (12.9)$$

Відобразимо геометрично максимінну стратегію і стратегію загрози першого гравця. Почнемо з максимінної стратегії. Нехай перший гравець обирає стратегію A_1 з ймовірністю x_1, $x_2 = 1 - x_1$. За умови вибору гравцем В стратегії B_1 середній виграш першого гравця становитиме $a_{11}x_1 + a_{21}(1-x_1)$.

За умови вибору гравцем В стратегії B_2 середній виграш першого першого гравця становитиме $a_{12}x_1 + a_{22}(1-x_1)$. Розв'язуючи ці рівняння, $a_{11}x_1 + a_{21}(1-x_1) = a_{12}x_1 + a_{22}(1-x_1) = v$, отримаємо оптимальне значення середнього виграшу гравця А у випадку застосування ним своєї максимінної стратегії. Аналогічні рівняння можна виписати також і для випадку стратегії загрози.

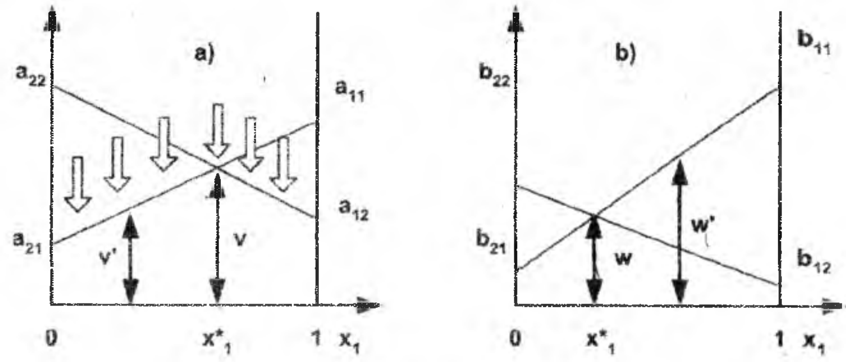


Рис 12.3. Геометрична ілюстрація максимінної стратегії та стратегії загрози

У випадку максимінної стратегії гравець А гарантує собі виграш, не менший за v (рис. 12.3а), але при цьому згоджується з тим, що припускає і більший виграш гравця В порівняно з застосуванням стратегії загрози. (значення w' на рис. 12.3b).

І навпаки, діючи з позиції стратегії загрози, гравець А гарантує для гравця В отримання мінімального виграшу, але разом з тим і зменшує свій середній виграш порівняно з максимінною стратегією (відомий підхід з точки зору „нехай мені буде гірше, але щоб у сусіда було найгірше").

Кооперативна гра двох осіб. Переговорна множина.

Розглянемо гру з наступною матрицею виплат:

$$\begin{array}{ccc} & y & 1-y \\ x & (2;1) & (-1;-1) \\ 1-x & (-1;-1) & (1;2) \end{array}.$$

Нехай гравець А використовує мішану стратегію $(x;1-x)$, В - стратегію $(y;1-y)$. Тоді середні виграші гравців становитимуть:

$$v = 2xy - 1x(1-y) - 1y(1-x) + 1(1-x)(1-y),$$

$$w = 1xy - 1x(1-y) - 1y(1-x) + 2(1-x)(1-y)$$

Таким чином, ми побудували відображення x,y в v,w. $0 \le x \le 1, 0 \le y \le 1$. Відображаючи всі можливі варіанти x,y в простір v,w, отримаємо наступну фігуру (рис 12.4), обмежену прямими, що проходять через пари точок $(-1;-1)$, $(1,2)$ і $(-1;-1)$, $(2,1)$, а також шматком параболи $5(v-w)^2 = 2(v+w)-1$. В ній є "провал", обмежений саме цією параболою.

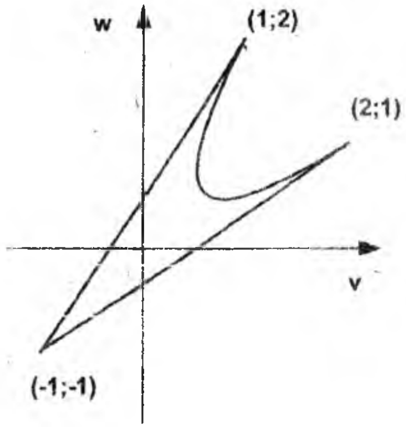


Рис 12.4. Відображення прикладу кооперативної гри за умови повного суперництва

А зараз повернемося до загального випадку гри двох осіб з матрицею виплат

$$\begin{pmatrix} (a_{11}, b_{11}) & (a_{12}, b_{12}) & \dots & (a_{1n}, b_{1n}) \\ (a_{21}, b_{21}) & (a_{22}, b_{22}) & \dots & (a_{2n}, b_{2n}) \\ \dots & \dots & \dots & \dots \\ (a_{m1}, b_{m1}) & (a_{m2}, b_{m2}) & \dots & (a_{mn}, b_{mn}) \end{pmatrix} \qquad (12.10)$$

і припустимо, що гравці мають нагоду домовлятися про сумісні дії. В чому полягатимуть ці сумісні дії?

Раніше стратегія A_i першого гравця обиралася з вірогідністю x_i, стратегія B_j другого гравця з вірогідністю y_j, і стратегії обох гравців були незалежні, так що комбінація (A_i, B_j) з'являлася з вірогідністю $x_i y_j$. Зараз

ходи обираються спільно і тому комбінація стратегій (A_i, B_j) з'являється з деякою сумісною вірогідністю p_{ij}. Сумісна гра зводиться, таким чином, до вибору сумісної мішаної стратегії p_{ij}. При цьому, очевидно:

$$\forall (i,j): p_{ij} \ge 0 \wedge \sum_{i=1}^{m} \sum_{j=1}^{n} p_{ij} = 1. \qquad (12.11)$$

При такій сумісній мішаній стратегії середні виграші першого і другого гравців рівні відповідно:

$$v = \sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij} p_{ij}, \quad w = \sum_{i=1}^{m} \sum_{j=1}^{n} b_{ij} p_{ij}. \qquad (12.12)$$

Представимо собі множину (v, w). Яку область заповнять значення, що отримані з наведених формул? Виявляється, що ця область є опуклою оболонкою образів точок з координатами (a_{ij}, b_{ij}).

Так, для наведеного вище прикладу гри область R є опукла оболонка точок (-1, -1), (2, 1) і (1, 2) (див. мал. 12.5).

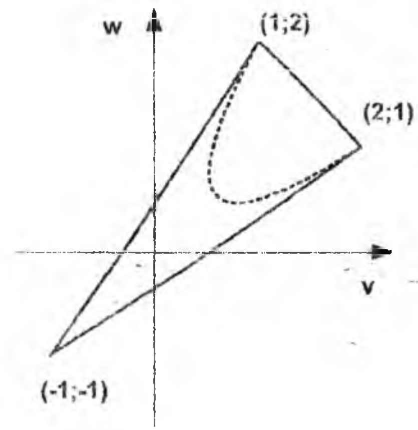


Рис.12.5. Відображення прикладу кооперативної гри за умови співробітництва

Порівняємо цю область з тією, яка зображена на рис. 12.4. Ми бачимо, що застосування сумісних стратегій дозволило заповнити ту "западину", яка була при некооперативній грі.

Про що ж тепер можуть домовитися гравці? Нехай v^* і w^* - максимінні виграші першого і другого гравців відповідно (Рис.12.6). Нанесемо на нашу множину R точку з координатами точкою (v^*, w^*). Ця точка називається точкою **status quo.** Очевидно, що жоден з гравців не погодиться одержувати в результаті сумісної гри менше ніж дає йому максимінна стратегія - навіщо йому така домовленість, якщо він може гарантувати собі без жодних домовленостей v^* або w^*.

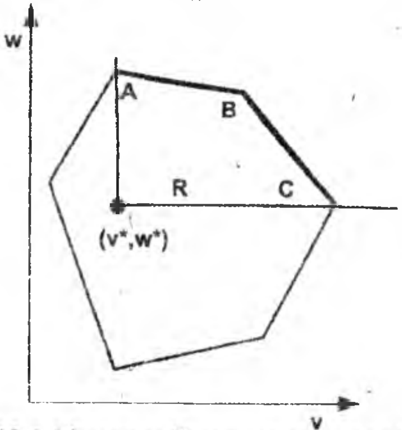


Рис 12.6. Множина Парето-оптимальних розв'язків гри

Тому з нашої множини розв'язків R відразу зникає область, де $v < v^*$ та $w < w^*$.

Розглянемо область, що тепер залишилася і визначається сумісною дією обмежень $(v \ge v^*) \wedge (w \ge w^*)$. Задачу можна розглядати як двокритерійну з критеріями (v, w). Як нескладно визначити, множину Парето-оптимальних (недомінованих) розв'язків утворюють всі точки, що знаходяться на ламаній А-В-С.

Очевидно, що якщо точка (v, w) домінується точкою (v', w'), то в процесі торгівлі гравці безболісно відмовляться від точки (v, w) на користь точки (v', w'), оскільки при такому переході хоча б одному стає краще, а іншому - не гірше. Очевидно також, що точки, які домінують точку (v, w), знаходяться правіше і вище на множині R. Тому множина Парето-оптимальних розв'язків називається також переговорною множиною для гравців А та В. Переговорна множина на рис 12.5.- це відрізок прямої, що сполучає точки (1; 2) і (2; 1).

До чого гравці домовляться - сказати наперед не можна, оскільки на цій множині інтереси гравців прямо протилежні. Результат залежить від уміння вести переговори і лежить за межами математичного дослідження.

Отже, в певному значенні, вирішити біматричну кооперативну гру двох осіб означає побудувати переговорну множину.

Схема алгоритму побудови переговорної множини.

Крок 1. Побудова образу всіх можливих мішаних стратегій x, y, $x = (x_1, \ldots, x_m)$,

$$x_i \ge 0,\ \sum_{i=1}^{m} x_i = 1,\ \overline{i = 1, m},\ y = (y_1, \ldots, y_n)\ y_j \ge 0,\ \sum_{j=1}^{n} y_j = 1,\ \overline{j = 1, n}$$

в просторі (v, w).

Крок 2. Побудова опуклої оболонки образу припустимих стратегій в просторі (v, w).

Крок 3. Знаходження максимінних виграшів кожного з гравців - точки status quo.

Крок 4. Побудова множини Парето-оптимальних розв'язків як „північно-східної границі" отриманого образу в просторі (v, w), для якої виконуються умови $(v \ge v^*) \wedge (w \ge w^*)$.

Слід відзначити, що конкретні реалізації алгоритмів для біматричної гри з числом стратегій, що більше 2, є достатньо складними.

Розглянемо ще декілька прикладів, що пояснюють різницю між станами рівноваги та Парето-оптимальними рішеннями в біматричних іграх.

До нерегульованого перехрестя на високій швидкості з різних доріг наближаються 2 автомобілі. В кожного водія є 2 стратегії: Б – безпечна – зменшити швидкість до безпечної; Р – ризикована – продовжувати їхати на високій швидкості. Ситуація, за якої 2 водії обирають безпечну стратегію, оцінюється для кожного водія в 1; якщо обидва водії обирають ризиковану стратегію, то вони зіштовхуються, наслідки якого оцінюються для кожного від'ємним значенням -9; якщо ж один з них поступиться, знизивши швидкість, в той час як інший продовжує їхати на високій швидкості, то такий результат оцінюється числом 0 для того, хто зменшив швидкість (за „втрату престижу"), а для іншого водія – значенням 3 (за „підвищення престижу").

Таким чином, отримуємо біматричну гру розміру 2×2, в якій водії є гравцями, а стратегії Б – безпечна, Р – ризикована.

$$\begin{array}{c c} & \begin{array}{cc} Б & Р \end{array} \\ \begin{array}{c} Б \\ Р \end{array} & \begin{pmatrix} (1;1) & (0;3) \\ (3;0) & (-9;-9) \end{pmatrix} \end{array}$$

Проаналізуємо ситуації цієї гри з точки зору стійкості. Ситуація $(Б;Б)$ є нестійкою, оскільки в цьому випадку гравець А може отримати кращий для себе результат, змінивши стратегію Б на стратегію Р – він отримає виграш 3 замість 1. Те ж справедливе в ситуації $(Б;Б)$ і для гравця В. Точно так само нестійкою є ситуація $(P;P)$.

Таким чином, нестійкість якоїсь ситуації виявляється в тому, що у випадку її виникнення їй загрожує розпад, обумовлений тим, що з'являються можливості в одного з гравців отримати кращий для себе результат шляхом однобічної зміни своєї стратегії.

Натомість ситуації $(Б;P)$ та $(P;Б)$ дві є стійкими: якщо вони виникли, то ні у кого з гравців немає підстав для зміни однобічно своєї стратегії. Кожен водій, дізнавшись про стратегію іншого, буде дотримуватись своєї обраної. Такі стійкі ситуації називаються ситуаціями рівноваги за Нешем (за прізвищем американського математика Джона Неша). В загальному випадку для біматричної гри рівноважність ситуації (A_{i_0}, B_{j_0}) означає, що для всіх $i=\overline{1,m}$, $j=\overline{1,n}$, $a_{ij_0} \le a_{i_0 j_0}$, $b_{i_0 j} \le b_{i_0 j_0}$. Рівноважні ситуації можна розглядати як оптимальні сумісні рішення, і оптимальність рівноважної ситуації виявляється в її стосунку щодо стійкості.

Однак природно виникає інше запитання: а наскільки добрими є рівноважні ситуації в іншому, насамперед, стосовно виграшів гравців? Кожен з гравців може розглядати вибір своєї стратегії як прийняття рішень в умовах невизначеності – і обрати, наприклад, рішення в результаті застосування максимінної стратегії – згідно з принципом гарантованого результату, це гарантуватиме йому максимальний виграш за найгірших умов. Чи не отримає він у цьому випадку виграш, більший ніж в ситуації рівноваги. Виявляється, що ні, і в цьому легко впевнитися, порівнявши результати на рис. 12.4 та 12.5.

Аналіз деяких інших задач дозволяє зробити також висновок про те, що між оптимальністю за Нешем та оптимальністю за Парето існує певна суперечність – ситуація може бути оптимальною за Нешем, але неоптимальною за Парето, і навпаки – оптимальна за Парето ситуація може бути неоптимальною за Нешем. Будь-яка Парето-оптимальна ситуація, оскільки є неполіпшуваною для всіх гравців відразу, є, таким чином, максимально вигідною для коаліції, до складу якої входять всі гравці, однак вона може виявитися невигідною для одного (чи декількох) гравців. Саме тому вибір гравцями Парето-оптимального розв'язку передбачає їх взаємодію (наприклад, обмін інформацією про рішення, що приймаються), в результаті чого колективний інтерес коаліції всіх гравців ставиться вище,

ніж інтерес окремого гравця. Якщо ж гравці обирають свої стратегії без взаємодії, керуючись лише приватними інтересами, то можна розраховувати лише на вибір ними ситуації, оптимальної за Нешем.

Якщо скористатися принципом децентралізації, то для децентралізованої системи природним є принцип оптимальності за Нешем, а для централізованої – за Парето.

При зовнішній парадоксальності суперечності між оптимальністю за Нешем та за Парето вони мають різні „ідейні основи" – підґрунтям для оптимальності за Парето є вигідність для системи загалом, а ґрунтом оптимальності за Нешем – стійкість системи, обумовлена інтересами та можливостями її окремих підсистем. Суперечність між оптимальністю за Парето та оптимальністю за Нешем є суперечністю між вигідністю та стійкістю.

Таким чином, на основі цього невеликого аналізу можна зробити наступні висновки:

☑ існування ситуацій рівноваги в безкоаліційних іграх не визначає, взагалі кажучи, їх розв'язків, і однозначні рекомендації для оперуючих сторін відсутні;

☑ в багатьох випадках корисні (і навіть необхідні) контакти та угоди між гравцями, а тому моделі, що дозволяють кооперування, мають переваги;

☑ часткові постановки не виключають можливості використання теорії безкоаліційних ігор, і питання про пошук ситуацій рівноваги повинно досліджуватися окремо в кожному окремому випадку.

Перехід конфліктуючих сторін до різних форм співробітництва (кооперування) створює якісно нові ситуації. В іграх з n учасниками розглядається три рівні взаємодії:

☑ обмін інформацією про перебіг гри та ситуації, що складаються;

☑ сумісний вибір стратегій на ґрунті загальної домовленості та взаємної інформованості;

☑ об'єднання активних засобів (ресурсів) з відповідною координацією дій.

Кожен ступінь кооперування базується на передачі якихось даних з боку одних учасників гри до інших. Умови нормального розвитку гри можуть бути порушені за рахунок передавання спотвореної інформації – обману суперника, який має неповну інформацію.

В більшості випадків вважаємо, що інформація, якою обмінюються учасники конфлікту, має об'єктивну вартість. До поширеного виду колективних дій належить випрацювання єдиного критерію коаліції, після чого вона може розглядатися як одна оперуюча сторона.

Важливим фактором, який визначає характер кооперування, є побічні виплати – "вступний внесок", "податок на кооперацію", "штраф за вихід з кооперації". В цих випадках йдеться про зміну виграшів в той чи інший бік порівняно з початковими умовами гри, а тому побічні виплати перетворюються в частину стратегій, що застосовуються, та впливають на результат конфлікту.

12.3. ПРИЙНЯТТЯ РІШЕНЬ В УМОВАХ НЕВИЗНАЧЕНОСТІ

Задачі прийняття рішень в умовах невизначеності близькі за ідеями та методами до теорії ігор, основною відмінністю є відсутність конфліктного забарвлення – ніхто нікому не протидіє, але наявний елемент невизначеності.

Таким чином, невідомі умови операції залежать не від свідомого суперника, а від об'єктивної реальності – "природи", яка є байдужою інстанцією. Поведінка природи невідома, але не протидіюча. Зовнішньо при наявності стратегій гравця та природи гра з природою представляється матрицею, але відсутність протидії робить ситуацію якісно іншою.

Найпростішим випадком є такий, коли одна зі стратегій гравця А домінує всі інші його стратегії – зрозуміло, що вона буде найкращою. Але не природи – гравця П (йому все одно, яку стратегію обирати). Таким чином, вибір можна звузити, виключивши з розгляду всі доміновані та еквівалентні стратегії гравця А.

Окрім того, бажано ввести такі показники, які б не просто давали виграш при даній стратегії в кожній ситуації, але й відображали б "вдалість" або "невдалість" вибору тієї чи іншої стратегії в конкретній ситуації. З цією метою вводиться поняття ризику, як різниці між виграшем, який можна було б отримати, якщо б ми знали умови природи – її стратегію $П_j$, та виграшем, який ми отримаємо, не знаючи їх та обираючи стратегію $А_i$,

$$r_{ij} = \beta_j - a_{ij}, \text{ де } \beta_j = \underset{i}{Max}\ a_{ij}. \tag{12.13}$$

Ризик — це, по суті, плата за відсутність інформації, і тому, звичайно, бажано би було мінімізувати ризик, що супроводжує вибір рішення. Таким чином, маємо 2 постановки задачі – в одній необхідно отримати максимальний виграш, а в іншій – мінімізувати ризик.

Найпростішим випадком невизначеності є "доброякісна" стохастична невизначеність. В цьому випадку стани природи характеризуються вірогідностями їх виникнення $p = (p_1, p_2, ..., p_n)$, і оптимальною стратегією $А_k$ буде та, для якої

$$k = \arg \underset{i \in 1,m}{Max} \sum_{j=1}^{n} a_{ij} p_j \tag{12.14}$$

(критерій $Q = \sum_{j=1}^{n} a_{ij} p_j \Rightarrow \underset{i \in 1,m}{Max}$). Відповідний критерій для ризику у випадку стохастичної невизначеності $R = \sum_{j=1}^{n} r_{ij} p_j \Rightarrow \underset{i \in 1,m}{Min}$.

Нехай тепер ймовірності природи існують, але невідомі нам.

Згідно з критерієм Лапласа, всі стани природи вважаються рівноймовірними. Однак застосовувати його в більшості випадків не рекомендується, оскільки в багатьох випадках апріорі більш-менш відомо, як відрізняються вірогідності. В цьому випадку, якщо є можливість, необхідно провести експертне опитування, або ж спробувати накопичити інформацію в результаті проведення декількох ігор з природою.

Якщо ж невизначеність "погана", якщо вірогідностей природи взагалі не існує, або ж вони не піддаються навіть приблизній оцінці, то в залежності від позиції дослідника застосовуються наступні критерії.

Максимінний критерій Вальда В. Згідно з цим критерієм, гра з природою ведеться як гра з агресивним та розумним суперником, і обирається стратегія з індексом k, для якої

$$k = \arg \underset{i}{Max} \left(\underset{j}{Min}\ a_{ij} \right). \tag{12.15}$$

Це є позиція крайнього песимізму, і стосовно природи є перестраховальною.

Критерій мінімального ризику Севіджа С. Цей критерій є теж вкрай песимістичним, але при виборі оптимальної стратегії орієнтує на мінімальний ризик. В якості оптимальної стратегії обирається така з індексом k, для якої величина ризику в найгірших умовах мінімальна – .

$$k = \arg \underset{i}{Min} \left(\underset{j}{Max}\ r_{ij} \right). \tag{12.16}$$

Критерій песимізму-оптимізму Гурвіца Н. Цей критерій рекомендує при виборі розв'язку не орієнтуватися ні на песимізм, ані на оптимізм, і має вигляд:

$$H = \lambda \times \underset{j}{Min}\ a_{ij} + (1-\lambda) \times \underset{j}{Max}\ a_{ij} \Rightarrow \underset{i \in 1,m}{Max}, \tag{12.17}$$

де $0 \le \lambda \le 1$ – коефіцієнт песимізму, коли рівний 1 – Гурвіца, 0 – скрайнього оптимізму.

ПРИКЛАДИ

Приклад 12.1. Побудова матриці гри.

Гра полягає в наступному. З чисел 1,2,3,4,5 випадковим чином (з рівною вірогідністю) обирається якесь одне x, і двом гравцям пропонується вказати верхню межу для обраного числа. Якщо вгадав лише один з гравців, то він отримує одиницю від суперника; якщо ж вгадали обоє, то одиницю від суперника отримує той, значення межі в якого строго менше ніж в іншого. В усіх інших випадках ніхто з гравців не отримує нічого. Гравцям відомо, що число $x \in \{1;2;3;4;5\}$ і що вибір реалізується рівноймовірно.

Розв'язання.

Припустимо, що гравець А називає число i, а гравець В - j. Позначимо середній виграш гравця А в цьому випадку як $K(i,j)$ (середній виграш рахується за умови, що гра повторюється певну кількість разів для названих значень чисел). Нехай випадково обране число рівне x, $x \in \{1;2;3;4;5\}$, $i \in \{1;2;3;4;5\}$, $j \in \{1;2;3;4;5\}$, та $P(x=k)=\frac{1}{5}$, $k=\overline{1,5}$, оскільки число обирається з рівною вірогідністю.

Обчислимо значення математичного сподівання виграшу гравця А при $i<j$. Можливим є лише один з трьох випадків:

$k \le i < j,\ i < k \le j,\ i < j < k$. Якщо справедлива перша нерівність, то гравець А перемагає, тобто з вірогідністю $P(k \le i)$ він отримує одиницю. В другому випадку він програє одиницю, і вірогідність цієї події рівна $P(i < k \le j)$. В третьому випадку нічия і нічого не трапляється. Таким чином, при $i<j$ очікуваний виграш гравця А становить:

$$K(i,j) = 1 \times P(k \le i) - 1 \times P(i < k \le j) = 2 \times P(k \le i) - P(k < j).$$

Аналогічно при $i > j$ його очікуваний виграш становить:

$$K(i,j) = -1 \times P(k \le j) + 1 \times P(j \le k \le i) = -2 \times P(k \le j) + P(k \le i).$$

І нарешті, $K(i,j)=0$ при $i=j$. Оскільки $i \in \{1;2;3;4;5\}$ та $j \in \{1;2;3;4;5\}$, матриця очікуваних виграшів буде наступною:

$$\begin{array}{c|c}
i\backslash j & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\
\hline
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} &
\begin{pmatrix}
0 & 0 & -\frac{1}{5} & -\frac{2}{5} & -\frac{3}{5} \\
0 & 0 & \frac{1}{5} & 0 & -\frac{1}{5} \\
\frac{1}{5} & -\frac{1}{5} & 0 & \frac{2}{5} & \frac{1}{5} \\
\frac{2}{5} & 0 & -\frac{2}{5} & 0 & \frac{3}{5} \\
\frac{3}{5} & \frac{1}{5} & -\frac{1}{5} & -\frac{3}{5} & 0
\end{pmatrix}
\end{array}$$

Приклад 12.2. Біматрична гра.

Нехай біматрична гра задана наступною матрицею. Проаналізувати цю гру.

$$\begin{array}{ccc}
 & B_1 & B_2 \\
A_1 & (5;2) & (0;0) \\
A_2 & (0;0) & (2;5)
\end{array}$$

Розв'язання.

В цій грі чисті стратегії (A_1, B_1) або (A_2, B_2) утворюють ситуації рівноваги (перевіряється безпосередньою підстановкою в формули), тобто відхилення від ситуації рівноваги приводить до зменшення виграшу. Однак перша ситуація значно краща для гравця А (виграє 5 одиниць, в той час як В – 2), в той час як друга (з точністю до навпаки) – для гравця В. В той же час один з гравців, що перебуватиме в гіршій ситуації, може прагнути погіршити її ще більш для себе – наприклад, гравець В, знаходячись в першій ситуації рівноваги, може прагнути змінити стратегію на B_2 – у цьому випадку його виграш дорівнюватиме нулю, але й виграш гравця А теж стане нульовим, і якщо А прагнутиме до виграшу, він може перейти до другої ситуації рівноваги. Таким чином, в безкоаліційній біматричній грі врешті-решт, незважаючи на існування ситуацій рівноваги, невідомо, до якого результату можуть прийти гравці (ситуацій рівноваги – декілька, і вони нерівноцінні для гравців !!!) і як вони повинні діяти. В той же час, якщо припустити можливість співпраці з оплатою іншому гравцеві, то, встановивши оплату для гравця B за рахунок гравця A в першій ситуації рівноваги в розмірі 1,5 одиниці та в іншій таку ж, але для гравця A за рахунок B, ми отримаємо дві еквівалентні стійкі ситуації рівноваги в грі.

Приклад 12.3. Біматрична гра.

Наступна гра, яка має платіжну матрицю

$$\begin{array}{cc} & \begin{array}{cc} y & 1-y \end{array} \\ \begin{array}{c} x \\ 1-x \end{array} & \begin{pmatrix} (2;1) & (-1;-1) \\ (-1;-1) & (1;2) \end{pmatrix} \end{array},$$

є варіантом "**родинної суперечки**". Чоловік (гравець A) і дружина (гравець B) можуть обирати одну з двох вечірніх розваг – футбол (i=1, j=1) або театр (i=2, j=2). Згідно зі звичайним стандартом, чоловік віддає перевагу футболу, а дружина – театру. Проте їм набагато важливіше йти разом, ніж дивитися своє видовище окремо. І якщо вони посваряться і підуть в різні боки (i=1, j=2 або i=2, j=1), то обидва програють, одержуючи (-1, -1).

Розв'язання.

Знайдемо стратегії першого гравця (очевидно, що через симетричність платіжної матриці стратегії другого гравця точно такі ж).

Розглядаючи максимінну стратегію першого гравця, коли він обирає хід i=1 з вірогідністю x, отримаємо, що його виграш буде рівний:

$2\times x-1\times(1-x)$ при $j=1$, $-1\times x+1\times(1-x)$ при $j=2$.

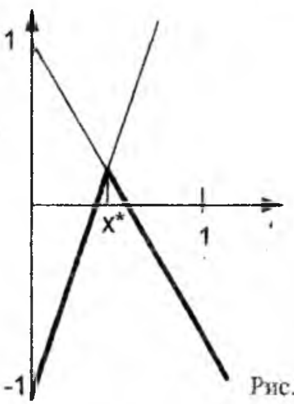


Рис.12.7. Максимінна стратегія першого гравця

Відповідні прямі зображені на рис. 12.7. Значення x^* знаходимо з умови

$$2x^*-(1-x^*)=-x^*+(1-x^*),$$

звідки отримуємо $x^*=2/5$, і мішана стратегія першого гравця є $(2/5;3/5)$ і його гарантований виграш дорівнює:

$$v=2\times\frac{2}{5}-1\times\left(1-\frac{2}{5}\right)=\frac{1}{5}.$$

Застосовуючи стратегію загрози, він рахує виграш другого гравця, який буде рівний $1\times x-1\times(1-x)$ при $j=1$, $-1\times x+2\times(1-x)$ при $j=2$.

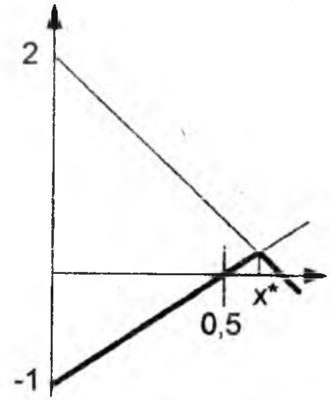


Рис. 12.8. Стратегія загрози

Відповідні прямі зображені на рис. 12. 8. Величина x^* знаходиться з умови:

$1\times x^*-(1-x^*)=-1\times x^*+2\times(1-x^*)$, звідки x*=3/5, так що змішана стратегія першого гравця є (3/5;2/5). При цьому виграш другого гравця буде у будь-якому випадку рівний:

$$w=1\times\frac{3}{5}-\left(1-\frac{3}{5}\right)=\frac{1}{5}.$$

Таким чином, застосовуючи максимінну стратегію, перший гравець може **гарантувати собі** виграш, рівний 1/5; застосовуючи стратегію загрози, він може бути упевнений, що **другий гравець** отримає **не більше** 1/5.

Приклад 12.4. Гра з природою.

Задана матриця гри з природою. Необхідно побудувати матрицю ризиків та обрати найкращу стратегію.

	$П_1$	$П_2$	$П_3$	$П_4$
A_1	1	4	5	9
A_2	3	8	4	3
A_3	4	6	6	2
β	4	8	6	9

Розв'язання.

Матриця ризиків буде мати вигляд:

	$П_1$	$П_2$	$П_3$	$П_4$
A_1	3	4	1	0
A_2	1	0	2	6
A_3	0	2	0	7

Розглядаючи матрицю "ризиків", зрозумілішими стають деякі риси гри з природою. Так, в другому рядку перший та останній елементи матриці виграшів рівні 3. Однак ці виграші нерівноцінні в сенсі вдалого вибору стратегії: при стані природи $П_1$ ми могли виграти найбільш 4 одиниці, і вибір A_2 для цієї ситуації майже добрий. За умови стану $П_4$ природи натомість ми могли б, обравши A_1, отримати на 6 одиниць більше, тобто вибір A_2 для цієї ситуації є поганим. Ризик – це, по суті, плата за відсутність інформації, і тому, звичайно, бажано би було мінімізувати ризик, що супроводжує вибір рішення.

Якщо відомий розподіл вірогідностей, то розраховуємо середні виграші. Для прикладу, якщо $p=(0.5,\ 0.2,\ 0.2,\ 0.1)$, то

$Q(A_1)=0.5\times1+0.2\times4+0.2\times5+0.1\times9=3.2$,

$Q(A_2)=0.5\times3+0.2\times8+0.2\times4+0.1\times3=4.2$,

$Q(A_3)=0.5\times4+0.2\times6+0.2\times6+0.1\times2=3.6$, і оптимальною буде стратегія A2.

Згідно з критерієм Лапласа, всі стани природи вважаються рівноймовірними. Згідно з цим критерієм, в прикладі рівноцінними є стратегії A_1 та A_2.

Приклад 12.5. Прийняття рішень в умовах невизначеності.

Нехай задана матриця гри з природою. Необхідно обрати альтернативи за критеріями Вальда, Гурвіца з λ =0.6, Севіджа.

	$П_1$	$П_2$	$П_3$	$П_4$
A_1	19	30	41	49
A_2	51	38	10	20
A_3	73	18	81	11

Розв'язання.

Розраховуємо значення критеріїв та матрицю ризиків:

	$П_1$	$П_2$	$П_3$	$П_4$	Min	Max	λ =0.6
A_1	19	30	41	49	19	49	31
A_2	51	38	10	20	10	51	26
A_3	73	18	81	11	11	81	38
β	73	38	81	11	19	81	38

Матриця ризиків

	$П_1$	$П_2$	$П_3$	$П_4$	Min
A_1	54	8	40	0	54
A_2	22	0	71	29	71
A_3	0	20	0	38	38

Таким чином, за критерієм Вальда обираємо стратегію A_1, Гурвіца з 0.6, - A_1, Севіджа – A_3.

РЕЗЮМЕ

12.1. Позиційна гра – це природне розширення матричної гри двох гравців з нульовою сумою, в якій може брати участь скінчена кількість гравців, кожен з яких може робити послідовно скінчену кількість ходів, причому деякі з них можуть бути випадковими, а інформація про них може змінюватися від одного до іншого ходу. Такі ігри можуть бути формалізовані, певним чином перетворені до гри, що еквівалентна деякій матричній грі двох осіб з нульовою сумою. Процес приведення позиційної гри до матричної називається нормалізацією, а отримана матрична гра – грою в нормальній формі.

12.2. Один із найважливіших висновків теорії ігор – це те, що певні форми кооперування гравців за умови зовнішньо різних їх прагнень дійсно мають сенс. Це частково пояснюється великою цінністю інформації, яка може бути передана від одного до іншого учасника гри, зростаючим значенням рішень, що приймаються спільно, синергічним ефектом від хоча *б часткового об'єднання ресурсів. Біматрична гра – це найпростіша ігрова модель, що припускає можливість співробітництва. Існують дві різновидності біматричних ігор – безкоаліційні, що забороняють будь-яке співробітництво, та кооперативні, що дозволяють співробітництво. Вибір гравцями Парето-оптимального розв'язку передбачає їх взаємодію (наприклад, обмін інформацією про рішення, що приймаються), в результаті чого колективний інтерес коаліції всіх гравців ставиться вище, ніж інтерес окремого гравця. Якщо ж гравці обирають свої стратегії без взаємодії, керуючись лише приватними інтересами, то можна розраховувати лише на вибір ними ситуації, оптимальної за Нешем. В іграх з n учасниками розглядається три рівні взаємодії: обмін інформацією про перебіг гри та ситуації, що складаються; сумісний вибір стратегій на ґрунті загальної домовленості та взаємної інформованості; об'єднання активних засобів (ресурсів) з відповідною координацією дій.*

12.3. Задачі прийняття рішень в умовах невизначеності близькі за ідеями та методами до теорії ігор, основною відмінністю є відсутність конфліктного забарвлення – ніхто нікому не протидіє, але наявний елемент невизначеності. Невідомі умови операції залежать не від свідомого суперника, а від об'єктивної реальності – "природи", яка є байдужою інстанцією. Ризик – це, по суті, плата за відсутність інформації, і тому, звичайно, бажано би було мінімізувати ризик, що супроводжує вибір рішення. Таким чином, маємо дві постановки задачі – в одній необхідно отримати максимальний виграш, а в іншій – мінімізувати ризик.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 12.1.

Розв'язати гру з прикладу 1.

Завдання 12.2.

Обрати найкращі альтернативи за критеріями Вальда, Севіджа, Гурвіца, Лапласа при значенні коефіцієнту песимізму 0.4 в грі з природою, що задана матрицею:

	$П_1$	$П_2$	$П_3$	$П_4$	$П_5$
A_1	10	5	3	6	12
A_2	3	8	2	9	4
A_3	12	6	1	10	9
A_4	2	4	6	15	3

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Дайте визначення позиційної гри.
2. Поясніть, яким чином позиційна гра зводиться до матричної.
3. Що таке інформаційна множина та який її вплив на ціну гри?
4. Дайте визначення біматричної гри та поясніть її особливості.
5. Які стратегії поведінки гравців розглядаються в біматричній грі та в чому вони полягають?
6. Що таке переговорна множина та як вона шукається?
7. Які рівні взаємодії існують в грі n осіб?
8. Які особливості прийняття рішень в умовах невизначеності?
9. Дайте визначення поняття ризик та розкрийте його змістовно.
10. За якими критеріями приймаються рішення в умовах невизначеності та які їх особливості?

РОЗДІЛ 6

МОДЕЛІ МАСОВОГО ОБСЛУГОВУВАННЯ В ДОСЛІДЖЕННІ ОПЕРАЦІЙ

ТЕМА 13. ОСНОВНІ ПОНЯТТЯ ТА ВИДИ СИСТЕМ МАСОВОГО ОБСЛУГОВУВАННЯ

Процеси утворення черг або затримок в обслуговуванні, які широко розповсюджені, ефективно аналізуються методами дослідження операцій. Практичне застосування моделей масового обслуговування економічно вигідне при розв'язуванні задач проектування й експлуатації систем, що складаються з великого числа тотожних або подібних елементів. Системи масового обслуговування, подані кількісними моделями, видаються на фоні реальних ситуацій сильно спрощеними. Але відносно прості моделі можуть бути використані для одержання якісного або наближеного кількісного уявлення про поведінку систем, що мають складнішу структуру. Основну увагу звертають на операційні характеристики моделей, до яких належать: середня довжина черги, середній час очікування на обслуговування, вірогідність того, що всі компоненти обслуговуючої системи виявляться зайнятими, а також інші показники функціональної ефективності системи. Після оцінювання цих характеристик можна переходити до побудови відповідної економічної моделі і до наступних процедур пошуку оптимальних керуючих рішень. У більшості випадків для розв'язання кожної конкретної задачі застосовується метод оптимізації з вузькою цільовою настановою, а якщо ж система виявляється занадто складною, застосовуються методи імітаційного моделювання.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ:

знати: ⇨структуру математичної моделі системи масового обслуговування (СМО); ⇨основні характеристики якості обслуговування в СМО; ⇨задачі, що розв'язуються при дослідженні СМО; ⇨класифікацію СМО;

вміти: ⇨будувати математичні моделі СМО для практичних ситуацій; ⇨аналізувати СМО з точки зору їх операційних характеристик; ⇨визначати кількісні характеристики вхідного потоку вимог та тривалості обслуговування.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ система масового обслуговування
- ☑ експоненційний закон розподілу
- ☑ середній час очікування на обслуговування
- ☑ середня довжина черги
- ☑ канал
- ☑ процес відновлення
- ☑ модель
- ☑ самообслуговування
- ☑ вірогідність втрати
- ☑ вимоги
- ☑ завантаження обслуговуючих апаратів
- ☑ вхідний потік вимог
- ☑ послідовність рекурентних подій
- ☑ модель чистої загибелі
- ☑ черга на обслуговування
- ☑ пуасонівський потік
- ☑ післядія
- ☑ обслуговуюча система
- ☑ ординарність
- ☑ вартісна модель СМО
- ☑ дисципліна черги
- ☑ стаціонарність
- ☑ механізм обслуговування

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

13.1.Основні поняття та класифікація систем масового обслуговування (СМО).

13.1.1. Області застосування систем масового обслуговування.

13.1.2. Особливості математичних моделей та операційні характеристики СМО.

13.1.3. Структура математичної моделі та класифікація СМО.

13.1.4. Характеристики якості та проблеми аналізу СМО.

13.2.Характеристики вхідного потоку вимог.

13.2.1. Найпростіший потік вимог.

13.2.2. Процеси чистого народження.

13.3.Розподіли вірогідностей для тривалостей обслуговування.

13.3.1. Експоненційний розподіл часу обслуговування.

13.3.2. Модель чистої загибелі.

13.3.3. Модель самообслуговування.

13.4. Термінологія і позначення систем масового обслуговування.

13.5. Функціонування СМО як марківський випадковий процес.

13.1. ОСНОВНІ ПОНЯТТЯ ТА КЛАСИФІКАЦІЯ СИСТЕМ МАСОВОГО ОБСЛУГОВУВАННЯ

13.1.1. ОБЛАСТІ ЗАСТОСУВАННЯ СИСТЕМ МАСОВОГО ОБСЛУГОВУВАННЯ

Основи теорії систем масового обслуговування були закладені в працях датського математика, співробітника Копенгагенської телефонної компанії А. К. Ерланга і отримали широкий розвиток у подальших дослідженнях.

Системи масового обслуговування (СМО) зустрічаються повсюди, і це пояснюється широкою розповсюдженістю черг. Черги виникають у кав'ярні, магазині, перукарні, бібліотеці, на бензозаправній станції тощо. До числа менш очевидних прикладів належать такі ситуації, коли доводиться затримуватися перед світлофором, очікувати одержання довідки по телефону або, скажімо, чекати на прибуття ранкової пошти. Для всіх цих ситуацій характерним є наявність індивідуумів або об'єктів, що потребують обслуговування, і виникнення затримок у тих випадках, коли обслуговуючий апарат зайнятий.

Такі процеси утворення черг або затримок в обслуговуванні ефективно аналізуються методами дослідження операцій. Проте витрати, пов'язані з науковим аналізом тієї чи іншої практичної задачі масового обслуговування, вважаються (як і в будь-якій іншій галузі організаційного управління) виправданими лише за умови, що економічні наслідки керуючих рішень в сфері, яка аналізується, мають істотний вплив. Як показує досвід, практичне застосування моделей масового обслуговування є економічно вигідним при розв'язуванні двох типів задач, між якими не можна провести чіткої межі, так що можуть існувати різноманітні проміжні варіанти.

До першого типу належать задачі проектування й експлуатації систем, що складаються з великого числа тотожних або подібних елементів. Як приклади, що ілюструють характер таких задач, можна розглядати:

- ☑ задачу визначення кількості касових апаратів у кожному з продовольчих магазинів, що належать фірмі, яка має розгалужену торгову мережу;
- ☑ задачу визначення кількості бензоколонок і чисельності обслуговуючого персоналу на кожній бензозаправній станції великої нафтової компанії ;
- ☑ задачу визначення кількості магістральних ліній зв'язку на кожній місцевій автоматичній телефонній станції.

Незважаючи на те, що умови функціонування різноманітних підсистем великої операційної системи масового обслуговування можуть виявитися неоднаковими, при аналізі, орієнтованому на оптимізацію кількісних показників, що належать до різноманітних однотипних компонентів системи (таких, як кількість вузлів обслуговування, чисельність обслуговуючого персоналу і т.п.), можна використовувати цілком *ідентичні процедури.* Отже, розроблені одноразово методологію дослідження і методи розв'язання задачі можна застосовувати багаторазово, тому що в кожному конкретному випадку фірмі потрібно лише врахувати відповідні чисельні значення параметрів, які фігурують у моделі, що використовується.

До другого типу належать задачі параметричної оптимізації окремої системи масового обслуговування. Наприклад, визначення кількості і вантажопідйомальності швидкісних ліфтів у багатоповерховому будинку, що проектується для адміністративних підрозділів фірми, задачу пошуку оптимального комплекту устаткування для великого сталеливарного заводу, задачу визначення кількості і габаритних характеристик злітно-посадкових смуг у великому аеропорту і т.п.

Разом з тим існують задачі, у яких сполучаються елементи й особливості як систем першого, так і систем другого типу.

13.1.2. ОСОБЛИВОСТІ МАТЕМАТИЧНИХ МОДЕЛЕЙ ТА ОПЕРАЦІЙНІ ХАРАКТЕРИСТИКИ СМО

В теорії СМО розглядаються ***моделі масового обслуговування, що піддаються кількісному аналізу.*** Системи масового обслуговування, подані цими моделями, видаються на фоні реальних ситуацій сильно спрощеними. Але ***відносно прості*** моделі можуть бути використані і для одержання ***якісного*** або ***наближеного кількісного*** уявлення про поведінку систем, що мають складнішу структуру.

Науковий аналіз процесів масового обслуговування в багатьох випадках є складним, тому що при оцінці впливів на режим функціонування системи таких показників, як частота надходження вимог на обслуговування, час обслуговування вимог, що надходять, кількість і розміщення різноманітних компонентів обслуговуючого комплексу і т.д., далеко не завжди можна покладатися на одну лише інтуїцію.

Основну увагу звертатимемо на ***операційні характеристики*** моделей. До них характеристик належать: середня довжина черги, середній час очікування на обслуговування, вірогідність того, що всі компоненти обслуговуючої системи виявляться зайнятими, а також інші показники функціональної ефективності системи. Після оцінювання цих характеристик можна переходити до побудови відповідної економічної моделі і до наступних процедур пошуку оптимальних керуючих рішень.

Ступінь складності задачі оптимізації залежить від структурних особливостей самої системи масового обслуговування і від того, наскільки широкий є діапазон альтернатив, які ми маємо намір проаналізувати. Так, наприклад, якщо потрібно обрати один із двох конкретних варіантів розв'язків, що визначають число касових апаратів у супермаркеті, то оптимальний розв'язок знаходиться за допомогою простого порівняння кількісних характеристик кожного з розглянутих варіантів. Проте, якщо мова йде, скажемо, про розробку системи керування повітряним рухом для центрального аеропорту, то для розв'язання задачі може знадобитися складніший метод оптимізації в порівнянні з методом, що полягає в розгляді кожного з припустимих варіантів, тому що число можливих варіантів у цьому випадку може виявитися безмежно великим.

На даний час не існує єдиного підходу до розв'язання задач оптимізації в сфері масового обслуговування. У більшості випадків для розв'язання кожної конкретної задачі застосовується метод оптимізації з вузькою цільовою настановою (тобто метод, придатний для розв'язання лише даного класу задач). Якщо ж система виявляється занадто складною, застосовуються методи імітаційного моделювання.

13.1.3. СТРУКТУРА МАТЕМАТИЧНОЇ МОДЕЛІ ТА КЛАСИФІКАЦІЯ СМО

Математична модель системи масового обслуговування (СМО) включає наступні основні елементи: потік вимог, що надходять на вхід системи (вхідний потік); чергу, що складається з вимог, які очікують на обслуговування; систему обслуговування; вихідні потоки обслужених, втрачених вимог та вимог, що надходять на повторне обслуговування; характеристики якості системи; механізм (дисципліну) обслуговування (рис. 13.1).

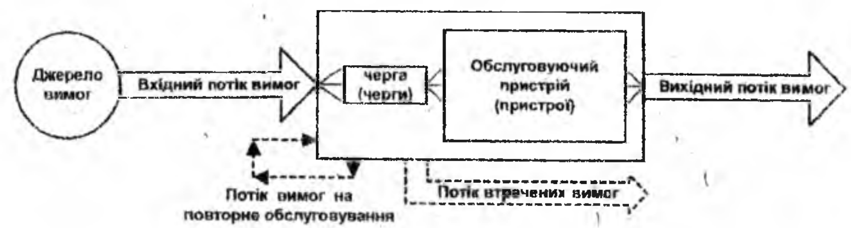


Рис.13.1. Структура СМО

Вхідний потік. Для описання вхідного потоку потрібно задати закон розподілу вірогідностей, що керує послідовністю моментів надходження вимог на обслуговування і зазначити кількість таких вимог у кожному черговому надходженні. Так, наприклад, вимоги на обслуговування в перукарні або в ресторані можуть надходити в середньому кожні 10 хв. При цьому в умовах перукарні щоразу надходить одинична вимога (клієнти приходять у перукарню по одному), а в умовах ресторану можуть надходити *як одиничні, так і групові вимоги* (відвідувачі можуть входити в ресторан як по одному, так і групами). Системи, у яких вимоги можуть надходити пакетами, що містять більш однієї вимоги, будемо називати *системами з груповим обслуговуванням.*) Тривалості інтервалів між послідовними надходженнями вимог у багатьох випадках практично є статистично незалежними і стаціонарними протягом тривалого періоду часу, хоча, зрозуміло, можливі ситуації і цілком іншого характеру. Діаметрально протилежними за своїм характером є, з одного боку, потоки, у яких моменти надходження вимог строго визначені, і, з іншого боку, потоки, у яких тривалості інтервалів між надходженнями вимог є цілком незалежними.

Джерело, що генерує вимоги, звичайно вважають невичерпним. Як приклад, що ілюструє СМО з джерелом вимог необмеженої потужності, можна навести велику залізничну станцію.

До числа СМО з джерелом вимог обмеженої потужності належать, наприклад, парк верстатів певного підприємства, ремонт яких при їхній несправності виконує спеціальна механічна майстерня.

У деяких випадках при наявності великої черги вимога може *відмовитися від чекання* (тобто в чергу не стає або відмовляється від очікування, простоявши певний час в черзі). У залежності від обставин вона може надійти на вхід обслуговуючої системи пізніше (на повторне обслуговування) або вибути назавжди (втрачена вимога). Так, у оптичному виробництві цілком можливі і дозволяються технологією повернення на попередні етапи (це є типовим явищем при шліфуванні великих лінз). У ряді випадків вимога не може стати в чергу через відсутність вільних місць у блоці чекання (черга з обмеженим числом місць). Таким чином, характеристики вхідного потоку (тобто потоку заявок на обслуговування) частково залежать від стану самої обслуговуючої системи.

Дисципліна черги. Ця характеристика дозволяє описати порядок обслуговування вимог, що надходять на вхід системи. Частіше усього використовується дисципліна черги типу: *першим прийшов - першим обслуговуєшся*. Такий порядок обслуговування з погляду математичного моделювання є найпростішим; слід також зауважити, що він стосується лише таких ситуацій, коли вимоги в чеканні обслуговування вибудовуються *в послідовну чергу*. Можливі численні види дисципліни черги, що відрізняються від згаданої вище. Іноді використовується дисципліна *„прийшов останнім - обслуговуєшся першим"* – це звичайний стек.

У деяких випадках порядок обслуговування є фактично *випадковим*. Він часто практикується, наприклад, шкільними вчителями при опитуванні учнів. Іноді дисципліна черги будується за деякою системою пріоритетів (так, наприклад, у випадку, коли приймаються заходи для порятунку пасажирів потопаючого корабля, у рятувальні шлюпки першими садять жінок і дітей). Нарешті, за тими чи іншими міркуваннями клієнт може відмовитися від чекання і прийняти рішення покинути чергу до того, як його встигнуть обслужити (тобто має місце черга з обмеженим часом чекання вимог, що надходять).

Механізм обслуговування. Обслуговуючий механізм характеризується тривалістю процедур обслуговування і кількістю вимог, що обслужені в результаті виконання кожної такої процедури. У наведеному вище прикладі, де мова йшла про обслуговування клієнтів у перукарні або в ресторані, процедура обслуговування вважається завершеною, коли клієнт (а у випадку обслуговування в ресторані, можливо, і ціла група клієнтів) залишає відповідний заклад після надання йому послуг. Тривалість інтервалу часу, необхідного для реалізації процедури обслуговування, частково залежить від запитів клієнта (або групи клієнтів). Але вона може залежати також і від стану самої обслуговуючої системи: так, наприклад, обслуговуючий персонал може форсувати процедури обслуговування, якщо

обслуговування очікує велике число клієнтів. Як і тривалості інтервалів між надходженнями вимог, тривалості обслуговування в кожній з обслуговуючих точок можуть (хоча це і не обов'язково) описуватися за допомогою незалежних випадкових змінних з ідентичними розподілами вірогідностей їхніх чисельних значень. У ряді випадків необхідно також враховувати вірогідність виходу *обслуговуючого приладу* з ладу після закінчення деякого обмеженого інтервалу часу.

Для описання механізму обслуговування потрібно також зазначити кількість і взаємне розташування *обслуговуючих приладів* або *каналів*. Так, наприклад, прилетівши з Нью-Йорка в аеропорт Лондона, пасажир повинен пройти там паспортну перевірку. При цьому всі пасажири вишиковуються в одну лінію і кожен із пасажирів, що дочекалися своєї черги, скеровується до одного з чиновників, що звільнилися і виконують перевірку паспорта (єдина черга). Цілком інша картина спостерігається, наприклад, у години пік на вокзалі біля білетних кас, коли черги утворюються до всіх без винятку касирів. У цьому випадку відвідувач повинен обрати одну з черг і очікувати обслуговування з боку цілком визначеного касира. Якщо Ваш вибір виявився невдалим, перебування в черзі може забрати у Вас навіть більше часу, ніж у деяких із тих, що прийшли після Вас, яким пощастило стати в „швидші" черги. (Якщо черги не занадто великі, можна перейти від обраного спочатку вікна до іншого, біля якого черга стала коротшою).

У кожному з наведених вище прикладів прилади (або канали) *функціонують паралельно*. Існує також множина СМО, у яких прилади розташовані послідовно, так що клієнт змушений переходити від одного приладу до іншого, іноді простоюючи біля кожного з них у черзі. До таких систем належать, наприклад, підприємство з дрібносерійним виробництвом, де для виготовлення партії замовлених виробів вони повинні пройти послідовне опрацювання в ряді цехів.

Ще одна ситуація, для котрої характерно послідовне розташування обслуговуючих приладів, виникає при русі автомобіля по одній з міських магістралей із великою кількістю регульованих перехресть, що викликає кількаразові вимушені зупинки машини перед світлофорами.

СМО класифікуються за різноманітними ознаками в залежності від ⇨характеристик перебування в черзі, ⇨вхідного потоку та ⇨ дисципліни обслуговування вимог.

За складом обслуговуючих пристроїв розрізняють одно (з одним обслуговуючим пристроєм) та багатоканальні (з багатьма обслуговуючими пристроями, що паралельно можуть обслуговувати вимоги) СМО. Якщо для обслуговування в СМО *вимога повинна послідовно пройти через декілька обслуговуючих пристроїв* (фаз обслуговування), *то така система є багатофазною*, якщо ж після проходження одного пристрою обслуговування вимога вважається обслуженою – то *однофазною*.

За характеристиками вхідного потоку розрізняються *системи з пріоритетами* (вимоги з вищим пріоритетом мають переваги як при визначенні місця в черзі, так і при обслуговуванні) та *системи без пріоритетів* (вимоги рівноправні).

В залежності від наявності чи відсутності черги СМО поділяються на *системи без черг* – з відмовами (якщо вимога, що надійшла на вхід СМО, не може бути обслужена, вона покидає систему) та *системи з чергами* (вимога має можливість стати в чергу і очікувати на обслуговування). В залежності від довжини черги СМО можуть бути з *обмеженою чергою* (кількість місць в черзі скінчена) та системи з *необмеженою чергою*. В залежності від часу перебування в черзі розрізняються *СМО з необмеженим часом перебування в черзі* (вимога, що потрапила в чергу, перебуває в ній до моменту початку обслуговування) та *СМО з обмеженим часом очікування* (вимога, що перебуває в черзі, вибуває з неї і виходить з системи після того, як час очікування перевищує певне критичне значення). В залежності від кількості черг системи поділяються на *системи з однією спільною чергою* та *системи з багатьма чергами*. В системах з багатьма чергами розглядаються підкласи систем *без переходів між чергами* та системи з *переходами між чергами*. СМО, з яких можуть вибувати необслужені вимоги, називаються *системами з втратами*, а системи, в яких вибування вимог до завершення обслуговування не передбачене – *без втрат*.

Якщо обслуговуючий пристрій перериває обслуговування вимоги з нижчим пріоритетом, коли надходить вимога з вищим пріоритетом, то СМО є *системою з перериваннями*, якщо ж вимога з вищим пріоритетом очікує моменту найшвидшого звільнення обслуговуючого пристрою – *системою без переривань*. Відмови в роботі обслуговуючого пристрою можна моделювати за допомогою потоку вимог найвищого пріоритету, що переривають обслуговування, і потребують для обслуговування певного часу (час ліквідації аварії – відмови обслуговуючого пристрою).

13.1.4. ХАРАКТЕРИСТИКИ ЯКОСТІ ТА ПРОБЛЕМИ АНАЛІЗУ СМО

До основних характеристик якості обслуговування належать:

1) вірогідність прямої або умовної втрати вимоги;

2) середній час обслуговування вимоги;

3) середня довжина черги;

4) вірогідність втрати вимоги, що надійшла;

5) завантаження обслуговуючих апаратів і ін.

Аналіз систем масового обслуговування. Кожен із можливих варіантів СМО неважко описати точною математичною моделлю, проте це часто майже нічого не дає, якщо оцінювати одержані на основі такого типу описань результати із практичної точки зору. Тому при аналізі СМО в більшості випадків практикується комбіноване застосування наступних двох підходів до розв'язання таких задач. Перший підхід полягає у використанні для наближених описань реальної системи простих математичних моделей. Потім, керуючись результатами аналізу первісних простих моделей і використовуючи ці результати як певний орієнтир, операційник може розробити імітаційну модель, що дозволить за допомогою комп'ютера врахувати ті аспекти задачі, які є істотними, але у той же час важко піддаються аналізу на першому етапі математичного моделювання.

Відповідь на питання про те, які операційні характеристики є найважливішими для формування керуючих рішень, зрозуміло, може бути дана лише з урахуванням конкретних умов задачі. Проте слід зазначити, що операційника, як правило, цікавлять розподіли вірогідностей для числа вимог, що надійшли в систему, і для тривалостей їх очікування, або принаймні середні значення випадкових змінних, що описують ці характеристики на великому відтинку часу. Крім того, іноді потрібно знати вірогідність того, що всі обслуговуючі прилади виявляться вільними або зайнятими; розподіл вірогідностей для тривалості вільних або, навпаки, зайнятих періодів; вірогідність того, що довжина черги (число вимог, що очікують на обслуговування в черзі) перевищить деяке наперед задане число, а також розподіл вірогідностей для інтервалу між послідовними моментами завершення процедур обслуговування. Якщо модель масового обслуговування не дуже складна, то для всіх згаданих вище характеристик вдасться одержати точні, аналітичні вирази, зручні для обчислень.

Операційні методи дослідження СМО орієнтовані на оптимізацію відповідних керуючих рішень. При аналізі прикладів виявляється, що практично в кожній із ситуацій керівник повинен брати до уваги всі три компоненти системи масового обслуговування: вхідний потік вимог на обслуговування, дисципліну черги і механізм обслуговування. Більш того, між різноманітними варіантами керуючих рішень існує ряд складних взаємозв'язків. Наприклад, середню тривалість очікування для вимоги на обслуговування можна зменшити шляхом зміни частоти надходжень вимог на вхід обслуговуючої системи, збільшенням кількості обслуговуючих приладів, використанням більш швидкодіючих приладів, а також шляхом скорочення розкиду часів обслуговування.

Результати дослідження системи обслуговування також можна використовувати для оптимізації моделі з вартісними характеристиками, в якій мінімізується сума витрат, пов'язаних з наданням послуг і втрат, обумовлених затримками в їх наданні. На рис. 13.2 зображена типова вартісна модель системи обслуговування (в грошових одиницях за одиницю часу), де витрати на обслуговування зростають із зростанням його рівня. В той же час втрати, обумовлені затримками в наданні послуг, зменшуються зі зростанням рівня обслуговування.

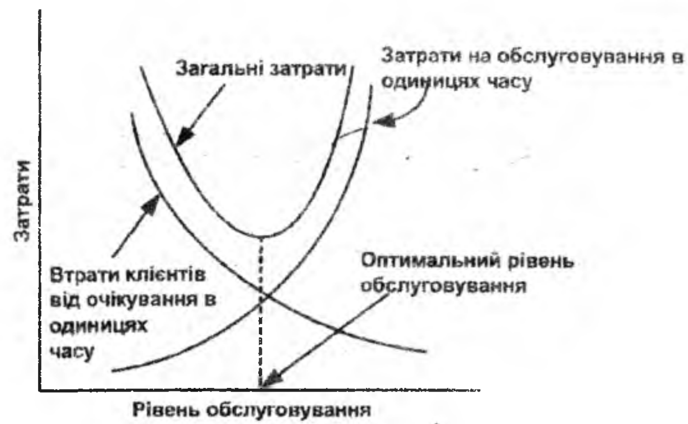


Рис. 13.2. Вартісна модель СМО

Головною проблемою, пов'язаною із застосуванням вартісних моделей, є складність оцінювання втрат в одиницю часу, обумовлених затримками

в наданні послуг. Зокрема, це особливо відчутно, коли послуги надаються індивідууму, чия поведінка може не співпадати з інтересами функціонування системи обслуговування.

При дослідженні СМО розв'язуються одна чи декілька наступних задач:

*1) **задачі аналізу СМО** - визначення характеристик якості обслуговування в залежності від параметрів і властивостей вхідного потоку вимог, параметрів і структури системи обслуговування і дисципліни обслуговування;*

*2) **задачі параметричного синтезу** - визначення параметрів системи обслуговування при її заданій структурі залежно від параметрів і властивостей потоку вимог, дисципліни і якості обслуговування.*

*3) **задачі синтезу структури системи з оптимізацією її параметрів** – необхідно досягти того, щоб при заданих потоках, дисципліні і якості обслуговування вартість СМО була мінімальною або були мінімальними втрати викликів при заданих потоках, дисципліні і вартості системи.*

13.2. ХАРАКТЕРИСТИКИ ВХІДНОГО ПОТОКУ ВИМОГ

13.2.1 НАЙПРОСТІШИЙ ПОТІК ВИМОГ

Починаючи з цього моменту і надалі, ми будемо ототожнювати подію, що описується терміном „надходження вимоги", із фактом прибуття в систему однієї вимоги, що потребує обслуговування. Механізм надходження вимог найзручніше описувати, задаючи розподіл вірогідностей для тривалостей інтервалів часу між послідовними надходженнями вимог на обслуговування.

Припустимо, що тривалості інтервалів між надходженнями вимог статистично незалежні, визначаються тим самим розподілом вірогідностей і описуються деякою неперервною функцією, що є щільністю розподілу. Вхідний потік вимог такого виду є типовим прикладом *процесу відновлення*, а послідовність надходжень є ілюстрацією послідовності *рекурентних подій*.

Нехай $f(t)$ є щільністю розподілу тривалостей інтервалів (t) між будь-якою парою суміжних надходжень (при цьому $t \geq 0$). Визначимо величину $1/\lambda$ як середнє значення тривалості інтервалу часу між надходженнями вимог, так що λ можна інтерпретувати як середнє число надходжень в одиницю часу або як середню частоту надходжень.

Якщо функція $f(t)$ задана, значення λ обчислюється через математичне сподівання $\int\limits_0^\infty tf(t)dt = 1/\lambda$. (13.1)

Так, наприклад, якщо одиницею часу є 1 г., а $\lambda = 4$ є середня кількість надходжень протягом години, то $1/\lambda = 0,25$ од., тобто в середньому протягом кожної чверті години в систему надходить одна додаткова вимога. Аналогічно, якщо кожні 10 хв. у систему надходить у середньому одна вимога, то частота надходження λ рівняється 0,1 вимог у хвилину.

Випадкові надходження (найпростіший, пуасонівський потік вимог). Найважливіший приклад розподілу тривалостей інтервалів між надходженнями вимог відповідає випадку **цілком випадкових надходжень.**

Потік цілком випадкових надходжень вимог має властивості стаціонарності, відсутності післядії і ординарності.

Стаціонарність означає, що з часом характеристики вірогідності потоку не змінюються. Стаціонарність потоку рівносильна постійній щільності вірогідності надходження вимог у будь-який момент часу, інакше кажучи, для стаціонарного потоку вірогідність надходження i вимог за проміжок часу завдовжки Δt залежить лише від величини проміжку Δt і не залежить від його розташування на осі часу, тобто

$$P_i(t, t+\Delta t) = P_i(t_1, t_1+\Delta t) = P_i(0, \Delta t).$$

Реальні потоки вимог можуть бути й нестаціонарними, в такому випадку їх апроксимують стаціонарними на деяких проміжках часу, складових інтервалу часу, в якому досліджується поведінка системи.

Післядія означає незалежність характеристик вірогідності потоку від попередніх подій. Іншими словами, вірогідність надходження і вимог в проміжок $[t_1,t_2]$ в загальному випадку залежить від числа, часу надходження і тривалості обслуговування викликів до моменту t_1. Для випадкового потоку без післядії умовна вірогідність надходження i вимог в проміжку $[t_1,t_2]$, обчислена при будь-яких припущеннях про перебіг процесу обслуговування викликів до моменту t_1, рівна безумовній $P_i([t_1,t_2] | \forall (t<t_1)) = P_i([t_1,t_2])$. Потік вимог, що надходять від достатньо великої групи джерел, близький за своїми властивостями до потоку без післядії, якщо при цьому не враховувати повторних викликів. Потік від малої групи, навпаки, володіє помітною післядією, оскільки число вільних джерел, своєю чергою, залежить від попередніх подій, чим і визначається післядія потоку.

Потік повторних викликів також є прикладом потоку з післядією, оскільки повторний виклик виникає як результат втрати попереднього виклику, тобто наявна залежність від попередніх подій.

Ординарність означає практичну неможливість групового надходження викликів. Інакше кажучи, вірогідність надходження двох або більш викликів за будь-який нескінченно малий проміжок часу є величиною нескінченно малою вищого порядку, ніж одиниці, тобто $P_{i\ge2}(\Delta t) = o(\Delta t)$.

Припущення про **цілком випадковий характер надходжень** приводить до експоненційного розподілу з середнім значенням $1/\lambda$ і дисперсією $1/\lambda^2$

$$f(t) = \lambda e^{-\lambda t},\ t \ge 0. \qquad (13.2)$$

Для перевірки того, що експоненційний є розподілом без післядії (не має пам'яті) – припустимо, що стартовою точкою є точка $t = 0$. Тоді вірогідність відсутності надходжень на інтервалі $(0,T)$ рівна вірогідності того, що перше надходження трапилося після моменту часу T і становить:

$$P\{t \ge T\} = \int\limits_T^\infty \lambda e^{-\lambda t} dt = e^{-\lambda T}. \qquad (13.3)$$

При цьому умовна вірогідність відсутності надходжень на інтервалі $(0, T+h)$ за умови, що не було жодного надходження на інтервалі $(0,T)$ за визначенням становить:

$$P[t \ge h] = \frac{\int\limits_{T+h}^\infty \lambda e^{-\lambda t} dt}{\int\limits_T^\infty \lambda e^{-\lambda t} dt} = \frac{e^{-\lambda(T+h)}}{e^{-\lambda T}} = e^{-\lambda h}, \qquad (13.4)$$

тобто залежить лише від h. Відповідно до цього, вірогідність відсутності надходжень на інтервалі $(T, T+h)$ залишається одною і тією ж незалежно від того, чи відсутні надходження на інтервалі $(0,T)$, чи в момент часу T має місце надходження вимоги і, отже, спостерігається дія поновлення потоку.

Існує й інший спосіб доведення того, що події в експоненційних процесах мають цілком випадковий характер. Викладемо ідею доведення. Нехай на інтервалі $(0,T)$ кількість надходжень становить n. Тоді, якщо тривалості інтервалів між послідовними надходженнями розподілені за експоненційним законом, моменти надходжень розподілені на інтервалі $(0,T)$ взаємонезалежно і рівномірно. Ці міркування можна покласти в основу цілого ряду статистичних випробувань, що дозволяють установити, наскільки адекватно експоненційний розподіл описує реальні процеси формування черги на вході СМО.

Властивості експоненційного розподілу тривалостей інтервалів між послідовними надходженнями стають прозорішими, якщо скористатися розкладанням $e^{-\lambda h}$ у ряд Тейлора: (13.5)

$$P\begin{bmatrix}\text{Відсутність надходжень}\\ \text{в будь-якому інтервалі,}\\ \text{що має тривалість } h\end{bmatrix}=e^{-\lambda h}=1-\lambda h+\frac{(-\lambda h)^2}{2!}+\frac{(-\lambda h)^3}{3!}+\dots$$

Для достатньо малих позитивних значень h складова $1-\lambda h$ у розкладі переверщує за своїм значенням суму інших членів ряду. Отже, для достатньо малих значень h вірогідність, обумовлена попереднім співвідношенням, може бути апроксимована сумою двох перших членів розкладання. Таким чином, для достатньо малих позитивних значень h будемо мати:

$$P\begin{bmatrix}\text{Відсутність надходжень}\\ \text{в будь-якому інтервалі,}\\ \text{що має тривалість } h\end{bmatrix}\approx 1-\lambda h. \quad (13.6)$$

Дуже наочний спосіб обґрунтування полягає в прийнятті такого припущення: на інтервалі достатньо малої тривалості h кількість надходжень не перевищує одиниці (тобто кількість надходжень усередині достатньо малого інтервалу $(T,T+h)$ дорівнює або нулю, або одиниці) – властивість ординарності потоку. Оскільки наближена формула для відсутності надходжень на інтервалі $(T,T+h)$ має попередній вигляд, ми приходимо до такого виразу:

$$P\begin{bmatrix}\text{Одне надходження в будь-якому}\\ \text{інтервалі, що має довжину } h\end{bmatrix}\approx\lambda h. \quad (13.7)$$

Припустимо, наприклад, що λ дорівнює чотирьом надходженням на годину. Тоді можливість відсутності надходжень на інтервалі 0,1 год., відповідно, дорівнює 0,96079, а відповідно до наближеної формули, 1 - 0,04 = 0,96; можливість одного надходження на зазначеному інтервалі дорівнює 0,04.

Якщо щільність розподілу тривалостей інтервалів між надходженнями вимог на обслуговування має експоненційний вигляд, то щільність розподілу повного часу для довільним способом обраного ряду з n послідовних надходжень визначається за гама-розподілом наступною формулою:

$$g(y)=\frac{\lambda(\lambda y)^{n-1}e^{-\lambda y}}{(n-1)!},\ y\ge 0, \quad (13.8)$$

де n - позитивне ціле число. Значення y можна інтепретувати як суму n незалежних вибірок з експоненційного розподілу. Тоді

$$P\begin{bmatrix}\text{Повний час для довільної}\\ \text{послідовності з n надходжень}\\ \text{дорівнює або менше T}\end{bmatrix}=\int_0^T g(y)dy=1-\sum_{j=0}^{n-1}\frac{(\lambda T)^j e^{-\lambda T}}{j!}, \quad (13.9)$$

в чому легко переконатися, якщо вдатися до багаторазового інтегрування по частинах.

Нарешті, слід зазначити, що припущення про експоненційний характер розподілів тривалостей інтервалів між надходженнями рівносильне такому твердженню: розподіл n надходжень у довільно обраному інтервалі тривалістю T є пуасонівським, тобто

$$P\begin{bmatrix}n\text{ надходжень у будь-якому}\\ \text{інтервалі, довжина, якого}\\ \text{дорівнює T}\end{bmatrix}=\frac{(\lambda T)^n e^{-\lambda T}}{n!}. \quad (13.10)$$

Тепер зрозуміло, чому терміни **„експоненційний закон розподілу вимог"** і **„пуасонівський процес"** є синонімами.

Легко переконатися, що з (13.9) та (13.10) наслідком є наступне:

$$P\begin{bmatrix}\text{Повний час для будь-якої}\\ \text{послідовності з n надходжень}\\ \text{дорівнює або менше T}\end{bmatrix}=P\begin{bmatrix}\text{Число надходжень у будь}\\ \text{-якому інтервалі тривалості}\\ \text{T дорівнює або більше n}\end{bmatrix}. \quad (13.11)$$

Ця рівність дійсна не лише для пуасонівського, але і для будь-якого рекурентного процесу формування вхідного потоку за умови, що стартова точка зазначеного інтервалу збігається з одним із моментів надходження.

13.2.2. ПРОЦЕСИ ЧИСТОГО НАРОДЖЕННЯ

Наведені вище співвідношення застосовуються безпосередньо для описання процесів **чистого народження.** Розглянемо систему, у якій у початковий момент 0 вимоги відсутні. Припустимо, що процес надходження (народження) є пуасонівським, і що ці вимоги, які надійшли в систему, на виході не з'являються (тобто залишаються в системі протягом нескінченно великого інтервалу часу). Тоді пуасонівський розподіл визначає можливість того, що в момент T в системі виявиться n вимог, а формула (13.9) є щільністю вірогідності для повного часу надходження перших n вимог.

Наведемо стисле обґрунтування того, що з деяких припущень щодо властивостей процесу надходження як наслідок випливає показниковий розподіл тривалостей інтервалів між надходженнями, і вхідний потік, таким чином, є пуасонівським. Виходитимемо з наступних припущень:

(А) Тривалості інтервалів між послідовними вимогами взаємо незалежні і розподілені ідентичним чином; при цьому можливість надходження вимоги в інтервалі $(T, T+h)$ залежить лише від h і не залежить від T. Щільність вірогідності, що відповідає такому характеру вхідного потоку, позначимо $f(t)$.

(Б) Існує деяка ненульова вірогідність надходження вимог протягом будь-якого інтервалу $h > 0$.

(В) При достатньо малих значеннях h кількість вимог, що надходять, не перевищує одиниці.

Припустимо, що система починає функціонувати, починаючи з моменту 0, причому перше надходження є у момент t $(t > 0)$. Звідси випливає, що $f(t)$ являє собою щільність вірогідності як для тривалості інтервалів між двома надходженнями, так і для фактичного часу появи першої вимоги.

Визначимо $r(T) = 1 - \int_0^T f(t)dt$, так що

$$r(T) = P\begin{bmatrix} \text{Перша вимога надходить} \\ \text{після моменту T} \end{bmatrix}. \quad (13.12)$$

Тоді внаслідок (А) і (Б)

$$r(T+h) = r(T) \times r(h) \quad (13.13)$$

для будь-яких T та $h > 0$. Легко довести, що єдиною функцією, для якої виконується ця умова, є:

$$r(T) = e^{-\lambda T}, \quad (13.14)$$

де λ є константа, причому $\lambda > 0$. Таким чином,

$$e^{-\lambda T} = 1 - \int_0^T f(t)dt \quad , \quad (13.15)$$

звідкіля отримуємо $f(t) = \lambda e^{-\lambda T}$, що і потрібно було довести.

Визначимо $P_n(T) = P\left[n \text{ надходжень в інтервалі } (0,\ T)\right]$.

Нехай $t = x$ є час першого надходження, причому, відповідно до постулату (В), у момент x надходить єдина вимога. Враховуючи постулат (А), ми можемо написати:

$$P_n(T) = \int_0^T P_{n-1}(T-x) f(x)dx = \int_0^T P_{n-1}(y) f(T-y)dy \quad (13.16)$$

для $n = 1, 2, \ldots$, де $y = T - x$. Використовуючи те, що $f(t) = \lambda e^{-\lambda T}$, отримаємо:

$$P_n(T) = \int_0^T P_{n-1}(y)\lambda e^{-\lambda(T-y)}dy. \quad (13.17)$$

Продиференціювавши $P_n(T)$ по T, отримаємо:

$$\frac{dP_n}{dT} = -\lambda P_n(T) + \lambda P_{n-1}(T), \; n = 1, 2, \ldots \quad (13.18)$$

Оскільки $P_0(T) = g(T) = e^{-\lambda T}$, це рівняння можна вирішити методом індукції, починаючи з $n = 1$; при цьому на кожному кроці нам доведеться розв'язувати лінійне диференційне рівняння першого порядку з постійними коефіцієнтами. У результаті ми прийдемо до розв'язку, який є пуасонівським розподілом:

$$P_n(T) = \frac{(\lambda T)^n e^{-\lambda T}}{n!}, \; n = 1, 2, \ldots \quad (13.19)$$

Легко переконатися в тому, що функція $P_n(T)$ дійсно задовільняє рівняння (13.18), для цього достатньо продиференціювати отриманий вираз для пуасонівського закону по T. Таким чином, доведена правильність формули (13.18).

Розподіл Ерланга. Іншим важливим окремим випадком розподілу тривалостей інтервалів між послідовними надходженнями вимог на обслуговування є розподіл Ерланга:

$$f(t)=\frac{(\lambda n)(\lambda nt)^{n-1}e^{-\lambda nt}}{(n-1)!},\quad t\geq 0 \quad ,\tag{13.20}$$

де n - позитивне ціле число. Для математичного сподівання і дисперсії відповідно маємо $M[t]=1/\lambda,\ D[t]=1/(n\lambda^2)$ (цей розподіл позначають як E_n). Виконавши в формулі (13.20) заміну $\lambda n \Rightarrow \lambda$, ми одержимо гама-розподіл.

Варіюючи належним чином значеннями λ та n, ми можемо використовувати розподіл Ерланга в якості доброго наближення розподілів інших видів. При $n=1$ розподіл Ерланга стає тотожним експоненційному розподілу. Відзначимо також, що при $n\to\infty$, коли $D[t]\to 0$, розподіл Ерланга відповідає випадку **регулярного** (або строго періодичного) надходження, тобто випадку, коли тривалості $1/\lambda$ інтервалів між надходженнями однакові і незмінні.

13.3. РОЗПОДІЛИ ВІРОГІДНОСТЕЙ ДЛЯ ТРИВАЛОСТЕЙ ОБСЛУГОВУВАННЯ

13.3.1. ЕКСПОНЕНЦІЙНИЙ РОЗПОДІЛ ЧАСУ ОБСЛУГОВУВАННЯ

Міркування, пов'язані з конкретизацією щільності вірогідності для тривалостей обслуговування, значною мірою схожі з міркуваннями, що дозволяють встановити форму розподілу тривалостей інтервалів між надходженнями вимог у вхідний контур системи. Ми виходимо з припущення, що кожний прилад обслуговування (або канал) може в один і той самий час обслуговувати лише одну вимогу.

Припустимо, що для кожного приладу обслуговування, який підлягає аналізу, інтервали обслуговування сусідніх вимог розподілені незалежним та ідентичним чином і можуть бути описані за допомогою щільності розподілу, що є неперервною функцією. Нехай

$$g(t)\equiv\left[\begin{array}{l}\text{Щільність розподілу ймовірностей}\\ \text{того, що в інтервалі, тривалість якого}\\ \text{дорівнює t, буде обслужена одна}\\ \text{вимога }(t\geq 0)\end{array}\right].\tag{13.21}$$

Нехай також

$$\left[\begin{array}{c}\text{Середній час}\\ \text{обслуговування}\end{array}\right]=\int_0^{\infty} tg(t)dt=\frac{1}{\mu},\tag{13.22}$$

так що

$$\mu=\left[\begin{array}{c}\text{Число вимог, обслужених за одиницю часу,}\\ \text{протягом якої прилад зайнятий обслуговуванням}\end{array}\right].\tag{13.23}$$

Так, наприклад, якщо за одиницю часу приймається 1 год. і якщо $\mu=5$, то протягом 1 год. чинний обслуговуючий прилад встигає обслужити п'ять вимог, і середній час обслуговування однієї вимоги становить $1/\mu=0,2$ год. Аналогічно, якщо на обслуговування однієї вимоги в середньому витрачається 30 хв., то швидкість обслуговування складає $\mu=2$ вимоги на годину (при цьому в розрахунок приймається тільки той час, коли прилад зайнятий обслуговуванням).

Часто припускають, що розподіл тривалостей обслуговування є експоненційним:

$$g(t)=\mu e^{-\mu t},t\geq 0.\tag{13.24}$$

Основна причина такого припущення полягає в прагненні спростити математичний бік питання. Проте припущення про те, що механізм обслуговування функціонує згідно з експоненційним розподілом, може одночасно служити деяким орієнтиром при аналізі операційних характеристик будь-якої СМО, оскільки в ньому знаходять висвітлення особливості систем „екстремального" типу, тобто систем, у яких обслуговуючі прилади не мають пам'яті. У випадку показникового розподілу тривалостей обслуговування вірогідність завершення обслуговування клієнта в будь-який наступний інтервал часу $(t,t+h)$ не залежить від того, скільки часу уже витрачено на обслуговування цієї вимоги.

Таким чином, якщо в момент t вимога вже обслуговувалася і ми розглядаємо систему в момент $(t+h)$, то внаслідок наведеного вище (13.24):

$$P\left[\begin{array}{c}\text{Обслуговування в інтервалі}\\ (t,t+h)\text{ не закінчується}\end{array}\right]=e^{-\mu h}.\tag{13.25}$$

Отже, при дуже малих позитивних значеннях h

$$P\begin{bmatrix}\text{Обслуговування в інтервалі}\\ (t,t+h)\ \text{не закінчується}\end{bmatrix}\approx 1-\mu h \quad (13.26)$$

$$P\begin{bmatrix}\text{Обслуговування в інтервалі}\\ (t,t+h)\ \text{закінчується}\end{bmatrix}\approx \mu h. \quad (13.27)$$

13.3.2. МОДЕЛЬ ЧИСТОЇ ЗАГИБЕЛІ

Розглянемо модель **чистої загибелі.** Нехай система функціонує, починаючи з моменту $t=0$. Припустимо, що в момент 0 в системі знаходиться M вимог і що після моменту 0 нові вимоги в систему більш не надходять. Припустимо, що в системі є єдиний прилад, швидкість обслуговування якого характеризується показниковим розподілом. Нехай

$$P_n(T)=\begin{bmatrix}\text{Вірогідність того, що в момент T к-ть}\\ \text{вимог, що знаходяться в системі} = n\end{bmatrix}. \quad (13.28)$$

З метою логічно послідовного виведення формули для $P_n(T)$ можна застосувати метод, що аналогічний використаному в попередньому питанні. Проте ми можемо досягти того ж результату, використовуючи лише співвідношення для дуже малих позитивних значеннях h; нижче ілюструється саме цей засіб одержання формули для $P_n(T)$, оскільки він виявляється дуже ефективним при аналізі складніших моделей.

Як і раніше, ми обчислюємо вірогідності приблизно, нехтуючи складовими вищого порядку малості. При цьому ми знову постулюємо, що протягом дуже малого інтервалу часу $(T,T+h)$ число подій, що полягають у виході вимог із системи, не може бути більше одної (властивість ординарності). Це означає, що якщо в момент $(T+h)$ у системі знаходиться n вимог, то нами враховуються лише:

а) вірогідність того, що в момент T в системі також знаходилося n вимог і жодна з них не покинула систему протягом інтервалу $(T,T+h)$;

б) вірогідність того, що в момент T в системі знаходилося $n+1$ вимог і протягом інтервалу $(T,T+h)$ одна вимога вибула із системи (тобто мав місце один випадок виходу).

Дійсно, при $h\to 0$ вірогідністю того, що в інтервалі $(T,T+h)$ спостерігається декілька (більш одного) випадків виходу вимог, можна знехтувати як величиною вищого порядку малості в порівнянні з вірогідністю того, що протягом зазначеного інтервалу не відбувається жодного випадку виходу або має місце єдиний випадок виходу. Тоді при $M>n\geq 1$

$$P_n(T+h)\approx(1-\mu h)P_n(T)+(\mu h)P_{n+1}(T). \quad (13.29)$$

Перша складова у правій частині цієї формули $(1-\mu h)P_n(T)$ є приблизно обчислена вірогідність того, що в інтервалі $(T,T+h)$ не була завершена жодна процедура обслуговування при n вимогах, що знаходилися в момент T всередині системи. Після перегрупування доданків в (13.29) отримаємо:

$$\frac{P_n(T+h)-P_n(T)}{h}\approx-\mu P_n(T)+\mu P_{n+1}(T), \quad (13.30)$$

так що при $h\to 0$ матимемо:

$$\frac{dP_n}{dT}=-\mu P_n(T)+\mu P_{n+1}(T),\quad M>n\geq 1. \quad (13.31)$$

Міркуючи аналогічним чином, можна довести, що

$$\frac{dP_M}{dT}=-\mu P_M(T)\ \text{ при } n=M. \quad (13.32)$$

Ця система лінійних диференційних рівнянь (13.31) та (13.32) має єдиний розв'язок:

$$P_n(T)=\frac{(\mu T)^{M-n}e^{-\mu T}}{(M-n)!}\ \text{ при } n=1,\dots,M \quad (13.33)$$

$$P_0(T)=1-\sum_{n=1}^{M}P_n(T)\ \text{ при } n=0. \quad (13.34)$$

Розподіл, заданий цими співвідношеннями, іноді називають усіченим пуасонівським розподілом.

Якщо М-й клієнт обслуговується останнім, то повний час у його перебування в системі, включаючи час, протягом котрого цей клієнт обслуговувався, має щільність розподілу (гама-розподіл):

$$h(y)=\frac{\mu(\mu y)^{M-1}e^{-\mu y}}{(M-1)!},\quad y\geq 0 \quad (13.35)$$

При цьому

$$M[y]=\frac{M}{\mu} \text{ і } D[y]=\frac{M}{\mu^2}. \qquad (13.36)$$

Хоча припущення про експоненційний характер розподілу тривалостей обслуговування і рекомендується з математичної точки зору як найзручніше, не варто забувати про існування й інших виглядів розподілу. Зокрема, для деяких простих моделей масового обслуговування вдається отримати досить корисні математичні результати в припущенні, що тривалості обслуговування розподілені за законом Ерланга.

13.3.3. МОДЕЛЬ САМООБСЛУГОВУВАННЯ

Припустимо тепер, що на відміну від розглянутого вище випадку (коли система містила єдиний обслуговуючий прилад) у момент 0 кожна із M вимог приступає до самообслуговування. Припустимо, що тривалості самообслуговування в кожної з вимог розподілені за експоненційним законом. Таке припущення в умовах самообслуговування є цілком логічним. Розглянемо дуже малий інтервал часу $(T,T+h)$ де $h>0$. Тоді, оскільки кожна вимога діє в процесі самообслуговування незалежним чином, за допомогою наближеної формули (13.6) отримаємо:

$$P\left[\begin{matrix}\text{Жодна з n вимог не}\\ \text{залишає систему}\end{matrix}\right]\approx(1-\mu h)^n=1-n\mu h. \qquad (13.37)$$

$$P\left[\begin{matrix}\text{Одна з n вимог залишає систему,}\\ \text{тобто закінчує самообслуговування}\end{matrix}\right]\approx n\mu h. \qquad (13.38)$$

При цьому, враховуючи, що $h\to 0$, ми в процесі обґрунтування можемо обмежитися розглядом лише таких подій, коли на інтервалі $(T,T+h)$ або жодна із n вимог не встигає закінчити процедуру самообслуговування, або закінчує самообслуговування одна із них, тобто ми нехтуємо вірогідностями того, що в даному інтервалі самообслуговування закінчують дві або більше вимог, рахуючи ці вірогідності величинами вищого порядку малості.

Іншими словами, якщо в момент $T+h$ у системі знаходиться n вимог, то передбачається, що можна враховувати лише:

а) вірогідність того, що в момент T в ній також було n вимог і не спостерігалося жодного випадку їхнього виходу із системи;

б) вірогідність того, що в момент T число вимог усередині системи дорівнювало $n+1$ і спостерігався випадок виходу однієї вимоги з даної системи (при цьому повинна виконуватися умова $M>n\geq 0$), тобто

$$P_n(T+h)=(1-h\mu n)P_n(T)+(n+1)\mu hP_{n+1}(T), \qquad (13.38)$$

де h - дуже мала величина.

Перенесемо $P_n(T)$ у ліву частину цього співвідношення, розділимо обидві частини отриманого співвідношення на h і спрямуємо h до нуля; після цього отримаємо:

$$\frac{dP_n}{dT}=-n\mu P_n(T)+(n+1)\mu P_{n+1}(T) \text{ при } M>n\geq 0. \qquad (13.39)$$

Міркуючи цілком аналогічно, отримаємо:

$$\frac{dP_M}{dT}=-M\mu P_M(T) \text{ при } n=M. \qquad (13.40)$$

Розв'язок системи рівнянь (13.39) та (13.40) має вигляд:

$$P_n(T)=C_M^n(e^{-\mu T})^n(1-e^{-\mu T})^{M-n} \text{ для } n=0,1,\ldots,M, \qquad (13.41)$$

(біноміальний розподіл). При цьому

$$M[n\,|\,T]=Me^{-\mu T}, \qquad D[n\,|\,T]=Me^{-\mu T}(1-e^{-\mu T}). \qquad (13.42)$$

13.4. ТЕРМІНОЛОГІЯ І ПОЗНАЧЕННЯ СИСТЕМ МАСОВОГО ОБСЛУГОВУВАННЯ

Термінологія в галузі СМО є непогано стандартизованою, а позначення (котрі часто називають позначеннями Кендала) уніфіковані. При цьому для позначення тієї або іншої моделі використовують три символи **A / B / m**, де **A** характеризує вхідний потік вимог, **B** - розподіл тривалостей обслуговування і **m** - число приладів в обслуговуючій системі. Нижче наведений перелік узвичаєних символів, що характеризують розподіли вірогідностей, які ставляться у відповідність моделям масового обслуговування:

M – експоненційний розподіл тривалостей інтервалів між надходженнями вимог або тривалостей обслуговування (від визначального слова «марківський»);

D - детермінований (або регулярний) розподіл тривалостей інтервалів між надходженнями вимог або тривалостей обслуговування;

E_m – n-фазний розподіл Ерланга для тривалостей інтервалів між надходженнями вимог або тривалостей обслуговування. (Ряд авторів використовують також символ **K_n**, позначаючи ним гама-розподіл).

GI – рекурентний характер вхідного потоку без яких-небудь спеціальних припущень щодо функції розподілу;

G – загальний вигляд розподілу тривалостей обслуговування (тобто не робиться ніяких припущень, що конкретизують функції розподілу).

Так, наприклад, для моделі з пуасонівським вхідним потоком, експоненційним розподілом тривалостей обслуговування і одним обслуговуючим приладом символічний запис має вигляд **M/M/1**. Якби вхідний потік був детермінований, а інші характеристики моделі залишалися б колишніми, символічне представлення моделі мало би вигляд **D/M/1**; якщо б замість одного обслуговуючого приладу тільки що згадана система мала m приладів, то в позначеннях Кендала символічне представлення моделі мало би вигляд **D/M/m**.

У випадках, коли необхідно також вказувати ємність накопичувача системи (число місць в черзі) К та кількість джерел М може використовуватися п'ятизначне позначення у вигляді **A/B/m/K/M**.

13.5. ФУНКЦІОНУВАННЯ СМО ЯК МАРКІВСЬКИЙ ВИПАДКОВИЙ ПРОЦЕС

Процес роботи СМО є випадковим процесом. **Випадковий процес** – це процес зміни в часі стану системи відповідно до закономірностей теорії вірогідностей. Процес **є процесом з дискретними станами**, якщо його можливі стани $S_1, S_2, \ldots$ можна перерахувати, а перехід системи зі стану в стан відбувається миттєво (стрибком). Процес є **процесом з неперервним часом**, якщо моменти можливих переходів системи зі стану в стан не фіксовані, а випадкові. Таким чином, процес функціонування СМО є випадковим процесом з дискретними станами та неперервним часом, тобто стани СМО змінюються стрибками в випадкові моменти появи подій. Випадковий процес є марківським, якщо він є процесом без післядії. При аналізі процесів з дискретними станами використовується їх представлення у вигляді графів переходів системи зі стану в стан (в СМО вони мають також назву діаграм інтенсивностей переходів). Стани системи зображаються вершинами графа, а можливі переходи – дугами, що з'єднують стани, причому переходи системи зі стану S_i в стан S_j відбуваються під дією найпростіших пуасонівських потоків з інтенсивностями λ_{ij} (див. приклад 13.5).

Вірогідністю i-го стану є вірогідність того, що в момент t система знаходитиметься в стані S_i, і $\sum_{i=0}^{K} p_i(t) = 1$.

Для вірогідностей станів можна побудувати систему диференційних рівнянь Колмогорова згідно з наступним правилом.

В лівій частині кожного рівняння знаходиться похідна вірогідності i-го стану, в правій – сума добутків вірогідностей всіх станів, з яких йдуть дуги зі стрілками, скерованими до i-го стану, на інтенсивності відповідних потоків подій, мінус суму інтенсивностей всіх потоків, що виводять систему з i-го стану, помножену на вірогідність i-го стану (див. приклад 13.6).

Для розв'язання диференційних рівнянь потрібно задати початкові умови та додати умову нормування вірогідностей, вилучивши з системи рівнянь Колмогорова одне залежне (довільне) рівняння, і, таким чином, отримати всі вірогідності станів як функції часу. Особливо цікавими з точки зору аналізу СМО є вірогідності перебування системи в кожному зі станів $p_i(t)$ в стаціонарному режимі, тобто $p_i = \lim_{t\to\infty} p_i(t)$. В теорії випадкових процесів доводиться, що якщо число станів системи є скінченим, і з кожного з них можна за скінчену кількість кроків перейти до будь-якого іншого, то граничні вірогідності, що відповідають стаціонарному режиму, існують.

Оскільки граничні вірогідності є сталими, то для отримання їх значень досить в системі рівнянь Колмогорова замінити їх похідні нульовими значеннями, щоб отримати систему лінійних алгебраїчних рівнянь, що описують стаціонарний режим.

Таким чином, застосування рівнянь Колмогорова для стаціонарного режиму дозволяє в ряді випадків суттєво полегшити аналіз СМО у встановленому режимі.

ПРИКЛАДИ

Приклад 13.1. Попередній аналіз СМО.

Пасажири вокзалу скаржаться на повільне обслуговування. На вокзалі працюють 3 каси з наявних семи. В результаті розслідування скарг пасажирів на повільне обслуговування та додаткових досліджень була отримана наступна емпірична залежність між кількістю касирів і часом очікування обслуговування.

Кількість касирів	1	2	3	4	5	6	7
Середній час очікування (хвилини)	16.2	10.3	6.9	4.8	2.9	1.9	1.3

Наведені дані свідчать про те, що при працюючих в даний час трьох касирах середній час очікування обслуговування приблизно рівний 7 хвилин. Керівник хоче зменшити його приблизно до 3 хвилин.

Розв'язання.

Як випливає з цих же даних, середній час очікування стає менше 3 хвилин, якщо кількість касирів більше або рівно п'яти. Дійсно, при п'яти працюючих касирах середній час очікування рівно 2.9 хвилини.

Приклад 13.2. Відсутність післядії.

Наведемо приклад, що ілюструє властивість експоненційного розподілу „не пам'ятати про минуле" і парадоксальну ситуацію, що може мати місце в СМО. Уявімо, що ми намагаємося впіймати таксі в будь-якому місці на одній з центральних вулиць міста.

Припустимо, що повз це місце кожні 30 с. проходить в середньому одне вільне таксі, тобто середня тривалість інтервалу між появами таксі в заданій точці дорівнює $1/\lambda = 30$ с. Припустимо, що ми намагаємося впіймати таксі, починаючи з деякого довільним чином обраного моменту часу. Скільки в середньому секунд нам доведеться чекати появи вільного таксі?

Розв'язання.

Багато хто відповість, що чекати доведеться 15 с. Чи так це?

Така відповідь була б вірною лише за умови, що вільні таксі з'являються біля місця, де ми їх зупиняємо, точно через кожні 30 с. Якщо має місце хоча б деякий розкид інтервалів між появами вільних таксі в обраному нами місці, тривалість нашого очікування завжди буде більшою за 15 с.

Можна довести, що якщо розглядати систему в довільний момент часу t, то

$$\begin{bmatrix}\text{Середня тривалість інтервалу}\\ \text{від моменту t до найближчого}\\ \text{моменту появи таксі}\end{bmatrix} = \frac{1}{2}\left(\frac{1}{\lambda} + \lambda \times \begin{bmatrix}\text{Дисперсія тривалості}\\ \text{інтервалу між появами таксі}\end{bmatrix}\right)$$

Отже, якщо дисперсія у цій формулі відмінна від нуля, то середня тривалість чекання з моменту t до появи першого таксі, перевищує $(1/2)\times(1/\lambda)$. Зауважимо, що у випадку експоненційного розподілу тривалостей інтервалів між послідовними появами дисперсія дорівнює

$1/\lambda^2$, і, отже, середній час чекання з моменту t до появи таксі дорівнює $1/\lambda$. Якщо ж дисперсія приймає значення, що перевищують $1/\lambda^2$, то середній час чекання вільного таксі (що відраховується з моменту t) виявляється насправді більшим, ніж середнє значення тривалості інтервалу між появами вільних таксі. Цей результат може здатися на перший погляд дещо незвичним.

Приклад 13.3. Відсутність післядії.

Те, що експоненціальний розподіл є абсолютно випадковим, ілюструється наступним прикладом. Якщо біжучий час 8:15 і деяка подія трапилась о 8:00, то,згідно з експоненційним законом розподілу, вірогідність того, що наступна аналогічна подія відбудеться о 8:30, є функцією лише інтервалу часу від 8:15 до 8:30 і не залежить від інтервалу часу, що пройшов із моменту настання останньої події (від 8:00 до 8:15). Це і є відсутність післядії або відсутність пам'яті.

Приклад 13.4. Модель чистого народження.

Діти народжуються в місті із інтенсивністю одне народження на кожні 20 хвилин. Час між послідовними народженнями розподілений за експоненційним законом. Необхідно визначити:

☑ середнє число народжень за рік;

☑ вірогідність того, що протягом одного дня не буде жодного народження;

☑ вірогідність видачі 30 свідоцтв про народження до кінця третьої години, якщо відомо, що протягом останніх двох годин було видано 25 таких свідоцтв.

Розв'язання.

Обчислимо інтенсивність народжень за добу: $\lambda = \dfrac{24\times 60}{20} = 72$ народжень за добу. Інтенсивність народжень в місті за рік становить $\lambda\times t = 72\times 365 = 26280$ народжень.

Вірогідність того, що протягом одного дня не буде жодного народження, обчислюється з використанням пуассонівського розподілу –

$$p_0(t) = \frac{(72\times 1)^0 e^{-72\times 1}}{0!} \approx 0\,.$$

Для обчислення вірогідності видачі 30 документів про народження до кінця третьої години за умови, що протягом останніх двох годин було видано 25 таких документів, зауважимо, що, оскільки розподіл числа народжень є пуассонівським, вірогідність зводиться до вірогідності появи $30-25=5$ народжень на одну $3-2=1$ годину. Оскільки $\lambda=\frac{72}{24}=3$ народження на годину, то $p_5(1)=\frac{(3\times1)^5e^{-3\times1}}{5!}=0{,}1017$.

Приклад 13.5. Модель системи у вигляді графу переходів.

Необхідно побудувати граф переходів наступної системи: система S складається з двох підсистем, кожна з яких може вийти з ладу в випадковий момент часу, після чого починається її ремонт, що триває випадковий час. Виходи з ладу підсистем є незалежними один від одного, вірогідністю одночасного виходу з ладу двох підсистем чи одночасного закінчення ремонту можна знехтувати (тобто процес є ординарним).

Розв'язання.

Визначимо можливі стани системи:

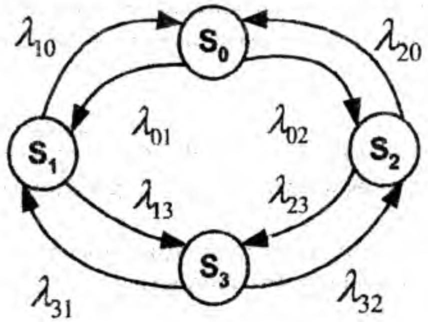
S_0 – дві підсистеми функціонують нормально;

S_1 – перша підсистема ремонтується, друга функціонує нормально;

S_2 – друга підсистема ремонтується, перша функціонує нормально;

S_3 – обидві підсистеми ремонтуються.

Позначимо інтенсивність переходу системи зі стану i в стан j як λ_{ij}. В результаті отримаємо наступний граф переходів системи:



Перехід системи зі стану S_0 до стану S_1 відбувається під дією потоку відмов першої підсистеми, а обернений з S_1 до S_0 – під дією потоку закінчень ремонтів першої підсистеми і т.д.

Приклад 13.6. Розрахунок граничних вірогідностей.

Для прикладу 13.5 побудувати систему диференційних рівнянь Колмогорова та визначити граничні вірогідності, якщо задані інтенсивності пуасонівських потоків

$\lambda_{01}=1;\ \lambda_{02}=2;\ \lambda_{10}=2;\ \lambda_{13}=2;\ \lambda_{20}=3;\ \lambda_{23}=1;\ \lambda_{31}=3;\ \lambda_{32}=2.$

Використовуючи правило побудови системи рівнянь Колмогорова, отримаємо:

$$\begin{cases}\frac{dp_0}{dt}=\lambda_{10}p_1+\lambda_{20}p_2-(\lambda_{01}+\lambda_{02})p_0,\\ \frac{dp_1}{dt}=\lambda_{01}p_0+\lambda_{31}p_3-(\lambda_{10}+\lambda_{13})p_1,\\ \frac{dp_2}{dt}=\lambda_{02}p_0+\lambda_{32}p_3-(\lambda_{20}+\lambda_{23})p_2,\\ \frac{dp_3}{dt}=\lambda_{13}p_1+\lambda_{23}p_2-(\lambda_{31}+\lambda_{32})p_3\end{cases}$$

і, прирівнявши похідні до нуля та перенісши складові з мінусом в ліві частини рівнянь, отримуємо систему для стаціонарного режиму:

$$\begin{cases}(\lambda_{01}+\lambda_{02})p_0=\lambda_{10}p_1+\lambda_{20}p_2,\\ (\lambda_{10}+\lambda_{13})p_1=\lambda_{01}p_0+\lambda_{31}p_3,\\ (\lambda_{20}+\lambda_{23})p_2=\lambda_{02}p_0+\lambda_{32}p_3,\\ (\lambda_{31}+\lambda_{32})p_3=\lambda_{13}p_1+\lambda_{23}p_2\ .\end{cases}$$

Підставивши числові значення та додавши умову нормування, вилучаємо останнє рівняння з системи Колмогорова (оскільки ця система рівнянь залежна), отримаємо:

$$\begin{cases}3p_0=2p_1+3p_2,\\ 4p_1=p_0+3p_3,\\ 4p_2=2p_0+2p_3,\\ p_0+p_1+p_2+p_3=1.\end{cases}$$

Розв'язком цієї системи є: $p_0=0,40;\ p_1=0,20;\ p_2=0,27;\ p_3=0,13$. Таким чином, в стаціонарному режимі система в середньому 40% часу буде працювати з двома справними підсистемами, 20% - ремонтуватиметься перша підсистема, а друга працюватиме, 27% - перша система працюватиме, а друга ремонтуватиметься, і 13% - ремонтуватимуться дві підсистеми.

Приклад 13.7. Аналіз варіантів системи на основі граничних вірогідностей.

Порівняти за ефективністю 2 варіанти системи з прикладів 13.5, 13.6 та обрати найкращий. В першому варіанті характеристики системи залишаються без змін, а в другому – середній час ремонту кожної з підсистем вдвічі зменшується за рахунок збільшення в два рази витрат на ремонт кожної з підсистем. Відомо, що робота першої підсистеми приносить дохід 10 грошових одиниць на одиницю часу, а другої – 6, витрати на їх ремонт становлять 4 та 2 грошові одиниці.

Розв'язання.

За результатами прикладу 13.7 розраховуємо, що в середньому перша підсистема працює частку часу, що становить:

$p_0 + p_2 = 0,40 + 0,27 = 0,67$, а друга підсистема -

$p_0 + p_1 = 0,40 + 0,20 = 0,60$. В той же час перша підсистема знаходиться в середньому в ремонті частку часу, що становить:

$p_1 + p_3 = 0,20 + 0,13 = 0,33$, а друга -

$p_2 + p_3 = 0,27 + 0,13 = 0,40$. Тому середній чистий дохід на одиницю часу для першого варіанту становитиме:

$$Q(1) = 0,67\times10 + 0,60\times6 - 0,33\times4 - 0,40\times2 = 8,18 \text{ грош. од./од.часу}$$

В другому варіанті зменшення в два рази середнього часу ремонту кожної з підсистем означатиме збільшення в два рази інтенсивностей потоків закінчень ремонтів, тобто значення інтенсивностей для цього випадку становитимуть:

$$\lambda_{01} = 1;\ \lambda_{02} = 2;\ \lambda_{10} = 4;\ \lambda_{13} = 2;\ \lambda_{20} = 6;\ \lambda_{23} = 1;\ \lambda_{31} = 6;\ \lambda_{32} = 4$$

і система рівнянь, що описують стаціонарний режим функціонування системи, видозміниться наступним чином:

$$\begin{cases} 3p_0 = 4p_1 + 6p_2, \\ 6p_1 = p_0 + 6p_3, \\ 7p_2 = 2p_0 + 4p_3, \\ p_0 + p_1 + p_2 + p_3 = 1. \end{cases}$$

Розв'язком цієї системи є:

$p_0 = 0,60;\ p_1 = 0,15;\ p_2 = 0,20;\ p_3 = 0,05$.

Для другого варіанту в середньому перша підсистема працює частку часу, що становить $p_0 + p_2 = 0,60 + 0,20 = 0,80$, а друга підсистема - $p_0 + p_1 = 0,60 + 0,15 = 0,75$. В той же час перша підсистема знаходиться в середньому в ремонті частку часу, що становить:

$p_1 + p_3 = 0,15 + 0,05 = 0,20$, а друга - $p_2 + p_3 = 0,20 + 0,05 = 0,25$.

Затрати на ремонт зростуть в два рази і становитимуть для першої підсистеми 8 грошових одиниць, а для другої – 4. Тому середній чистий дохід за одиницю часу для другого варіанту становитиме:

$$Q(2) = 0,80\times10 + 0,75\times6 - 0,20\times8 - 0,25\times4 = 9,9 \text{ грош. од./од.часу}$$

Оскільки $Q(2) > Q(1)$, то доцільніше зменшити час ремонту вдвоє за рахунок збільшення витрат на ремонт у два рази.

РЕЗЮМЕ

13.1. Системи масового обслуговування (СМО) є широко розповсюдженим класом систем. Такого виду процеси утворення черг або затримок в обслуговуванні ефективно аналізуються методами дослідження операцій. Проте витрати, пов'язані з науковим аналізом тієї чи іншої практичної задачі масового обслуговування, вважаються (як і в будь-якій іншій галузі організаційного управління) виправданими лише за умови, що економічні наслідки керуючих рішень в сфері, що аналізується, мають істотний вплив. Відносно прості моделі СМО можуть бути використані і для одержання якісного або наближеного кількісного уявлення про поведінку систем, що мають складнішу структуру. До операційних характеристик належать: середня довжина черги, середній час очікування на обслуговування, вірогідність того, що всі компоненти обслуговуючої системи виявляться зайнятими та ін. Після оцінювання цих характеристик можна переходити до побудови відповідної економічної моделі і до наступних процедур пошуку оптимальних керуючих рішень. Математична модель системи масового обслуговування (СМО) включає наступні основні елементи: потік вимог, що надходять на вхід системи (вхідний потік); чергу, що складається з вимог, які очікують на обслуговування; систему обслуговування; вихідні потоки обслужених, втрачених вимог та вимог, що надходять на повторне обслуговування, характеристики якості системи і механізм (дисципліну) обслуговування. Операційні методи дослідження СМО орієнтовані на оптимізацію відповідних керуючих рішень. При аналізі прикладів виявляється, що практично *в кожній із ситуацій керівник повинен брати до уваги всі три компоненти системи масового обслуговування: вхідний потік вимог на обслуговування, дисципліну черги і механізм обслуговування. Результати дослідження системи обслуговування також можна використовувати для оптимізації моделі з вартісними характеристиками, в якій мінімізується сума витрат, пов'язаних з наданням послуг, і втрат, обумовлених затримками в їх наданні. При дослідженні СМО можуть розв'язуватися: задачі аналізу СМО - визначення характеристик якості обслуговування в залежності від параметрів і властивостей вхідного потоку вимог, параметрів і структури системи обслуговування і дисципліни обслуговування; задачі параметричного синтезу - визначення параметрів системи обслуговування при її заданій структурі залежно від параметрів і властивостей потоку вимог, дисципліни і якості обслуговування; задачі синтезу структури системи з оптимізацією її параметрів – зробити так, щоб при заданих потоках, дисципліні і якості обслуговування вартість СМО була мінімальною, або були мінімальними втрати викликів при заданих потоках, дисципліні і вартості системи.*

13.2. Механізм надходження вимог найзручніше описувати, задаючи розподіл вірогідностей для тривалостей інтервалів часу між послідовними надходженнями вимог на обслуговування. Якщо тривалості інтервалів між надходженнями вимог статистично незалежні, визначаються тим самим розподілом вірогідностей і описуються деякою неперервною функцією, що є щільністю розподілу то вхідний потік вимог такого виду є типовим прикладом процесу відновлення, а послідовність надходжень є ілюстрацією послідовності рекурентних подій. Найважливіший приклад розподілу тривалостей інтервалів між надходженнями вимог відповідає випадку цілком випадкових надходжень. Потік цілком випадкових надходжень вимог має властивості стаціонарності, відсутності післядії і ординарності. Події в експоненційних процесах мають цілком випадковий характер. Якщо щільність розподілу тривалостей інтервалів між надходженнями вимог на обслуговування має експоненційний вигляд, то щільність розподілу повного часу для довільним способом обраного ряду з n послідовних надходжень визначається за гама-розподілом. Припущення про експоненційний характер розподілів тривалостей інтервалів між надходженнями рівносильне такому твердженню: розподіл n надходжень у довільно обраному інтервалі тривалістю T є пуасонівським. З деяких припущень щодо властивостей процесу надходження як наслідок

випливає показниковий розподіл тривалостей інтервалів між надходженнями, і вхідний потік, таким чином, є пуасонівським. Важливим окремим випадком розподілу тривалостей інтервалів між послідовними надходженнями вимог на обслуговування є розподіл Ерланга, який ми можемо використовувати в якості доброго наближення розподілів інших видів.

13.3. Основна причина припущення про те, що розподіл тривалостей обслуговування є експоненційним, полягає в прагненні спростити математичний бік питання. Однак припущення про те, що механізм обслуговування функціонує згідно з експоненційним розподілом, може одночасно служити деяким орієнтиром при аналізі операційних характеристик будь-якої СМО, оскільки в ньому знаходять висвітлення особливості систем „екстремального" типу, тобто систем, у яких обслуговуючі прилади не мають пам'яті. Для деяких простих моделей масового обслуговування вдається отримати досить корисні математичні результати в припущенні, що тривалості обслуговування розподілені за законом Ерланга.

13.4. Термінологія в галузі СМО є непогано стандартизованою, а позначення Кендала уніфіковані. При цьому для позначення тієї або іншої моделі використовують три символи $A/B/m$, *де A характеризує вхідний потік вимог, B - розподіл тривалостей обслуговування і m - кількість приладів в обслуговуючій системі. У випадках, коли необхідно також вказувати ємність накопичувача системи (кількість місць у черзі) K та кількість джерел M, може використовуватися позначення з п'яти букв у вигляді A/B/m/K/M.*

13.5. Процес функціонування СМО є випадковим процесом з дискретними станами та неперервним часом, тобто стани СМО змінюються стрибками в випадкові моменти появи подій. При аналізі процесів з дискретними станами використовується їх представлення у вигляді графів переходів системи зі стану в стан. Для вірогідностей станів можна побудувати систему диференційних рівнянь Колмогорова. Особливо цікавими з точки зору аналізу СМО є вірогідності перебування системи в кожному зі станів в стаціонарному режимі. Оскільки граничні вірогідності є сталими, то для отримання їх значень досить в системі рівнянь Колмогорова замінити їх похідні нульовими значеннями, щоб отримати систему лінійних алгебраїчних рівнянь, що описують стаціонарний режим. Таким чином, застосування рівнянь Колмогорова для стаціонарного режиму дозволяє в ряді випадків суттєво полегшити *аналіз СМО у встановленому режимі.*

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 13.1. Попередній аналіз СМО.

Припустимо, що подальший аналіз роботи вокзалу привів до наступних додаткових даних.

Кількість касирів	1	2	3	4	5	6	7
Простій (%)	0	8	12	18	29	36	42

Необхідно відповісти на наступні запитання.

Яка ефективність обслуговування (у відсотках часу, коли касири були зайняті роботою), якщо їх за кількістю п'ять?

Керівник бажає підтримувати одночасно середній час очікування на рівні приблизно трьох хвилин і ефективність використовування персоналу на рівні 90%. Чи може він досягти цього? Наведіть належне пояснення.

Завдання 13.2. Попередній аналіз СМО.

Кондитерське підприємство купує машину для випікання кондвиробів. Наявними до вибору є моделі A і B, експлуатаційні витрати при їх використанні становлять 18 і 25 гривень на годину відповідно. Машина моделі A працює повільніше, ніж машина моделі B. Аналіз систем обслуговування з такими пристроями показує, що у разі використання моделі A середня довжина черги робіт, що чекають на обслуговування, рівна чотирьом, що на 30% вище за аналогічний показник при використанні моделі B. Затримка у виконанні роботи приводить до втрати прибутку, який оцінюється в 10 гривень на годину очікування в черзі. Яку модель машини слід придбати підприємству?

Завдання 13.3. Відсутність післядії.

При обслуговуванні складного агрегату завжди є запасний блок для негайної заміни у разі поломки. Час виходу з ладу агрегату є випадковою величиною, розподіленою за експоненційним законом, і в середньому відбувається кожні 30 хвилин.

Необхідно визначити наступні показники:

☑ середню інтенсивність відмов агрегату (к-ть відмов за годину);

☑ навести вигляд функції щільності густини експоненційного розподілу,

☑ середнє число відмов за один тиждень, якщо система обслуговування функціонує 5 днів на тиждень по 18 годин щодня.

☑ вірогідність принаймні однієї відмови за 3 години;

☑ вірогідність того, що наступна відмова не відбудеться протягом двох годин;

☑ якщо після останньої відмови протягом трьох годин інших відмов не було, то яка вірогідність того, що час між послідовними відмовами системи становитиме принаймні 4 години?

Завдання 13.4. Властивості експоненційного розподілу

В місті функціонують дві трамвайні лінії: маршрут №1 та маршрут №2. Лінія №1 обслуговує західну частину міста, №2 – східну. Трамвайні лінії мають спільну трамвайну зупинку в центрі міста. Час між прибуттями трамваїв маршруту №1 на спільну трамвайну зупинку є випадковою величиною, розподіленою за експоненційним законом з середнім значенням 15 хвилин. Аналогічний показник для трамваїв маршруту №2 становить 9 хвилин.

Визначте розподіл часу очікування пасажира на спільній трамвайній зупинці, який прибуває по лінії №1 для пересадки на лінію №2.

Завдання 13.5. Модель чистого народження.

Час між послідовними прибуттями відвідувачів в кав'ярню розподілений за експоненційним законом з середнім значенням 3 хвилини. Кав'ярня починає роботу об 11:00. Час перебування відвідувача в кав'ярні не менший, ніж 40 хвилин. Необхідно визначити:

Вірогідність того, що об 11:20 у кав'ярні опиниться 10 відвідувачів за умови, що об 11:15 у кав'ярні було 8 відвідувачів.

Вірогідність того, що новий відвідувач надійде в інтервалі між 11:28 і 11:33, якщо відомо, що попередній зайшов до кав'ярні об 11:25.

Завдання 13.5. Модель чистої загибелі.

Для покриття поточних потреб студенту щомісячно необхідно 300 грн. Отримання студентом грошей за випадкові додаткові роботи відбувається випадковим чином по 70 гривень протягом місяця, згідно з експоненціальним законом з середнім значенням 1 раз на тиждень. Визначте вірогідність того, що до кінця місяця у студента не буде грошей на поточні витрати, вважаючи що місяць складається з 4-х тижнів.

Завдання 13.6. Модель чистої загибелі.

Об'єм попиту на певний товар є випадковою величиною, що розподілена за законом Пуасона з середнім значенням 3 одиниці на день. Максимальна місткість комори становить 25 одиниць. Комора повністю заповнюється кожний понеділок відразу ж після отримання нового замовлення. Об'єм замовлення залежить від кількості виробів, що залишилися до кінця тижня в суботу (неділя – вихідний день). Необхідно визначити наступні значення:

☑ вірогідність відсутності запасу вранці в п'ятницю;

☑ вірогідність того, що тижневий об'єм замовлення перевищить 10 одиниць.

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Яку структуру має система масового обслуговування?
2. Задачі яких типів можуть моделюватися у вигляді СМО?
3. Перерахуйте операційні характеристики моделей СМО та дайте їм змістовну інтерпретацію.
4. Якими особливостями характеризується вхідний потік вимог СМО?
5. Що описує дисципліна черги?
6. Чим характеризується обслуговуючий механізм?
7. Назвіть основні характеристики якості обслуговування.
8. В чому полягає суть вартісної моделі СМО?
9. Які задачі розв'язуються при дослідженні СМО?
10. Чим характеризується вхідний потік вимог в СМО?
11. Який потік вимог є стаціонарним?
12. В чому полягає властивість відсутності післядії?
13. Визначте: істинні чи помилкові наступні твердження та обґрунтуйте відповідь:

а). В умовах експоненційного розподілу протягом нескінченно малого інтервалу часу можуть відбутися дві події.

б). Якщо інтервали часу між послідовними надходженнями вимог в систему розподілені за експоненційним законом, то вірогідність того, що вимога не надійде протягом інтервалу А, позитивна, незалежно від того, наскільки малим є інтервал А.

14. Які властивості має пуасонівський (найпростіший) потік?
15. В чому полягають основні припущення при описанні розподілів вірогідностей для тривалостей обслуговування?
16. Поясніть формальні співвідношення для моделі чистої загибелі.

17. Розкрийте особливості моделі самообслуговування.
18. Яким чином позначаються класи СМО?
19. В чому виявляється властивість ординарності потоку вимог?

ТЕМА 14. ХАРАКТЕРИСТИКИ ТА АНАЛІЗ МОДЕЛЕЙ СИСТЕМ МАСОВОГО ОБСЛУГОВУВАННЯ

Побудова операційної моделі для описання реальної ситуації завжди пов'язана з необхідністю прийняття ряду апроксимуючих припущень. У випадку розв'язання задачі масового обслуговування апроксимації є неминучими незалежно від того, якого типу модель при цьому використовується – математична, імітаційна або комбінована. Часто вдається одержати приблизне уявлення про операційні характеристики складної системи шляхом аналізу деяких „екстремальних" або граничних випадків. При розгляді загальних систем масового обслуговування передбачається, що система функціонує протягом достатньо тривалого інтервалу часу, і після закінчення перехідного періоду переходить у стаціонарний режим. Цей режим функціонування обслуговуючої системи протиставляється перехідному (або несталому) режиму, який превалює в початковий період функціонування системи. У загальній моделі системи масового обслуговування передбачається, що і інтенсивність надходження вимог, і інтенсивність вихідного потоку залежать від стану системи, що означає їх залежність від числа вимог в системі обслуговування. Найпростішими моделями СМО є одно- та багатоканальні моделі з обмеженою чергою та без, що характеризуються показниковим розподілом як тривалостей інтервалів між надходженнями вимог, так і тривалостей обслуговування. Для таких моделей операційні характеристики визначаються аналітичним шляхом. Деякі з отриманих результатів застосовні і до найпростіших моделей з довільними законами розподілу. Для СМО складної структури результати можуть бути отримані шляхом імітаційного моделювання з використанням проблемно-орієнтованих мов, як, наприклад, GPSS.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ:

знати: ⇨проблеми та прийоми апроксимації реальних систем за допомогою систем масового обслуговування; ⇨структуру СМО та способи визначення її операційних характеристик; ⇨особливості різних класів СМО та способи їх аналізу;

вміти: ⇨будувати моделі СМО для реальних систем та їх досліджувати; ⇨розраховувати операційні характеристики для декількох варіантів СМО; ⇨аналізувати СМО на чутливість до зміни значень її параметрів.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ дисципліна черги
- ☑ вимога
- ☑ умова стаціонарності
- ☑ пріоритет
- ☑ діаграма інтенсивностей переходів
- ☑ процес загибелі та народження
- ☑ апроксимація СМО
- ☑ очікувана інтенсивність
- ☑ трафік-інтенсивність
- ☑ стаціонарний режим
- ☑ рівняння балансу
- ☑ операційні характеристики
- ☑ середня довжина черги
- ☑ інтенсивність потоку
- ☑ багатоканальна СМО
- ☑ період зайнятості
- ☑ ординарність
- ☑ скінчена черга
- ☑ середній час перебування в системі
- ☑ середній час перебування в черзі
- ☑ середня тривалість обслуговування
- ☑ імітаційне моделювання
- ☑ граничні вірогідності
- ☑ одноканальна СМО

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

14.1. Особливості апроксимації реальних систем за допомогою СМО.

14.2. Одноканальна модель з пуасонівським вхідним потоком і експоненційним розподілом тривалостей обслуговування.

14.2.1. Найпростіша одноканальна система масового обслуговування.

14.2.2. Операційні характеристики системи М/М/1.

14.3. Розширення системи М/М/1: скінчена черга, довільний розподіл тривалостей обслуговування, вимоги з пріоритетами.

14.4. Багатоканальна модель з пуасонівським вхідним потоком і експоненційним розподілом тривалостей обслуговування.

14.5.Інші моделі систем масового обслуговування.

14.1. ОСОБЛИВОСТІ АПРОКСИМАЦІЇ РЕАЛЬНИХ СИСТЕМ ЗА ДОПОМОГОЮ СМО

У більшості систем масового обслуговування є декілька обслуговуючих приладів, а дисципліна черги, як правило, виявляється дуже складною. Так, наприклад, перед тим, як обрати касу у супермаркеті, покупець спочатку подивиться, скільки осіб знаходиться в кожній черзі і яка кількість різноманітних продуктів знаходиться у стоячих. Крім того, він спробує вирішити, котра із кас знаходиться найближче до потрібного йому виходу, і відзначить, який із касирів працює швидше за інших. Аналогічними міркуваннями керується і водій, що обирає один із пунктів збирання за проїзд платною автомагістраллю з таким розрахунком, щоб витратити на всю процедуру мінімальний час. Але іноді, дійсно, діє дисципліна „першим прийшов – першим обслуговуєшся". Такого характеру черги виникають, наприклад, на бензозаправних станціях, біля кас кінотеатрів, у майстернях, де відбувається терміновий ремонт взуття і т.п.

Як уже відзначалося, побудова операційної моделі для описання реальної ситуації завжди пов'язана з необхідністю прийняття ряду апроксимуючих припущень. У випадку розв'язання задачі масового обслуговування апроксимації є неминучими незалежно від того, якого типу модель при цьому використовується – математична, імітаційна або комбінована. Часто вдається одержати приблизне уявлення про операційні характеристики складної системи шляхом аналізу деяких „екстремальних" або граничних випадків. Один із таких наближених методів полягає в наступному: СМО, що нараховує n обслуговуючих приладів, розглядається як „механічне" об'єднання n одноканальних систем, що функціонують незалежно одна від іншої. Нехай, наприклад, обслуговуюча система складається з п'ятьох приладів, а інтенсивність вхідного потоку дорівнює 20 вимог/год. Тоді таку систему приблизно можна розглядати як сукупність п'ятьох автономних систем з одним обслуговуючим приладом, кожна з яких характеризується вхідним потоком з інтенсивністю 4 вимоги/год. Цей метод є наближеним з двох причин:

☑ по-перше, передбачається, що вимога може потрапити на вхід будь-якої із згаданих одноканальних підсистем з однаковою вірогідністю (незалежно від довжини відповідної черги);

☑ по-друге, потрапивши в одну з черг, вимога повинна залишатися саме в цій обраній спочатку черзі.

Очікувана кількість вимог, що знаходяться у всіх автономних підсистемах цієї гіпотетичної системи, і середній час перебування вимоги в ній зазвичай перевищують значення відповідних операційних характеристик реальних багатоканальних систем. Це пояснюється тим, що якщо б ми застосували нашу гіпотетичну систему практично, то виявилося б, що частіше, ніж це можна припустити, один з обслуговуючих приладів знаходився б у вільному стані, у той час як вимоги простоювали у чергах до інших приладів. Якщо ж припустити, що в системі з декількома обслуговуючими приладами черга є єдиною і характеризується дисципліною "першим прийшов - першим обслуговуєшся", то відповідні апроксимуючі оцінки очікуваної кількості вимог у системі і середнього часу, витраченого кожною вимогою на очікування обслуговування, виявляться заниженими в порівнянні з фактичними значеннями цих показників.

Можна відзначити, що системи з одним обслуговуючим приладом, як і множина багатоканальних систем із єдиною чергою, що характеризується дисципліною „першим прийшов - першим обслуговуєшся" зазвичай піддаються математичному описанню і кількісному аналізу.

При розгляді загальних систем масового обслуговування передбачається, що система функціонує протягом достатньо великого інтервалу часу, і після закінчення перехідного періоду переходить в стаціонарний режим. Цей режим функціонування обслуговуючої системи протиставляється перехідному (або несталому) режиму, який превалює в початковий період функціонування системи. Ми не розглядатимемо перехідні режими роботи систем масового обслуговування, оскільки на практиці системи звичайно призначаються для роботи протягом тривалого часу.

В загальній моделі системи масового обслуговування передбачається, що і інтенсивність надходження вимог, і інтенсивність вихідного потоку ***залежать від стану системи***, що означає їх залежність від числа вимог в системі обслуговування. Наприклад, в цеху з фіксованою кількістю верстатів інтенсивність їх виходу з ладу зменшується зі зростанням числа аварійних верстатів, тому що лише працюючі верстати можуть виходити із ладу. Вважатимемо, що n – число вимог в системі обслуговування (в черзі і на обслуговуванні); λ_n – інтенсивність надходження в систему вимог за умови, що в системі вже знаходиться n вимог; μ_n – інтенсивність вихідного потоку обслужених вимог за умови, що в системі знаходиться n вимог; p_n – вірогідність того, що в системі знаходиться n вимог.

В загальній моделі системи масового обслуговування встановлюється функціональна залежність ймовірності p_n від λ_n і μ_n. Ця залежність використовується потім при визначенні функціональних характеристик обслуговуючої системи, таких як середня довжина черги, середній час очікування і середній коефіцієнт використання обслуговуючих пристроїв.

Вірогідність p_n визначається з діаграми інтенсивностей переходів, представленої на рис. 14.1. Обслуговуюча система знаходиться в стані n, якщо в ній є n клієнтів. Вірогідність появи більш ніж одного нового клієнта протягом малого проміжку часу h прямує до нуля при $h \to 0$ (ординарність). Це означає, що при $n > 0$ стан n може бути змінений в двох можливих напрямах: $(n-1)$, коли з інтенсивністю μ_n обслужений клієнт вибуває з системи, і $(n+1)$, коли має місце надходження клієнта з інтенсивністю λ_n. Стан 0 може змінитися лише до стану 1, коли має місце надходження клієнта з інтенсивністю λ_0. Відзначимо, що m_0 не визначене, оскільки не може відбуватися вибування клієнтів з порожньої системи обслуговування.

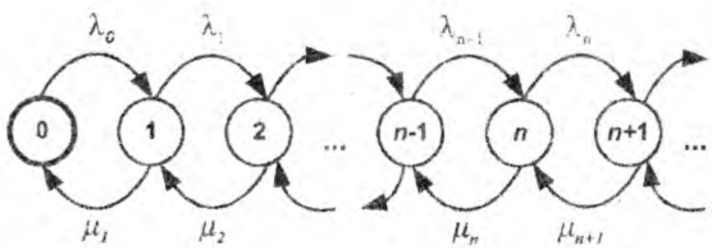


Рис. 14.1. Діаграма інтенсивностей переходів

При виконанні умов стаціонарності ***очікувані*** інтенсивності вхідного і вихідного потоків в стані n $(n > 0)$ повинні бути рівні. Оскільки стан n може змінюватися лише до станів $(n-1)$ і $(n+1)$, отримаємо співвідношення:

$$\begin{pmatrix} \text{Очікувана інтенсивність} \\ \text{вхідного потоку в стані } n \end{pmatrix} = \lambda_{n+1} p_{n-1} + \mu_{n+1} p_{n-1}. \qquad (14.1)$$

Аналогічно,

$$\begin{pmatrix} \text{Очікувана інтенсивність} \\ \text{вихідного потоку в } n \end{pmatrix} = (\lambda_n + \mu_n) p_n. \qquad (14.2)$$

Звідси, прирівнюючи ці дві інтенсивності, одержуємо наступне рівняння балансу:

$$\lambda_{n-1} p_{n-1} + \mu_{n+1} p_{n+1} = (\lambda_n + \mu_n) p_n, \quad n = 1, 2, \dots . \qquad (14.3)$$

Як видно з рис. 14.1, рівняння балансу, відповідне $n = 0$, має вигляд:

$$\lambda_0 p_0 = \mu_1 p_1. \qquad (14.4)$$

Рівняння балансу розв'язуються рекурентно, послідовно висловлюючи вірогідність p_i через p_0. Таким чином, для $n = 0$ отримаємо:

$$p_1 = \left(\frac{\lambda_0}{\mu_1}\right) p_0. \qquad (14.5)$$

Далі, для $n = 1$ одержуємо:

$$\lambda_0 p_0 + \mu_2 p_2 = (\lambda_1 + \mu_1) p_1. \qquad (14.6)$$

Підставляючи сюди $p_1 = (\lambda_0 / \mu_1) p_0$ і спрощуючи отриманий вираз, маємо:

$$p_2 = \left(\frac{\lambda_1 \lambda_0}{\mu_2 \mu_1}\right) p_0. \qquad (14.7)$$

Використовуючи індукцію, приходимо до виразу:

$$p_n = \left(\frac{\lambda_{n-1} \lambda_{n-2} \dots \lambda_0}{\mu_n \mu_{n-1} \dots \mu_1}\right) p_0, \quad n = 1, 2, \dots \qquad (14.8)$$

Значення p_0 визначається із рівняння $\sum_{n=0}^{\infty} p_n = 1$. Такий процес є процесом загибелі та народження, оскільки об'єднує в собі ці два процеси і описує функціонування СМО.

14.2. ОДНОКАНАЛЬНА МОДЕЛЬ З ПУАСОНІВСЬКИМ ВХІДНИМ ПОТОКОМ І ЕКСПОНЕНЦІЙНИМ РОЗПОДІЛОМ ТРИВАЛОСТЕЙ ОБСЛУГОВУВАННЯ

14.2.1. НАЙПРОСТІША ОДНОКАНАЛЬНА СИСТЕМА МАСОВОГО ОБСЛУГОВУВАННЯ

Найпростішою одноканальною моделлю з вірогіднісним вхідним потоком і процедурою обслуговування є модель, що характеризується показниковим розподілом як тривалостей інтервалів між надходженнями вимог, так і тривалостей обслуговування (тобто модель типу **M/M/1**). Таким чином, передбачається, що щільність розподілу тривалостей інтервалів між надходженнями вимог та щільність розподілу тривалостей обслуговування мають вигляд відповідно:

$$f(t)=\lambda e^{-\lambda t}\quad g(t)=\mu e^{-\mu t},t\geq 0. \tag{14.9}$$

Діаграма інтенсивностей переходів зображена на рис. 14.2. Прихід вимоги в систему здійснюється з інтенсивністю λ, а вихід – з інтенсивністю μ, оскільки обслуговування здійснюється за допомогою одного обслуговуючого пристрою.

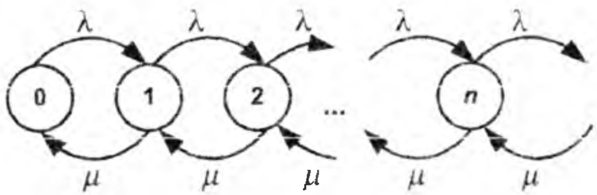


Рис. 14.2. Діаграма інтенсивностей переходів для одноканальної системи з необмеженою чергою

Нехай у деякий момент часу число вимог, що знаходяться в системі, включаючи вже ті вимоги, що обслуговуються, дорівнює n. Припустимо, що система починає функціонувати з моменту $t=0$. Позначимо

$$P_n(T)=\begin{bmatrix}\text{Вірогідність того, що в момент T}\\ \text{в системі знаходиться } n \text{ вимог}\end{bmatrix}. \tag{14.10}$$

Взагалі кажучи, $P_n(T)$ залежить від кількості вимог *i*, що знаходилися в системі в момент 0; в формулі (14.10) відповідний індекс для зменшення захаращеності не наводиться.

Нехай $h>0$ є інтервал часу дуже малої тривалості. Якщо в момент $T+h$ кількість вимог у системі дорівнює n, то ми будемо вважати, що кількість вимог, що могли знаходитися в системі в момент T, дорівнює або $(n-1)$, або n, або $(n+1)$; всіма іншими можливостями ми нехтуємо як величинами вищого порядку малості. Таким чином, при $n>0$ для малих значень h отримаємо:

$$\begin{aligned}P_n(T+h)\approx(\lambda h)(1-\mu h)P_{n-1}(T)+(1-\lambda h)(1-\mu h)P_n(T)+\\ +(\lambda h)(\mu h)P_n(T)+(1-\lambda h)(\mu h)P_{n+1}(T).\end{aligned} \tag{14.11}$$

Перший доданок у правій частині (14.11) відповідає можливості одного надходження за відсутності виходів із системи у випадку знаходження $(n-1)$ вимог усередині системи в момент T. Другий і третій доданки відображають відповідно можливість відсутності як надходжень, так і виходів і можливість одного надходження й одного виходу у випадку перебування усередині системи в момент T n вимог. Остання складова у правій частині співвідношення (14.3) відповідає можливості одного виходу із системи при відсутності надходжень у випадку перебування усередині системи в момент T n вимог. Як показує символ $\approx$, вираз (14.3) є наближеним (точний вираз для $P_n(T+h)$ містив би складові з коефіцієнтами h^k, де $k\geq 2$).

Перенесемо $P_n(T)$ із правої частини співвідношення (14.11) у ліву, розділимо обидві частини цього співвідношення на h і спрямуємо h до нуля. За допомогою цих перетворень одержимо наступний вираз:

$$\frac{dP_n}{dT}=\lambda P_{n-1}(T)-(\lambda+\mu)P_n(T)+\mu P_{n+1}(T)\quad \text{при } n>0. \tag{14.12}$$

Рівняння (14.4) є точним, оскільки нами виконаний граничний перехід $h\to 0$. Аналогічно неважко одержати рівняння:

$$\frac{dP_0}{dT}=-\lambda P_0(T)+\mu P_1(T)\quad \text{для } n=0. \tag{14.13}$$

Система лінійних диференційних рівнянь (14.12) і (14.13) має єдиний розв'язок, якщо задані відповідні початкові умови (тобто кількість вимог *i*, що знаходилися в системі у момент 0). Цей розв'язок називається **несталим**, оскільки він безпосередньо залежить від T (нестаціонарний режим роботи).

Припустимо, що нами розглядаються значення $P_n(T)$ при $T\to\infty$. Якщо $P_n(T)$ прямує при цьому до деякого граничного значення (позначимо його через P_n) і якщо для даного граничного розподілу $M|n|$ виявляється скінченим, то система досягає стану статистичної рівноваги. Назвемо граничні значення P_n сталими або стаціонарними вірогідностями. Прикметник „стаціонарний" відображає наступну властивість системи: якщо кількість вимог, що знаходяться в ній, визначається в довільно обраний момент часу t за допомогою розподілу P_n, то для будь-якого

$h > 0$ P_n є також вірогідністю виявлення в системі n вимог у момент $T + h$. Значення P_n можна також інтепретувати як частку часу довільно великого періоду, протягом якого в системі знаходиться n вимог.

Якщо виконується умова

$$\rho = \frac{\lambda}{\mu} < 1, \tag{14.14}$$

то стаціонарні вірогідності P_n завжди існують. Значення ρ є трафік-інтенсивністю.

Розв'язок, що відповідає рівноважному стану $P_n(T) = P_n$ при будь-якому Т, легко знайти з умови $dP_n / dT = 0$, яка відображає той факт, що P_n не залежить від Т. Таким чином, для визначення P_n потрібно лише прирівняти до нуля похідні, що стоять у лівих частинах рівнянь (14.12) і (14.13). У результаті отримаємо:

$$0 = \lambda P_{n-1} - (\lambda + \mu) P_n + \mu P_{n+1} \quad \text{при } n = 1,2,3,..., \tag{14.15}$$

$$0 = -\lambda P_0 + \mu P_1 \quad \text{при } n = 0. \tag{14.16}$$

Система рівнянь у скінчених різницях (14.15) і (14.16) легко вирішується методом математичної індукції. Почавши з розгляду (14.16), отримуємо:

$$P_1 = P_0 \frac{\lambda}{\mu} = P_0 \rho. \tag{14.17}$$

Потім переходимо до (14.15), розглядаючи послідовно значення n = 1, 2, 3, . . . ;

у результаті отримаємо:

$$P_n = P_0 \rho^n. \tag{14.18}$$

$$\sum_{n=0}^{\infty} P_n = P_0 \sum_{n=0}^{\infty} \rho^n = \frac{P_0}{1-\rho} = 1, \tag{14.19}$$

звідкіля випливає, що $P_0 = 1 - \rho$, так що

$$P_n = (1 - \rho)\rho^n, n = 0,1,2,... \tag{14.20}$$

(геометричний розподіл), причому

$$M\left[\begin{array}{l}\text{Кількість вимог}\\ \text{в системі}\end{array}\right] = M[n] = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu - \lambda} \tag{14.21}$$

$$D[n] = \frac{\rho}{(1\text{-}\rho)^2}, \qquad P[n \geq N] = \rho^N.$$

Розподіл вірогідностей (14.20) залежить лише від трафік-інтенсивності $\rho = \lambda / \mu$. Оскільки $\rho = (1 - P_0)$ також є тією часткою повного часу з початку функціонування системи, протягом якої прилад не простоює, то цю величину називають іноді коефіцієнтом використання або коефіцієнтом завантаженості приладу. Істотним є те, що така інтерпретація зберігає силу навіть у тому випадку, коли не вводиться ніяких конкретизуючих припущень ні щодо розподілу тривалостей інтервалів між надходженнями, ні щодо розподілу тривалостей обслуговування (тобто коли модель належить до типу $GI / G / 1$).

Якщо через $P_n(T \setminus i)$ позначити розв'язання рівнянь (14.20) і (14.21) за умови, що в початковий момент часу $t = 0$ у черзі знаходилося i вимог, то можна показати, що

$$\left|P_n(T \mid i) - P_n\right| \leq e^{-T(\lambda + \mu - 2\sqrt{\lambda\mu})}.$$

Отже, $P_n(T|i)$ прямує до P_n не повільніше, ніж при зміні за експоненційним законом. Зауважимо, що при $\lambda \to \mu$ коефіцієнт при T в показнику прямує до нуля. Отже, інтервал часу T, протягом якого значення $P_n(T \mid i)$ стане майже рівним P_n, може бути дуже тривалим: ця властивість „повільного прямування до стаціонарного режиму" виявляється особливо помітною при великих значеннях ρ і малих значеннях i.

14.2.2. ОПЕРАЦІЙНІ ХАРАКТЕРИСТИКИ СИСТЕМИ M/M/1

Математичне сподівання довжини черги можна обчислити, враховуючи, що

$$\text{Довжина черги} = \begin{cases} \text{Число вимог в системі , якщо } & n = 0, \\ \text{Число вимог в системі -1, якщо } & n > 0, \end{cases} \tag{14.22}$$

так що M [Довжина черги] $= 0 \times P_0 + \sum_{n=1}^{\infty} (n-1) P_n = \sum_{n=0}^{\infty} n P_n - \sum_{n=1}^{\infty} P_n =$

$$= M[n] - (1 - P_0) = \frac{\rho^2}{1-\rho} = \frac{\lambda^2}{\mu(\mu - \lambda)}. \tag{14.23}$$

Розглянемо тепер інтервали часу, коли прилад не зайнятий обслуговуванням (простоює). Оскільки інтервали простою обслуговуючого приладу починаються з моментів завершення обслуговування і закінчуються при надходженні нової вимоги, тривалості простою приладів мають такий же розподіл, як і тривалості інтервалів між надходженнями вимог, тобто характеризуються розподілом із математичним сподіванням $1/\lambda$. Нехай T настільки велике, що ми можемо оперувати середніми значеннями операційних характеристик без яких-небудь побоювань. При цьому число окремих періодів часу, протягом яких прилад опиняється в межах T незайнятим, дорівнює $T(1-\rho)/(1/\lambda) = \lambda T(1-\rho)$. Оскільки періоди обслуговування і періоди простою строго чергуються, то $\lambda T(1-\rho)$ визначає також число зайнятих періодів протягом повного періоду T, а ρT є сумарною тривалістю періодів, коли прилад зайнятий обслуговуванням Отже,

$$M\begin{bmatrix}\text{Тривалість періоду,}\\ \text{коли прилад зайнятий}\end{bmatrix} = \frac{\rho T}{\lambda T(1-\rho)} = \frac{1}{\mu-\lambda}, \tag{14.24}$$

$$P\begin{bmatrix}\text{Число вимог , обслужених за}\\ \text{час періоду зайнятості приладу}\end{bmatrix} = \mu M\begin{bmatrix}\text{Тривалість періоду}\\ \text{зайнятості приладу}\end{bmatrix} = \frac{1}{1-\rho} \tag{14.25}$$

Формули (14.24) і (14.25), взагалі кажучи, справедливі при будь-якому розподілі тривалостей обслуговування (тобто застосовні й у випадку моделі M/G/1).

Розглянемо тепер щільність розподілу для тривалості перебування вимоги в системі обслуговування. Час, протягом якого вимога знаходиться в системі, складається з тривалості чекання нею на обслуговування в черзі і тривалості самої процедури обслуговування. Тепер припустимо, що система знаходиться в стані статистичної рівноваги, так що кожна з вимог, які знову надходять, з ймовірністю P_n, визначеною за формулою (14.20), має перед собою n вимог, що очікують на обслуговування. Припустимо, що для черги діє дисципліна „першим прийшов - першим обслуговуєшся". У цьому випадку повний час перебування вимоги в системі є сумою $n+1$ незалежних вибірок із множини значень, що приймає випадкова змінна, яка характеризується експоненційним розподілом, тобто описується гамма-розподілом із густиною :

$$\frac{\mu(\mu y)^n e^{-\mu y}}{n!},\ y \ge 0. \tag{14.26}$$

Отже, щільність розподілу для часу перебування в системі вимоги, що надійшла в довільно обраний момент часу, визначається наступною формулою:

$$h(w) = \sum_{n=0}^{\infty}(1-\rho)\rho^n\left[\frac{\mu(\mu w)^n e^{-\mu w}}{n!}\right] = \mu(1-\rho)e^{-\mu(1-\rho)w} \tag{14.27}$$

(експоненційний розподіл), при чому

$$M\begin{bmatrix}\text{Час перебування}\\ \text{в системі}\end{bmatrix} = M[w] = \frac{1}{\mu(1-\rho)} = \frac{1}{\mu-\lambda}, \tag{14.28}$$

а

$$M\begin{bmatrix}\text{Час перебу-}\\ \text{вання в черзі}\end{bmatrix} = M\begin{bmatrix}\text{Час перебуван-}\\ \text{ня в системі}\end{bmatrix} - M\begin{bmatrix}\text{Час обслу-}\\ \text{говування}\end{bmatrix} =$$

$$= \frac{1}{\mu}\frac{\rho}{1-\rho} = \frac{\lambda}{\mu(\mu-\lambda)}. \tag{14.29}$$

При фіксованому значенні ρ середня тривалість перебування вимоги в системі, так само як і середня тривалість перебування її в черзі, обернено пропорційна швидкості обслуговування μ.

Припустимо тепер, що нас цікавить лише час, протягом якого вимога змушена очікувати, знаходячись у черзі, тобто виключимо з розгляду як час обслуговування кожної з вимог, так і ті випадки, коли вимога, що надійшла, застає прилад незайнятим, тобто готовим до негайного обслуговування. Доведено, що умовна вірогідність для тривалості перебування в черзі тих клієнтів, яким доводиться витрачати час на очікування, визначається щільністю розподілу (14.27) при будь-якому розподілі тривалостей інтервалів між надходженнями (тобто формула (14.27) справедлива і для моделей типу *GI/M/1*). Отже, математичне сподівання $M[w]$, визначене за формулою (14.28), є також умовним математичним сподіванням тривалості перебування вимоги в черзі за умови, що вона дійсно змушена гаяти час на чекання.

Аналіз СМО на чутливість.

Основні операційні характеристики одноканальної системи масового обслуговування з дисципліною черги „першим прийшов - першим обслуговуєшся" є функціями ρ і μ.

При зростанні трафік-інтенсивності ρ очікувані значення таких параметрів, як число вимог, що знаходяться в системі, довжина черги, повний час перебування вимоги в системі і чистий час її перебування в черзі також починають швидко зростати. Хоча всі перераховані нами показники при достатньо великих $\rho<1$ можуть бути як завгодно великими, може пройти дуже багато часу перед тим, як система досягне рівноважного стану.

При заданій швидкості обслуговування μ, коли трафік-інтенсивність ρ незначна, основна частка середнього часу перебування вимоги в системі пов'язана із самою процедурою обслуговування (середня тривалість процедури обслуговування дорівнює $1/\mu$); проте при зростанні ρ, тобто при збільшенні інтенсивності вхідного потоку λ велика частина часу перебування вимоги в системі (у сенсі середнього значення) обумовлена чеканням.

14.3. РОЗШИРЕННЯ СИСТЕМИ M/M/1: СКІНЧЕНА ЧЕРГА, ДОВІЛЬНИЙ РОЗПОДІЛ ТРИВАЛОСТЕЙ ОБСЛУГОВУВАННЯ, ВИМОГИ З ПРІОРИТЕТАМИ

Поки що ми не накладали ніяких обмежень на сумарну кількість вимог, що може виявитися в обслуговуючій системі в той або інший момент часу. Припустимо тепер, що в системі може бути не більш M вимог і, отже, у будь-який момент часу в черзі дозволяється знаходитися не більш, ніж $M-1$ вимогам. (Як приклад, що ілюструє такого типу умову, можна навести бензозаправну станцію з одною бензоколонкою, розташованою у вузькому провулку, що виходить на одну з центральних вулиць міста). Якщо вимога надходить у момент, коли в обслуговуючій системі вже знаходиться M об'єктів, то такій вимозі відмовляється в обслуговуванні і, таким чином, вона втрачається. З цієї причини такого виду моделі іноді називають моделями масового обслуговування з відмовами (втратами). Істотна різниця між попередньою моделлю і моделлю зі скінченою довжиною черги полягає в тому, що в останньому випадку статистична рівновага досягається при довільному значенні $\rho=\lambda/\mu$.

Рівняння в скінчених різницях (14.15) і (14.16), які відповідають сталому режиму при $n=0,1,\dots,M-1$, справедливі й у випадку скінченої черги, а при $n=M$ справедливе рівняння

$$0=\lambda P_{M-1}-\mu P_M. \tag{14.30}$$

Спільний розв'язок рівняння (14.30) і згаданих вище рівнянь (14.15) і (14.16) при $n=0,1,\dots,M-1$ є наступним:

$$P_n=\begin{cases}\left(\dfrac{1-\rho}{1-\rho^{M+1}}\right)\rho^n & \text{при } \lambda\neq\mu,\\[2ex] \dfrac{1}{M+1} & \text{при } \lambda=\mu.\end{cases} \tag{14.31}$$

Зрозуміло, що при $\mu>\lambda$ і $M\to\infty$ вираз (14.31) зводяться до (14.20). При $\lambda\neq\mu$,

$$M\begin{bmatrix}\text{Кількість вимог}\\ \text{в системі обслуговування}\end{bmatrix}=M[n]=$$

$$=\frac{\rho}{1-\rho}\left[\frac{1-(M+1)\rho^M+M\rho^{M+1}}{1-\rho^{M+1}}\right]=\frac{\rho}{1-\rho}-\frac{(M+1)\rho^{M+1}}{1-\rho^{M+1}} \tag{14.32}$$

Зауважимо, що при $\lambda<\mu$ середня кількість вимог, що знаходяться в системі, є меншою порівняно з відповідним показником, обчисленим для моделі з необмеженою довжиною черги (тобто за допомогою формули (14.13)). Аналогічним чином доводиться, що при $\lambda=\mu$

$$M\begin{bmatrix}\text{Кількість вимог}\\ \text{в системі обслуговування}\end{bmatrix}=M[n]=\frac{M}{2}. \tag{14.33}$$

Деякі тонкощі необхідно враховувати як при визначенні часу перебування вимоги в системі обслуговування, так і тривалості її чекання в черзі. Якщо вимога надходить у той момент, коли в системі вже знаходиться M інших вимог, то вона не може стати до черги, і, отже, час її перебування в системі дорівнює нулю. Тому середній час перебування в системі можна визначити або з огляду на усі вимоги незалежно від того, була для них можливість приєднатися до черги чи ні, або приймаючи в розрахунок лише ті вимоги, котрим вхід був дійсно дозволений. Ми приймемо другий варіант, оскільки в переважній більшості випадків нас цікавлять затримки в обслуговуванні лише тих вимог, що мали можливість ввійти в систему. Тоді за умови, що вимога, яка надходить у деякий довільно обраний момент часу, одержує право на чекання в черзі і що черга характеризується дисципліною „першим прийшов – першим обслуговується”, для середньої тривалості перебування даної вимоги в системі обслуговування одержуємо наступний вираз:

$$M\begin{bmatrix}\text{Тривалість перебування вимоги}\\ \text{в системі обслуговування}\end{bmatrix}=$$

$$M[w]=\frac{\rho}{\mu(1-\rho)}\left[\frac{1-\mu\rho^{M-1}+(M-1)\rho^M}{1-\rho^M}\right]+\frac{1}{\mu}=$$

$$=\frac{1}{\mu(1-\rho)}-\frac{M\rho^M}{\mu(1-\rho^M)} \qquad \text{при} \quad \lambda\neq\mu. \tag{14.34}$$

а

$$M\begin{bmatrix}\text{Тривалість пребування вимоги}\\ \text{в системі обслуговування}\end{bmatrix}=M[w]=\frac{1}{\mu}\frac{M+1}{2} \quad \text{при} \quad \lambda=\mu. \tag{14.35}$$

За допомогою витончених методів аналізу вдається також одержати формули, що визначають операційні характеристики моделі масового обслуговування зі скінченою чергою в умовах нестаціонарного режиму. У цьому випадку вигляд розв'язку значно ускладнюється, тому що система функціонує в перехідному режимі (тобто ще не досягла стану статистичної рівноваги). Позначимо через i кількість вимог, що знаходяться в системі в момент 0. Тоді при будь-яких λ і μ у випадку $M=1$, тобто коли доступ в обслуговуючу систему забезпечується лише при незайнятому обслуговуючому приладі, у момент 0 кількість вимог у системі дорівнює 0, і формула для перехідного режиму набуває простішого вигляду,

$$P_0(T\mid i=0)=\frac{\mu}{\lambda+\mu}+\frac{\lambda}{\lambda+\mu}e^{-(\lambda+\mu)T},$$

$$P_1(T\mid i=0)=\frac{\mu}{\lambda+\mu}-\frac{\lambda}{\lambda+\mu}e^{-(\lambda+\mu)T}$$

при будь-яких λ і μ. Таким чином, при $T\to\infty$ різниця між значеннями вірогідностей в умовах перехідного періоду і граничними значеннями (тобто значеннями, що відповідають стану статистичної рівноваги) зменшується за експоненційним законом.

Випадки довільного розподілу тривалостей обслуговування. Інші

варіанти одноканальних моделей, у яких розподіл тривалостей інтервалів між надходженнями вимог і розподіл тривалостей інтервалів обслуговування не є експоненційними, є доволі складними і тут не розглядаються. Проте нескладним виявляється обчислення очікуваної кількості вимог у СМО і середньої тривалості їх перебування в ній у тому випадку, коли вхідний потік вимог характеризується експоненційним розподілом, а щодо вигляду розподілу тривалостей обслуговування не робиться ніяких спеціальних припущень (тобто у випадку, коли модель ставиться до типу *M/G/1*). Нехай

$$V=\mathrm{D}\begin{bmatrix}\text{Тривалість}\\ \text{обслуговування}\end{bmatrix}=\int_0^\infty\left(t-\frac{1}{\mu}\right)^2 g(t)dt, \tag{14.36}$$

де, як і раніше, $1/\mu$ — середнє значення тривалості процедури одного обслуговування, а $g(t)$ - щільність розподілу тривалостей обслуговування. Тоді

$$M\begin{bmatrix}\text{Кількість вимог}\\ \text{в системі обслуговування}\end{bmatrix}=\rho+\frac{\lambda^2V+\rho^2}{2(1-\rho)}, \tag{14.37}$$

$$M[\text{Довжина черги}]=\frac{\lambda^2V+\rho^2}{2(1-\rho)}, \tag{14.38}$$

$$M[\text{Прилад простоює}]=P_0=1-\rho, \tag{14.39}$$

де, як звичайно, $\rho=\lambda/\mu<1$.

Якщо припустити, що діє дисципліна черги „першим прийшов – першим обслуговується”, то для середньої тривалості перебування вимоги в системі обслуговування буде справедливим наступне співвідношення:

$$M\begin{bmatrix}\text{Тривалість пребування вимоги}\\ \text{в системі обслуговування}\end{bmatrix}=M\begin{bmatrix}\text{Тривалість очікування}\\ \text{вимоги в черзі}\end{bmatrix}+ \tag{14.40}$$

$$+M\begin{bmatrix}\text{Тривалість процедури}\\ \text{обслуговування}\end{bmatrix}=\frac{\lambda}{\mu^2}\left[\frac{\mu^2V+1}{2(1-\rho)}\right]+\frac{1}{\mu}=\frac{1}{\lambda}\left[\rho+\frac{\lambda^2V+\rho^2}{2(1-\rho)}\right]$$

Співвідношення (14.37) і (14.40) є формулами Полачека - Хінчіна.

Неважко перевірити, що якщо щільність розподілу $g(t)$ є експоненційною і, отже, $D=1/\mu^2$, то формули (14.37), (14.38) і (14.40) перетворяться відповідно у формули (14.21), (14.23) і (14.28). Ці формули визначають також відповідні математичні сподівання у випадку, коли швидкість обслуговування постійна, тому що при цьому D=0. Іншими словами, при D=0 отримані вище формули описують модель типу M/D/1. Середні значення всіх зазначених вище показників є лінійними функціями від D і залежать лише від частоти надходження вимог λ, графік-інтенсивності ρ і дисперсії тривалостей обслуговування D і не залежать ні від яких інших параметрів, що характеризують розподіл тривалостей інтервалів між надходженнями і розподіл тривалостей обслуговування.

Зауважимо далі, що в умовах статистичної рівноваги

$$M\left[\begin{matrix}\text{Кількість вимог}\\ \text{в системі обслуговування}\end{matrix}\right]=\lambda M\left[\begin{matrix}\text{Тривалість перебування вимоги}\\ \text{в системі обслуговування}\end{matrix}\right]. \quad (14.41)$$

Співвідношення (14.41), взагалі кажучи, дійсне і при загальніших припущеннях щодо режиму функціонування СМО; зокрема, воно виконується й у випадку багатоканальної моделі.

Наведемо начерк одержання формули (14.37). Щоб спростити схему міркувань, припустимо, що система розглядається лише в ті моменти часу, коли процедура обслуговування виявляється завершеною і відповідна вимога щойно покинула систему. Якщо система знаходиться в стані рівноваги, співвідношення (14.37) діє й у будь-який інший довільно обраний момент часу; проте доведення формули (14.37) без використання тільки що прийнятого нами припущення щодо послідовності моментів спостереження за станом системи є набагато складнішим.

Нехай n - число вимог, що знаходяться в системі в момент T, що збігається з моментом, коли систему щойно покинула остання обслужена вимога, а n' - число вимог, що знаходяться в системі в момент T', що збігається з моментом, коли систему тільки що покинула така обслужена вимога. Позначимо через j число вимог, що надходять у систему за час із моменту T до моменту T'. Тоді, як неважко отримати,

$$n'=\begin{cases} j, \text{ якщо } n=0,\\ n-1+j, \text{ якщо } n>0.\end{cases} \quad (14.42)$$

Співвідношення (14.34) можна записати в зручнішому вигляді, ввівши в розгляд дельта-функцію Дірака:

$$\delta(n)=\begin{cases}0 & \text{при } n=0,\\ 1 & \text{при } n\neq 1.\end{cases} \quad (14.43)$$

Дійсно, з урахуванням (14.43) співвідношення (14.42) (І) набуде вигляду:

$$n'=n-\delta(n)+j \qquad \text{при будь-якому } n\geq 0. \quad (14.44)$$

Перед тим, як приступати до обчислення $M[n]$, пригадаємо, що кількість надходжень j на заданому інтервалі обслуговування тривалістю t має пуасонівський розподіл із середнім значенням λt. Отже,

$$M[j]=\int_0^\infty M[j\,|\,t]\mathrm{g}(t)dt=\int_0^\infty \lambda\, t\mathrm{g}(t)dt=\frac{\lambda}{\mu}=\rho, \quad (14.45)$$

де $M[j\,|\,t]$ - умовне математичне сподівання j при заданому t, а $g(t)$ - щільність розподілу тривалостей обслуговування. Аналогічним чином знаходимо:

$$M\left[j^2\right]=\int_0^\infty M\left[j^2\,|\,t\right]\mathrm{g}(t)dt=\int_0^\infty\left[\lambda t+(\lambda t)^2\right]\mathrm{g}(t)dt=\frac{\lambda}{\mu}+$$

$$+\lambda^2\left\{\{D\ [\text{Тривалість обслуговування}]+\left(\frac{1}{\mu}\right)^2\right\}=\rho+\lambda^2V+\rho^2. \quad (14.46)$$

Можна довести, що

$$P_0=1-\rho \ \text{ і } \ M[\delta]=M\left[\delta^2\right]=\rho. \quad (14.47)$$

Піднесемо тепер до квадрату обидві частини співвідношення (14.36):

$$M\left[(n')^2\right]=M\left[n^2\right]+M\left[\delta^2\right]+M\left[j^2\right]-2M[n\delta(n)]+$$

$$+2M[nj]-2M[\delta(n)j]$$

Після необхідних перетворень одержимо:

$$M\left[(n')^2\right]=M\left[n^2\right]+\rho+M\left[j^2\right]-2M[n]+2M[n]M[j]-2\rho M[j]. \quad (14.48)$$

При цьому нами було враховано, що $M[\delta]=M\left[\delta^2\right]=\rho$ і $n\delta(n)=n$, а також використана умова взаємої незалежності j і n. Оскільки у стаціонарному режимі $M\left[(n')^2\right]=M\left[n^2\right]$, ми можемо, використовуючи (14.45), записати (14.48) у такому вигляді:

$$M[n] = \frac{M\left[j^2\right] + \rho - 2\rho^2}{2(1-\rho)}. \qquad (14.49)$$

Після спрощень одержуємо:

$$M[n] = \rho + \frac{\lambda^2 V + \rho^2}{2(1-\rho)}. \qquad (14.50)$$

Дисципліна черги при наявності пріоритету. У багатьох реальних ситуаціях дисципліна черги не узгоджується з правилом „першим прийшов першим обслуговується". Припустимо, що вимоги, які надходять на вхід системи обслуговування, можна підрозділяти на r різних категорій, кожній із яких приписується пріоритет k ($k = 1, 2, \ldots, r$), відображений відповідним номером. Припустимо, що пріоритет падає зі збільшенням номера, тобто пріоритет 1 виявляється найвищим, а пріоритет r - найнижчим. Як тільки прилад закінчує обслуговування тієї або іншої вимоги, він негайно переходить до обслуговування, віддаючи при цьому перевагу тій з вимог, що знаходяться в черзі, пріоритет якої виявляється найвищим (якщо в черзі знаходиться декілька вимог з однаковими пріоритетами, черговість їх обслуговування визначається правилом „першим прийшов – першим обслуговується".) У деяких випадках, крім того, передбачається, що при надходженні в систему обслуговування вимоги з вищим пріоритетом у порівнянні з пріоритетом вимоги, що у цей момент знаходиться в процесі обслуговування, система кидає вже почату процедуру обслуговування і перемикається на обслуговування вимоги з вищим пріоритетом

Припустимо, що потік вимог, які мають пріоритети $k = 1, 2, \ldots, r$ є пуасонівським, причому відповідні середні частоти надходжень дорівнюють λ_k. Припустимо також, що кожний пріоритет k ($k = 1, 2, \ldots, r$) характеризується розподілом тривалостей обслуговування з щільністю $g_k(t)$ цілком довільного вигляду і

$$M\left[\begin{array}{l}\text{Тривалість процедури}\\ \text{обслуговування}\end{array}\middle|\,\text{Пріоритет} = k\right] = 1/\mu_k, \qquad (14.51)$$

$$D_k = D\left[\begin{array}{l}\text{Тривалість процедури}\\ \text{обслуговування}\end{array}\middle|\,\text{Пріоритет} = k\right].$$

Визначимо $\delta_k = \sum_{j=1}^{k} \lambda_j / \mu_j$, $\delta_0 = 0$, і будемо припускати, що $\delta_r < 1$. (Це гарантує можливість поступового переходу системи в стан статистичної рівноваги).

Розглянемо систему в момент часу безпосередньо за завершенням чергової процедури обслуговування. Тоді для вимоги, що знову надійшла, із пріоритетом k середній час чекання в черзі визначається формулою:

$$M\left[\begin{array}{l}\text{Тривалість очікування}\\ \text{в черзі}\end{array}\middle|\,\text{пріоритет } k\right] = \frac{\sum_{j=1}^{r} \lambda_j \left[V_j + \left(-\frac{1}{\mu_j}\right)^2\right]}{2(1-\delta_{k-1})(1-\delta_k)}. \qquad (14.52)$$

Неважко переконатися в тому, що при r=1 вираз (14.52) зводиться до (14.40).

Оскільки $\lambda_k / \sum \lambda_j$, є можливість того, що вимога, яка знову надійшла, має пріоритет k, середній час перебування в черзі довільним способом обраної вимоги обчислюється за наступною формулою: (14.53)

$$M\left[\begin{array}{l}\text{Тривалість очікування}\\ \text{в черзі}\end{array}\right] = \frac{\sum_{k=1}^{r} \lambda_k M\left[\begin{array}{l}\text{Тривалість очікування}\\ \text{в черзі}\,|\,\text{пріоритет } k\end{array}\right]}{\sum_{k=1}^{r} \lambda_k}$$

тобто математичне сподівання тривалості перебування в черзі, асоційоване з повним ансамблем вимог, є сумою пiдансамблів вимог, кожна з яких має пріоритет k ($k = 1, 2, \ldots, r$).

У деяких задачах тривалість процедури обслуговування залежить від номера (або, як кажуть, від рівня) пріоритету. Тоді, забезпечуючи вимогам із вищими рівнями пріоритету прискорене обслуговування, ми тим самим скорочуємо середній час очікування в черзі, обчислений за сумарним вихідним потоком обслужених вимог. У якості ілюстрації цього твердження звернемося до випадку, коли $\lambda = 18$, а $\mu = 20$. (так що коефіцієнт завантаженості обслуговуючої системи при цьому дорівнює 0,9, а середній час чекання в черзі 0,45). Перейдемо тепер до схеми обслуговування з двома рівнями пріоритету і припустимо, що $\lambda_1 = \lambda_2 = 9$, а $\mu_1 = 30$ і $\mu_2 = 15$, так що коефіцієнт завантаженості $\delta_r = 9/30 + 9/15 = 0{,}9$ (тобто залишається таким же, як і в схемі обслуговування за відсутності пріоритетів). За допомогою вищенаведених формул легко переконатися, що в цьому випадку середня тривалість перебування в черзі вимоги з пріоритетом 1 дорівнює 1/14 = 0,0714, а вимоги з пріоритетом 2 — 10/14=0.7142, тобто менше ніж у випадку, коли $\mu_1 = \mu_2 = \mu = 20$. При цьому середня тривалість перебування в черзі, обчислена на сумарному ансамблі вимог, що обслуговуються, дорівнює 0,3928, тобто виявляється меншою, у порівнянні з аналогічною операційною характеристикою системи без пріоритетів, а також у порівнянні з випадком, коли $\mu_1 = \mu_2 = 20$.

14.4. БАГАТОКАНАЛЬНА МОДЕЛЬ З ПУАСОНІВСЬКИМ ВХІДНИМ ПОТОКОМ І ЕКСПОНЕНЦІЙНИМ РОЗПОДІЛОМ ТРИВАЛОСТЕЙ ОБСЛУГОВУВАННЯ

Зрозуміло, що в переважній більшості випадків СМО є багатоканальними, і, отже, моделі з S обслуговуючими приладами (де S > 1) є важливими. Узагальнимо отримані результати на випадок декількох обслуговуючих пристроїв. Постульована при цьому дисципліна черги виглядає дещо спрощено для більшості ситуацій, із якими доводиться зустрічатися в дійсності. Проте отримані результати можна розглядати як дуже корисні, оскільки вони принаймні дозволяють у найпершому наближенні оцінити функціональні характеристики складніших СМО.

Нехай S = число обслуговуючих приладів. (14.54)

Будемо припускати, що

а) щільність розподілу тривалостей інтервалів між надходженнями вимог має вигляд $f(t) = \lambda e^{-\lambda t}$. (14.55)

б) щільність розподілу тривалостей обслуговування кожним із приладів має вигляд $l(t) = \mu e^{-\mu t}$, причому тривалості обслуговування взаємонезалежні як для окремо узятого приладу, так і для системи загалом.

Діаграма інтенсивностей переходів для цієї системи наведена на рис. 14.3.

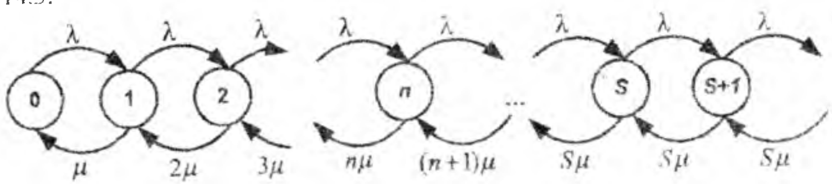


Рис. 14.3. Діаграма інтенсивностей переходів для багатоканальної системи з необмеженою чергою

інтенсивність потоку обслуговувань не залишається сталою, а зі збільшення кількості вимог в СМО від 0 до S збільшується від величини μ до $S\mu$, оскільки відповідно збільшується число каналів обслуговування. При числі вимог в СМО, більшому за S, інтенсивність потоку обслуго-вувань зберігається рівною $S\mu$.

Рівняння в скінчених різницях, аналогічні рівнянням для випадку одноканальної СМО, що визначають P_n в умовах сталого режиму, мають такий вигляд:

$$\begin{aligned} &0 = \lambda P_{n-1} - (\lambda + n\mu)P_n + (n+1)\mu P_{n+1} \quad \text{при } S > n \geq 1, \\ &0 = \lambda P_{n-1} - (\lambda + S\mu)P_n + S\mu P_{n+1} \quad \text{при } n \geq S. \end{aligned} \tag{14.56}$$

Розв'язки системи рівнянь (14.48) мають наступний вигляд:

$$\begin{aligned} &P_n = \frac{\rho^n}{n!} \cdot P_0 \quad \text{при } S > n \geq 0, \\ &P_n = \frac{\rho^n}{S!S^{n-S}} \cdot P_0 \quad \text{при } n \geq S, \end{aligned} \tag{14.57}$$

де $\delta = \lambda / \mu$, а $P_0 = \dfrac{1}{\sum_{j=0}^{S-1} \dfrac{\rho^j}{j!} + \dfrac{\rho^S}{S!\left[1 - (\rho / S)\right]}}$. (14.58)

Сталий режим функціонування СМО, що характеризується співвідношеннями (14.57) і (14.58), можливий за умови $\lambda < \mu S$. (або $\rho < S$).

У випадку необмеженої кількості обслуговуючих приладів (наприклад, в умовах самообслуговування) перша з формул (14.57) стає застосовною для будь-якого значення n. Отже, у цьому випадку P_n приймає пуасонівський вигляд, причому $M[n] = \rho$. Для такої моделі використовують символічне позначення $M/M/\infty$. При $S = \infty$ пуасонівський характер P_n має місце фактично при будь-якому вигляді розподілу тривалостей обслуговування, тобто й у випадку $M/G/\infty$.

Формули (14.57) застосовні й у тому випадку, коли діє обмеження на сумарну кількість вимог $M(\geq S)$, що може знаходитися в системі. При цьому $n \leq M$, а P_0 визначається з умови $\sum_{n=0}^{M} P_n = 1$.

Відзначимо, крім того, що формули (14.57) залишаються при цьому справедливими й у випадку, коли $\lambda > \mu S$.

Операційні характеристики. Коли P_n знайдені, більшість операційних характеристик цієї моделі обчислюються за допомогою елементарних алгебраїчних операцій. До числа дуже важливих характеристик СМО належить вірогідність того, що всі прилади виявляться зайнятими:

$$P\begin{bmatrix}\text{Повне завантаження}\\ \text{всіх каналів (приладів)}\end{bmatrix} = P[n \ge S] = \frac{\rho^S \mu S}{S!(\mu S - \lambda)} \cdot P_0 = \frac{\rho^S}{S!(1-\rho/S)} \cdot P_0. \quad (14.59)$$

Величина $P[n \ge S+1]$ визначає частку часу, протягом котрої вимоги фактично перебувають у системі.

Введемо в розгляд наступні величини:

$$P[x = s \mid \rho] = \frac{\rho^S e^{-\rho}}{S!} \quad \text{і} \quad P[x < s \mid \rho] = \frac{\rho^x e^{-\rho}}{x!}. \quad (14.60)$$

Тоді (14.51) можна записати в наступному вигляді:

$$P\begin{bmatrix}\text{Повне завантаження}\\ \text{всіх каналів}\end{bmatrix} = \frac{P(x = S \mid \rho)}{P(x = S \mid \rho) + (1-\rho/S)P(x < S \mid \rho)}. \quad (14.61)$$

З урахуванням (14.44), (14.53)отримаємо:

$$M\begin{bmatrix}\text{Довжина}\\ \text{черги}\end{bmatrix} = P\begin{bmatrix}\text{Повне завантаження}\\ \text{всіх каналів}\end{bmatrix} \cdot \frac{\rho}{S-\rho} = \frac{\rho^S \lambda \mu S}{(\mu S - \lambda)^2} \cdot P_0, \quad (14.62)$$

$$M\begin{bmatrix}\text{Кількість обслуговуваних}\\ \text{вимог}\end{bmatrix} = \sum_{n=0}^{S-1} nP_n + S\sum_{n=S}^{\infty} P_n = \rho, \quad (14.63)$$

$$M\begin{bmatrix}\text{Кількість вимог}\\ \text{в СМО}\end{bmatrix} = M[n] =$$

$$M\begin{bmatrix}\text{Довжина}\\ \text{черги}\end{bmatrix} + M\begin{bmatrix}\text{Кількість обслуговуваних}\\ \text{вимог}\end{bmatrix}. \quad (14.64)$$

Для розглянутої моделі

$$M\begin{bmatrix}\text{Тривалість перебування}\\ \text{в системі обслуговування}\end{bmatrix} = M\begin{bmatrix}\text{Кількість}\\ \text{вимог в СМО}\end{bmatrix} \quad (14.65)$$

і, отже, розділивши обидві частини співвідношення (14.65) на l, отримаємо:

$$M\begin{bmatrix}\text{Тривалість пере-}\\ \text{бування в СМО}\end{bmatrix} = \frac{P\begin{bmatrix}\text{Повне заванта-}\\ \text{ження всіх каналів}\end{bmatrix}}{\mu S - \lambda} + \frac{1}{\mu}. \quad (14.66)$$

У правій частині (14.66) перший доданок є тривалістю очікування в черзі, а другий – тривалістю процедури обслуговування.

Важливою особливістю такої системи є те, що її вихідний потік на інтервалі T має пуасонівський розподіл із середнім значенням λT вимог, що обслуговуються, за одиницю часу. Розглянемо тепер велику СМО, що складається з груп послідовно включених приладів, тобто організованих таким чином, що вихідний потік однієї групи приладів є вхідним для іншої групи приладів. Якщо кожну з подібних груп приладів можна описати за допомогою багатоканальної моделі розглянутого вище типу, то середні значення операційних характеристик системи легко обчислити, проаналізувавши спочатку кожну з груп як цілком автономну, а потім склавши отримані результати в припущенні, що частоти надходження вимог λ на вході кожній із груп однакові.

Аналіз на чутливість.

Результати наведеного вище порівняння різноманітних варіантів і режимів функціонування СМО справедливі і для інших значень λ, μ і S. Зокрема, при заданих значеннях λ і μS P [всі обслуговуючі прилади зайняті] зростає при спаданні S, те ж саме відбувається при цьому і із середньою кількістю вимог, що очікують у черзі, і із середньою тривалістю очікування початку обслуговування. Проте середня кількість вимог, що знаходяться в системі, а також і середній час перебування вимоги в системі скорочуються при спаданні S.

Таким чином, в залежності від характеру практичної задачі, що розв'язується, керівник може варіювати значеннями λ, μ і S і в результаті знайти необхідний оптимальний варіант. Якщо задача має просту структуру, розв'язок може бути отриманий в аналітичному вигляді, поданий за допомогою таблиць і графіків, що полегшують процес оптимізації.

Дисципліна черги на грунті системи пріоритетів. У цьому випадку вимоги кожної із r категорій надходять у систему відповідно до пуасонівського розподілу. З пріоритетом k (k = 1,2, . . . , r) пов'язана частота надходження λ_k, причому найвищим є пріоритет 1, а найнижчим - пріоритет r. Ми припускатимемо, що розподіл тривалостей обслуговування для вимог кожної з r категорій носить експоненційний характер при середній швидкості обслуговування μ. Як і в попередньому випадку, визначимо:

$$\delta_k = \sum_{j=1}^{k} \frac{\lambda_j}{\mu_j} = \frac{\sum_{j=1}^{k} \lambda_j}{\mu}, \qquad \delta = 0. \tag{14.67}$$

Нехай $\delta_r < S$, що гарантує можливість асимптотичного переходу системи в стан рівноваги. Тоді (14.68)

$$M\begin{bmatrix}\text{Тривалість очікування} \\ \text{в черзі} \mid \text{пріоритет } k\end{bmatrix} = \frac{S/\mu}{(S-\delta_{k-1})(S-\delta_k} \cdot P\begin{bmatrix}\text{Всі обслуговуючі} \\ \text{прилади зайняті}\end{bmatrix}.$$

Середній час перебування в системі довільної вимоги (тобто середня тривалість перебування в системі, обчислена на сумарному ансамблі, що містить вимоги всіх категорій) залишається таким же, як і в системі без пріоритетів при середній частоті надходження $\lambda = \sum \lambda_j$.

Розглянемо, наприклад, випадок, коли λ =36, S=2 і μ =20. У системі без пріоритетів середня тривалість очікування в черзі дорівнює 0,2132. Припустимо тепер, що мають місце два рівні пріоритетів, причому $\lambda_1 = \lambda_2 = 18$. За допомогою (14.68) легко переконатися, що середня тривалість очікування в черзі вимоги з пріоритетом 1 дорівнює 0,0387 (=0,8526/22), а вимоги з пріоритетом 2 дорівнює 0,3875 (=0,8526/2,2). Чим менше значення λ_1, тим менше буде середня тривалість очікування для вимог обох категорій.

14.5. ІНШІ МОДЕЛІ СИСТЕМ МАСОВОГО ОБСЛУГОВУВАННЯ

Майже всі результати, розглянуті в цій темі, стосуються моделей масового обслуговування, у яких процес надходження вимог є пуасонівським, а тривалості інтервалів обслуговування однієї вимоги мають експоненційний розподіл. В усіх розглянутих нами моделях передбачалося, що існує лише одна черга з дисципліною „першим прийшов – першим обслуговуєшся" (ситуації з декількома чергами аналізувалися нами виходячи з припущення, що всю систему загалом можна уявити в вигляді сукупності підсистем, що функціонують паралельно і незалежно одна від іншої).

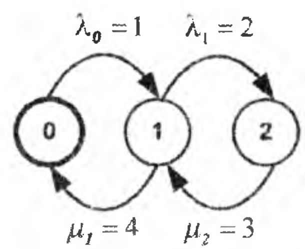
Ці результати не так важко узагальнити на випадки, коли умови задачі дещо видозмінені. Так, наприклад, дуже просто вдається врахувати можливість відмов (тобто тенденцію клієнтів-вимог утримуватися від приєднання до черги в міру того, як її довжина зростає), а також можливість приєднання до черги клієнтів з обмеженим часом чекання (таких, що вибувають із системи обслуговування до того, як їх встигнуть обслужити). В принципі можна одержати формули, що дозволили б враховувати різноманітні, як правило, лише дуже прості вигляди дисципліни черги, такі, як обслуговування „із випадковим вибором вимог" і обслуговування за схемою „прийшов останнім – обслуговуєшся першим".

Якщо мати на увазі конкретні практичні застосування теорії масового обслуговування, то випадки, коли модель являє собою точну копію реального процесу, є скоріше винятком, ніж правилом. Тому математичні моделі варто використовувати головним чином із метою досягнення кращого розуміння особливостей розв'язуваної задачі і заради визначення ступеня чутливості функціональної ефективності системи до варіацій керуючих рішень. Якщо попереднє дослідження (що ґрунтується на застосуванні того або іншого наближеного методу) показує, що негативні економічні наслідки помилкового керуючого рішення виявляються дуже серйозними, то виникає необхідність у проведенні ретельнішого аналізу задачі з застосуванням комп'ютерного імітаційного моделювання досліджуваних процесів.

ПРИКЛАДИ

Приклад 14.1. Процес народження та загибелі.

Функціонування СМО представлене за допомогою діаграми інтенсивностей переходів.



Необхідно знайти граничні вірогідності станів.

Розв'язання.

$p_0+p_1+p_2=1$. Випишемо вирази для p_1 та p_2, використовуючи формулу (14.8), $p_1=\left(\frac{\lambda_0}{\mu_1}\right)p_0$, $p_2=\left(\frac{\lambda_1\lambda_0}{\mu_2\mu_1}\right)p_0$, і отримаємо:

$$p_0+\left(\frac{\lambda_0}{\mu_1}\right)p_0+\left(\frac{\lambda_1\lambda_0}{\mu_2\mu_1}\right)p_0=1,\quad \text{звідки}\quad p_0=\left(1+\left(\frac{\lambda_0}{\mu_1}\right)+\left(\frac{\lambda_1\lambda_0}{\mu_2\mu_1}\right)\right)^{-1}.$$

Підставляючи значення умови, отримаємо: $p_0=\left(1+\frac{1}{4}+\frac{2\times1}{3\times4}\right)^{-1}=0,706$,

$p_1=\frac{1}{4}\times0,706=0,176$, $p_2=\frac{2\times1}{3\times4}\times0,706=0,118$. Знаючи ці вірогідності, надалі за їх допомогою можна визначити деякі операційні характеристики системи.

Приклад 14.2. Процес народження та загибелі.

В крамниці 3 каси, одна з яких працює постійно, а функціонування двох інших залежить від наявності покупців. Якщо число покупців в будь-якій черзі перевищуватиме 3, то у цей момент буде відкрита додаткова каса. Таким чином, якщо число покупців в черзі менше чотирьох, то працювати буде лише одна каса. Якщо число покупців від чотирьох до шести, то працюватиме дві каси. Якщо є більше шести покупців, буде відкрито всі три каси. Покупці підходять до кас, згідно з розподілом Пуассона, з математичним сподіванням 10 осіб на годину. Час розрахунку одного покупця в касі розподілений за експоненційним законом з середнім 12 хвилин.

Необхідно визначити вірогідність знаходження в стаціонарному режимі вірогідність p_n знаходження n покупців біля кас.

Розв'язання.

З умови задачі зрозуміло, що значення параметрів системи становлять $\lambda_n=\lambda=10$ покупців на годину, $n=0,1,\ldots$, $\mu_n=60/12=5$, $n=0,1,2,3$; $\mu_n=2\times5=10$, $n=4,5,6$; $\mu_n=3\times5=15$, $n=7,8,\ldots$.

Отже, $p_1=\left(\frac{10}{5}\right)p_0=2p_0$, $p_2=\left(\frac{10}{5}\right)^2p_0=4p_0$, $p_3=\left(\frac{10}{5}\right)^3p_0=8p_0$

$$p_4=\left(\frac{10}{5}\right)^3\left(\frac{10}{10}\right)p_0=8p_0,\ p_5=\left(\frac{10}{5}\right)^3\left(\frac{10}{10}\right)^2p_0=8p_0,$$

$$p_6=\left(\frac{10}{5}\right)^3\left(\frac{10}{10}\right)^3p_0=8p_0,$$

$$p_n=\left(\frac{10}{5}\right)^3\left(\frac{10}{10}\right)^3\left(\frac{10}{15}\right)^{n-6}\times p_0=8\left(\frac{2}{3}\right)^{n-6}\times p_0,\ n=7,8,\ldots.$$

Значення p_0 визначається із рівняння:

$$p_0+p_0\left[2+4+8+8+8+8+8\left(\frac{2}{3}\right)+8\left(\frac{2}{3}\right)^2+8\left(\frac{2}{3}\right)^3+\ldots\right]=1,$$

або, що рівносильно,

$$p_0\left\{31+8\left[1+\left(\frac{2}{3}\right)+\left(\frac{2}{3}\right)^2+\ldots\right]\right\}=1.$$

Використовуючи формулу суми нескінченної геометричної прогресії

$$p_0\left\{31+8\left[1+\left(\frac{2}{3}\right)+\left(\frac{2}{3}\right)^2+\ldots\right]\right\}=1,$$

отримаємо:

$$p_0\left[31+8\left(\frac{1}{1-\frac{2}{3}}\right)\right]=1.$$

Отже, $p_0=1/55$.

Знаючи p_0, визначимо інші. Наприклад, вірогідність того, що працюватиме лише одна каса, обчислюється як вірогідність знаходження в системі не більше трьох клієнтів, тобто

$$p_0+p_1+p_2+p_3=(1+2+4+8)\times\frac{1}{55}=0,2727.$$

Вірогідності p_n можна використовувати для визначення чисельних значень операційних характеристик системи. Наприклад, середня кількість непрацюючих кас становитиме:

$$3p_0 + 2(p_1 + p_2 + p_3) + 1(p_4 + p_5 + p_6) + 0(p_7 + p_8 + ...) = 1 \text{ каса.}$$

Приклад 14.3. Одноканальна СМО з необмеженою чергою.

На перевалочну базу прибувають вантажівки з товаром з інтенсивністю 0,4 вантажівки на годину. Середній час розвантаження однієї вантажівки становить 2 години, причому поки попередня вантажівка не розвантажена, наступну не розвантажують. На довжину черги обмежень не накладається, тобто прибула вантажівка без розвантаження не може покинути систему. Необхідно визначити операційні характеристики системи та вірогідність того, що розвантаження очікуватимуть не більш ніж 2 вантажівки.

Розв'язання.

Розраховуємо трафік-інтенсивність:

$\rho = \lambda / \mu = \lambda \times M[\text{час обслуговування}] = 0,4 \times 2 = 0,8.$ Оскільки $\rho = 0,8 < 1$, то черга на розвантаження не може зростати безмежно, і існують граничні вірогідності (див. 14.14). Розрахуємо значення граничних вірогідностей. Вірогідність того, що розвантажувальна площа вільна, становить $p_0 = 1 - 0,8 = 0,2$ (14.20). Відповідно вірогідність того, що площа зайнята – 0,8. Значення інших вірогідностей (що в системі знаходяться 1, 2, 3 вантажівки, або, що еквівалентне очікують на розвантаження 0, 1, 2 вантажівки) розраховуємо за формулою (14.24): $p_1 = 0,8(1-0,8) = 0,16$; $p_2 = 0,8^2(1-0,8) = 0,128$; $p_3 = 0,8^3(1-0,8) = 0,1024$.

Вірогідність того, що на розвантаження очікують не більш ніж 2 вантажівки становить:

$$P = p_1 + p_2 + p_3 = 0,16 + 0,128 + 0,1024 = 0,3904.$$

Середню кількість вантажівок в черзі визначаємо за (14.23),

M [Довжина черги]$= \rho^2 / (1-\rho) = 0,8^2 / (1-0,8) = 3,2$, середній час очікування в черзі (14.29),

$M\begin{bmatrix}\text{Час перебу-}\\ \text{вання в черзі}\end{bmatrix} = \dfrac{1}{\mu}\dfrac{\rho}{1-\rho} = 2 \times \dfrac{0,8}{1-0,8} = 8$ год., середня кількість вимог в системі (14.21), $M\begin{bmatrix}\text{Кількість вимог}\\ \text{в системі}\end{bmatrix} = \dfrac{\rho}{1-\rho} = \dfrac{0,8}{1-0,8} = 4$, а середній час перебування вантажівки в системі (14.28)

$M\begin{bmatrix}\text{Час перебування}\\ \text{в системі}\end{bmatrix} = \dfrac{1}{\lambda} \times \dfrac{\rho}{1-\rho} = \dfrac{1}{0,4} \times \dfrac{0,8}{1-0,8} = 10$ год. Аналізуючи отримані дані, можна зробити висновок, що ефективність розвантаження невисока, а тому покращення її можна досягнути двома шляхами: збільшити інтенсивність розвантаження або збільшити кількість майданчиків для розвантаження.

Приклад 14.4. Одноканальна СМО з обмеженою чергою.

Розрахувати середню кількість вантажівок в системі та середню тривалість перебування вантажівки в системі з прикладу 14.3 за умови, що прибула вантажівка покидає систему без розвантаження, якщо в черзі на розвантаження знаходиться більш ніж 3 вантажівки.

Розв'язання.

За умовою кількість вимог в системі M=3+1=4. Вірогідність того, що розвантажувальна площа вільна, становить (14.31):

$$p_0 = \frac{1-\rho}{1-\rho^{M+1}} = \frac{1-0,8}{1-0,8^{4+1}} = \frac{0,2}{1-0,32768} = 0,297.$$

Вірогідність того, що вантажівка покине систему без розвантаження:

$p_{\text{відм.}} = \rho^M p_0 = 0,8^4 \times 0,297 = 0,122.$

Середня кількість вимог в системі (14.32):

$$M\begin{bmatrix}\text{Кількість вимог}\\ \text{в системі обслуговування}\end{bmatrix} = \frac{\rho}{1-\rho} - \frac{(M+1)\rho^{M+1}}{1-\rho^{M+1}} = 1,56,$$

середня тривалість перебування вимоги в системі (14.34)

$$M\begin{bmatrix}\text{Тривалість перебування вимоги}\\ \text{в системі обслуговування}\end{bmatrix} = \frac{1}{\mu(1-\rho)} - \frac{M\rho^M}{\mu(1-\rho^M)} = 4,44 \text{ год.}$$

РЕЗЮМЕ

14.1 У випадку розв'язання задачі масового обслуговування апроксимації є неминучими незалежно від того, якого типу модель при цьому використовується – математична, імітаційна або комбінована. Часто вдається одержати приблизне уявлення про операційні характеристики складної системи шляхом аналізу деяких „екстремальних" або граничних випадків. Один із таких наближених методів полягає в наступному: СМО, що нараховує n обслуговуючих приладів, розглядається як „механічне" об'єднання n одноканальних систем, що функціонують незалежно одна від іншої. Системи з одним обслуговуючим приладом, як і множина багатоканальних систем із єдиною чергою, що характеризується дисципліною „першим прийшов - першим обслуговуєшся" зазвичай піддаються математичному описанню і кількісному аналізу. При розгляді загальних систем масового обслуговування передбачається, що система функціонує протягом достатньо великого інтервалу часу, і після закінчення перехідного періоду переходить в стаціонарний режим.

14.2. Найпростішою одноканальною моделлю з вірогідністним вхідним потоком і процедурою обслуговування є модель, що характеризується показниковим розподілом як тривалостей інтервалів між надходженнями вимог, так і тривалостей обслуговування Система лінійних диференційних рівнянь, що описує найпростішу модель СМО, має єдиний розв'язок, якщо задані відповідні початкові умови. Цей розв'язок є несталим, оскільки він безпосередньо залежить від часу. Спрямувавши час до безмежності, отримуємо стаціонарні характеристики системи, причому граничні стаціонарні вірогідності завжди існують, коли виконується умова стаціонарності $\rho < 1$. Операційні характеристики найпростішої СМО розраховуються аналітично.

14.3. Розширеннями найпростішої СМО є СМО з обмеженням на довжину черги, введення пріоритетів вимог. Ці узагальнення дозволяють з певним наближенням описувати практично важливі класи систем, і водночас для таких моделей в більшості випадків існують аналітичні розв'язки, що уможливлює проведення аналізу систем за умови зміни значень їх параметрів. Це, своєю чергою, дозволяє порівняти варіанти систем за їх ефективністю, не залучаючи додаткових засобів. Нескладним виявляється також обчислення очікуваної кількості вимог у СМО і середньої тривалості їх перебування в ній у тому випадку, коли вхідної потік вимог характеризується експоненційним розпо-ділом, а щодо вигляду

розподілу тривалостей обслуговування не робиться ніяких спеціальних припущень.

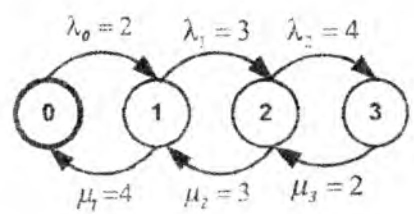
14.4. В переважній більшості випадків СМО є багатоканальними, і, отже, моделі з декількома паралельними обслуговуючими приладами є важливими. Постульована при цьому дисципліна черги виглядає дещо спрощено для більшості ситуацій, із якими доводиться зустрічатися в дійсності. Проте отримані результати можна розглядати як дуже корисні, оскільки вони принаймні дозволяють у найпершому наближенні оцінити функціональні характеристики складніших СМО.

14.5. Майже всі отримані результати стосуються моделей масового обслуговування, у яких процес надходження вимог є пуасонівським, а тривалості інтервалів обслуговування однієї вимоги мають експоненційний розподіл. Ці результати не так важко узагальнити на випадки, коли умови задачі дещо видозмінені. Якщо мати на увазі конкретні практичні застосування теорії масового обслуговування, то випадки, коли модель є точною копією реального процесу, є скоріше винятком, ніж правилом. Тому математичні моделі варто використовувати головним чином із метою досягнення кращого розуміння особливостей розв'язуваної задачі і заради визначення ступеня чутливості функціональної ефективності системи до варіацій керуючих рішень. Для детальнішого аналізу складніших СМО застосовуються методи імітаційного моделювання, а побудова моделі і реалізація експериментів на ній значно полегшується за рахунок застосування спеціалізованих мов моделювання, як, наприклад, GPSS.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 14.1. Процес народження та загибелі.

Знайти граничні вірогідності станів для системи, функціонування якої представлене за допомогою наступної діаграми інтенсивностей переходів.



Завдання 14.2. Процес народження та загибелі.

В бюро довідкового сервісного обслуговування є 4 працівники, один з яких працює постійно, а функціонування трьох інших залежить від наявності клієнтів. Якщо число клієнтів в будь-якій черзі перевищуватиме 2, то у цей момент підключається додатковий працівник. Таким чином, якщо число клієнтів в черзі менше трьох, то працювати буде лише один працівник. Якщо число клієнтів від трьох до чотирьох, то працюватиме два працівники. Якщо ж число клієнтів від п'яти до шести, обслуговувати будуть три працівники. Якщо ж число клієнтів більше шести, то всі працівники будуть обслуговувати клієнтів. Запити клієнтів надходять, згідно з розподілом Пуассона, з математичним сподіванням 20 запитів на годину. Час надання однієї довідки розподілений за експоненційним законом з середнім 6 хвилин.

Необхідно визначити вірогідність знаходження в стаціонарному режимі p_n знаходження n запитів в черзі.

Завдання 14.3. Діаграма інтенсивностей переходів.

Побудуйте діаграму інтенсивностей переходів для прикладу 14.2.

Завдання 14.4. Одноканальна СМО з необмеженою чергою.

Розрахуйте операційні характеристики системи з прикладу 14.3 за умови, що: а) інтенсивність розвантаження зростає вдвічі; б) кількість місць для розвантаження становитиме 2. Порівняйте отримані результати між собою та з початковим варіантом системи.

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Розкрийте проблеми апроксимації реальних систем за допомогою СМО.
2. Від чого залежать основні параметри СМО в загальній моделі?
3. Виведіть формули граничних вірогідностей для процесу народження та загибелі.
4. Які особливості має найпростіша модель СМО?
5. Виведіть основні співвідношення, що описують функціонування найпростішої СМО.
6. Перерахуйте операційні характеристики найпростішої СМО та поясніть, яким чином вони обчислюються.
7. В чому суть аналізу СМО на чутливість?

8. Яким чином виявляється обмеження на довжину черги в найпростішій СМО?
9. У чому полягають особливості одноканальних моделей СМО з довільним законом розподілом тривалостей обслуговування?
10. Які співвідношення описуються формулами Полачека – Хінчіна?
11. Яким чином відображається на функціонуванні СМО наявність пріоритетів вимог?
12. Яким чином функціонує багатоканальна СМО?
13. Намалюйте діаграму інтенсивностей переходів для багатоканальної системи з необмеженою чергою та поясніть її.
14. Яким чином впливає наявність пріоритетів вимог на характеристики багатоканальної СМО?
15. З якою метою використовуються математичні моделі СМО?

РОЗДІЛ 7

МОДЕЛІ УПРАВЛІННЯ ЗАПАСАМИ

ТЕМА 15. ОСНОВНІ ПОНЯТТЯ ТА ДЕТЕРМІНОВАНІ МОДЕЛІ УПРАВЛІННЯ ЗАПАСАМИ

Основна проблема, що виникає при розв'язанні задачі управління запасами, полягає в створенні ефективної і надійної системи управління рівнем наявних запасів. Задача управління запасами виникає, коли необхідно створити запас матеріальних ресурсів або предметів споживання з метою задоволення попиту на певному заданому інтервалі часу (скінченому або нескінченому). Для забезпечення неперервного і ефективного функціонування практично будь-якої організації необхідне створення запасів. В будь-якій задачі управління запасами потрібно визначити кількість замовленої продукції і строки розміщення замовлень. Попит можна задовольнити шляхом одноразового створення запасу на весь потрібний період часу або шляхом створення запасу для кожної одиниці часу цього періоду. Ці два випадки відповідають залишковому запасу (стосовно одиниці часу) і недостатньому запасу (стосовно повного періоду часу). При залишковому запасі потрібні вищі виділені (стосовно одиниці часу) капітальні вкладення, але дефіцит виникає рідше і частота розміщення замовлень менша. З іншого боку, при недостатньому запасі виділені капітальні вкладення зменшуються, але частота розміщення замовлень і загроза дефіциту зростає. Для будь-якого з вказаних граничних випадків характерні значні економічні втрати. Таким чином, рішення щодо розміру замовлення і моменту його розміщення можуть базуватися на мінімізації відповідної функції загальних затрат, що обумовлені втратами від залишкового запасу і дефіциту.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати:⇨основні функції системи управління запасами та їх розподіл за ієрархічними рівнями;⇨структуру узагальненої моделі управління

запасами та основні типи моделей управління запасами;⇨особливості та види детермінованих моделей управління запасами;

вміти:⇨визначати тип моделі управління запасами, що відповідає реальній практичній ситуації;⇨ визначати оптимальний рівень запасів для детермінованих моделей управління запасами, а саме: однопродуктної статичної моделі без та з врахуванням запізнення, з „розривами" цін, багатопродуктної статичної моделі з обмеженнями на ємність складських приміщень

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ витрати на зберігання замовлення
- ☑ витрати на оформлення замовлення
- ☑ пункт накопичення запасу
- ☑ розмір замовлення
- ☑ рівень (об'єм) запасів
- ☑ залишковий запас
- ☑ втрати від дефіциту
- ☑ детермінований попит
- ☑ стохастичний попит
- ☑ строки розміщення замовлень
- ☑ періодичний контроль стану запасу
- ☑ багатопродуктна статична модель
- ☑ точка відновлення замовлення
- ☑ неперервний контроль стану запасу
- ☑ однопродуктна статична модель
- ☑ функція загальних затрат
- ☑ витрати на придбання
- ☑ модель з розривами цін

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

15.1. Поняття та проблематика управління запасами.
15.2. Узагальнена модель управління запасами.
15.3. Типи моделей управління запасами.
15.4. Детерміновані моделі управління запасами.

15.1. ПОНЯТТЯ ТА ПРОБЛЕМАТИКА УПРАВЛІННЯ ЗАПАСАМИ

Основна проблема, що виникає при розв'язанні задачі управління запасами, полягає в створенні ефективної і надійної системи управління рівнем наявних запасів. Відповідно до функцій, що реалізуються в системі управління запасами, її структуру зручно розглядати як трирівневу ієрархічну систему, кожен з рівнів якої розв'язує свої завдання.

На третьому (нижньому) рівні здійснюється опрацювання, ведення обліку і збереження інформації про запаси. Основною інформацією, необхідною для успішного функціонування системи управління запасами, є рівень наявних запасів і запасів, що будуть створені за рахунок розміщення замовлень, а також накопичення за замовленнями споживачів (якщо є попередні замовлення). Крім того, облік інформації щодо будь-якої наявної в запасах продукції включає такі показники, як її вартість, час випередження, одиниця виміру, джерело одержання й ін. Таким чином, обсяг оперативної облікової інформації є істотним, і значне підвищення ефективності опрацювання інформації досягається шляхом побудови та використання відповідних комп'ютерних інформаційних систем. Оперативний цикл третього рівня системи зазвичай становить день.

Завданням другого (середнього) рівня є розроблення та застосування правил ухвалення рішень, на ґрунті яких установлюються терміни і розміри замовлень, необхідних для поповнення запасів. Кожне правило прийняття рішень є результатом вирішення деякої часткової задачі оптимізації управління запасами. Задачі, що доводиться вирішувати користувачу в практичних умовах, є різними, тому що конкретні системи мають різні параметри. Тому досліднику систем необхідно знати умови, за яких одне правило прийняття рішень виявляється кращим за інше. Формально ці рішення є оптимальними, проте для успішного застосування необхідно враховувати реальні умови та досвід персоналу, оскільки не завжди виправданим є уявлення про "єдино правильне" рішення. Оперативний цикл середнього рівня системи становить місяць. В цьому випадку слід особливу увагу звернути на достовірність облікових даних, тому що навіть найкраща модель не може генерувати оптимальні чи навіть прийнятні рішення за умови відсутності достовірної первинної інформації.

Перший (вищий) рівень дозволяє на основі розроблених правил прийняття рішень побудувати модель системи управління і відповідно до цієї моделі визначити стратегію функціонування (зазвичай горизонт планування в цьому випадку є квартал або рік). Поточний контроль за виконанням плану дає можливість органам управління вирішувати питання про доцільність втручання у функціонування системи, і якщо так, то яким чином і коли необхідно здійснювати управління системою загалом, а не запасами конкретної продукції.

Існують дві точки зору щодо необхідності створення запасів. Згідно з однією з них, наявність запасів дозволяє швидко задовольняти запити споживачів; згідно з іншою, наявність запасів дозволяє постачальнику нейтралізувати коливання попиту в умовах рівномірного виробництва продукції.

У системі управління запасами, що орієнтована на споживача, рішення про збільшення розміру запасу, звичайно, приймається з урахуванням кількості замовлень на ту чи іншу продукцію. Якщо протягом часу T на деяку продукцію надійшло принаймні N заявок, то створюється запас цієї продукції. При такій стратегії органи управління, відповідальні за сферу збуту, можуть скласти таблицю рішень, що передбачають витрати при заданих значеннях (N, T) шляхом класифікації запасів. Існує продукція визначеного типу (запасні частини для нового обладнання й основні запаси для виробництва нової продукції), для яких доцільно мати запас до надходження перших замовлень.

Оскільки дохід від реалізації наявної в запасі продукції протягом тривалого часу може бути лише частково оцінений у грошовому вираженні (і подібна оцінка, як правило, є дуже суб'єктивною), то, очевидно, не має сенсу розробляти складні моделі системи управління запасами з урахуванням витрат на збереження додаткового обсягу продукції. Органи управління, що приймають рішення, повинні мати уявлення про економічний ефект прийнятого рішення, однак модель, що використовується для такого прогнозу, може бути гранично простою і враховувати плановий обсяг інвестицій у товарно-матеріальні запаси, розміри складських приміщень, а також витрати на функціонування першого рівня системи управління (системи обліку).

У системі управління запасами, що орієнтована на постачальника, існує реальна економічна основа для ухвалення рішення щодо того, організовувати власний випуск певної продукції чи закуповувати її на стороні, якщо вона користується попитом.

Моделювання систем управління запасами приводить до необхідності розроблення підходів до розв'язання задачі економічно доцільного розміщення продукції на складах розподільної мережі.

Для галузі промисловості, що випускає товари широкого вжитку, характерна ситуація, коли на складі є багато подібних типів продукції, кожен з яких міг би задовольнити запити споживачів. У цьому випадку в більшості підходів (пов'язаних з визначенням величини партії продукції, необхідної для поповнення запасів і збереження рівня резервних запасів) теорія управління запасами розглядає всю агреговану групу взаємозамінної продукції як елемент планування. Таким чином, поки існує загальний запас придатної продукції, споживач буде обирати і купувати що-небудь з того, що є на складі.

15.2. УЗАГАЛЬНЕНА МОДЕЛЬ УПРАВЛІННЯ ЗАПАСАМИ

Таким чином, задача управління запасами виникає, коли необхідно створити запас матеріальних ресурсів або предметів споживання з метою задоволення попиту на певному заданому інтервалі часу (скінченому або нескінченому). Для забезпечення неперервного і ефективного функціонування практично будь-якої організації необхідне створення запасів. В будь-якій задачі управління запасами потрібно визначати кількість замовленої продукції і строки розміщення замовлень. Попит можна задовольнити шляхом одноразового створення запасу на весь потрібний період часу або шляхом створення запасу для кожної одиниці часу цього періоду. Ці два випадки відповідають залишковому запасу (стосовно одиниці часу) і недостатньому запасу (стосовно повного періоду часу).

При залишковому запасі потрібні вищі виділені (стосовно одиниці часу) капітальні вкладення, але дефіцит виникає рідше і частота розміщення замовлень менша. З іншого боку, при недостатньому запасі виділені капітальні вкладення зменшуються, але частота розміщення замовлень і загроза дефіциту зростає. Для будь-якого з вказаних крайніх випадків характерні значні економічні втрати. Таким чином, рішення щодо розміру замовлення і моменту його розміщення можуть базуватися на мінімізації відповідної функції загальних затрат, що обумовлені втратами від залишкового запасу і дефіциту.

Довільна модель управління запасами остаточно повинна дати відповідь на два запитання: Яку кількість продукції замовляти? Коли замовляти?

Відповідь на перше питання відображається через ***розмір замовлення***, що визначає оптимальну кількість ресурсів, яку необхідно постачати кожен раз, коли відбувається розміщення замовлення. В залежності від ситуації розмір замовлення може змінюватися з часом. Відповідь на друге питання залежить від типу системи управління запасами. Якщо система передбачає ***періодичний контроль*** стану запасу через рівні проміжки часу (наприклад, кожної неділі або кожного місяця), момент надходження нового замовлення, зазвичай, співпадає з початком кожного інтервалу часу. Якщо ж у системі

передбаченні ***неперервний контроль*** стану запасу, ***точка замовлення*** зазвичай визначається ***рівнем запасу***, при якому необхідно розміщувати нове замовлення.

Таким чином, розв'язання узагальненої задачі управління запасами визначається наступним:

У випадку ***періодичного контролю стану запасу*** потрібно забезпечувати надходження нової кількості ресурсів об'ємом в ***розмір замовлення*** через рівні проміжки часу.

У випадку ***неперервного контролю стану запасу*** необхідно розміщувати нове замовлення розміром в ***об'єм запасу***, коли його рівень досягає ***точки замовлення***.

Розмір і точка замовлення зазвичай визначаються з умов мінімізації сумарних витрат системи управління запасами, які можна представити у вигляді функції цих двох змінних. Сумарні витрати системи управління запасами відображаються у вигляді функції їх основних компонент наступним чином :

$$\begin{pmatrix}\text{Сумарні}\\ \text{витрати}\\ \text{системи}\\ \text{управління}\\ \text{запасами}\end{pmatrix}=\begin{pmatrix}\text{Витрати}\\ \text{на прид-}\\ \text{бання}\end{pmatrix}+\begin{pmatrix}\text{Витрати на}\\ \text{оформлення}\\ \text{замовлення}\end{pmatrix}+\begin{pmatrix}\text{Витрати на}\\ \text{зберігання}\\ \text{замовлення}\end{pmatrix}+\begin{pmatrix}\text{Втрати}\\ \text{від}\\ \text{дефіциту}\end{pmatrix}$$

Витрати на придбання стають важливим фактором, коли ціна одиниці продукції залежить від розміру замовлення, що, зазвичай, відображається в виді ***гуртових знижок*** у тих випадках, коли ціна одиниці продукції зменшується зі збільшенням розміру замовлення.

Витрати на оформлення замовлення – це постійні витрати, що пов'язані з його розміщенням. Таким чином, при задоволенні попиту протягом заданого періоду часу шляхом розміщення менших замовлень (частіше) витрати збільшуються порівняно з випадком, коли попит задовольняється розміщенням більших замовлень (і, відповідно, рідше).

Витрати на зберігання запасу – це витрати на зберігання запасу на складі (наприклад, процент на інвестований капітал, витрати на переробку, амортизаційні витрати і експлуатаційні витрати), зазвичай збільшуються з зростанням рівня запасу.

Втрати від дефіциту є витрати, що обумовлені відсутністю запасу необхідної продукції. Зазвичай вони пов'язані з погіршенням репутації постачальника у споживача і з потенційними втратами.

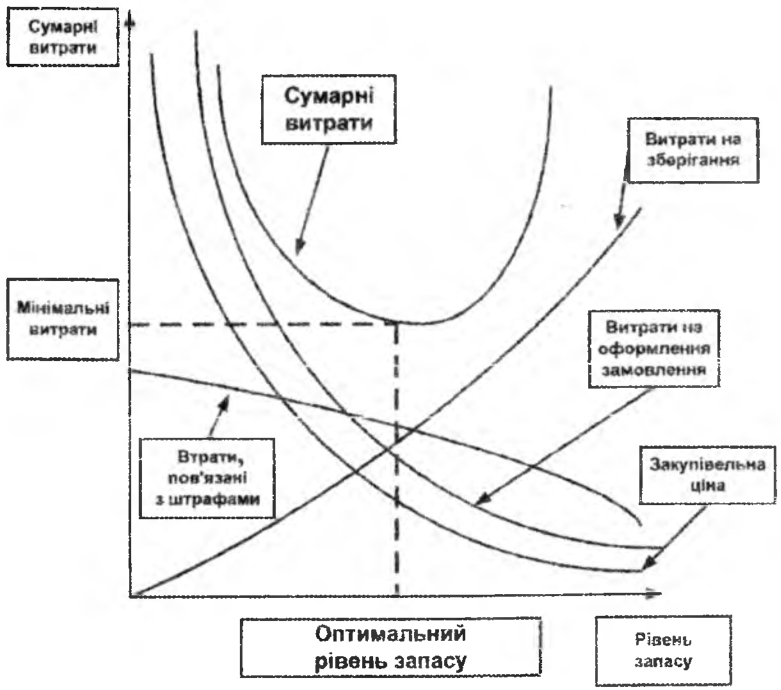


Рис. 15.1. Залежність сумарних витрат та їх складових від рівня запасу

На малюнку відображені залежності чотирьох компонент витрат узагальненої моделі управління запасами від рівня запасу. Оптимальний рівень запасу відповідає мінімуму сумарних витрат. Зазначимо, що модель управління запасами не обов'язково повинна включати всі чотири види витрат, так як деякі з них можуть бути незначними, а деколи врахування всіх видів витрат дуже ускладнює функцію сумарних витрат. На практиці будь-яку компоненту витрат можна не враховувати за умови, що вона не містить важливу частину загальних витрат. Цей фактор необхідно мати на увазі при вивченні різноманітних моделей.

15.3. ТИПИ МОДЕЛЕЙ УПРАВЛІННЯ ЗАПАСАМИ

Узагальнена модель управління запасами, описана вище, є досить простою. Але різноманітність моделей цього класу й методів розв'язування відповідних задач, які базуються на різному математичному апараті – від простих схем диференціального і інтегрального числення до складних алгоритмів динамічного і інших видів математичного програмування, – визначається характером попиту, який може бути детермінованим або стохастичним. На рис. 15.2 наведена схема класифікації попиту, який, зазвичай, використовується в моделях управління запасами.

Детермінований попит може бути ***статичним***, в тому сенсі, що інтенсивність споживання залишається незмінною з часом, або ***динамічним***, коли попит відомий достовірно, але змінюється в залежності від часу.

Стохастичний попит є ***стаціонарним***, якщо функція щільності імовірності попиту незмінна в часі, і ***нестаціонарним***, коли функція щільності попиту змінюється в часі.

Рис. 15.2. Класифікація задач управління запасами за видом попиту

В реальних умовах випадок детермінованого статичного попиту зустрічається доволі рідко. Такий випадок є найпростішим. Так, наприклад, хоча попит на деякі продукти масового споживання, як хліб, може змінюватися від одного дня до іншого, ці зміни можуть бути настільки незначними, що припущення про статичність попиту несуттєво спотворює дійсність.

Найточніше характер попиту може бути описаний за допомогою нестаціонарних розподілів вірогідностей. Однак із математичної точки зору модель значно ускладнюється, особливо при збільшенні періоду часу, що розглядається. На рис. 15.2 ілюструється зростання математичної складності моделі управління запасами при переході від детермінованого статичного попиту до вірогідніского нестаціонарного попиту. По суті, цю класифікацію можна вважати представленням різних ***рівнів абстрагування*** при описанні попиту.

На першому рівні абстрагування припускається, що розподіл вірогідності попиту стаціонарний у часі. Це означає, що для описання попиту протягом усіх періодів часу, що досліджуються, використовується одна і та ж функція розподілу вірогідностей. При такому припущенні вплив квартальних коливань попиту в моделі не враховується.

На другому рівні абстракції враховуються зміни попиту від одного періоду до іншого. Однак при цьому функції розподілу не застосовуються, а потреби в кожному періоді описуються середньою величиною попиту. Це спрощення означає, що елемент ризику в управлінні запасами не враховується. Однак це дозволяє досліджувати квартальні коливання попиту, які внаслідок аналітичних і обчислювальних труднощів не можна врахувати у вірогідніснй моделі. Іншими словами, тут виникає певний компроміс: можна використовувати, з одного боку, стаціонарні розподіли ймовірностей, а з іншого – змінну, але відому функцію попиту при припущенні "визначеності".

На третьому рівні спрощення виключаються як елементи ризику, так і зміни попиту. Тим самим попит протягом будь-якого періоду вважається рівним середньому значенню відомого (за припущенням) попиту для всіх періодів, що розглядаються. В результаті цього спрощення попит можна оцінити його ***сталою*** інтенсивністю.

Хоча характер попиту є одним з основних факторів при побудові моделі управління запасами, є й інші фактори, що впливають на вибір типу моделі, а саме:

Запізнення надходжень виконання замовлень. Після розміщення замовлення воно може бути поставлене відразу ж або буде потрібний деякий час на його виконання. Інтервал часу між моментом розміщення замовлення і його надходженням називається запізненням замовлення або терміном виконання замовлень. Ця величина може бути детермінованою або стохастичною.

Поповнення замовлення. Хоча система управління запасами може функціонувати при запізненні надходжень, процес збільшення запасу може здійснюватися миттєво або рівномірно в часі. Миттєве збільшення запасу може бути реалізоване за умови, коли замовлення надходять від зовнішнього джерела. Рівномірне збільшення може бути тоді, коли

продукція, що запасається, виробляється самою організацією. В загальному випадку система може функціонувати при позитивному запізненні надходження і рівномірному збільшенні запасу.

Період часу визначає інтервал, протягом якого здійснюється зміна рівня запасу. В залежності від проміжку часу, на якому можна надійно прогнозувати період, що розглядається, вважається скінченим або нескінченим.

Кількість пунктів накопичення запасу. До складу системи управління запасами може входити декілька пунктів зберігання запасу. В деяких випадках ці пункти організовані таким чином, що один є постачальником для іншого. Ця схема деколи реалізується на різноманітних рівнях, так що пункт-споживач одного рівня може стати пунктом-постачальником на іншому. В такому випадку це є система управління запасами з розгалуженою структурою.

Кількість видів продукції. В системі управління запасами може фігурувати більше ніж один вид продукції. Цей фактор враховується за умови наявності деякої залежності між різними видами продукції. Так, для різних виробів може використовуватися одне і те ж складське приміщення або ж виробництво може здійснюватися при обмеженнях на загальні виробничі фонди.

15.4. ДЕТЕРМІНОВАНІ МОДЕЛІ УПРАВЛІННЯ ЗАПАСАМИ

Узагальнену модель управління запасами, яка враховувала б усі різновиди умов, що спостерігаються в реальних системах, побудувати важко. Але якщо вдалося побудувати достатньо універсальну модель, на ній неможливо отримати аналітичні розв'язки. Наведені нижче моделі відповідають деяким спрощеним системам управління запасами. Малоймовірно, що ці моделі зможуть точно відповідати реальним умовам, однак вони наведені з метою пояснення різних підходів до розв'язання деяких конкретних задач управління запасами.

Більшість із моделей однопродуктні, і тільки в одній з них враховується вплив деяких "конкуруючих" видів продукції. Основна відмінність між моделями визначається припущенням про характер попиту (статичний або динамічний). Важливим фактором із точки зору формулювання й розв'язання задачі є також вид функції витрат. Для розв'язування можуть використовуватися різні методи, які включають класичну схему оптимізації, лінійне й динамічне програмування.

Однопродуктна статична модель.

Модель управління запасами простого типу характеризується сталим у часі попитом, миттєвим збільшенням запасу і відсутністю дефіциту.

На рис. 15.3 показано зміну рівня запасу з часом. Припускається, що інтенсивність попиту (в одиницю часу) дорівнює β.

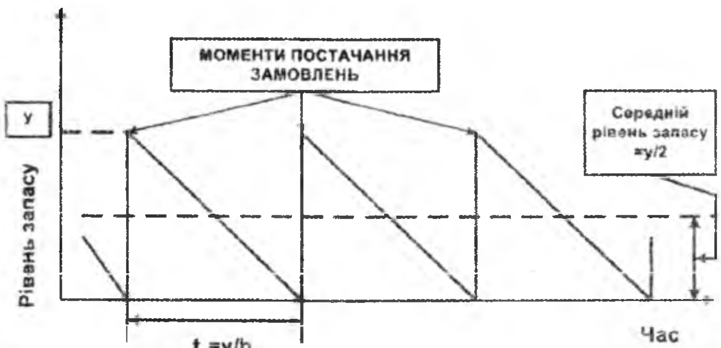


Рис. 15.3 Зміна рівня запасу з часом в однопродуктовій статичній моделі

Найвищого рівня запас досягає в момент постачання замовлення розміром y.

(Припускається, що запізнення постачання є заданою константою). Рівень запасу досягає нуля протягом y/β одиниць часу після отримання замовлення розміром y.

Чим менший розмір замовлення y, тим частіше потрібно розміщувати нові замовлення. Однак при цьому середній рівень запасу буде зменшуватися. З іншого боку, зі збільшенням розміру замовлень рівень запасу збільшується, але замовлення розміщуються рідше (рис. 15.4). Так як витрати залежать від частоти розміщення замовлення й об'єму запасу, що зберігається, то величина y обирається згідно з умовою забезпечення оптимального балансу між двома видами витрат. Це лежить в основі побудови відповідної моделі управління запасами.

Нехай K – витрати на оформлення замовлення, що мають місце щоразу при його розміщенні в припущенні, що витрати на зберігання одиниці замовлення *в одиницю часу* рівні h.

Рис. 15.4. Зміна рівня запасу з часом в залежності від частоти розміщення замовлень

Отже, сумарні витрати ***в одиницю часу*** $C(y)$ як функцію від y можна представити у вигляді:

сумарні витрати ***в одиницю часу*** $C(y)$ = *Витрати на оформлення замовлення за одиницю часу + Витрати на зберігання запасів за одиницю часу.*

Таким чином, $$C(y)=\frac{K\beta}{y}+h\times\frac{y}{2}. \qquad (15.1)$$

Тривалість циклу руху замовлення складає $t_0=y/\beta$ і середній рівень запасу становить $y/2$.

Оптимальне значення y отримується в результаті мінімізації $C(y)$ по y. Таким чином, у припущенні, що y неперервна змінна, одержуємо:

$$\frac{dC(y)}{dy}=-\frac{K\beta}{y^2}+\frac{h}{2}=0,$$

звідки оптимальне значення розміру замовлення визначається виразом

$$y^*=\sqrt{\frac{2K\beta}{h}}. \qquad (15.2)$$

Оскільки друга похідна в точці y^* строго додатна, досягається мінімум. Отриманий вираз для розміру замовлення, зазвичай, називають **формулою економічного розміру замовлення Уілсона**.

Оптимальна стратегія моделі передбачає замовлення y^* через кожні

$t_0^* = y^* / \beta$ одиниць часу. Оптимальні витрати $C(y^*)$, отримані шляхом безпосередньої підстановки, складають $\sqrt{2K\beta h}$.

Рис. 15. 5. Функціонування системи з запізненням

Для більшості реальних ситуацій існує (позитивний) **термін виконання замовлення** (тимчасове запізнення) L від моменту розміщення замовлення до його дійсної поставки. Стратегія розміщення замовлень у наведеній моделі повинна визначати **точку відновлення замовлення**. Рис. 15.5 ілюструє випадок, коли точка відновлення замовлення повинна випереджати на L одиниць часу очікуване надходження. У практичних цілях цю інформацію можна просто перетворювати, визначивши ***точку відновлення замовлення*** через ***рівень запасу***, який відповідає моменту відновлення замовлення. На практиці це реалізується шляхом неперервного контролю рівня запасу до моменту досягнення чергової точки відновлення замовлення. Можливо, з цієї причини модель економічного розміру замовлення деколи називають **моделлю неперервного контролю стану замовлення**. Потрібно відзначити, що з точки зору аналізу в умовах стабілізації системи термін виконання замовлення L можна завжди прийняти меншим за тривалість циклу t_0^*.

Прийняті в розглянутій вище моделі припущення можуть не відповідати деяким реальним умовам унаслідок вірогіднісного характеру попиту. На практиці отримав розповсюдження наближений метод, який зберігає простоту моделі економічного розміру замовлення і в той же час у якійсь мірі враховує вірогіднісний характер попиту. Ідея методу надзвичайно проста. Вона передбачає створення деякого (постійного) буферного запасу на всьому горизонті планування. Розмір резерву визначається таким чином,

щоб вірогідність зменшення запасу протягом ***періоду виконання замовлення*** L не перевищувала наперед заданої величини. Припустимо, що $f(r)$ – щільність розподілу вірогідності попиту протягом ***цього терміну***. Надалі припустимо, що вірогідність зменшення запасу протягом періоду L не повинна перевищувати a. Тоді розмір резервного запасу B визначиться з умови $P(r \geq B + L\beta) \leq a$, де $L\beta$ є споживання протягом часу L. Зміна запасу при наявності резерву показана на рис. 15.6.

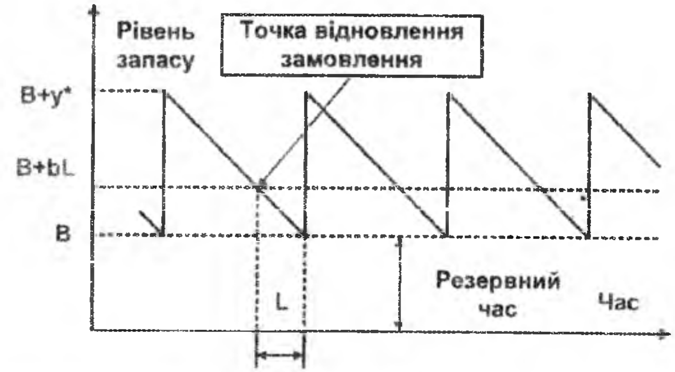


Рис. 15.6. Зміна запасу при наявності резерву

Немає причин припускати, що загальний результат використання процедур визначення B й економічного розміру замовлення обов'язково оптимальний або близький до оптимального. Відхилення від оптимуму пояснюється тим, що на початку деяка суттєва інформація не враховується, а потім використовується зовсім неявно на останньому етапі обчислень. По суті, витрати на зберігання резерву B можна розглядати просто як деяку "ціну" за те, що вся наявна інформація у процесі аналізу одночасно не використовується.

Різновиди моделей економічного розміру замовлення (партії) припускають можливість дефіциту і рівномірного (а не миттєвого) збільшення запасу. Останній випадок є типовим для виробничих систем, в яких інтенсивність збільшення є функцією інтенсивності виробництва. В цих ситуаціях у моделях управління запасами, як і раніше, співставляються витрати на зберігання запасів і оформлення замовлень. В функцію сумарних витрат включаються також втрати від дефіциту, якщо він має місце. В загальному випадку втрати від дефіциту припускаються пропорційними середній величині дефіциту.

Однопродуктова статична модель з "розривами" цін.

В попередніх моделях не враховуються окремі витрати на придбання товарів, так як вони сталі і не впливають на рівень запасу. Однак в багатьох випадках ціна одиниці продукції залежить від розмірів закупленої партії. У таких випадках ціни змінюються стрибкоподібно або надаються гуртові знижки. При цьому в моделі управління запасами необхідно враховувати витрати на придбання.

Розглянемо модель управління запасами з миттєвим збільшенням запасу за відсутності дефіциту. Припустимо, що ціна одиниці продукції дорівнює c_1 при $y < q$ і рівна c_2 при $y \geq q$, де $c_1 > c_2$ і q –розмір замовлення, при перевищенні якого надається знижка. Тоді сумарні витрати за цикл, незважаючи на затримки в оформленні замовлення і зберігання запасу, повинні включати затримки придбання.

Сумарні витрати в ***одиницю часу*** при $y < q$ становлять:

$C_1(y) = \beta c_1 + \dfrac{K\beta}{y} + \dfrac{h}{2}y$, при $y \geq q$ ці витрати становлять: (15.3)

$$C_2(y) = \beta c_2 + \frac{K\beta}{y} + \frac{h}{2}y.$$

Графіки цих двох функцій наведені на рис. 15.7 нижче. Нехтуючи впливом зниження цін, позначимо через y_m розмір замовлення, при якому досягається мінімум величин C_1 і C_2. Тоді $y_m = \sqrt{2K\beta / h}$.

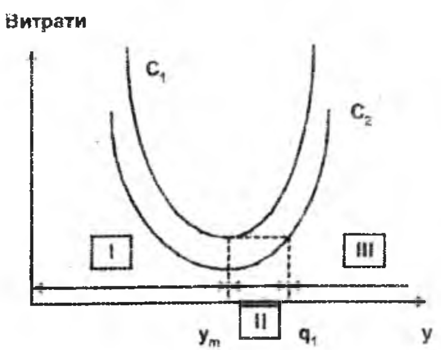


Рис. 15.7. Графіки сумарних витрат в одиницю часу

З вигляду функцій витрат C_1 і C_2 робимо висновок, що оптимальний розмір замовлення y^* залежить від того, де саме стосовно трьох показаних на рисунку зон I, II і III знаходиться точка розриву ціни q. Ці зони знаходяться наступним чином: Зона I: $0 \leq q < y_m$; Зона II: $y_m \leq q < q_1$; Зона III: $q \geq q_1$.

На рис. 15.8 наведене графічне розв'язання рівняння для розглянутого випадку, яке залежить від того, де знаходиться q відносно зон I, II і III. В результаті оптимальний розмір замовлення y^* визначається наступним чином:

$$y^* = \begin{cases} y_m, \text{ якщо } 0 \leq q < y_m \text{ (зона I)}, \\ q, \text{ якщо } y_m \leq q < q_1 \text{ (зона II)}, \\ y_m, \text{ якщо } q \geq q_1 \text{ (зона III)}, \end{cases} \tag{15.4}$$

Алгоритм визначення y^* має наступні основні кроки:

Крок 1. Визначити $y_m = \sqrt{2K\beta / h}$. Якщо $q < y_m$ (зона I), то $y^* = y_m$ і **стоп**. В іншому випадку перейти до кроку 2.

Крок 2. Визначити q_1 з рівняння $C_1(y_m) = C(q_1)$ і встановити, де саме відносно зон II і III знаходиться значення q.

а. Якщо $y_m \leq q \leq q_1$ (зона II), то $y^* = q$.

б. Якщо $q \geq q_1$ (зона III), то $y^* = y_m$.

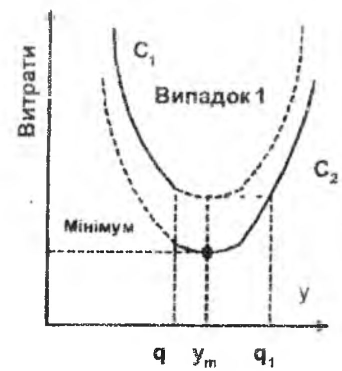


Рис. 15.8. Модель з розривами цін. Випадок 1

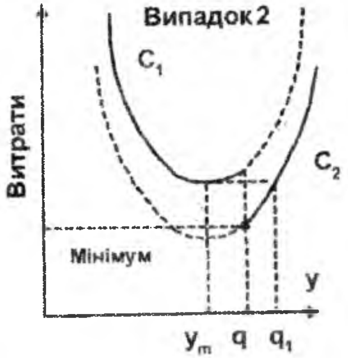


Рис. 15.9. Модель з розривами цін. Випадок 2

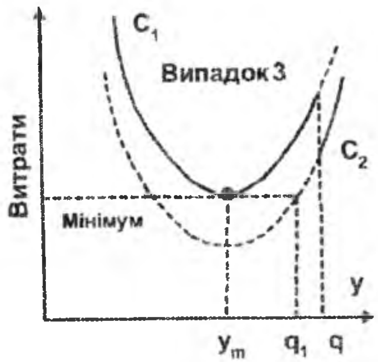


Рис. 15.10. Модель з розривами цін. Випадок 3

Випадок 1: q потрапляє в зону I, $y^* = y_m$. Випадок 2: q потрапляє в зону II, $y^* = q$. Випадок 3: q потрапляє в зону III, $y^* = y_m$.

Багатопродуктна статична модель з обмеженнями на ємність складських приміщень

Ця модель призначена для системи управління запасами, яка містить $n > 1$ видів продукції, що зберігається на одному складі з обмеженою площею. Дана умова визначає взаємозв'язок між різними видами продукції і може бути включена в модель як обмеження.

Нехай A – максимальна припустима площа приміщення для складу для n видів продукції; припустимо, що площа, необхідна для зберігання одиниці продукції i – го виду, становить a_i. Якщо y_i – розмір замовлення на продукцію i – го виду, то обмеження на споживання в складі мають вигляд $\sum_{i=1}^{n} a_i y_i \le A$.

Припустимо, що запас продукції кожного виду поповнюється миттєво і знижки на ціни відсутні. Припустимо далі, що дефіцит не припускається. Нехай β_i, K_i і h_i – інтенсивність попиту, витрати на оформлення замовлення і витрати на зберігання одиниці продукції за одиницю часу для i – го виду продукції відповідно. Загальні витрати по продукції кожного виду, по суті, будуть такими самими, що і в випадку еквівалентної однопродуктової моделі. Таким чином, розглянута задача має вигляд:

$$C(y,\dots,y_n) = \sum_{i=1}^{n}\left(\frac{K_i\beta_i}{y_i} + \frac{h_i y_i}{2}\right) \Rightarrow Min \qquad (15.5)$$

$$\sum_{i=1}^{n} a_i y_i \le A,\ \forall\left(i = \overline{1,n}\right): y_i \ge 0.$$

Загальний розв'язок цієї задачі знаходиться за допомогою методу множників Лагранжа. Але перед тим, як застосовувати цей метод, необхідно встановити, чи діє вказане обмеження, перевіривши виконання обмеження на площу складу для розв'язку $y_i^* = \sqrt{\dfrac{2K_i\beta_i}{h_i}}$ необмеженої задачі. Якщо обмеження виконується, то воно зайве, і ним можна знехтувати.

Обмеження діє, якщо воно не виконується для значень y_i^*. В такому випадку потрібно знайти нове оптимальне значення y_i, що задовольняє обмеження на площу складу в ***вигляді рівності.*** Цей результат досягається побудовою функції Лагранжа виду:

$$L\ (\lambda\ , y_1, y_2, \dots, y_n) = C(y_1,\dots,y_n) - \lambda\ \left(\sum_{i=1}^{n} a_i y_i - A\right) =$$

$$= \sum_{i=1}^{n}\left(\frac{K_i\beta_i}{y_i} + \frac{h_i y_i}{2}\right) - \lambda\ \left(\sum_{i=1}^{n} a_i y_i - A\right) \qquad (15.6)$$

де $\hat{\lambda} < 0$ – множник Лагранжа.

Оптимальне значення y_i і λ можна знайти, прирівнявши до нуля відповідні часткові похідні, що дає:

$$\frac{\partial L}{\partial y_i} = -\frac{K_i\beta_i}{y_i^2} + \frac{h_i}{2} - \lambda \quad a_i = 0,$$
$$\frac{\partial L}{\partial \lambda} = -\sum_{i=1}^{n} a_i y_i + A = 0. \qquad (15.7)$$

З другого рівняння випливає, що значення y_i^* має задовільняти обмеження на площу складу в вигляді рівності.

З першого рівняння випливає, що $y_i^* = \sqrt{\dfrac{2K_i\beta_i}{h_i - 2\lambda^* a_i}}$. $\qquad$ (15.8)

Відзначимо, що y_i^* залежить від оптимального значення λ^* множника λ. Крім того, при $\lambda^* = 0$ значення y_i^* є розв'язком задачі без обмеження.

Значення λ^* можна знайти методом проб і помилок або ж скерованого перебору. Так, за визначенням в поставленій вище задачі мінімізації $\lambda < 0$, то при послідовній перевірці від'ємних значень λ знайдене значення λ^* буде одночасно визначати значення y^*, які задовільняють задане обмеження в вигляді рівності. Таким чином, в результаті визначення λ^* автоматично отримуються значення y_i^*.

ПРИКЛАДИ

Приклад 15.1. Модель Уілсона.

Об'єм продаж крамниці складає 500 одиниць товару в рік. Величина попиту рівномірно розподілена протягом року. Ціна купівлі однієї одиниці товару становить 2 грошові одиниці. За доставку замовлення власник крамниці повинен заплатити 10 грошових одиниць. Час доставки замовлення від постачальника складає 12 робочих днів (при 6-денному робочому тижні). Витрати на зберігання складають 20% середньорічної вартості запасів. Необхідно визначити: 1) яку кількість одиниць товару повинен замовляти власник крамниці для однієї поставки; 2) частоту замовлень; 3) точку замовлення. Відомо, що крамниця працює 300 днів на рік.

Розв'язання.

Плановим періодом є рік, $\beta = 500$ пакетів в рік, $K = 10$ грошових одиниць, витрати на зберігання однієї одиниці товару на рік складають 20% від вартості запасу в одну одиницю товару, тобто $h = 0,2 \times 2 = 0,4$

грошові одиниці. Тоді

$$y^* = \sqrt{\frac{2K\beta}{h}} = \sqrt{\frac{2\times 10\times 500}{0,4}} = 158,11 \text{ одиниць товару.}$$

Оскільки число одиниць товару повинне бути цілим, то замовлятимемо по 158 одиниць. При такому замовленні річні витрати становлять:

$$C = K\cdot\frac{\beta}{y} + h\cdot\frac{y}{2} = 10\cdot\frac{500}{158} + 0,4\cdot\frac{158}{2} = 63,25 \text{ грошових одиниць на рік.}$$

Подачу кожного нового замовлення власник магазину повинен здійснювати через

$$t_0^* = \frac{y^*}{\beta} = \frac{158}{500} = 0,316$$

року. Оскільки відомо, що в цьому випадку рік становить 300 робочих днів, то $t = 0,316\times 300 = 94,8 \approx 95$ робочих днів. Замовлення слід подавати при рівні запасу, що становить $y_0 = \beta T_r = \dfrac{500}{300}\cdot 12 = 20$ одиниць товару, тобто ці 20 одиниць буде продано протягом 12 днів, поки доставлятиметься замовлення.

Приклад 15.2. Однопродуктна статична модель з запізненням.

Щоденний попит на деякий товар складає близько 100 од. Витрати на розміщення кожного замовлення сталі й рівні 100 грошових одиниць. Щоденні витрати на зберігання одиниці запасу становлять 0,02 грошових одиниці. Потрібно визначити економічний розмір партії й точку замовлення при терміні виконання замовлення, що дорівнює 12 днів.

Розв'язання.

З наведених вище формул для економічного розміру партії отримаємо

$$y^* = \sqrt{\frac{2K\beta}{h}} = \sqrt{\frac{2\times 100\times 100}{0.02}} = 1000 \text{ од.}$$

Відповідна оптимальна тривалість циклу складає:

$t_0^* = y^* / \beta = 1000/100 = 10$ днів.

Оскільки термін виконання замовлення дорівнює 12 днів і тривалість циклу становить 10 днів, відновлення замовлення відбувається, коли рівень запасу достатній для задоволення попиту на два (=12-10) дні. Таким чином, замовлення розміром $y^* = 1000$ розміщується, коли рівень запасу досягає $2\times100 = 200$ одиниць.

Потрібно відзначити, що "ефективний" термін виконання замовлення приймається рівним 2, а не 12 дням. Це пояснюється тим, що цей термін більший, ніж t_0^*. Однак після стабілізації системи (у цьому прикладі вона досягається за два цикли) можна вважати, що термін виконання замовлення рівний $L - t_0^*$ при $L > t_0^*$. В описаних умовах у будь-який момент часу є більше ніж одне розміщене, але ще не виконане замовлення.

Приклад 15.3. Зміна запасу при наявності резерву.

Припустимо, що попит у попередньому прикладі у дійсності є апроксимацією випадкової ситуації, при якій ***щоденний*** попит розподілений нормально із середнім $\mu = 100$ і середнім квадратичним відхиленням $\sigma = 10$. Необхідно визначити розмір резервного запасу таким чином, щоб імовірність зменшення запасу протягом терміну виконання замовлення не перевищувала 0,05.

Розв'язання.

В попередньому прикладі цей термін дорівнює 2 дні. Так як щоденний попит розподілений нормально, залізнення попиту x_L також має нормальний розподіл з середнім $\mu_L = 2\times100 = 200$ одиниць і середнім квадратичним відхиленням $\sigma_L = \sqrt{2\times10^2} = 14{,}14$.

Таким чином, $P(x_L \ge \mu_L + B) \le \alpha$,

$$P\left\{\frac{x_L-\mu_L}{\sigma_L} \ge \frac{B}{\sigma_L}\right\} \le \alpha \text{, або } P\left\{\frac{x_L-\mu_L}{\sigma_L} \ge \frac{B}{14.14}\right\} \le 0.05 .$$

З таблиць нормального розподілу отримуємо $B/14.14 \ge 1.64$ або $B \ge 23.2$.

У прикладі цікавим є той факт, що B не залежить від μ_L. Цей результат є очікуваним, так як визначаючим фактором є середнє квадратичне відхилення. Дійсно, якщо середньоквадратичне відхилення рівне нулю (детермінований випадок), розмір резервного запасу повинен бути нульовим.

Приклад 15.4. Однопродуктова статична модель з "розривами" цін.

Витрати на замовлення становлять 10 грошових одиниць, витрати на зберігання продукції - 1 грошова одиниця на добу, інтенсивність споживання товару 5 одиниць на день, ціна товару – 2 грошові одиниці за одиницю товару, а при об'ємі закупівлі 15 одиниць і більше – 1 грошова одиниця. Необхідно визначити оптимальний розмір замовлення, ціну покупки і загальні витрати на управління запасами.

Розв'язання.

Починаємо розв'язання з приблизної побудови (пунктирними лініями) графіків двох функцій загальних витрат, кожна з яких відповідає одній з двох цін. (рис.15.11).

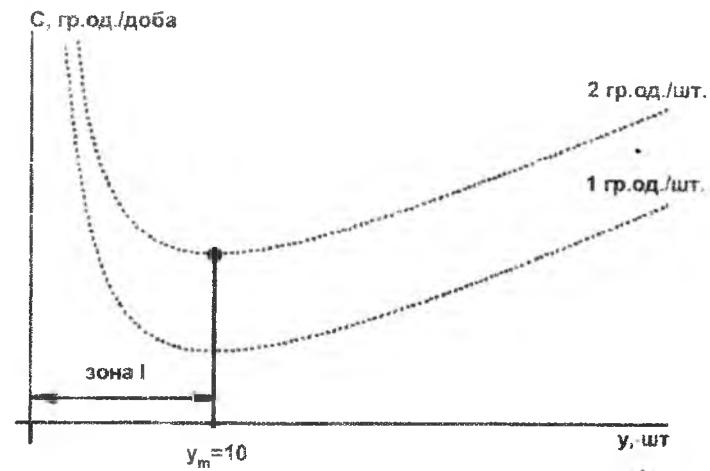


Рис. 15.11. Залежності загальних витрат від ціни товару

Використовуючи параметри $K = 10$ грошових одиниць, $\beta = 5$ од. за день, $h = 1$ грошова одиниця за 1 шт. в добу, обчислюємо значення y_m та відображаємо його на рис. 15.11.: $y_m = \sqrt{\frac{2\cdot10\cdot5}{1}} = 10$ [шт.].

Очевидно, що в зону I $q = 15$ шт. не потрапляє, оскільки $q > y_m$. Таким чином, q може потрапити в зони II або III. Межею між цими зонами служить розмір замовлення, що зрівнює загальні витрати при ціні із знижкою 1 грош.один.шт. і витрати при замовленні за початковою ціною 2 грош.один./шт. (рис.15.12).

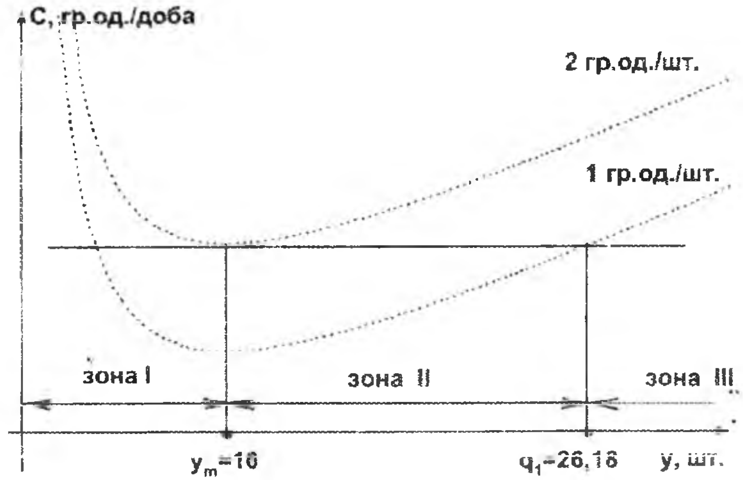


Рис. 15.12. Визначення меж зон

Запишемо вираз, що показує рівність витрат і дозволяє визначити межу зони II та зони III – $C_1(y_m) = C(q_1)$ з чисельними значеннями параметрів: $C_1(10) = C(q_1)$

$$C_1(10) = 10\times\frac{5}{10} + 1\times\frac{10}{2} + 2\times5 = 20,$$

$$C(q_1) = 10\times\frac{5}{q_1} + 1\times\frac{q_1}{2} + 1\times5 = \frac{50}{q_1} + \frac{q_1}{2} + 5,$$

$\frac{50}{q_1} + \frac{q_1}{2} + 5 = 20$, $q_1^2 - 30q_1 + 100 = 0$, $q_1 = 26{,}18$ шт. або $q_1 = 3{,}82$.

Обираємо більший з коренів, оскільки менший за значенням корінь не дає нам інформації про межу областей II і III.

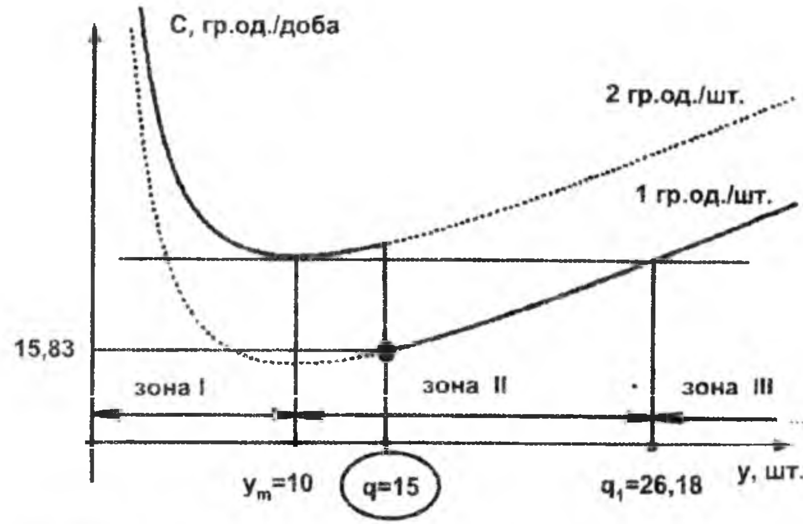


Рис. 15.13. Точка розриву цін

Таким чином, точка розриву цін $q = 15$ потрапляє в область II, оскільки $10 \le 15 \le 26{,}18$. Суцільною лінією обведені ті ділянки обох функцій витрат, які відповідають діючим цінам, тобто до об'єму $q = 15$ обведемо верхню лінію витрат, а після – нижню.

Таким чином, оптимальним є об'єм замовлення $y^* = 15$ шт. за ціною 1 грош.один./шт. Таким чином, у даній ситуації знижкою користуватися вигідно. Загальні витрати при цьому складають:

$$C_1(15) = 10\cdot\frac{5}{15} + 1\cdot\frac{15}{2} + 1\cdot5 = 15{,}83 \text{ [грош. од/добу]}.$$

Якщо б замовляли по 10 шт. товару, то загальні витрати склали б 20 грошових одиниць, тобто при замовленні в 15 шт. економія засобів складає 4,17 грошових одиниць на добу.

Приклад 15.5. Однопродуктова статична модель з "розривами" цін.

Модель управління запасами має наступні вхідні дані: $K = 10$ грошових одиниць, $h = 1$ грошова одиниця, $\beta = 5$, $c_1 = 2$ грошові одиниці, $c_2 = 1$

грошова одиниця і $q=15$ од. Необхідно визначити оптимальний рівень запасу.

Розв'язання.

Обчислимо спочатку значення y_m:

$$y_m=\sqrt{\frac{2K\beta}{h}}=\sqrt{\frac{2\times10\times5}{1}}=10 \text{ од.}$$

Оскільки $q>y_m$, необхідно визначити, де знаходиться q: в зоні II або III. Значення q_1 знаходиться з рівняння $C_1(y_m)=C_2(q_1)$ або

$$c_1\beta+\frac{K\beta}{y_m}+\frac{hy_m}{2}=c_2\beta+\frac{K\beta}{q_1}+\frac{hq_1}{2}$$

В результаті підстановки отримаємо:

$$2\times5+\frac{10\times5}{10}+\frac{1\times10}{2}=1\times5+\frac{10\times5}{q_1}+\frac{1\times q_1}{2}, \text{ або } q_1^2-30q_1+100=0.$$

Звідси отримаємо $q_1=26,18$ або $q_1=3,82$. За визначенням в якості q_1 обирається більше значення. Так як $y_m<q<q_1$, величина q знаходиться в зоні II. Таким чином, $y^*=q=15$ од. Сумарні витрати за одиницю часу визначаються наступним чином:

$$C_1=(y^*)=C_2(15)=c_2\beta+\frac{K\beta}{15}+\frac{h\times15}{2}=1\times5+\frac{10\times5}{15}+\frac{1\times15}{2}=15,83$$

грошових одиниць/день.

Приклад 15.6. Багатопродуктова статична модель з обмеженнями на ємність складських приміщень.

Розглянемо задачу управління запасами для випадку трьох видів продукції $(n=3)$, вихідні дані якої наведені в таблиці.

Вид продукції, i	K_i, гр.од.	β_i, од.	h_i, гр.од.	a_i, $і^2$
1	10	2	0,3	1
2	5	4	0,1	1
3	15	4	0.2	1

Припустимо, що загальна площа складу становить $A=25і^2$. Необхідно визначити оптимальні запаси продукції 3-х видів.

Розв'язання.

Виходячи з формули $y_i^*=\sqrt{\dfrac{2K_i\beta_i}{h_i-2\lambda^* a_i}}$, побудуємо наступну таблицю:

λ	y_1	y_2	y_3	$\sum_{i=1}^{3}a_iy_i-A$
0	11,5	20,0	24,5	+31
-0,05	10,0	14,1	17,3	+16,4
-0,10	9,0	11,5	14,9	+10,4
-0,15	8,2	10,0	13,4	+6,6
-0,20	7,6	8,9	12,2	+3,7
-0,25	7,1	8,2	11,3	+1,6
-0,30	6,7	7,6	10,6	-0,1

При $A=25м^2$ обмеження на складську площу задовольняється в вигляді рівності при деякому значенні λ, що лежить між -0,25 і -0,3. Це значення рівне λ^*, і його можна оцінити з допомогою лінійної інтерполяції. Відповідні значення y_i визначають значення y_i^*. Оскільки з таблиці видно, що значення λ^* дуже близьке до -0,3, то оптимальні значення y_i^* приблизно дорівнюють $y_1^*=6,7,\ y_2^*=7,6,\ і\ \ y_3^*=10,6$.

Якщо $A\geq52,4$, то значення y_i без врахування обмеження, відповідні $\lambda=0$, визначають y_i^*. В цьому випадку обмеження надлишкове.

РЕЗЮМЕ

15.1. Основна проблема, що виникає при розв'язанні задачі управління запасами, полягає в створенні ефективної і надійної системи управління рівнем наявних запасів. Основною інформацією, необхідною для успішного функціонування системи управління запасами, є рівень наявних запасів і запасів, що будуть створені за рахунок розміщення замовлень, а також накопичення за замовленнями споживачів. Кожне правило прийняття рішень є результатом вирішення деякої часткової задачі оптимізації управління запасами. Задачі, що доводиться вирішувати користувачу в практичних умовах, є різними, тому що конкретні системи мають різні параметри. Тому досліднику систем необхідно знати умови, за яких одне правило прийняття рішень виявляється кращим за інше. Поточний *контроль за виконанням плану дає можливість органам управління вирішувати питання про доцільність втручання у функціонування системи, тобто яким чином і коли необхідно здійснювати управління системою загалом, а не запасами конкретної продукції.*

15.2. Задача управління запасами виникає, коли необхідно створити запас матеріальних ресурсів або предметів споживання з метою задоволення попиту на певному заданому інтервалі часу. В будь-якій задачі управління запасами потрібно визначати кількість замовленої продукції і строки розміщення замовлень. Попит можна задовольнити шляхом одноразового створення запасу на весь потрібний період часу або шляхом створення запасу для кожної одиниці часу цього періоду. Рішення щодо розміру замовлення і моменту його розміщення можуть базуватися на мінімізації відповідної функції загальних затрат, що обумовлені втратами від залишкового запасу і дефіциту. Довільна модель управління запасами остаточно повинна дати відповідь на два запитання: Яку кількість продукції замовляти? Коли замовляти? У випадку періодичного контролю стану запасу потрібно забезпечувати надходження нової кількості ресурсів об'ємом в розмір замовлення через рівні проміжки часу. У випадку неперервного контролю стану запасу необхідно розміщувати нове замовлення розміром в об'єм запасу, коли його рівень досягає точки замовлення.

15.3. Детермінований попит може бути статичним, в тому сенсі, що інтенсивність споживання залишається незмінною з часом, або динамічним, коли попит відомий достовірно, але змінюється в залежності від часу. Стохастичний попит є стаціонарним, якщо функція щільності імовірності попиту незмінна в часі, і нестаціонарним, коли функція щільності попиту змінюється в часі. Найточніше характер попиту може бути описаний за допомогою нестаціонарних розподілів вірогідностей. Однак із математичної точки зору модель значно ускладнюється, особливо при збільшенні періоду часу, що розглядається. На першому рівні абстрагування припускається, що розподіл вірогідності попиту стаціонарний у часі. На другому рівні абстракції враховуються зміни попиту від одного періоду до іншого. На третьому рівні спрощення виключаються як елементи ризику, так і зміни попиту.

15.4. Узагальнену модель управління запасами, яка враховувала б усі різновиди умов, що спостерігаються в реальних системах, побудувати важко. Але, якщо вдалося побудувати достатньо універсальну модель, на ній неможливо отримати аналітичні розв'язки. Для розв'язування

можуть використовуватися різні методи, які включають класичну схему оптимізації, лінійне й динамічне програмування. На практиці отримав розповсюдження наближений метод, який зберігає простоту моделі економічного розміру замовлення і в той же час у якійсь мірі враховує вірогідніший характер попиту. Ідея методу надзвичайно проста. Вона передбачає створення деякого (постійного) буферного запасу на всьому горизонті планування. Розмір резерву визначається таким чином, щоб вірогідність зменшення запасу протягом періоду виконання замовлення L не перевищувала наперед заданої величини. Різновиди моделей економічного розміру замовлення (партії) допускають можливість дефіциту і рівномірного (а не миттєвого) збільшення запасу. Останній випадок є типовим для виробничих систем, в яких інтенсивність збільшення є функцією інтенсивності виробництва. В цих ситуаціях у моделях управління запасами, як і раніше, співставляються витрати на зберігання запасів і оформлення замовлень. У функцію сумарних витрат включаються також втрати від дефіциту, якщо він має місце. В загальному випадку втрати від дефіциту припускаються пропорційними середній величині дефіциту. В багатьох випадках ціна одиниці продукції залежить від розмірів закупленої партії. У таких випадках ціни змінюються стрибкоподібно або надаються гуртові знижки. При цьому в моделі управління запасами необхідно враховувати витрати на придбання. Загальний розв'язок багатопродуктової задачі знаходиться за допомогою методу множників Лагранжа.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 15.1. Модель Уілсона.

При будівництві ділянки автомобільного шляху завдовжки 500 м використовують гравій, витрати якого складають 120 кг/м. Терміни будівництва становлять 17 діб. Робота виконується в одну зміну. Витрата гравію рівномірна. Гравій довозиться вантажними машинами місткістю 7 т протягом 4 годин. Витрати на один рейс вантажівки становлять 15 грошових одиниць. Витрати на зберігання однієї тонни гравію на місці будівництва складають 1,1 грошових одиниць за добу.

Визначити параметри управління запасами: оптимальний об'єм замовлення, кількість вантажних машин, що використовуються для доставки, період поставок, точку замовлення, загальні витрати на управління запасами для будівництва ділянки шляху в 500 м.

Завдання 15.2. Модель з розривами цін.

Для задачі прикладу 15.1 визначити, чи вигідно власнику крамниці скористатися однією із знижок, що надаються постачальником? Які при цьому будуть розмір замовлення і загальні витрати?

Розміри знижок наведені в таблиці.

Розмір замовлення	Ціна, грош. од./шт.
1–199	2
200–499	1,96 (2% знижки)
500 і більш	1,92 (4% знижки)

Зауваження. При розв'язуванні задач з двома знижками спочатку шукається оптимальный об'єм замовлення з врахуванням першої знижки, а потім розглядається наступна знижка.

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. У чому суть основної проблеми управління запасами?
2. Які завдання розв'язуються на кожному з рівнів грирівневої системи управління запасами?
3. Які особливості має система управління запасами, орієнтована на споживача?
4. Що є характерним для системи управління запасами, орієнтованої на постачальника?
5. Розкрийте сенс відповідей на основні запитання, які повинна давати система управління запасами.
6. Виходячи з яких міркувань визначаються розмір і точка замовлення?
7. Від чого залежать сумарні витрати системи управління запасами?
8. Розкрийте сенс вартісної моделі управління запасами.
9. Які типи моделей управління запасами існують і як зростає ступінь математичної складності відповідних задач?
10. У чому полягають загальні особливості детермінованих моделей управління запасами?
11. Як визначаються характеристики однопродуктної моделя управління запасами?
12. В яких випадках виникають „розриви цін" та як визначаються точки розриву?

13. Які особливості властиві багатопродуктовій статичній моделі управління запасами з обмеженнями на ємність складських приміщень?
14. За допомогою яких методів та яким чином розраховуються характеристики багатопродуктових моделей управління запасами?

ТЕМА 16. АНАЛІЗ НА МОДЕЛЯХ УПРАВЛІННЯ ЗАПАСАМИ

При необхідності поповнити запаси продукції розв'язується задача визначення розміру замовлення (чи розміру партії) продукції. Відновлення рівня запасів може здійснюватися шляхом виконання замовлення на виготовлення деякого обсягу продукції. Вирішуючі правила розробляються зазвичай на другому рівні трирівневої системи управління запасами для визначення економічно обґрунтованого оптимального розміру замовлення. Ці правила використовуються при визначенні запланованого чи максимального рівня запасу для конкретного типу чи типів продукції. Різні підходи до розв'язання задачі визначення економічного розміру замовлення базуються на тих чи інших можливих припущеннях щодо таких показників, як спосіб постачання замовленої продукції, інтенсивність реалізації продукції, витрати на освоєння виробництва нової продукції, собівартість одиниці продукції і поточні витрати на збереження запасів. Спочатку для кожного з перерахованих факторів робиться найпростіше припущення, а потім розглядаються умови, при яких можливі припущення виявляються кращими, ніж первісне. Оскільки припущення щодо одного з факторів не пов'язані з припущеннями щодо інших факторів, то можливі принаймні сотня комбінацій, що приводять до різних вирішуючих правил. З такої множини вирішуючих правил можна обрати найприйнятніше, однак доцільність того чи іншого вибору визначається з урахуванням конкретної ситуації, у якій ці правила будуть реалізовані. Рішення щодо подачі замовлення на поповнення запасів приймається системою управління нижнього рівня і залежить від результатів порівняння поточного стану запасу з максимальним значенням розумних потреб на інтервалі виконання, що зберігається як точка замовлення. Для визначення коефіцієнта резервного запасу використовуються такі правила, як правило тритижневого запасу, правило постійного коефіцієнта резервного запасу, правило мінімального числа невиконаних замовлень, правило мінімального числа випадків дефіциту запасів та інші. Матеріальні запаси служать для того, щоб згладити безпосередню залежність між динамікою виробництва продукції та її споживанням. Наявність запасів дозволяє налагодити виробництво продукції оптимальними партіями, а також визначити оптимальні партії поставок кожного продукту. З точки зору затримок функціонування системи важливе не просто безумовне (100%), безперервне постачання продукцією.

При такому постачанні потрібно б було мати дуже великі запаси. Виникає задача визначення оптимального рівня запасів. Оптимальний рівень запасів складається з економічно оптимальної партії поставки плюс деякий страховий запас. Необхідність страхового запасу диктується випадковими явищами, які органічно наявні в будь-якій системі матеріально-технічного постачання. Комбінуючи елементарні операції управління та вирішуючі правила, можна створити моделі систем управління запасами різної структури, і шляхом моделювання з використанням реальної інформації за минулі періоди обрати найвідповідніший варіант для реалізації в конкретних умовах.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати: ⇨проблематику обґрунтування економічно вигідного розміру партії; ⇨вирішуючі правила, що визначають оптимальний рівень резервного запасу; ⇨основні співвідношення для визначення рівня страхового запасу в умовах ризику;

вміти: ⇨розраховувати значення економічно вигідного розміру партії для конкретних прикладів систем управління запасами; ⇨ застосовувати правила прийняття рішень щодо рівня резервного запасу до конкретних систем; ⇨визначати основні характеристики системи управління запасами в стохастичному середовищі.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ стохастичні моделі
- ☑ функція розподілу
- ☑ час попередження
- ☑ випадковий попит
- ☑ розмір партії
- ☑ вирішуюче правило
- ☑ рівень запасу
- ☑ відновлення рівня запасів
- ☑ затрати на зберігання
- ☑ модель максимум/мінімум
- ☑ інтенсивність споживання
- ☑ модель з фіксованим інтервалом
- ☑ поповнююча партія продукції
- ☑ поточні витрати на зберігання
- ☑ сумарні затрати
- ☑ резервний запас
- ☑ штраф за дефіцит
- ☑ щільність розподілу
- ☑ коефіцієнт ризику
- ☑ нерівномірність попиту
- ☑ модель точки замовлення та розміру партії

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

16.1. Стохастичні моделі управління запасами.

16.2. Проблеми аналізу та вибору економічно вигідного розміру партії.

16.3. Прийняття рішень щодо рівня резервного запасу.

16.4. Структура систем управління запасами.

16.1. СТОХАСТИЧНІ МОДЕЛІ УПРАВЛІННЯ ЗАПАСАМИ

В стохастичних моделях управління запасами попит є випадковою величиною, що описується законами теорії вірогідностей. Врахування випадковості суттєво ускладнює аналіз та отримання рішень на таких моделях, а тому розглянемо найпростіші з них.

Припустимо, що попит r за інтервал часу T є випадковим і заданий його закон розподілу (дискретний) $p(r)$ або ж щільність розподілу вірогідності $f(r)$. Якщо попит r є нижчий, ніж рівень запасу s, то закупівля (збереження, продаж) залишку продукту потребуватимуть додаткових затрат c_2 на одиницю продукту. Якщо ж попит вищий за рівень запасу, то це приводить до штрафу за дефіцит c_3 на одиницю продукції. Критерієм затрат, оскільки вони є випадковою величиною, вважатимемо математичне сподівання сумарних затрат $M[s]$.

Для моделі, що розглядається, матимемо у випадку дискретного попиту:

$$M[s]=c_2\sum_{r=0}^{s}(s-r)p(r)+c_3\sum_{r=s+1}^{\infty}(r-s)p(r). \qquad (16.1)$$

Перша складова враховує затрати на набуття (зберігання) надлишку $(s-r)$ одиниць продукту (при $r \le s$), а друга – штраф за дефіцит в $(r-s)$ одиниць продукту (при $r>s$). У випадку неперервного випадкового попиту, що задається щільністю розподілу $f(r)$, для математичного сподівання сумарних витрат отримаємо:

$$M[s]=c_2\int_0^s(s-r)f(r)dr+c_3\int_s^{\infty}(r-s)f(r)dr. \qquad (16.2)$$

Таким чином, задача полягає у визначенні такого запасу s, при якому математичне сподівання сумарних витрат є найменшим. Якщо s^* – запас, що мінімізує $M[s]$, то справедливі наступні співвідношення:

$F(s^*)<\eta<F(s^*+1)$ для дискретного розподілу, (16.3)

і $F(s^*)=\eta$ - для неперервного, (16.4)

де $F(s)=p(r<s)$ - інтегральна функція розподілу попиту,

$$\eta=\frac{c_3}{c_2+c_3} \qquad (16.5)$$

– щільність збитків внаслідок незадоволеного попиту. В управлінні запасами η має велике значення, $0 \le \eta \le 1$: якщо значення c_3 мале порівняно з c_2, то η близьке до нуля, якщо ж, навпаки, значно перевищує його, то η близьке до одиниці. Таким чином, η можна розглядати як міру припустимості дефіциту – якщо дефіцит не допускається, то $c_3=\infty$, тобто $\eta=1$.

Припустимо тепер, що витрачання запасу здійснюється неперервно з однаковою інтенсивністю (рис. 16.1).

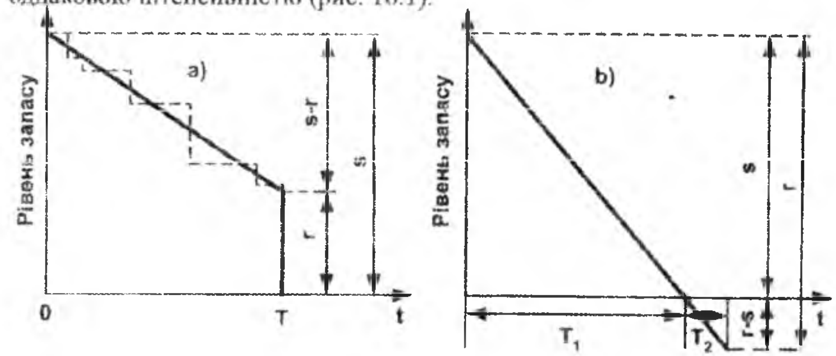


Рис. 16.1. Витрачання запасу з постійною інтенсивністю

Варіант a) відповідає випадку, коли попит не перевищує запасу $(r \le s)$, а b) – коли попит перевищує запас $(r>s)$. Насправді витрачання запасу здійснюється невеличкими стрибками (пунктирна ламана на рис. 16.1a), але пряма є непоганим наближенням до цієї ламаної.

Середній запас, що відповідає варіанту a), становить:

$$s_1=\frac{1}{2}\left(s+(s-r)\right)=s-\frac{1}{2}r. \qquad (16.6)$$

Середній запас, що відповідає варіанту b) з врахуванням того, що

$T_1=\frac{s}{r}T,\ T_2=\frac{r-s}{r}$, становить $s_2=\frac{1}{2}s\frac{T_1}{T}=\frac{s^2}{2r}$. (16.7)

Таким чином, середній дефіцит продукту за період T_2 для випадку b) буде $s_3=\frac{1}{2}(r-s)\frac{T_2}{T_1}=\frac{1}{2}\frac{(r-s)^2}{r}$, (16.8) і математичне сподівання сумарних витрат становитиме:

$$M[s]=c_2\sum_{r=0}^{s}\left(s-\frac{r}{2}\right)p(r)+c_2\sum_{r=s+1}^{\infty}\frac{s^2}{r}p(r)+c_3\sum_{r=0}^{\infty}\frac{1}{2}\frac{(r-s)^2}{r}p(r). \quad (16.9)$$

Доведено, що в цьому випадку математичне сподівання сумарних витрат є мінімальним при рівні запасу s^*, для якого справедлива нерівність:

$$L(s^*)\le\eta\le L(s^*+1), \quad (16.10)$$

де $L(s)=F(s)+\left(s-\frac{1}{2}\right)\sum_{r=s}^{\infty}\frac{p(r)}{r}$, $F(s)=p(r<s)$. (16.11)

Ця модель є достатньо ідеалізованою, оскільки вважається, що поповнення запасу відбувається миттєво. Однак в багатьох задачах час затримання поставок виявляється значним, і його необхідно враховувати в моделі.

Нехай за час затримання поставок θ вже замовлені n партій по одній в кожному з періодів тривалістю $T=\theta/n$. Позначимо s_0 - початковий рівень запасу (до початку першого періоду), s_i, r_i, q_i – відповідно запас, попит та поповнення запасу за i-й період.

В цьому випадку до завершення n-го періоду в комору надійде $\sum_{i=1}^{n}q_i$ одиниць продукту, а використано буде $\sum_{i=1}^{n}r_i$ одиниць, і, таким чином,

$$s_n=s_0+\sum_{i=1}^{n}q_i-\sum_{i=1}^{n}r_i\text{, або } s_n=s-r, \quad (16.12)$$

де $s=s_0+\sum_{i=1}^{n}q_i$, $r=\sum_{i=1}^{n}r_i$.

Необхідно визначити оптимальний об'єм партії замовлення, який потрібно зробити на n-й період.

Математичне сподівання сумарних витрат визначимо як:

$$M[s]=c_2\sum_{r=0}^{s}(s-r)p(r)+c_3\sum_{r=s+1}^{\infty}(r-s)p(r), \quad (16.13)$$

а оптимальний запас – за співвідношенням:

$$F(s^*)<\eta<F(s^*+1). \quad (16.14)$$

Знайшовши оптимальний запас s^*, та знаючи $q_1, q_2,...,q_{n-1}$, обчислимо:

$$q_n=s^*-\left(s_0+\sum_{i=1}^{n-1}q_i\right). \quad (16.15)$$

Аналітично визначити оптимальні значення точки запасу та об'єму партії вдається лише в найпростіших випадках. Якщо ж система має складну структуру з багатьма видами продукції та коморами, то єдиним способом її аналізу виявляється імітаційне моделювання.

16.2. ПРОБЛЕМИ АНАЛІЗУ ТА ВИБОРУ ЕКОНОМІЧНО ВИГІДНОГО РОЗМІРУ ПАРТІЇ

При необхідності поповнити запаси продукції розв'язується задача визначення розміру замовлення (чи розміру партії) продукції. Відновлення рівня запасів може здійснюватися або шляхом виконання замовлення на виготовлення деякого фіксованого обсягу продукції, зареєстрованого системою нижнього рівня, або шляхом виконання замовлення на виготовлення об'єму продукції, рівного різниці між рівнем розміщуваного запасу (який дорівнює: Наявний запас + Об'єм замовлень – Очікувані витрати продукту) і максимальним рівнем запасів, дані про який також зберігаються в системі обліку.

Вирішуючі правила розробляються зазвичай на другому рівні трирівневої системи управління запасами для визначення економічно обґрунтованого оптимального розміру замовлення. Ці правила використовуються при визначенні запланованого чи максимального рівня запасу для конкретного типу чи типів продукції.

Аналіз та формування припущень щодо умов функціонування системи керування запасами.

Різні підходи до розв'язання задачі визначення економічного розміру замовлення (партії продукції) базуються на тих чи інших можливих припущеннях щодо таких показників, як спосіб постачання замовленої продукції, інтенсивність реалізації продукції, витрати на освоєння виробництва нової продукції, собівартість одиниці продукції і поточні витрати на збереження запасів. Спочатку для кожного з перерахованих факторів робляться найпростіші припущення, а потім розглядаються умови, за яких можливі припущення виявляються кращими, ніж первісне. Оскільки вважається, що припущення щодо одного з факторів не пов'язані з припущеннями щодо інших факторів, то можливі сотні комбінацій факторів, що приводять до різних вирішуючих правил. З такої множини вирішуючих правил можна обрати найприйнятніше, однак доцільність того чи іншого вибору визначається з урахуванням конкретної ситуації, у якій ці правила будуть реалізовані.

Розглянемо деякі можливі припущення щодо зазначених факторів.

Спосіб постачання замовленої продукції.

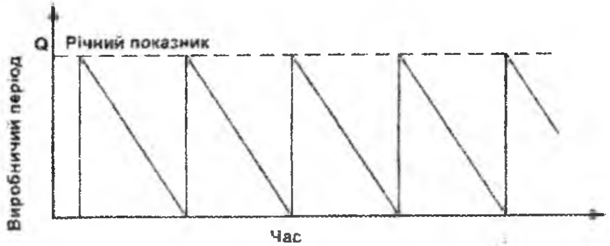


Рис. 16.2. Наявний запас при одержанні замовлення однієї партії

1). Уся замовлена продукція постачається у вигляді однієї партії. Відповідно до моделі, наявний запас зростає від мінімального рівня таким чином, що до початку наступного циклу досягає розміру партії.

2). Продукція надходить в комору з моменту початку виробничого циклу протягом інтервалу часу, достатнього для реалізації продукції до постачання всієї партії (рис. 16.3).

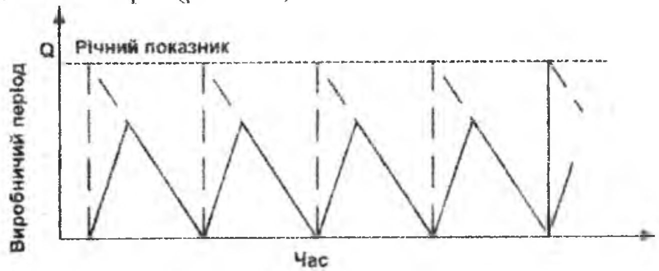


Рис. 16.3. Розподіл у часі запасу виробленої продукції

Інтенсивність і терміни реалізації продукції.

1). Продукція реалізується з постійною інтенсивністю протягом необмеженого часу. Це вимагає багаторазового поповнення запасів за рахунок замовлень однакового розміру, причому наявний запас буде зменшуватися за лінійним законом (рис. 16.2 і 16.3). Практично припущення про необмежений час реалізації означає, що необхідно принаймні більш ніж три поповнюючі партії продукції.

2). Продукція реалізується з відомою чи прогнозованою інтенсивністю, що змінюється в часі. У цьому випадку для одержання точного розв'язку потрібно застосовувати методи динамічного програмування.

Якщо потреба в продукції, що користається високим попитом, припадає на кінець певного інтервалу періоду планування (як це є з продукцією, що швидко виходить з моди, чи з новою продукцією, для якої передбачається експоненційне зростання попиту), то не можна використовувати жоден із запропонованих наближених методів. Якщо протягом періоду планування можливі і високий, і низький попит, то розходження між можливими наближеннями незначні й оцінки їх, звичайно, робляться на основі окремих випадкових пробних вибірок, а не на основі принципових відмінностей.

3). При значних випадкових коливаннях попиту мінімальний рівень запасу відмінний від нуля (а не дорівнює нулю, як це показано на рис. 16.2 і 16.3). У цьому випадку існує резервний запас. Якщо розмір партії продукції, що замовляється, збільшується в порівнянні з тим, що відповідає випадку виконання замовлень, то вірогідність дефіциту продукції протягом року зменшується. Отже, резервний запас повинен бути таким, щоб забезпечити бажаний рівень обслуговування споживачів. Загальна економія в результаті одноразового визначення рівня резервного запасу і розміру партії продукції для його поповнення здійснюється в тих випадках, коли розмір партії продукції, обчислений за умови відсутності дефіциту, менший, ніж стандартне відхилення помилок прогнозу на інтервалі випередження.

Розглянемо випадок, коли занадто дорога продукція, освоєння виробництва якої вимагає малих витрат, може бути випущена одномоментно в обсязі тижневого постачання, але реалізація її виявляється настільки нерівномірною, що стандартне відхилення еквівалентне розміру місячного постачання. Точне розв'язання задачі вимагає ітеративного процесу обчислень. Практично задовільним наближенням рішення є збільшення розміру партії принаймні до величини стандартного відхилення. Однак подібний розв'язок вимагає виконання значного обсягу обчислень, і тому подальше поліпшення результату може виявитися невиправданим з точки зору затрат на отримання розв'язку.

4). Продукція реалізується доти, поки не з'явиться непередбачена необхідність повного списання запасу, що залишився. У цьому випадку варто брати до уваги технологічні зміни, що плануються на майбутнє, для того щоб поступово звести до нуля поточні запаси. Для цього остання партія виробів, що постачається, повинна мати об'єм, достатній для споживання лише в межах планового періоду, що передує моменту запланованого нововведення.

5). Якщо інтенсивність реалізації продукції протягом деякого інтервалу часу періоду планування впаде до нуля (закінчення контракту, зміна моделі виробу, порушення неперервності і т.д.), то розмір останньої партії повинен бути таким, щоб компенсувати різницю між наявним запасом і загальним незадоволеним попитом. Для вирішення питання про те, чи економічно вигідним є виготовлення виробів у кількості, що задовольняє попит, який залишився, однією чи декількома партіями, потрібно використання моделей динамічного програмування. Якщо передбачається задоволення попиту шляхом виготовлення декількох партій виробів, то необхідно визначити розмір першої партії. Різниця між розміром партії в оптимальній послідовності і розміром партії без врахування спадів істотна лише при постачанні для задоволення попиту, що залишився, у вигляді не більш трьох партій.

Затрати на запуск у виробництво.

Ці затрати можуть включати витрати на налагодження виробничої лінії, організацію виконання календарного графіка виробництва і затрати на розміщення замовлення, які, врешті-решт, залежать від кількості виконаних протягом року замовлень. У цьому випадку можливі наступні припущення:

1). Освоєння виробництва продукції кожного нового типу незалежно від тих, що випускалися раніше, вимагає нових затрат.

2). Основні затрати пов'язані з розміщенням замовлення на виготовлення сукупності виробів, і незначні затрати пов'язані з включенням у поточне замовлення додаткових одиниць цієї ж продукції. Прикладами, що ілюструють ці випадки, можуть бути перелік товарів, придбаних в одного постачальника, різні продукти, що розливаються на конвеєрі в пляшки однакової ємності.

Варіант об'єднаних замовлень виявляється ефективним, якщо основні затрати на освоєння виробництва нової продукції перевищують загальні додаткові затрати по всій сукупності виробів.

Собівартість одиниці продукції.

При визначенні об'єму капіталовкладень у запаси враховуються припущення щодо собівартості продукції, що приводить до доцільності випуску більших партій виробів, ніж це потрібно в даний момент. Тут можливі наступні припущення.

1). Може бути вироблений будь-який обсяг продукції при постійній її собівартості.

2). Існують непередбачені зміни цін на продукцію. У цьому випадку, якщо затрати на розміщення майбутніх замовлень можуть зрости в порівнянні з існуючою собівартістю продукції, то поточний розмір партії,

що замовляється, повинний бути більшим, ніж той, котрий визначається величиною вірогідності невиконання замовлень споживачів; це дозволяє віддалити точку збільшення затрат на придбання виробів за підвищеною ціною. Економічно доцільніше збільшення обсягу разового постачання пропорційно очікуваному відносному збільшенню цієї ціни. На коефіцієнт пропорційності в основному впливають поточні управлінські витрати чи значення коефіцієнта приведення майбутніх витрат до біжучого моменту часу. Ефективність такого рішення може виявитися значно вищою, аніж при виборі стратегії повторення замовлень рівного розміру, що визначається умовою відсутності дефіциту без врахування зміни навколишніх умов.

3). Дисконтування об'єму замовлення. У тих випадках, коли збільшення розміру партії виробів понад обумовлений вірогідністю невиконання замовлень споживачів приводить до зниження собівартості продукції, можна визначити економічний розмір замовлення на основі співвідношення загальних витрат при недостачі запасів з витратами в кожен момент зміни ціни.

Можна рекомендувати зручне евристичне правило, відповідно до якого економічно доцільно кожні *M* місяців замовляти партію продукції, розмір якої визначається з умови відсутності дефіциту запасів, якщо зниження питомих витрат (собівартості) буде не менш ніж на *2M%* (виходячи з типового положення, коли щорічні поточні управлінські видатки лежать у межах 20–24%).

Поточні витрати на зберігання запасів.

Подібні затрати є керованою змінною, котра може бути збільшена, щоб зменшити об'єм капіталовкладень у запаси за рахунок виконання більшої кількості замовлень на рік. Величина поточних затрат, що могла б теоретично забезпечити максимальну віддачу капіталовкладень, дорівнює поточному значенню прибутку фірми на її нетто-активи, тобто повинна знаходитися в межах установленого діапазону 20–24% річних витрат. Однак реальні поточні витрати повинні бути більшими, ніж це значення, якщо керуючі органи вважають, що вкладення капіталу в запаси пов'язане з великим ризиком чи меншою ліквідністю, ніж за іншими об'єктами капіталовкладень фірми. Звідси випливає, що для певної великої корпорації поточні затрати на збереження запасів сировини можуть бути меншими, ніж на збереження запасів готової продукції, а поточні витрати на зберігання периферійних запасів будуть більшими, ніж на збереження запасів готової продукції в коморі підприємства. Можливі наступні варіанти.

1). Поточні витрати визначаються політикою капіталовкладень, що –

ґрунтується на врахуванні ризику виникнення дефіциту запасів.

2). Враховується займаний простір. Якщо є вироби великих габаритів і поряд з ними вироби, що розташовуються на складі досить компактно, то керуючі органи можуть в якості другої характеристики стратегії обрати затрати, що пов'язані з об'ємом комори, який зайнятий продукцією, і це повинно привести для великогабаритних виробів до зменшення розміру партії в порівнянні з тим, що був би у випадку відсутності дефіциту.

Зменшення поточних затрат приводить до збільшення капіталовкладень у запаси, але зате зменшує щорічні затрати, пов'язані з розміщенням замовлень на поповнення запасів, що у цьому випадку є рідшими.

У цьому сенсі існує максимальне практично доцільне значення поточних затрат. Коли для підтримки потрібного рівня обслуговування замовників застосовуються резервні запаси, стратегія дрібніших і частіших поповнюючих замовлень вимагає додаткового збільшення резервних запасів. Починаючи з деякого значення величини поточних затрат на збереження запасів, зменшення обсягу поточних запасів супроводжується збільшенням необхідного страхового запасу, і тому за межами зазначеної точки збільшення поточних затрат нічого не дає, крім збільшення затрат на розміщення замовлень без компенсації їх за рахунок зниження загального обсягу капітальних витрат.

16.3. ПРИЙНЯТТЯ РІШЕНЬ ЩОДО РІВНЯ РЕЗЕРВНОГО ЗАПАСУ

Таким чином, матеріальні запаси служать для того, щоб згладити безпосередню залежність між динамікою виробництва продукції та її споживанням. Наявність запасів дозволяє налагодити виробництво продукції оптимальними партіями, а також визначити оптимальні партії поставок по кожному продукту.

Матеріальний запас – це демпфер, що згладжує залежність споживача від можливих коливань випуску продукції, послаблює залежність виробничого процесу постачальника від нерівномірностей споживання цієї продукції споживачами; запаси згладжують залежність окремих цехів, робочих місць і т.д. один від одного. Завдяки створенню запасів вирівнюються та здешевлюються виробничі процеси. Запаси роблять систему стійкішою, вони створюють необхідні передумови для забезпечення неперервності виробничого процесу.

З точки зору затримок функціонування системи важливим є не просто

безумовне (100%), неперервне постачання продукцією. При такому постачанні потрібно було б мати надзвичайно великі запаси. Виникає задача визначення оптимального рівня запасів. Оптимальний рівень запасів складається з економічно оптимальної партії поставки плюс деякий страховий запас. Необхідність страхового запасу диктується випадковими явищами, які органічно наявні в будь-якій системі матеріально-технічного постачання.

Нехай k – витрати на оформлення замовлення, що мають місце щоразу при його розміщенні в припущенні, що витрати на зберігання одиниці замовлення ***в одиницю часу*** рівні S. Припускається, що інтенсивність попиту (в одиницю часу) дорівнює μ.

Нехай υ – випадкова величина з заданим законом розподілу $f(\upsilon)$ і скінченим математичним сподіванням μ. Якщо величину партії поставки визначати за формулою Уілсона $q=\sqrt{\dfrac{2k\mu}{S}}$, то період повторення замовлень становить $T=\sqrt{\dfrac{2k}{\mu S}}$. Але оскільки попит є випадковою величиною, то приблизно в 50% випадків буде спостерігатися дефіцит. Для гарантії від випадкових коливань в попиті або інших непередбачених ситуацій в системі необхідно мати деякі додаткові страхові запаси. При достатньо великому рівні страхового запасу практично можна досягнути бездефіцитного постачання. Але при цьому можливі значні витрати від іммобілізації засобів в запасах. При малому страховому запасі можливі втрати від дефіциту. Позначимо через P вірогідність того, що потреби в інтервалі T перебільшать наявний запас, тобто вірогідність того, що величина фактичної потреби Q буде більша ніж сума страхового запасу R і партії поставки $y-q$.

$$p(Q>q+R)=p \tag{16.16}$$

Щоб знайти R, необхідно знати розподіл випадкової величини Q.

1. Нехай випадкова величина Q має нормальний розподіл з математичним сподіванням q і середнім квадратичним відхиленням σ. Як відомо, функція щільності вірогідності при цьому буде мати вигляд:

$$f(Q)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(Q-q)^2}{2\sigma^2}}. \tag{16.17}$$

В якості випадкової величини введемо відхилення випадкової змінної Q від її математичного сподівання q, висловленого в середньоквадратичному відхиленні:

$$u = \frac{Q - q}{\sigma}. \quad (16.18)$$

Задача при цьому полягає в тому, щоб знайти таке значення u_p, для якого справедлива рівність:

$$\frac{1}{\sqrt{2\pi}} \int_{u_p}^{\infty} l^{-\frac{u^2}{2}} \alpha \;\; u = p. \quad (16.19)$$

Значення u_p, яке задовольняє рівності (16.19),можна знайти з таблиць нормального розподілу. З рівнянь (16.17) і (16.18) випливає, що величина страхового запасу, відповідно рівню обслуговування p, повинна задовольняти рівність:

$$R = Q - q = u_p \sigma. \quad (16.20)$$

Наприклад, при $u_p = 3$, тобто при $R = 3\sigma$ коефіцієнт ризику $p = 0{,}003$; при $R = 2\sigma$ $p = 0{,}046$; при $R = \sigma$ $p = 0{,}317$.

Таким чином, при нормальному розподілі величину страхового запасу повністю визначають заданий коефіцієнт обслуговування, тобто вірогідність того, що заданий рівень запасу практично буде достатнім.

Враховуючи рівність (16.20), середній рівень запасу, який задовольняє потреби з ймовірністю $(1 - p)$, висловиться формулою:

$$\overline{I} = \sqrt{\frac{k\mu}{2S}} + u_p \sigma. \quad (16.21)$$

2. Нехай випадкова величина попиту розподілена за законом Пуасона. Функція густини вірогідності при розподілі за цим законом має вигляд:

$$f(Q) = \frac{e^{-q} q^Q}{Q!}. \quad (16.22)$$

Можна показати, що якщо $q \to \infty$, то розподіл за законом Пуасона зводиться до особливого виду нормального розподілу з математичним сподіванням q і ссереднім квадратичним відхиленням $\sigma = \sqrt{q}$.

Тому величина страхового запасу становить $R = u_p \sigma = u_p \sqrt{q}$. (16.23)

3. Розглянемо випадок рівномірного розподілу попиту. Як відомо, розподіл ймовірностей називається рівномірним, якщо на інтервалі, якому належать всі можливі значення випадкової величини, функція розподілу випадкової величини має стале значення. Нехай Q_{max} і Q_{min} відповідно максимальне і мінімальне значення випадкової величини Q. Так як $f(Q)(Q_{max} - Q_{min}) = 1$, то $f(Q) = \frac{1}{Q_{max} - Q_{min}}$. (16.24)

Математичне сподівання рівномірно розподіленої випадкової величини рівне:

$$q = \int_{-\infty}^{+\infty} Qf(Q)\,dQ = \frac{1}{2}(Q_{max} - Q_{min}). \quad (16.25)$$

Величина страхового запасу визначається з рівності (16.16). Знайдемо величину страхового запасу. Нехай Q_p – величина споживання з заданою ймовірністю p. Так як p – це вірогідність того, що створений страховий запас R буде недостатнім, то

$$Q_p = (1 - p)(Q_{max} - Q_{min}) \quad (16.26)$$

і, відповідно, (16.27)

$$R = Q_p - q = (1 - p)(Q_{max} - Q_{min}) - \frac{1}{2}(Q_{max} - Q_{min}) = \left(\frac{1}{2} - p\right)(Q_{max} - Q_{min}),$$

тобто при рівномірному розподілі ймовірностей розмір страхового запасу прямо пропорційний різниці між максимально і мінімально можливою потребою.

4. Встановимо зв'язок між коефіцієнтом ризику p і окремими затримками функціонування системи. Позначимо через S окремі затримки. Мінімізуємо математичне сподівання сумарних затримок, пов'язаних з зберіганням запасу і можливими втратами від дефіциту:

$$L = S\int_{-\infty}^{R} \left[R - (Q - q)\right] f(Q)\,dQ + d\int_{R}^{\infty} \left[(Q - q) - R\right] f(Q)\,dQ. \quad (16.28)$$

Оптимальне значення R знаходимо , розв'язуючи рівняння

$$\frac{\partial L}{\partial R} = S\int_{-\infty}^{R} f(Q)\,dQ - d\int_{R}^{\infty} f(Q)\,dQ = 0, \quad (16.29)$$

звідси

$$\frac{\int_{-\infty}^{R} f(Q)\,dQ}{\int_{R}^{\infty} f(Q)\,dQ} = \frac{d}{S}. \quad (16.30)$$

Відзначимо, що $\int_{R}^{\infty} f(Q)\,dQ$ — це вірогідність того, що $Q - q > R$, але $p(Q - q > R) = p$, отже, $\int_{-\infty}^{R} f(Q)\,dQ = 1 - p$. Таким чином, з рівняння (16.29) отримаємо:

$$\frac{1 - p}{p} = \frac{d}{S}, \text{ звідси } p = \frac{S}{d + S}. \quad (16.31)$$

Знаючи d і S, із формули (16.31) можна визначити припустимий коефіцієнт ризику. Своєю чергою, вираз (16.31) може застосовуватись для оцінки втрат від дефіциту.

Рішення щодо подачі замовлення на поповнення запасів приймається системою управління нижнього рівня і залежить від результатів порівняння поточного стану запасу (Наявний запас + Замовлений обсяг продукції – Заявки на замовлення) з максимальним значенням розумних потреб на інтервалі виконання, що зберігається як точка замовлення на нижньому рівні інформаційної системи. Відлік часу виконання постачання виробу в запаси (час випередження) починається з моменту одержання заявки, що зменшує наявний запас виробу нижче точки замовлення. Час постачання включає власне час опрацювання замовлення, час, необхідний постачальнику на виконання замовлення, і деякий необхідний надалі час на одержання матеріалів і передачу їх для використання.

Максимальний розумний попит є сумою прогнозованого попиту протягом часу попередження і резервного запасу. Резервний запас дорівнює добутку коефіцієнта резервного запасу на стандартне відхилення помилок прогнозування попиту протягом часу виконання замовлення на поповнення запасу, тобто $k\sigma$.

Якщо час попередження відомий з достатнім ступенем точності, то величина стандартного відхилення σ може бути оцінена з урахуванням помилок прогнозування попиту. Якщо час випередження змінюється непередбачено, то для порівняння розподілу цього часу з розподілом помилок прогнозування небезпечно використовувати будь-яку модель. Результуючий розподіл дуже чутливий стосовно припущень щодо статистичної залежності (часто припускають незалежність), що існує між цими розподілами.

Між цими розподілами існує позитивна або негативна кореляція і реальні дані практично не дають основ для припущення про незалежність цих розподілів. Інтервали випередження для продукції різного типу, природно, різні. Перше завдання полягає в тому, щоб знайти причину розходжень і, якщо можливо, усунути її. Багато компаній домовляються з постачальником про конкретні дати постачань, завдяки чому час попередження стає відомим, що, своєю чергою, дозволяє визначити необхідний рівень резервного запасу.

Якщо практично неможливо керувати часом випередження, то варто оцінювати попит безпосередньо протягом цього часу. Відповідно до визначення, момент подачі замовлення на поповнення запасу є початком відліку

часу випередження.

У розглянутих нижче випадках припускаємо, що розподіл помилок прогнозу є нормальним з нульовим середнім значенням. Добрий прогноз має розподіл помилок з нульовим середнім, і якщо інтервал між послідовними перевірками прогнозу набагато менше, ніж одна третина часу випередження, то розподіл близький до нормального. Однак у тих випадках, коли час випередження менший, ніж інтервали між прогнозами, і, крім того, рівні запасів близькі до запитів споживачів, доцільно визначити вид розподілу помилок прогнозу для даного моменту часу, тому що з часом він може істотно змінюватися. Результати використання розглянутих правил прийняття рішень, що продукуються середнім рівнем системи управління запасами, застосовні до усіх випадків, якщо відповідно коректувати розподіл вірогідностей. Для зручності, звичайно, попит представляється як неперервна випадкова змінна.

Показники рівня обслуговування.

Резервний буферний запас, що додається до запасу, який визначається відповідно до прогнозу часу попередження до поповнення запасу, висловлюється через $k\sigma$, де k– коефіцієнт резервного запасу, що, звичайно, обмежений невід'ємними значеннями, визначеними при виборі стратегії. Однак для балансування запасів наприкінці сезону збуту може знадобитися припущення про негативність k.

Вірогідність виникнення дефіциту продукції відразу після одержання партії, що поповнює запас, практично дорівнює нулю. Якщо ж рівень запасу відповідає нижній межі розміру партії безпосередньо перед одержанням нової партії, то виникає дефіцит продукції. Таким чином, показники рівня обслуговування залежать від кількості поповнень запасів у рік і, отже, коефіцієнт резервного запасу також залежить від частоти поповнення запасів.

Криві розподілу.

Кожне правило прийняття рішень з керування запасами містить ту чи іншу керовану змінну, котра може бути використана для регулювання капіталовкладень у резервні запаси у залежності від деякого показника рівня обслуговування споживачів, що має розмірність [1/рік]. Таким показником можуть бути річне число випадків дефіциту запасів продукції, річна кількість (у грошових одиницях) врахованих, але не виконаних замовлень чи річна кількість замовлень споживачів, що не може бути задоволена з наявних запасів. Вибір конкретного значення керованої змінної в одному випадку може привести до високого обсягу капіталовкладень і високого рівня обслуговування споживачів, а в іншому випадку – до зменшення об'єму капіталовкладень і погіршення рівня обслуговування.

Одну з важливих особливостей вищого рівня системи управління запасами візуалізують криві розподілу які ілюструють результати вибору можливих значень керованих змінних.

На рис. 16.4 показані криві, отримані для двох можливих варіантів правил прийняття рішень по керуванню запасами однієї і тієї ж продукції. Для загальності відображені правила будемо називати правилами "А" і "В". Очевидно, що якщо об'єм врахованого, але не задоволеного попиту відбиває рівень обслуговування, то вирішуюче правило "А" виявляється ліпшим, ніж правило "В"; якщо ж число випадків дефіциту запасів – фактор важливіший, то перевага надається правилу "В".

Враховані, але невиконані замовлення (чи економічні втрати) найчастіше є основними показниками рівня обслуговування систем управління остаточними запасами, що задовольняють попит споживачів і в який запаси відносно ізольовані від постачальника. Число випадків недостачі запасів може бути більш придатною оцінкою для систем, у яких фактичний попит може настільки перевищувати прогнозований, що прийнятний резервний запас виявляється недостатнім, але є можливість швидко виготовити необхідний об'єм продукції й одержати матеріали раніше, ніж встановлений термін постачання.

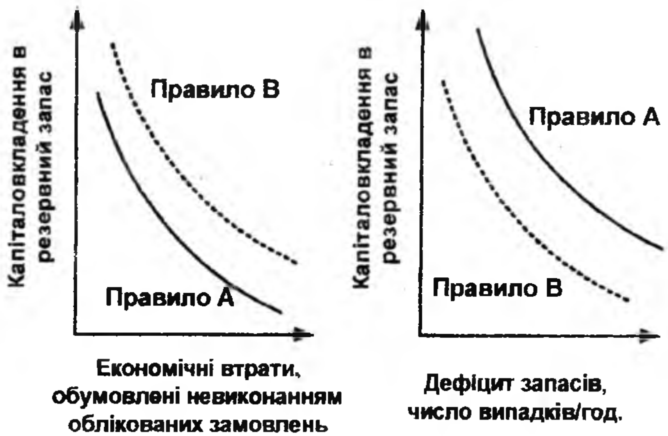


Рис. 16.4. Два можливих варіанти правил прийняття рішень по керуванню запасами однієї і тієї ж продукції

Криві розподілу витрат будуються для деяких фіксованих можливих значень керуючих змінних. Як показано на рис. 16.5, порівняння поточних капіталовкладень у резервний запас і поточного фактичного рівня обслуговування допомагає керуючим органам вчасно прийняти рішення щодо майбутніх обсягу капіталовкладень і рівня обслуговування. Оскільки криві розподілу витрат наочно демонструють фізичну сутність різних варіантів рішень, то саме вони можуть бути використані для подолання суперечностей між органами управління, відповідальними за фінанси, і органами управління, відповідальними за збут.

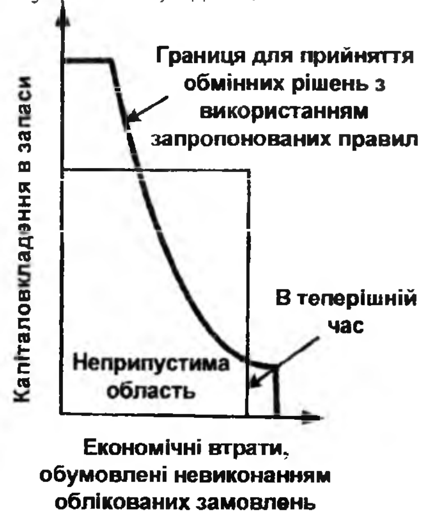


Рис. 16.5. Крива розподілу витрат

Визначення необхідного об'єму капіталовкладень для ефективності збуту є прерогативою керівника, що йде на ризик, і цей об'єм не має "ідеального" значення, котре можна було б об'єктивно оцінити за допомогою методів дослідження операцій.

Оцінка ефективності вирішуючих правил.

Оскільки кожне вирішуюче правило при подальшому уточненні постановки задачі може пов'язуватися з досягненням множини цілей, то при оцінці можливих правил варто враховувати побічні ефекти, такі, як середній запас у системі загалом, число рішень з розподілу ресурсів протягом року, очікуване число випадків дефіциту запасів і величина цього дефіциту. Може виявитися, наприклад, що таке просте правило, як виділення рівного числа одиниць запасу для кожного процесу, буде мати зовсім ясний фізичний зміст і давати непогані результати відразу у всіх аспектах, у той час як якесь, більш тонке правило дасть можливість детально враховувати якийсь один фактор, але буде приводити до небажаного побічного явища. Такі "невигідні" правила можуть бути виключені за допомогою імітаційної моделі системи, якщо моделюючий алгоритм буде налаштований на оцінювання відразу декількох аспектів функціонування системи при використанні того чи іншого вирішаючого правила.

16.4. СТРУКТУРА СИСТЕМ УПРАВЛІННЯ ЗАПАСАМИ

Комбінуючи елементарні операції управління та вирішуючі правила, можна створити моделі систем управління запасами різної структури, і шляхом моделювання з використанням реальної інформації за минулі періоди обрати найвідповідніший варіант для реалізації в конкретних умовах.

Досить глибоке розуміння теоретичних основ розробки правил ухвалення рішення щодо поповнення запасів може бути досягнуте при дослідженні моделей системи управління запасами з нульовим часом постачання чи замовленнями з детермінованим попитом. При використанні систем управління запасами, як правило, доводиться мати справу з випадковим попитом протягом часу постачань замовлень. Більшість теоретичних результатів отримано в припущенні, що дане вирішуюче правило буде застосовуватися багаторазово, однак у дійсності реальні інтервали постачань матеріалів, затрати і розподіл попиту змінюються в часі. Таким чином, прийняте в даний момент рішення щодо поповнення запасів є найкращим лише для цього моменту.

Розглянемо деякі можливі методи рішення задач нижнього рівня системи, тобто задач розміщення чергового замовлення на поповнення запасу деякої продукції і визначення розміру цього замовлення.

Подача незалежних замовлень на поповнення запасів.

У тих випадках, коли можна замовити будь-яку продукцію в будь-який час, незалежно від стан справ із продукцією інших типів, можна використовувати наступні моделі для визначення обсягу і термінів замовлення.

Модель точки замовлення і розміру партії. Точка замовлення чи рівень запасу, при якому подається замовлення на поповнення, обчислюється для кожного типу продукції на основі обліку поточної прогнозованої потреби в ній на інтервалі часу постачання поповнення і деякого резервного запасу.

Останній дорівнює добутку стандартного відхилення розподілу помилок прогнозу протягом часу постачання на коефіцієнт резервного запасу. Коефіцієнт страхового запасу може бути обчислений у результаті застосування вирішуючих правил чи їхньої модифікації.

Система першого рівня відслідковує стан запасу, що дорівнює: Рівень наявного запасу + Об'єм замовлених постачань – Об'єм врахованих, але не виконаних заявок чи Об'єм продукції по інших зобов'язаннях. У тих випадках, коли рівень наявного запасу знижується в результаті задоволення попиту до чи нижче точки замовлення, подається замовлення на поповнення запасу. Об'єм замовлення обчислюється на підставі придатного вирішуючого правила, що розробляється на другому рівні системи управління запасами. Отриманий об'єм замовлення може бути округлений до величини, зручної з погляду одержання постачань чи кратного числа упакувань без істотного скорочення економічної вигоди. Вирішуючі правила другого рівня звичайно перевіряються через певні інтервали часу, наприклад, один раз на місяць, і результати фіксуються на нижньому рівні системи керування запасами.

Такий найпростіший для розуміння і застосування спосіб досить часто виявляється й ефективним. Наприклад, у виробництві фарб і ліків розмір цистерн фізично обмежує об'єм замовлення, так що має сенс лише визначений заздалегідь об'єм партії, що замовляється.

Моделі типу максимум/мінімум. Мінімальний рівень запасів обчислюється аналогічно тому, як обчислюється точка замовлення в попередній моделі. Максимальний рівень дорівнює сумі мінімального рівня й об'єму замовлення, що одержується за допомогою відповідного вирішаючого правила визначення розміру партії. Значення цих рівнів обчислюються через однакові проміжки часу на другому рівні системи, і результати фіксуються на нижньому рівні.

Система нижнього рівня неперервно стежить за рівнем наявного запасу, і якщо він знижується до величини s, то замовляється обсяг продукції, рівний різниці між максимальним і наявним рівнями запасу. Якщо є можливість зробити нове замовлення після задоволення кожної одиниці попиту (у результаті чого наявний запас знижується точно до мінімального рівня), то система функціонує аналогічно до попередньої. Однак, якщо інформація про наявний запас передається через великі інтервали часу чи якщо існує одиночний попит на великий об'єм продукції, то існує можливість того, що до моменту розпізнавання необхідності поповнення запасу рівень наявного запасу знизиться нижче точки замовлення і система усуне цей дефіцит. Цей спосіб є кращим у порівнянні з моделлю "точка замовлення і розмір партії" при наявності істотного дефіциту протягом часу виконання замовлення на поповнення запасів. У цьому випадку об'єм партії, що замовляється, також може бути визначений на підставі припущення про його кратність числу упакувань, числу постачань чи кількості одиниць устаткування.

Модель з фіксованим інтервалом. Якщо постачальник приймає замовлення лише протягом певного інтервалу часу, то прийом замовлень між цими інтервалами відсутній. Наприклад, машинобудівний завод може приймати замовлення тільки до 24 числа кожного місяця, оскільки до цього часу закінчується формування плану виробництва на черговий місяць. Інший приклад – корабель-постачальник, що постачає кораблі деякої флотилії і прибуває до них один раз на 10 днів. В обох зазначених прикладах споживач змушений чекати, поки постачальник буде готовий до прийому нового замовлення. У цьому випадку точка замовлення чи мінімальний рівень запасу обчислюється на основі прогнозу потреб протягом часу постачання поповнюючого запасу і додаткового інтервалу чекання можливості подачі замовлення з врахуванням, як і раніше, резервного запасу, що компенсує помилки прогнозу потреб протягом усього розглянутого інтервалу.

Якщо наявний запас у момент перевірки менший, ніж нова точка замовлення, то поповнююче замовлення виконується або в заздалегідь встановленому об'ємі, або в об'ємі запасу, визначеного різницею між максимальним і наявним рівнями. Зазначимо, що мінімальний об'єм продукції, що замовляється, повиннен бути достатнім принаймні для того, щоб в інтервалі чекання можливості подачі замовлення на поповнення запасу він не вичерпувався цілком. Ця схема використовується також у ряді випадків як апроксимація об'єднаних замовлень, коли необхідне регулювання запасів відразу декількох видів. У загальному випадку значення точки замовлення чи мінімального рівня запасу й обсяг замовлення чи максимальний рівень запасу повинні бути перераховані кожен раз при перегляді прогнозу. У деяких системах точка замовлення переглядається з кожним новим прогнозом, але обсяг замовлення чи максимальний рівень запасу обчислюються лише безпосередньо перед розміщенням замовлення.

Нерівномірність попиту.

Попит на більшість видів запасів (зокрема, запасних частин для машин) може мати нерівномірний характер, тобто спостерігаються періоди різної

тривалості, протягом яких попит практично дорівнює нулю, незначний чи різко зростає. Якщо різке зростання попиту (яке може бути наслідком спеціальних проектів чи контрактів) може бути передбачено, то в цьому випадку можливо спланувати постачання продукції, у іншому випадку застосування класичних методів управління запасами (зокрема, прогнозування) виявляється дуже складним. У цій ситуації успішно використовується наступна система.

Запас поповнюється відповідно до максимінної системи. При визначенні розміру замовлення, що використовується для визначення максимального рівня запасу, враховується взаємозв'язок між розмірами партії і резервних запасів, при цьому розмір партії повинний бути не менше, ніж стандартне відхилення σ. Точка замовлення повинна перевищувати середній мінімум на величину очікуваного дефіциту, чи на середню величину різниці точки замовлення і наявного запасу в моменти подачі замовлень на поповнення запасу. Оцінка величини цього очікуваного дефіциту становить:

$M\left[d^2\right] \big/ 2M\left[d\right]$, де d – число одиниць продукції в окремій заявці чи в партії, що відправляється на склад. Відзначимо, що якщо всі заявки однакового об'єму (рівномірний попит), то очікуваний дефіцит скорочується до величини, рівній половині очікуваного об'єму попиту. Для випадку рівномірного попиту цей дефіцит значно більший.

Таким чином, у випадку звичайної продукції система управління з нерівномірним попитом перетворюється в класичну мінімаксну систему.

ПРИКЛАДИ

Приклад 16.1. Стохастична модель управління запасами.

Підприємство купує агрегат з запасними блоками до нього. Вартість одного блоку становить 5 грошових одиниць. У випадку виходу з ладу агрегату внаслідок поломки блоку, який відсутній в запасі, простій агрегата та раптове замовлення нового блока до нього становитимуть 100 грошових одиниць. Емпіричний розподіл агрегатів за числом блоків, що потребуватимуть заміну, встановлений та відображений в таблиці.

Число замінених блоків	0	1	2	3	4	5	6
Статистична вірогідність (частка агрегатів) $p(r)$, для яких потрібно було замінити r блоків	0,90	0,05	0,02	0,01	0,01	0,01	0,00

Розв'язання.

За умовою $c_2 = 5$, $c_3 = 100$. Обчислимо щільність збитків внаслідок незадоволеного попиту $\eta = \frac{c_3}{c_2 + c_3} = \frac{100}{100+5} = 0,952$. На основі співвідношення для інтегральної функції розподілу $F(s) = p(r < s)$ розрахуємо значення функції розподілу попиту та зведемо їх в таблицю:

s	0	1	2	3	4	5	6	>6
r	0	1	2	3	4	5	6	>6
F(s)	0,00	0,00	0.90	0,95	0,97	0,98	0,99	1,00

Оскільки розподіл дискретний, для визначення оптимального рівня запасу використаємо співвідношення $F(s^*) < \eta < F(s^*+1)$, $F(3) < 0,952 < F(4)$, і, таким чином, оптимальний рівень запасу становитиме $s^* = 3$.

Приклад 16.2. Стохастична модель управління запасами.

Розв'язати задачу з прикладу 16.1 за умови, що попит є випадковим, але розподілений за неперервним експоненційним законом з інтегральною функцією розподілу $F(r) = 1 - e^{-\lambda r}$ зі значенням $\lambda = 0,8$.

Розв'язання.

Оптимальну кількість запасних блоків визначимо зі співвідношення для неперервного розподілу $F(s^*) = \eta = 1 - e^{-\lambda s^*}$, звідки:

$$s^* = -\frac{1}{\lambda}\ln(1-\eta) = -\frac{1}{0,8}\ln(1-0,952) = \frac{3,036}{0,98} = 3,795 \approx 4.$$

Приклад 16.3. Стохастична модель з неперервною витратою запасу.

Вироби, що є в коморі, витрачаються рівномірно протягом місяця. Затрати на зберігання одного виробу становлять 5 грошових одиниць, а штраф за дефіцит одного виробу становить 100 грошових одиниць.

На основі вивчення попиту був побудований розподіл числа виробів, що споживаються за місяць, наведений в таблиці:

Попит *r*	0	1	2	3	4	5	>=6
Статистична вірогідність *p(r)*	0,1	0,2	0,2	0,3	0,1	0,1	0,0

Необхідно визначити оптимальний місячний запас в коморі.

Розв'язання.

Для визначення оптимального запасу розрахуємо значення функції $L(r)$ і зведемо результати в таблицю:

s	r	$p(r)$	$\frac{p(r)}{r}$	$\sum\limits_{r=s}^{\infty}\frac{p(r)}{r}$	$\left(s-\frac{1}{2}\right)\sum\limits_{r=s}^{\infty}\frac{p(r)}{r}$	$F(s)$	$L(s)$
0	0	0,1	-	-	-	0,0	-
1	1	0,2	0,200	0,445	0,2225	0,1	0,3225
2	2	0,2	0,100	0,245	0,3675	0,3	0,6675
3	3	0,3	0,100	0,145	0,3625	0,5	0,8625
4	4	0,1	0,025	0,045	0,1575	0,8	0,9575
5	5	0,1	0,020	0,020	0.0900	0,9	0,9900
>=6	>=6	0.0	0,000	0,000	0,0000	1,0	1.0000

Оптимальний рівень запасу відповідає $s^* = 3$, тому що $L(3) < 0,952 < L(4)$.

Приклад 16.4. Стохастична модель з фіксованим часом затримки поставок.

Товар, що швидко псується, замовляється крамницею щоденно, а надходить 7 днів після замовлення (тобто замовлення робиться на тиждень наперед). В момент чергового замовлення запас товару склав в вартісному еквіваленті 10 грошових одиниць. Товар, що продається в день виготовлення, дає дохід 0.95 грошової одиниці, а не проданий в цей день товар може бути реалізованим зі збитком в 0.1 грошову одиницю.

В результаті опрацювання дослідних даних отриманий наступний розподіл попиту на товар, наведений в таблиці:

r	0	10	20	30	40	50	60	70	80	90	100
p(r)	0,00	0.00	0,01	0,02	0.05	0,06	0.08	0,09	0.12	0,13	0,10
r	110	120	130	140	150	160	170	180	190	200	>200
p(r)	0,08	0,05	0.02	0.02	0,02	0,02	0,01	0,01	0,01	0.00	0,00

Необхідно визначити оптимальний об'єм замовленого товару q_7 на сьомий день після замовлення.

Розв'язання.

Щільність збитків, зумовлена дефіцитом товару становить:

$$\eta = \frac{c_3}{c_2 + c_3} = \frac{0,95}{0,10 + 0,95} = 0.905.$$

Враховуючи, що $F(s) = p(r < s)$, розрахуємо значення функції розподілу попиту і зведемо отримані результати в таблицю:

s	0	10	20	30	40	50	60	70	80	90
r	0	10	20	30	40	50	60	70	80	90
F(s)	0,00	0,00	0.01	0,03	0,08	0,14	0,22	0,31	0.43	0,66
s	100	110	120	130	140	150	160	170	180	>=190
r	100	110	120	130	140	150	160	170	180	>=190
F(s)	0,76	0,84	0,89	0,91	0,93	0,95	0,97	0,98	0,99	1,00

Умова $F(s^*)<\eta<F(s^*+1)$ задовільняється у випадку $F(120)<0,905<F(130)$, тобто $s^*=120$. Таким чином, оптимальний запас товару за 7 днів повинен бути на суму 120 грошових одиниць, звідки отримуємо оптимальний об'єм замовленого товару на сьомий день, що становить:

$$q_7 = s^* - \left(s_0 + \sum_{i=1}^{6} q_i\right) = 120 - \left(10 + (10+20+10+10+20+10)\right) = 30 \text{ грош од.}$$

РЕЗЮМЕ

16.1. В стохастичних моделях управління запасами попит є випадковою величиною, що описується законами теорії вірогідностей. Врахування випадковості суттєво ускладнює аналіз та отримання рішень на таких моделях, а тому аналітичні рішення можливо отримати лише для найпростіших випадків. Якщо ж система має складну структуру з багатьма видами продукції та коморами, то єдиним способом її аналізу виявляється імітаційне моделювання.

16.2. При необхідності поповнити запаси продукції розв'язується задача визначення розміру замовлення (чи розміру партії) продукції. Відновлення рівня запасів може здійснюватися або шляхом виконання замовлення на виготовлення деякого фіксованого обсягу продукції, зареєстрованого системою нижнього рівня, або шляхом виконання замовлення на виготовлення певного об'єму продукції. Вирішуючі правила розробляються, зазвичай, на другому рівні трирівневої системи управління запасами для визначення економічно обґрунтованого оптимального розміру замовлення. Ці правила використовуються при визначенні запланованого чи максимального рівня запасу для конкретного типу чи типів продукції. Різні підходи до розв'язання задачі визначення економічного розміру замовлення (партії продукції) базуються на тих чи інших можливих припущеннях щодо таких показників, як спосіб постачання замовленої продукції, інтенсивність реалізації продукції, витрати на освоєння виробництва нової продукції, собівартість одиниці

продукції і поточні витрати на збереження запасів. Спочатку для кожного з перерахованих факторів робляться найпростіші припущення, а потім розглядаються умови, за яких можливі припущення виявляються кращими, ніж первісне. Оскільки вважається, що припущення щодо одного з факторів не пов'язані з припущеннями щодо інших факторів, то можливі сотні комбінацій факторів, що приводять до різних вирішуючих правил. З такої множини вирішуючих правил можна обрати найприйнятніше, однак доцільність того чи іншого вибору визначається з урахуванням конкретної ситуації, у якій ці правила будуть реалізовані.

16.3. Матеріальні запаси служать для того, щоб згладити безпосередню залежність між динамікою виробництва продукції та її споживанням. Наявність запасів дозволяє налагодити виробництво продукції оптимальними партіями, а також визначити оптимальні партії поставок по кожному продукту. Матеріальний запас – це демпфер, що згладжує залежність споживача від можливих коливань випуску продукції, послаблює залежність виробничого процесу постачальника від нерівномірностей споживання цієї продукції споживачами; запаси згладжують залежності окремих цехів, робочих місць і т.д. один від одного. Завдяки створенню запасів вирівнюються та здешевлюються виробничі процеси. Запаси роблять систему стійкішою, вони створюють необхідні передумови для забезпечення неперервності виробничого процесу. Оптимальний рівень запасів складається з економічно оптимальної партії поставки плюс деякий страховий запас. Необхідність страхового запасу диктується випадковими явищами, які органічно наявні в будь-якій системі матеріально-технічного постачання. Час постачання включає власне час опрацювання замовлення, час, необхідний постачальнику на виконання замовлення, і деякий необхідний надалі час на одержання матеріалів і передачу їх для використання. Максимальний розумний попит є сумою прогнозованого попиту протягом часу попередження і резервного запасу. Резервний запас дорівнює добутку коефіцієнта резервного запасу на стандартне відхилення помилок прогнозування попиту протягом часу виконання замовлення на поповнення запасу. Оскільки кожне вирішуюче правило при подальшому уточненні постановки задачі може пов'язуватися з досягненням множини цілей, то при оцінці можливих правил варто враховувати побічні ефекти, такі, як середній запас у системі загалом, число рішень з розподілу ресурсів протягом року, очікуване число випадків дефіциту запасів і величина цього дефіциту.

16.4. Комбінуючи елементарні операції управління та вирішуючі *правила, можна створити моделі систем управління запасами різної структури, і шляхом моделювання з використанням реальної інформації за минулі періоди обрати найвідповідніший варіант для реалізації в конкретних умовах.*

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 16.1.

Цукерня торгує тортами власного виготовлення. Кожен кілограм торта приносить 20 грошових одиниць прибутку. Наступного дня торти продаються зі знижкою в 2 грошові одиниці. В результаті опрацювання статистичних даних отриманий наступний розподіл попиту на торти, заданий таблицею:

Попит *r*	0	10	20	30	40	50	>=60
Статистична вірогідність *p(r)*	0.1	0.2	0.2	0.3	0.1	0.1	0,0

Визначити, який денний виробіток тортів буде оптимальним.

Завдання 16.2.

Розв'яжіть завдання 16.1 за умови, що попит на торти є випадковою величиною, що розподілена за показниковим законом з параметром $\lambda = 0,9$.

Завдання 16.2.

Комора поповнюється щомісяця певними виробами. Протягом перших п'яти місяців року об'єми поповнення склали 10, 20, 20, 20, та 30 виробів. За статистичними даними отриманий розподіл попиту на товар, представлений наступною таблицею:

r	0	10	20	30	40	50	60	70	80	90	100
p(r)	0,00	0,00	0,01	0,02	0,05	0,07	0,07	0,09	0,10	0,13	0,12
r	110	120	130	140	150	160	170	180	190	200	>200
p(r)	0,06	0,06	0,03	0.02	0,02	0,02	0,01	0,01	0,01	0,00	0,00

Зсув в часі між замовленням на поповнення та доставкою становить 6 місяців. Витрати від надлишку в розрахунку на один виріб становлять 10 грошових одиниць, а від недостачі – 120 грошових одиниць. Необхідно визначити оптимальне поповнення комори на 6-й місяць

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Які особливості стохастичних моделей управління запасами?
2. Який вигляд має критерій якості для стохастичних моделей?
3. Яким чином визначається оптимальний рівень запасу для найпростіших стохастичних моделей управління запасами?
4. Які фактори впливають на формування вирішуючих правил?
5. Які міркування використовуються при визначенні рівня резервного запасу?
6. Які функції виконує матеріальний запас?
7. Яким чином оцінюється ефективність вирішуючих правил?
8. Розкрийте сенс функціонування системи управління запасами на основі моделі точки замовлення та розміру партії.
9. В чому суть моделі максимум/мінімум?
10. Розкрийте особливості функціонування моделі з фіксованим інтервалом.

РОЗДІЛ 8

ДИНАМІЧНЕ ПРОГРАМУВАННЯ

ТЕМА 17. ЗАДАЧІ ДИНАМІЧНОГО ПРОГРАМУВАННЯ

В задачах динамічного програмування процес залежить від часу, і тому розв'язування зводиться до багатоетапного або багатостадійного процесу прийняття рішень. Розв'язання задач динамічного програмування дозволяє виробити оптимальну стратегію дій, в той час як статичні задачі дозволяють отримати розв'язок, оптимальний з точки зору умов, що склалися, тобто тактичний. За допомогою методу динамічного програмування здійснюється оптимізація багатоетапних процесів, виходячи з інтересів системи загалом, а не кожної її стадії як окремого елементу, тому розв'язок на кожному кроці повинен відповідати вимогам оптимізації процесу загалом. Для того, щоб для розв'язування задачі можна було застосувати метод динамічного програмування, необхідно наступне: функція мети була адитивною (мультиплікативною); в процесі руху системи не було післядії. Метод динамічного програмування ґрунтується на принципі оптимальності Р. Белмана, що визначає порядок покрокового розв'язування задачі, яка може бути піддана декомпозиції, за допомогою рекурентних обчислювальних процедур. Згідно з цим принципом, оптимальне керування має наступну основоположну властивість, якими не були би початковий стан та прийняте початкове рішення, наступні рішення повинні утворювати оптимальне керування відносно стану, що виник в результаті попереднього рішення.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ:

знати: ⇨особливості багатоетапних керованих процесів та можливості застосування до них динамічного програмування; ⇨ загальну постановку задачі динамічного програмування; ⇨формулювання принципу оптимальності Р. Белмана та послідовність його застосування;

вміти: ⇨будувати формальні моделі багатокрокових керованих процесів; ⇨застосовувати принцип оптимальності з метою отримання рекурентних співвідношень;⇨розв'язувати динамічні багатокрокові задачі за допомогою методу динамічного програмування.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ динамічне програмування
- ☑ адитивність
- ☑ оптимальна стратегія
- ☑ керований процес
- ☑ мультиплікативність
- ☑ проблема розмірності
- ☑ декомпозиція процесу
- ☑ критерій якості
- ☑ вектор стану процесу
- ☑ стан процесу
- ☑ функціональні рівняння
- ☑ траєкторія руху системи
- ☑ принцип оптимальності
- ☑ однопродуктова динамічна модель
- ☑ змінні стану
- ☑ післядія
- ☑ рекурентні співвідношення
- ☑ керовані змінні

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

17.1. Поняття динамічного програмування та загальна постановка задачі ДП.
17.2. Принцип оптимальності.
17.3. Метод функціональних рівнянь.
17.4. Динамічні моделі управління запасами.

17.1. ПОНЯТТЯ ДИНАМІЧНОГО ПРОГРАМУВАННЯ ТА ЗАГАЛЬНА ПОСТАНОВКА ЗАДАЧІ ДП

В задачах лінійного та нелінійного програмування розглядаються статичні процеси, і оптимальний розв'язок знаходиться лише на один етап планування, одномоментно. Такі задачі називаються однокроковими.

В задачах динамічного програмування процес залежить від часу, і тому розв'язування зводиться до багатоетапного або багатостадійного процесу прийняття рішень. Розв'язання задач динамічного програмування дозволяє виробити оптимальну стратегію дій, в той час як статичні задачі дозволяють отримати розв'язок, оптимальний з точки зору умов, що склалися, тобто тактичний. Однак ця межа не є чітко визначеною, тому що широкі класи динамічних задач в принципі можна звести до однокрокових, але цей факт має скоріше теоретичний інтерес, тому що розв'язати такі зведені задачі внаслідок надзвичайно великого зростання розмірності просто неможливо. З іншого боку, деякі статичні задачі можна представити у вигляді багатокрокових і розв'язувати методом динамічного програмування. Таким чином, **динамічне програмування (ДП) є математичним апаратом, що дозволяє здійснити оптимальне планування багатокрокових керованих процесів та процесів, які залежать від часу**.

Процес називається керованим, якщо наявна можливість впливу на перебіг його розвитку. **Керуванням** називатимемо сукупність розв'язків, що приймаються на кожному з етапів з метою впливу на хід процесу.

Випуск продукції підприємством – це керований процес, що визначається змінами попиту ринку, складу обладнання, величиною попереднього прибутку, тенденціями розвитку, відсотком коштів, що спрямовуються у виробництво та наукову діяльність і ін. Сукупність рішень, що приймаються на початку кожного з відтинків певного періоду (тижнів місяця, місяців року та ін.) з питань забезпечення сировиною, обладнанням, розмірами фінансування, є керуванням. Планування на місяць здійснюється з врахуванням того, що на кінець року повинні бути досягнуті певні цілі.

Багатоетапні керовані процеси мають наступні спільні риси:

1. Процес може бути підданий декомпозиції, тобто розбитий на складові елементи – кроки або етапи. Якщо процес розглядається в часі, то природним є розбиття за періодами часу. Виробничі процеси можуть бути розбиті за стадіями відповідно до їх технологічних особливостей, ресурси можуть бути розподілені між споживачами і т. ін.

2. Кожний етап характеризується станом, який визначається значеннями факторів, або змінних, кількість яких для кожного з етапів може бути різною.

3. З загального числа змінних виділяються керовані, тобто ті, значення яких можна спрямовано змінювати і цими змінами впливати на стан процесу, **та змінні стану.**

4. На кожному кроці існує залежність між керованими змінними, змінними стану та функцією мети, яка представляється за допомогою рівняння, системи рівнянь або таблично. За допомогою динамічного програмування для кожного кроку встановлюються такі значення керованих змінних, які забезпечують екстремальне значення функції мети

для процесу загалом.

5. Критерій оптимальності повинен бути адитивним (або мультиплікативним), тобто значення функції мети для всього керованого процесу повинно складатися з елементарних значень цієї функції, отриманих для кожного окремого етапу.

Таким чином, ***за допомогою методу динамічного програмування здійснюється оптимізація багатоетапних процесів, виходячи з інтересів системи загалом, а не кожної її стадії як окремого елементу, тому розв'язок на кожному кроці повинен відповідати вимогам оптимізації процесу загалом.***

В загальному вигляді постановка задачі динамічного програмування формулюється наступним чином.

Нехай деяка фізично керована система S знаходиться в початковому стані s_0. Під дією керування u_i на кожному i-му кроці вона переходить через послідовність проміжних станів в остаточний стан s_n:

$$s_0 \Rightarrow s_1 \Rightarrow s_2 \Rightarrow \ldots \Rightarrow s_n .$$

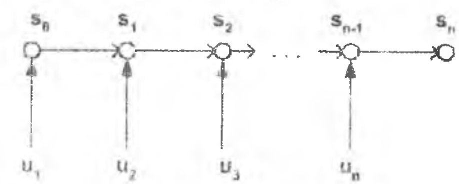


Рис. 17.1. Траєкторія руху динамічної системи

При відомому керуванні на i-му кроці стан s_i визначається як

$$s_i = v(s_{i-1}, u_i). \qquad (17.1)$$

Критерій якості

$$Q(s_0, u_1, u_2, \ldots, u_n) \Rightarrow Max , \qquad (17.2)$$

або, позначивши $u = (u_1, u_2, \ldots, u_n)$, $Q(s_0, u) \Rightarrow Max$. Якщо в результаті здійснення i-го кроку на етапі забезпечується певний виграш (значення функції мети на етапі) $W_i(s_{i-1}, u_i)$, що залежить від початкового стану системи на цьому кроці s_{i-1} та обраного управління з числа можливих на цьому кроці $u_i \in U_i$ то загальний виграш (значення функції мети для процесу загалом), згідно з вимогою адитивності, становитиме:

$$Q = \sum_{i=1}^{n} W_i(s_{i-1}, u_i) . \qquad (17.3)$$

17.2. ПРИНЦИП ОПТИМАЛЬНОСТІ

Виникає природне запитання: яким чином потрібно діяти, щоб при побудові стратегії врахувати на кожному кроці вимоги процесу загалом і які вимоги висуваються до постановки задачі багатокрокового керування?

Задача для того, щоб можна було для її розв`язування застосувати метод динамічного програмування, повинна відповідати двом наступним вимогам:

- ☑ функція мети повинна бути адитивною (мультиплікативною);
- ☑ в процесі руху системи повинна бути відсутня післядія: стан s_i, в який перейшла система, залежить лише від значення попереднього стану s_{i-1} та від управління u_i і не залежить від того, яким шляхом система потрапила в стан s_{i-1}.

Оптимальною називається така стратегія $u^*=(u_1^*,u_2^*,...,u_n^*)$ (послідовність управлінь на кожному кроці процесу), що переводить систему з стану s_0 в стан s_n за n кроків, і для котрої $u^*=\arg \underset{u\in U}{Max}\ Q(s_0,u)$.

Метод динамічного програмування ґрунтується на принципі оптимальності Р. Белмана, що визначає порядок покрокового розв`язування задачі, яка може бути піддана декомпозиції, за допомогою рекурентних обчислювальних процедур.

Принцип оптимальності формулюється наступним чином .

Оптимальне керування має наступну основоположну властивість: яким не були би початковий стан та прийняте початкове рішення, наступні рішення повинні утворювати оптимальне керування відносно стану, що виник в результаті попереднього рішення. Іншими словами, процес $(s_0,s_1,s_2,...,s_n)$ оптимальний тоді і лише тоді, коли оптимальний кожен з процесів $(s_i,s_{i+1},s_{i+2},...,s_n)$, $i=1,2,...,n-1$.

Таким чином, динамічне програмування є поетапним плануванням багатокрокового процесу з оптимізацією на один крок , яке враховує на кожному кроці розвиток процесу загалом, тобто при прийнятті рішення враховується майбутнє. Безпосередньо це зробити неможливо, і власне принцип оптимальності дає нам можливість це зробити. В кожному процесі є останній, n-й крок, прийняття рішення на якому не залежить від майбутнього. Здійснивши планування цього кроку, до нього можна приєднати $(n-1)$-й крок, до яких в свою чергу $(n-2)$-й, приходячи таким чином до початкового стану s_0. Процес динамічного програмування розгортається з кінця до початку – від останнього стану до початкового.

Для того ж, щоб спланувати n-й крок, необхідно знати стан системи на $(n-1)$-му кроці. Якщо стан системи на $(n-1)$-му кроці невідомий (а так є завжди), то, виходячи з характеристик процесу, роблять припущення про можливі стани системи на цьому кроці. Для кожного припущення визначаємо оптимальне керування на останньому, n-му кроці. Таке оптимальне керування називається умовно-оптимальним (за умови певного значення стану на попередньому кроці). Аналогічно діємо на $(n-2)$-му кроці, але умовно оптимальні керування обираємо, враховуючи вже обрані умовно-оптимальні керування на $(n-1)$-й крок.

Для першого кроку (на відміну від інших) припущень про можливий стан системи не робимо – стан s_0 відомий, і знаходимо оптимальне керування з врахуванням всіх умовно-оптимальних керувань, знайдених для другого кроку. І, нарешті, проходячи від s_0 до s_n в прямому напрямку виконання процесу, отримуємо оптимальне керування для процесу загалом.

У динамічному програмуванні існує *проблема розмірності* (або *прокляття розмірності* за словами Р.Белмана), суть якої полягає в наступному.

В моделях динамічного програмування стани можуть бути описані за допомогою $(n\geq 1)$ змінних, які утворюють вектор стану. Збільшення кількості змінних викликає зростання числа можливих варіантів розв'язування, асоційованих з кожним з етапів. Це особливо помітно при проведенні обчислень з використанням таблиць. Оскільки при розв'язуванні задач динамічного програмування зазвичай використовуються відповідні комп'ютерні програми, зростання кількості змінних стану не лише збільшує час обчислень, але й може викликати переповнення пам'яті комп'ютера.

Якщо ж говорити про складність задачі, то вона належить до класу NP повних задач, оскільки її складність експоненційно залежить від складності задачі, $S = ae^n$, де n – розмірність задачі. З іншого боку, кількість варіантів є значно меншою, ніж у випадку повного перебору.

17.3. МЕТОД ФУНКЦІОНАЛЬНИХ РІВНЯНЬ

При застосуванні методу функціональних рівнянь знаходження оптимального розв'язку (стратегії) багатокрокового процесу отримується шляхом розв'язування функціональних рівнянь.

Проілюструємо основні ідеї методу функціональних рівнянь на прикладі простого багатоетапного процесу розподілу.

Нехай наявна деяка кількість засобів x_0, які необхідно вкласти в розвиток 2-х неоднорідних підприємств. Якщо в перше підприємство вкласти y_0 засобів, то для 2-го залишиться $(x_0 - y_0)$ засобів.

При цьому будуть отримані відповідні доходи $g(y_0)$ та $h(x_0 - y_0)$.

Необхідно розподілити засоби x (тобто обрати величину y) таким чином, щоб максимізувати загальний дохід від вкладення x одиниць засобів, тобто $Q_1(x_0, y_0) = g(y_0) + h(x_0 - y_0) \Rightarrow Max,\ y_0 \in [0, x_0]$. (17.4)

Нехай g та h неперервні для всіх скінчених $x \geq 0$. Тоді максимальне значення Q існує завжди та визначає максимальне значення доходу в одноетапному процесі.

Розглянемо двохетапний процес.

Припустимо, що за рахунок витрат, необхідних для отримання доходу $g(y)$, кількість первісних засобів y зменшиться до значення $\varphi(y_0) = ay$, $0 \leq a \leq 1$. Аналогічно для отримання доходу $h(x - y)$ кількість первісних засобів $(x_0 - y_0)$ зменшиться до значення $h(x_0 - y_0)$, тобто

$\psi(x_0 - y_0) = b(x_0 - y_0)$, $0 \leq b \leq 1$.

Таким чином, після завершення 1-го етапу залишок засобів становитиме

$$x_1 = ay_0 + b(x_0 - y_0). \tag{17.5}$$

Повторимо процес розподілу сумарного залишку на другому кроці:

$$x_1 = y_1 + (x_1 - y_1),\ 0 \leq y_1 \leq x_1. \tag{17.6}$$

При цьому дохід на другому кроці становитиме $g(y_1) + h(x_1 - y_1)$. Таким чином повний дохід за два кроки становитиме

$$Q(x_0, y_0, y_1) = g(y_0) + h(x_0 - y_0) + g(y_1) + h(x_1 - y_1) \Rightarrow Max,$$

$$x_1 = ay_0 + b(x_0 - y_0),\ 0 \leq y_0 \leq x_0, \tag{17.7}$$

$$x_1 = \varphi(y_0) + \psi(x_0 - y_0),\ 0 \leq y_1 \leq x_1.$$

Розповсюджуючи отримані результати на N кроків, отримаємо:

$$Q(x_0, y_0, \ldots, y_{N-1}) = g(y_0) + h(x_0 - y_0) + g(y_1) + \ldots$$

$$+ g(y_{N-1}) + h(x_{N-1} - y_{N-1}) = \sum_{i=0}^{N-1}\big(g(y_i) + h(x_i - y_i)\big) \Rightarrow Max, \tag{17.8}$$

$$x_i = \varphi(y_{i-1}) + \psi(x_{i-1} - y_{i-1}),\ 0 \leq y_i \leq x_i,\ i = \overline{0, N-1}.$$

Співставивши з загальною постановкою задачі, отримаємо:

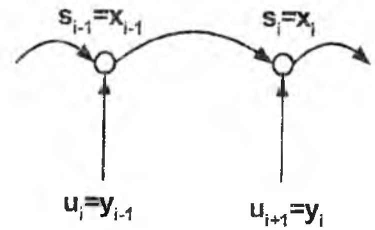


Рис. 17.2. Співставлення розподільчої задачі з загальною задачею динамічного програмування

$$s_i = x_i = \varphi(y_{i-1}) + \psi(x_{i-1} - y_{i-1}) = \varphi(u_i) + \psi(s_{i-1} - u_i),$$

$$Q_i = g(y_{i-1}) + h(x_{i-1} - y_{i-1}) = g(u_i) + h(s_{i-1} - u_i). \tag{17.9}$$

Застосуємо принцип оптимальності Белмана для розв'язування цієї задачі.

На N-му кроці розподілу підлягає значення:

$$x_{N-1} = y_{N-1} + (x_{N-1} - y_{N-1}), \tag{17.10}$$

де y_{N-1} визначається зі співвідношення (оскільки немає майбутнього !!!)

$$f_N(x_{N-1}) = \underset{0 \leq y_{N-1} \leq x_{N-1}}{Max}\{g(y_{N-1}) + h(x_{N-1} - y_{N-1})\}$$

$$= \underset{0 \leq y_{N-1} \leq x_N}{Max}\ \Phi_N(x_{N-1} - y_{N-1}). \tag{17.11}$$

За два останні етапи найбільший прибуток визначиться, як:

$$f_{N-1}(x_{N-2}) = \underset{0 \leq y_{N-2} \leq x_{N-2}}{Max}\{g(y_{N-2}) + h(x_{N-2} - y_{N-2}) + f_N(x_{N-1})\},$$

$$f_{N-1}(x_{N-2}) = \underset{0 \leq y_{N-2} \leq x_{N-2}}{Max}\ \Phi_{N-1}(x_{N-2} - y_{N-2}), \tag{17.12}$$

$$x_{N-1} = \varphi(y_{N-2}) + \psi(x_{N-2} - y_{N-2}),$$

звідки

$$f_{N-1}(x_{N-2}) = \underset{0 \leq y_{N-2} \leq x_{N-2}}{Max}\left\{\begin{matrix} g(y_{N-2}) + h(x_{N-2} - y_{N-2}) + \\ f_N\big(\varphi(y_{N-2}) + \psi(x_{N-2} - y_{N-2})\big) \end{matrix}\right\}. \tag{17.13}$$

Таким чином, оптимальний сумарний прибуток з i-го по N-й крок визначиться, як:

$$f_i(x_{i-1}) = \underset{0 \leq y_{i-1} \leq x_{i-1}}{Max}\{g(y_{i-1}) + h(x_{i-1} - y_{i-1}) + f_{i+1}(x_i)\}, \tag{17.14}$$

або $f_i(x_{i-1}) = \underset{0 \leq y_{i-1} \leq x_{i-1}}{Max}\ \Phi_i(x_{i-1} - y_{i-1})$, і $x_i = \varphi(y_{i-1}) + \psi(x_{i-1} - y_{i-1})$. (17.15)

17.4. ДИНАМІЧНІ МОДЕЛІ УПРАВЛІННЯ ЗАПАСАМИ. ОДНОПРОДУКТОВА ДИНАМІЧНА МОДЕЛЬ

В цій моделі припускається, що хоча й попит достовірно відомий, він може змінюватися від етапу до етапу. Рівень запасу контролюється *періодично*. Хоч запізнення поставки (відображене в фіксованому числі періодів) припустиме, в моделі вважається, що збільшення запасу проходить миттєво на початку етапу. Нарешті, дефіцит не припускається.

Побудова динамічної детермінованої моделі зводиться до дослідження скінченого проміжку часу. Це пояснюється тим, що для отримання чисельного розв'язку відповідних задач потрібно використовувати метод динамічного програмування, який в *цьому випадку* можна практично застосовувати на кінцевому етапі (кроці). Однак це не є важливою перешкодою, так як попит в майбутньому суттєво не впливає на рішення, прийняті для скінченого проміжку часу, що розглядається. Крім цього, зазвичай, не має сенсу припускати, що продукція буде зберігатися в вигляді запасу безмежно довго.

Визначимо для етапу i, $i = 1,2,\dots,N$, наступні величини:

z_i – кількість замовленої продукції (розмір замовлення);

ξ_i – потреба в продукції (попит);

x_i – початковий запас (на момент початку етапу i);

h_i – витрати на зберігання одиниці запасу, що надходить з етапу i в етап $i+1$;

K_i – витрати на оформлення замовлення;

$c_i(z_i)$ – функція граничних витрат, пов'язаних з купівлею (виробництвом) при заданому значенні z_i.

Нехай

$$C_i(z_i) = \delta_i K_i + c_i(z_i), \text{ де } \quad \delta_i = \begin{cases} 0, & z_i = 0, \\ 1, & z_i > 0. \end{cases} \tag{17.16}$$

Функція $c_i(z_i)$ нас цікавить лише тоді, коли витрати на купівлю одиниці продукції змінюються з часом або існують розриви ціни.

Так як дефіцит не допускається, то потрібно знайти оптимальне значення z_i, мінімізувавши загальні витрати на оформлення замовлень, купівлі і зберігання на всіх N етапах. Витрати на зберігання вважаються пропорційними величині

$$x_{i+1} = x_i + z_i - \xi_i, \tag{17.17}$$

що є об'ємом запасу, який переходить з етапу i в етап $i+1$. В результаті

витрати на зберігання на етапі i рівні $h_i x_{i+1}$. Це вводиться виключно з метою спрощення, так як модель можна узагальнити на випадок довільної функції витрат $H_i(x_{i+1})$, замінивши $h_i x_{i+1}$ на $H_i(x_{i+1})$. Аналогічно для оцінювання витрат на зберігання можна використати величини x_i або $(x_i + x_{i+1})/2$.

Побудова моделі динамічного програмування спрощується, коли представити задачу схематично, як показано на рис.1.1. Кожний етап відповідає одному кроку. Використовуючи зворотнє рекурентне рівняння, визначимо стан системи на кроці i як об'єм вихідного запасу x_i.

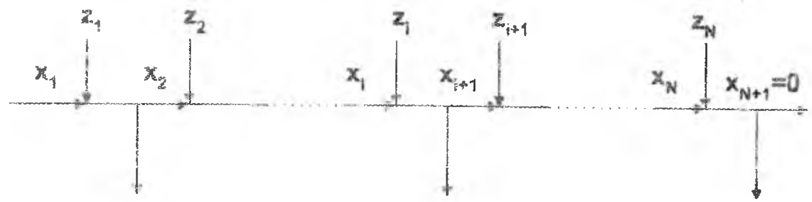


Рис. 17.2. Схематичне зображення динамічної моделі управління запасами

Нехай $f_i(x_i)$ – мінімальні витрати на етапах $i, i+1, \dots, N$. Рекурентне рівняння має вигляд:

$$f_N(x_N) = \min_{z_N + x_N = \xi_N,\ z_N \ge 0} \{C_N(z_N)\}, \tag{17.18}$$

$$f_i(x_i) = \min_{\xi_i \le x_i + z_i \le \xi_i + \dots + \xi_N,\ z_i \ge 0} \{C_i(z_i) + h_i(x_i + z_i - \xi_i) + f_{i+1}(x_i + z_i - \xi_i)\},$$

$i = 1,\ 2,\ \dots,\ N-1.$

Пряме рекурентне рівняння можна отримати, визначивши стан на кроці i як об'єм запасу на кінець етапу i. На рис. 17.2. ці стани задані величинами x_{i+1}. На будь-якому кроці на величини x_{i+1} накладені наступні обмеження:

$$0 \le x_{i+1} \le \xi_{i+1} + \dots + \xi_N. \tag{17.19}$$

Таким чином, в найгіршому випадку об'єм замовленої продукції z_i на етапі i може бути настільки великий, що запас x_{i+1} задовольняє попит на всіх наступних етапах.

Нехай $f_i(x_{i+1})$ – мінімальні загальні витрати на етапах $1, 2, \dots, i$ при заданій величині запасу x_{i+1} на кінець етапу i. Тоді рекурентне рівняння буде мати вигляд:

$$f_1(x_2) = \min_{0 \le z_1 \le \xi_1 + x_2} \{C_1(z_1) + h_1 x_2\},$$

$$f_i(x_{i+1}) = \min_{0 \le z_i \le \xi_i + x_{i+1}} \{C_i(z_i) + h_i x_{i+1} + f_{i-1}(x_{i+1} + \xi_i - z_i)\}. \qquad (17.20)$$

$i = 2, 3, \ldots, N$.

Пряма і зворотня постановки задачі з обчислювальної точки зору еквівалентні. Але (див. Приклад 17.6) прямий алгоритм (алгоритм прямого проходження) ефективніший при аналізі важливого часткового випадку розглянутої вище моделі. У наведеному прикладі для ілюстрації обчислюваної процедури використовується прямий алгоритм.

Модель відновлення при нескінченій кількості кроків.

У моделях цього типу система функціонує не протягом скінченого числа дискретних періодів, а протягом такого відтинку часу, що прямує до нескінченості.

У них задачах момент відновлення настає у той момент часу, коли система переходить у початковий стан. Тут керуючими змінними є послідовні інтервали між моментами заміни.

Розглянемо загальну модель задачі відновлення.

Скінчений плановий період.

Нехай у процесі прийняття рішень можна вибрати один з альтернативних варіантів, котрим присвоєні індекси $k=1, 2, \ldots, N$, де N – кількість відрізків часу між сусідніми моментами відновлення.

Пов'яжемо з кожним варіантом k затрати R_k при їх оцінці на початку періоду відновлення (наприклад, вартість заміни обладнання). Позначимо через f_n інтегральні дисконтні затрати (ІДЗ) для оптимальної стратегії відновлення, при якій один з альтернативних варіантів має бути обраний за n відтинків до кінця планового періода.

Припустимо, що обрали альтернативу k. Тоді відразу з'являються затрати R_k, і якщо припустити, що у наступний момент відновлення, тобто на відтинку $n-k$ також приймається оптимальна стратегія, то надалі виникають затрати $\alpha_k f_{n-k}$, де α_k використовується для приведення майбутніх затрат f_{n-k} до теперішнього моменту часу. Далі, за n відрізків до кінця планового періода оптимальною буде така стратегія, при якій мінімізується сума $(R_k + \alpha_k f_{n-k})$,

і ми приходимо до наступного рекурентного співвідношення:

$$f_n = \min_{k=1,2,\ldots,N} [\alpha_k f_{n-k} + R_k], \qquad (n \ge N) \qquad (17.21)$$

Нескінчений плановий період

Тепер розглянемо випадок нескінченого планового періоду. Вважаючи $n \to \infty$ і пропустивши індекс n у функції f_n, приходимо до співвідношення $f = \min_{k=1,\ldots,N} [\alpha_k f + R_k] \quad 0 < \alpha \le 1$.

Це функціональне рівняння має однозначний розв'язок виду:

$$f = \min_k \left[R_k / 1 - \alpha_k \right].$$

ПРИКЛАДИ

Приклад 17.1. Використання принципу оптимальності.

Необхідно знайти керування, що дозволяє перемістити точку з s_0 в s_n на площині за мінімальний час, який є пропорційний тривалості шляху, що обмежений лініями f_1 та f_2, причому точка на кожному з етапів може проходити лише через цілочисельні точки (можливі стани).

Розв'язання.

Починаємо зі стану s_n і послідовно рухаючись до s_0, будуємо оптимальні підтраєкторії, розглядаючи множини припустимих станів від S_6 до S_1, закінчуючи s_0.

Рухаючись від s_0 до s_n в прямому напрямку, встановлюємо оптимальну траєкторію (зображена на рис. 17.3 стрілками):

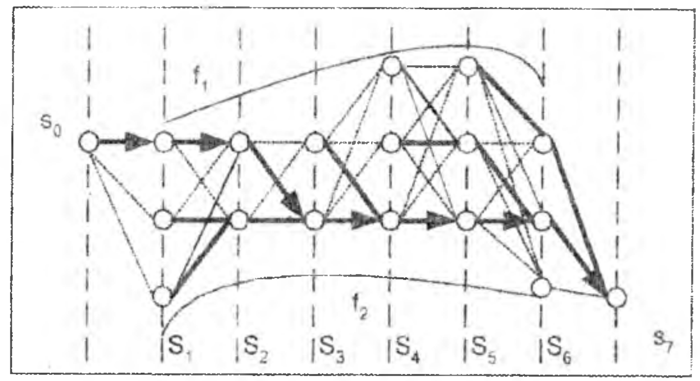


Рис.17.3. Графічна ілюстрація умови задачі прикладу 17.1

Приклад 17.2. Задача про оптимальні витрати пального.

Однією з найпростіших задач, до якої можна застосувати метод динамічного програмування, є задача про оптимальні витрати пального при набиранні висоти літаком.

Літак, що знаходиться на висоті h_0 та летить зі швидкістю v_0, повинен піднятися на висоту h_k та досягнути швидкості v_k. Досягнення цих значень здійснюється за k кроків, причому на кожному кроці можливий вибір однієї з двох альтернатив – набирати висоту при постійній швидкості або набирати швидкість при постійній висоті (можливе узагальнення цієї задачі з декількома, не обов'язково двома режимами). Для всіх режимів відомі витрати пального. Необхідно визначити таку траєкторію польоту, щоб досягнути заданої швидкості та висоти польоту за k кроків.

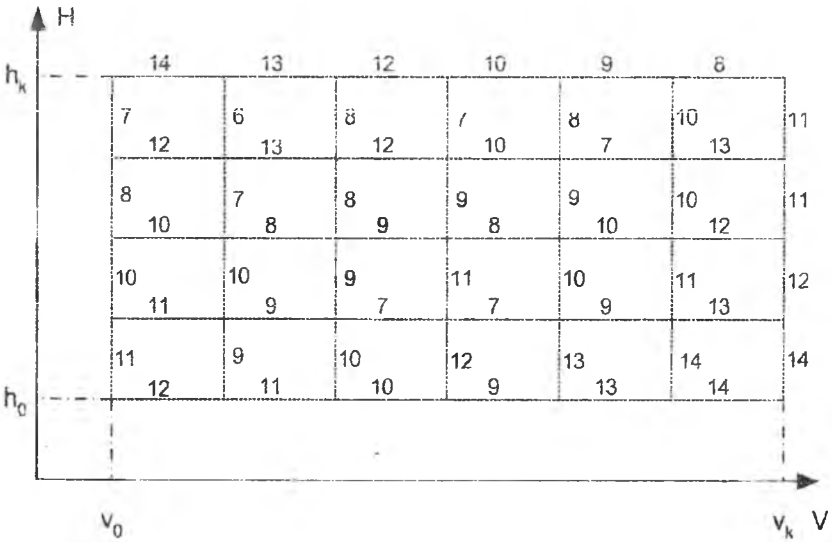


Рис. 17.4. Задача про оптимальні витрати пального

Розв'язання.

Стан $s_i=(h_i,v_i)$ характеризується двома складовими – висотою та швидкістю польоту. Відобразимо всі можливі стани та витрати пального при однокроковому переході зі стану в стан: розіб'ємо відрізки (h_0,h_k) та (v_0,v_k) на h_1 та h_2 інтервалів відповідно. В цьому випадку досягнення кінцевого стану здійснюється за $k=h_1+h_2$ кроків.

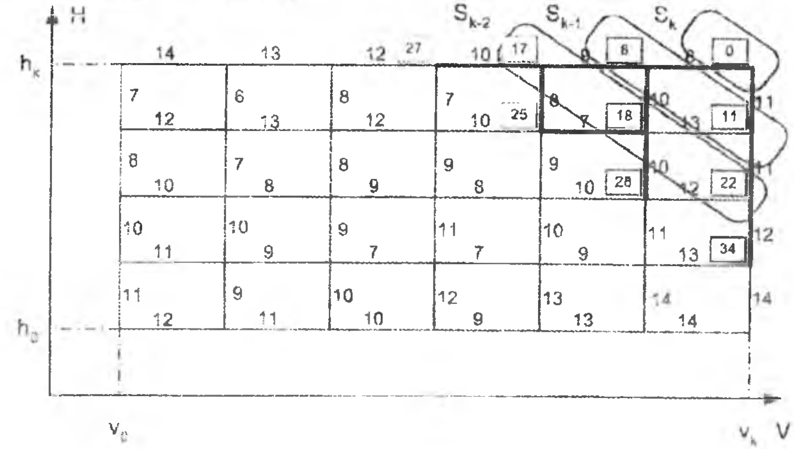


Рис. 17.5. Послідовність руху в процесі розв'язання задачі

Рухаючись від останнього стану до початкового, будуємо пучок оптимальних траєкторій, рухаючись якими з кожного стану досягаємо останнього з мінімальними витратами пального. Числа в квадратиках є оптимальними витратами пального для кожної з відповідних траєкторій.

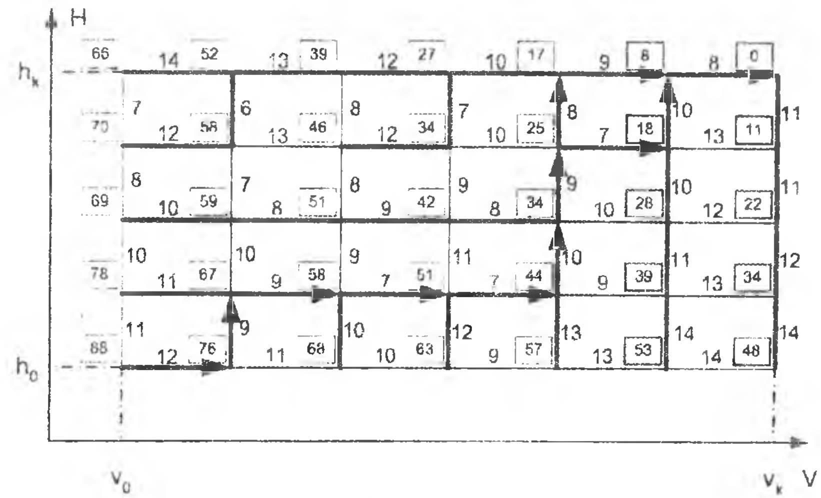


Рис. 17.6. Оптимальна траєкторія

Після досягнення початкового стану рухаємося в прямому напрямку і поновлюємо оптимальну траєкторію, що забезпечує мінімальні витрати пального. Мінімальні витрати пального при русі з початкового стану становитимуть 88 одиниць.

Позначивши 0 – зміну швидкості, 1 – зміну висоти, будуємо оптимальну траєкторію. Оптимальною буде траєкторія 0-1-0-0-0-1-1-1-0-0.

Також оптимальною буде траєкторія 0-1-0-0-0-1-1-0-1-0.

Приклад 17.3. Складання функціональних рівнянь. Задача про фонд робочої сили.

Планується робота виробництва на n місяців наперед. Виробниче завдання кожного місяця відоме та визначає необхідну кількість робітників m_j, $j = 1,2,...,n$. Якщо б керівник міг звільняти та приймати нових робітників без додаткових витрат, то він міг би кожен j-й місяць приймати рівно m_j робітників($j = 1,2,...,n$).

Припустимо, що робота j-го місяця може бути виконана і меншою кількістю робітників при понаднормовій праці. Нехай x_j - фактична кількість робітників в j-й місяць. Витрати на зміну чисельності робітників при переході від ($j-1$)-го місяця до j-го визначаються функцією $f_j(x_j - x_{j-1})$.

В залежності від знаку величини $x_j - x_{j-1}$ функція $f(x_j - x_{j-1})$ визначає витрати на наймання чи звільнення. Очевидно, що $f_j(0) = 0$. Відхилення чисельності робітників від m_j приводить до витрат $g_j(x_j - m_j)$, причому $g_j(0) = 0$, $j = 1,2,...,n$. Вважаємо, що в початковий момент кількість робітників дорівнює m_0. Необхідно побудувати рекурентні співвідношення, що дозволять за допомогою методу динамічного програмування визначити кількість робітників у кожному з періодів.

Розв'язання.

Функція мети задачі z визначається таким співвідношенням:

$$z = \sum_{j=1}^{n}\left[f_j(x_j - x_{j-1}) + g_j(x_j - m_j) \right], \quad x_0 = m_0 . \qquad (17.22)$$

Це задача з початковою фіксованою умовою.

Виведемо основне рекурентне співвідношення:

$$z = \min_{x_1,...,x_n} \left\{ f_1(x_1 - m_0) + g_1(x_1 - m_1) + \sum_{j=2}^{n} f_j(x_j - x_{j-1}) + g_j(x_j - m_j) \right\}; \qquad (17.23)$$

Задачу розв'язуватимемо у зворотному напрямку. Позначимо:

$$A_n(\xi) = \min_{x_n} \left[f_n(x_n - \xi) + g_n(x_n - m_n) \right], \quad \xi = x_{n-1}. \qquad (17.24)$$

Основне рекуренте співвідношення в цьому випадку матиме вигляд:

$$A_k(\xi) = \min_{x_k} \left[f_k(x_k - \xi) + g_k(x_k - m_k) + A_{k+1}(x_k) \right]. \qquad (17.25)$$

$A_k(\xi)$ - мінімальні витрати за місяці з k-го до n-го включно, якщо кількість робітників у $(k-1)$-й місяць рівна ξ.

Приклад 17.4. Метод функціональних рівнянь. Розв'язування задачі в числовому вигляді.

Конкретизуємо загальну постановку задачі з питання теми 17.3 наступним чином та розв'яжемо її. Нехай наявна кількість засобів до розподілу $x_0 = 8$, існує 2 підприємства, якщо в 1-е вкласти y засобів, то отриманий дохід становитиме:

$$g(y) = 10y - y^2, \qquad (17.26)$$

відповідно для другого залишиться $(x-y)$ та дохід становитиме відповідно $h(x-y) = 4(x-y)$ (17.27)

грошових одиниць. Кількість первісних засобів y зменшиться до значення $\varphi(y) = 0.6y$. (17.28)

а $(x-y)$ - до $\psi(x-y) = 0.8(x-y)$ (17.29)

грошових одиниць.

Необхідно розподілити ресурси на 3 роки таким чином, щоб отримати максимальний сумарний прибуток. Починаємо з останнього, третього року.

На початок 3-го року маємо до розподілу x_2 одиниць ресурсів. Здійснюємо розподіл:

$x_2 = y_2 + (x_2 - y_2)$, $0 \le y_2 \le x_2$,

$\Phi_3(x_2, y_2) = 10y_2 - y_2^2 + 4(x_2 - y_2)$,

$$f_3(x_2) = \underset{0\le y_2\le x_2}{Max}\, \Phi_3(x_2, y_2) = \underset{0\le y_2\le x_2}{Max}\, (6y_2 + 4x_2 - y_2^2), \qquad (17.30)$$

$$\frac{\partial \Phi_3}{\partial y_2} = -2y_2 + 6 = 0, \quad \frac{\partial^2 \Phi_3}{\partial y_2^2} = -2 \le 0 \Rightarrow Max, \quad y^*_2 = 3,$$

$f_3(x_2) = 6y^*_2 - y^{*2}_2 + 4x_2 = 9 + 4x_2, \quad x_2 \ge 3$

На початок 2-го року та 3-й рік разом:

$$\Phi_2(x_1, y_1) = 10y_1 - y_1^2 + 4(x_1 - y_1) + f_3(x_2)$$
$$= 10y_1 - y_1^2 + 4(x_1 - y_1) + 9 + 4x_2,$$
$$x_2 = 0.6y_1 + 0.8(x_1 - y_1),\ \Phi_2(x_1, y_1) = -(y_1 - 2.6)^2 + 2.6^2 + 9 + 7.2x_1. \quad (17.31)$$

Далі отримуємо:

$$f_2(x_1) = \underset{0 \le y_1 \le x_1}{Max} \quad \Phi_2(x_1, y_1), \quad \frac{\partial \Phi_2}{\partial y_1} = -2(y_1 - 2.6) = 0, \quad \frac{\partial^2 \Phi_2}{\partial y_1^2} = -2 \le 0,$$
$$y^* = 2.6, \quad f_2(x_1) = 15.76 + 7.2x_1. \quad (17.32)$$

На 1-й — 3-й роки (тобто за 3 роки):

$$\Phi_1(x_0, y_0) = 10y_0 - y_0^2 + 4(x_0 - y_0) + 15.76 + 7.2x_1,$$
$$x_1 = 0.6y_0 + 0.8(x_0 - y_0),$$
$$\Phi_1(x_0, y_0) = (y_0 - 2.28)^2 + 2.28^2 + 9.76x_0 + 15.76, \quad y^*_0 = 2.28, \quad (17.33)$$
$$f_1(x_0) = \underset{0 \le y_0 \le x_0}{Max} \quad \Phi_1(x_0, y_0), \quad x_0 = 8.$$

Оптимальний план розподілу ресурсів встановлюємо шляхом прямого проходу:

1). $x_0 = 8, \quad y_0 = 2.28, \quad x_0 - y_0 = 5.72 \quad Q_1 = 40.48.$ (17.34)

Залишок становить:

$$x_1 = 0.6 \times 2.28 + 0.8 \times 5.72 = 5.944 \quad . \quad (17.35)$$

2). $x_1 = 5.944, \quad y_1 = 2.6, \quad x_1 - y_1 = 3.344,$

$$Q_2 = \left(10 \times 2.6 - 2.6^2\right) + 4 \times 3.444 = 32.616,$$
$$x_2 = 0.6 \times 2.6 + 0.8 \times 3.44 = 4.2352 \quad (17.36)$$

3). $x_2 = 4.2353, \quad y_2 = 3, \quad x_2 - y_2 = 4.2352 - 3 = 1.2352,$

$$Q_3 = 25.9408, \ Q = f_1(x_0) = Q_1 + Q_2 + Q_3 = 99.0384. \quad (17.37)$$

Приклад 17.5. Задача про оптимальний розподіл капіталовкладень.

До складу фірми входить 4 виробничі підрозділи (підприємства). Для модернізації виробництва фірма виділила 6 млн грошових одиниць. Кожне з підприємств розробило по декілька проектів модернізації виробництва. Кожен з цих варіантів характеризується певними величинами затрат на

переобладнання та величиною очікуваного доходу.

Необхідно скласти план модернізації фірми таким чином, щоб отримати максимальний сумарний дохід для фірми загалом, не перевищивши виділену суму грошей. Розподіл грошей здійснюється з дискретністю 1 млн грошових одиниць. Характеристики варіантів проектів наведені в таблиці. Якщо підприємство не модернізується, то відповідні витрати та дохід є нульовими. Окрім того, можливі варіанти, коли обов'язково необхідно зробити мінімальні капіталовкладення, які можуть принести дохід, або ж не принести його і служитимуть для підтримання діяльності підприємства на певному рівні.

Таблиця 17.1. Умова задачі про оптимальний розподіл капіталовкладень.

№	Підпр. 1		Підпр. 2		Підпр. 3		Підпр. 4	
вар.	Витр.	Дох.	Витр.	Дох.	Витр.	Дох.	Витр.	Дох.
1	0	0	0	0	1	0	0	0
2	1	4	1	5	3	8	1	3
3	2	7	2	6	4	9	-	-
4	-	-	-	-	5	12	-	-

Для наведеної вище задачі варіант 1 для підприємства 3 є саме таким мінімальним, що обов'язково вимагатиме капіталовкладень в 1 млн.

Таким чином, задачу можна спростити - вважаючи, що в підприємство 3 ми вже вклали 1 млн – необхідно розподілити 6-1=5 млн і відповідно скоригувати капіталовкладення в це підприємство.

В результаті отримаємо наступну модифіковану таблицю.

Таблиця 17.2. Модифікована умова задачі про оптимальний розподіл капіталовкладень.

№	Підпр. 1		Підпр. 2		Підпр. 3		Підпр. 4	
вар.	Витр.	Дох.	Витр.	Дох.	Витр.	Дох.	Витр.	Дох.
1	0	0	0	0	0	0	0	0
2	1	4	1	5	2	8	1	3
3	2	7	2	6	3	9	-	-
4	-	-	-	-	4	12	-	-

Розв'яжемо задачу з використанням методу динамічного програмування, здійснивши розподіл грошових ресурсів за чотири етапи: спочатку для четвертого підприємства, потім для третього та четвертого разом, для другого, третього та четвертого разом та для першого, другого, третього та четвертого підприємств. Кількість етапів таким чином становитиме (за $N = 4$ кількістю підприємств):

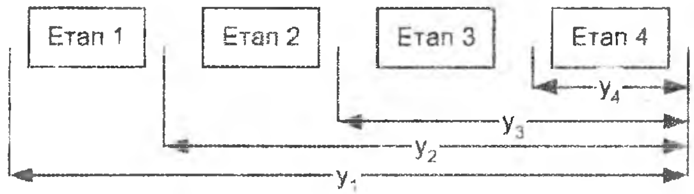
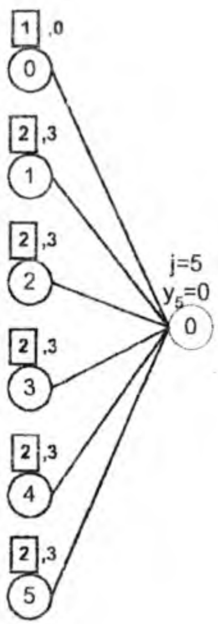


Рис 17.7. Послідовність розв'язання задачі про розподіл капіталовкладень

Позначимо: y_j – об'єм капіталовкладень, що розподілятимуться від j-го до N-го етапу; $f_j\left(y_j\right)$ – оптимальний дохід від j-го до N-го етапу за умови вкладення y_j коштів; $c_j\left(k_j\right)$ – капіталовкладення в проект k_j на j-му етапі; $R_j\left(k_j\right)$ – дохід від капіталовкладень на j-му етапі; k_j – варіант проекту, обраний на j-му етапі.

Розрахунки почнемо, ввівши штучний, 5-й етап, на якому всі грошові ресурси будуть вже розподілені, $N=5$, $f_N\left(y_N\right)=0$. Наступні розрахунки проводитимемо, рухаючись до першого етапу за формулами:

$$f_j\left(y_j\right)=\underset{k_j\in K_j}{Max}\left\{R_j\left(k_j\right)+f_{j+1}\left(y_{j+1}\right)\right\},\ y_j=y_{j+1}+c_j\left(k_j\right). \qquad (17.38)$$

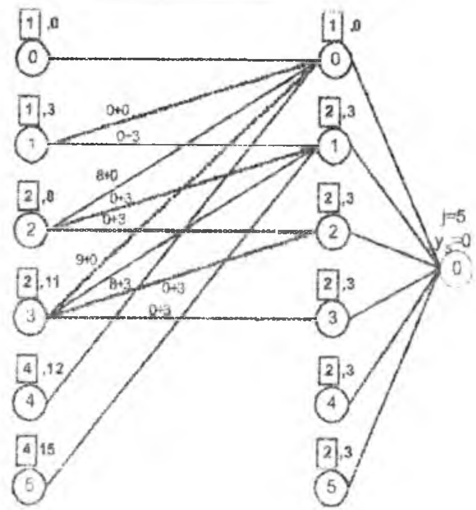


Процес розв'язання задачі нижче відображений графічно.

На першому етапі розподіляємо грошові засоби, що залишилися для 4-го підприємства. Кількість цих засобів може становити від 0 до 5 млн грошових одиниць. Однак, оскільки грошовий ресурс є обмежений, то реально до розподілу на останньому кроці може залишитися або 0, або 1 млн грошових одиниць. Якщо ж їх буде більше, то все одно вибір буде лише між двома варіантами – проект, що потребує 0 чи 1 млн грошових одиниць капіталовкладень. У першому випадку (0 млн) оптимальним буде вибір проекту 1 (в квадраті на рисунку), а у всіх інших – варіанту 2 (що дає прибуток 3 млн грошових одиниць).

Переходимо до наступного етапу – і знову ж таки до можливих варіантів належатимуть такі, коли до розподілу з попередніх етапів може залишитися від 0 до 5 млн грошових одиниць з дискретністю 1 млн.

Приєднуючи до розподілу 3-є підприємство, визначаємо умовно-оптимальні розподіли коштів на



два останніх етапи відразу – для 3-го та 4-го підприємства разом. У цьому випадку знову ж таки вважаємо, що з попередніх етапів можуть залишитися до розподілу від 0 до 5 млн грошових одиниць з дискретністю 1 млн. Переглядаючи можливі варіанти, добудовуємо пучок оптимальних траєкторій на два останні етапи (3-тє та 4-те підприємство) разом.

На наступних двох етапах добудовуємо пучок оптимальних траєкторій на 2-е та 1-е підприємство, і рухаючись у прямому напрямку, поновлюємо оптимальну траєкторію.

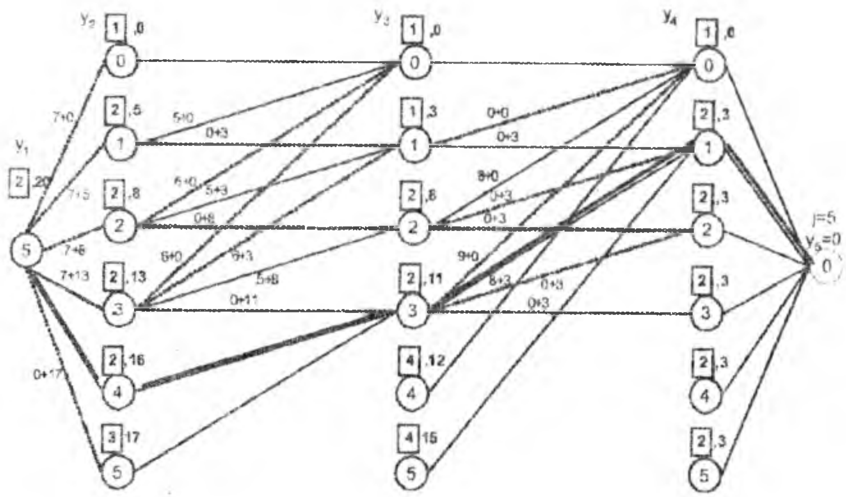


Рис. 17.8. Процес розв'язування задачі про оптимальний розподіл капіталовкладень

Таким чином, оптимальним розподілом для розв'язаної задачі є наступний: Для 1-го підприємства обирається 3-й варіант проекту, для 2-го – 2-й, для 3-го – також 2-й, і для 4-го – також 2-й. При цьому досягається максимальний загальний дохід, що становить 20 млн грошових одиниць.

Приклад 17.6. Динамічна задача управління запасами.

Розглянемо триетапну систему управління запасами з дискретною продукцією і динамічним детермінованим попитом. Вихідні дані задачі наступні:

Період i	Попит ξ_i, од.	Витрати на оформлення замовлення K_i, гр.од.	Витрати на зберігання h_i, гр. од.
1	3	3,00	1,00
2	2	7.00	3,00
3	4	6,00	2,00

Запас x_1 для етапу 1 дорівнює 1 од. продукції. Припускається, що граничні витрати на придбання продукції становлять 10 гр. од. за кожну одиницю часу для перших трьох одиниць і 20 гр. од. для кожної додаткової одиниці. В цьому випадку функція виробничих витрат для періоду і становитиме: $C_i(z_i)=K_i+c_i(z_i)$ для $z_i>0$,

$$c_i(z_i)=\begin{cases}10\quad z_i, & 0\le z_i\le 3,\\ 30+20(z_i-3), & z_i\ge 4.\end{cases}$$

Результати покрокових обчислень для прямого алгоритму наступні.

Крок 1: $\xi_1=3,\ 0\le x_2\le 2+4=6.$

		$f_1(z_1\|x_2)=C_1(z_1)+h_1x_2$							Оптимальний розв'язок	
		$z_1=2$	3	4	5	6	7	8		
x_2	h_1x_2	$C_1(z_1)=23$	33	53	73	93	113	133	$f_1(x_2)$	z_1^*
0	0	23							23	2
1	1		34						34	3
2	2			55					55	4
3	3				76				76	5
4	4					97			97	6
5	5						118		118	7
6	6							139	139	8

Так як $x_1=1$, мінімальне значення z_1 становить $\xi_1-x_1=3-1=2$.

Крок 2: $\xi_2=2,\ 0;\ 0\le x_3\le 4.$

		$f_2(z_2\|x_3)=C_2(z_2)+h_2x_3+f_1(x_3+\xi_2-z_2)$							Оптимальний розв'язок	
		$z_2=0$	1	2	3	4	5	6		
x_3	h_2x_3	$C_2(z_2)=0$	17	27	37	57	77	97	$f_2(x_3)$	z_2^*
0	0	0+55=55	17+34=51	27+23=50					50	2
1	3	3+76=79	20+55=75	30+34=64	40+23=63				63	3
2	6	6+97=103	23+76=99	33+55=88	43+34=77	63+23=86			77	3
3	9	9+118=127	26+97=123	36+76=112	46+55=101	66+34=100	86+23=109		100	4
4	12	12+139=151	23+118=147	39+97=136	49+76=125	69+55=124	89+34=123	109+23=132	123	5

Крок 3: $\xi_3=4,\ x_4=0.$

		$f_3(z_3\|x_4)=C_3(z_3)+h_3x_4+f_2(x_4+\xi_3-z_3)$					Оптимальний розв'язок	
		$z_1=0$	1	2	3	4		
x_4	h_3x_4	$C_3(z_3)=0$	16	26	36	56	$f_3(x_4)$	z_3^*
0	0	0+123=123	16+100=116	26+77=103	36+63=99	56+50=106	99	3

Розв'язок визначається наступними значеннями шуканих змінних: $z_1^*=2,\quad z_2^*=3,$ і $z_3^*=3$, загальні витрати становлять 99 грош.одиниць.

РЕЗЮМЕ

1. Динамічне програмування є математичним апаратом, що дозволяє здійснити оптимальне планування багатокрокових керованих процесів та процесів, які залежать від часу. Багатоетапні керовані процеси мають наступні спільні риси: можуть бути піддані декомпозиції, кожен етап характеризується станом, який визначається значеннями факторів, або змінних, кількість яких для кожного з етапів може бути різною. З загального числа змінних виділяються керовані, тобто ті, значення яких можна спрямовано змінювати і цими змінами впливати на стан процесу, та змінні стану. На кожному кроці існує залежність між керованими *змінними, змінними стану та функцією мети, яка представляється за допомогою рівняння, системи рівнянь або таблично. Критерій оптимальності повинен бути адитивним (або мультиплікативним). За допомогою методу динамічного програмування здійснюється оптимізація багатоетапних процесів, виходячи з інтересів системи загалом, а не кожної її стадії як окремого елементу, тому розв'язок на кожному кроці повинен відповідати вимогам оптимізації процесу загалом.*

2. Задача для того, щоб можна було для її розв'язування застосувати метод динамічного програмування, повинна відповідати наступним вимогам: функція мети повинна бути адитивною (мультиплікативною); в процесі руху системи повинна бути відсутня післядія. Принцип оптимальності дозволяє розв'язати багатокрокові задачі формування оптимальної стратегії поведінки системи і формулюється наступним чином – оптимальне керування має наступну основоположну властивість: яким не були би початковий стан та прийняте початкове рішення, наступні рішення повинні утворювати оптимальне керування відносно стану, що виник в результаті попереднього рішення.

3. При застосуванні методу функціональних рівнянь знаходження оптимального розв'язку (стратегії) багатокрокового процесу отримується шляхом розв'язування деяких функціональних рівнянь. Ці рівняння мають рекурентний вигляд та відображають розгортання процесу розв'язування від остаточного стану до первісного згідно з принципом оптимуму.

4. Багатокрокові задачі управління запасами успішно розв'язуються методом динамічного програмування. В однопродуктовій динамічній моделі припускається, що попит достовірно відомий, але він може змінюватися від етапу до етапу, а дефіцит є неприпустимим. Рівень запасу контролюється періодично. Хоч можливе запізнення поставки в фіксоване число періодів, в моделі вважається, що збільшення запасу проходить миттєво на початку етапу.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 17.1. Задача про оптимальні витрати пального.

Літак, що знаходиться на висоті h_0 та летить зі швидкістю v_0, повинен піднятися на висоту h_k та досягнути швидкості v_k. Досягнення цих значень здійснюється за k кроків, причому на кожному кроці можливий вибір однієї з трьох альтернатив - набирати висоту при постійній швидкості, набирати

швидкість при постійній висоті або одночасно змінювати швидкість та висоту. Для всіх режимів відомі витрати пального (рис. 17.9). Необхідно визначити таку траєкторію польоту, щоб досягнути заданої швидкості та висоти польоту за k кроків.

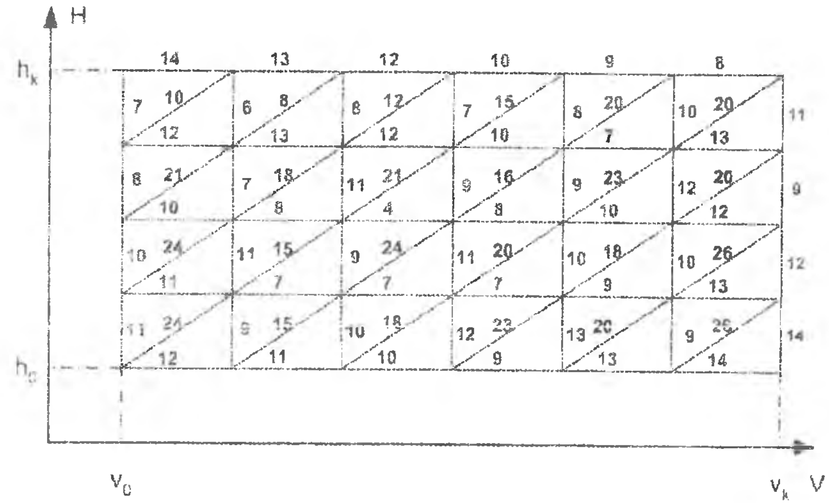


Рис 17.9. Задача про оптимальні витрати пального

Завдання 17.2. Задача оптимального розподілу ресурсів.

Наявно $x_0 = 100$ млн грошових одиниць, які необхідно вкласти в розвиток 2-х підрозділів фірми (вкладення проводяться раз в рік, на початку року), щоб за 3 роки загальний дохід фірми був максимальним. Якщо за умови наявності на початок року до розподілу x млн грошових одиниць в 1-й підрозділ вкласти y млн грошових одиниць, то річний дохід від нього становитиме $g(y) = 4y$ млн грошових одиниць, якщо ж в 2-й вкласти $(x-y)$ одиниць, що залишаться, то дохід від цього вкладення становитиме $h(x-y) = 3(x-y)$. Кількість первісних засобів за рік зменшиться до значення для y – $\varphi(y) = 0.2y$, а $(x-y)$ – до $\psi(x-y) = 0.6(x-y)$ млн грошових одиниць.

Завдання 17.3. Задача про оптимальний розподіл капіталовкладень.

Визначити за допомогою методу динамічного програмування оптимальну стратегію інвестування для наступної задачі.

Фірма виділяє для впровадження нових інформаційних технологій на

3-х своїх філіях 10 мільйонів грошових одиниць. Для кожної з філій можлива реалізація проектів інформаційних систем в декількох варіантах, з відповідно оціненими значеннями витрат та очікуваного доходу для кожного варіанту кожної філії - числові дані наведені в таблиці. Розподіл виділених грошових ресурсів для реалізації проектів можливий з дискретністю, не меншою ніж 1 мільйон грошових одиниць. Необхідно визначити оптимальний варіант розподілу інвестиції між філіями фірми.

№ проекту	Філія 1		Філія 2		Філія 3	
	Витр.	Дохід	Витр.	Дохід	Витр.	Дохід
1	0	0	3	4	1	1
2	2	3	4	7	2	4
3	4	6			4	8
4	6	8				
5	10	12				

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. В чому полягають особливості задач динамічного програмування?
2. Що таке керування та який процес називається керованим?
3. Які спільні риси мають багатоетапні керовані процеси?
4. Дайте загальну постановку задачі динамічного програмування.
5. Яким вимогам повинна відповідати задача, щоб для її розв'язування можливо було застосувати метод динамічного програмування?
6. Дайте визначення оптимальної стратегії.
7. Сформулюйте принцип оптимальності Белмана та проілюструйте його застосування на прикладах.
8. В чому полягає проблема розмірності в методі динамічного програмування?
9. Проілюструйте основні ідеї методу функціональних рівнянь на прикладі простого багатоетапного процесу розподілу.
10. Поясніть рекурентні співвідношення для однопродуктової динамічної моделі управління запасами.

РОЗДІЛ 9 ЧИСЕЛЬНІ МЕТОДИ ПОШУКУ ОПТИМАЛЬНИХ РІШЕНЬ

ТЕМА 18. ОСНОВНІ ПІДХОДИ ДО ПОШУКУ ОПТИМАЛЬНИХ РІШЕНЬ В НЕЛІНІЙНИХ ЗАДАЧАХ

Розв'язання задач нелінійної оптимізації без обмежень базується на результатах математичного аналізу, але навіть в цьому відносно нескладному випадку знаходження екстремумів функції n змінних без обмежень виникають системи нелінійних рівнянь, аналітичне розв'язання яких можливе лише для найпростіших випадків. Для задач з обмеженнями можливе застосування методу множників Лагранжа, а для обмежень загального типу формулюються фундаментальні умови Куна-Такера. Однак це є в більшості випадків не чим іншим, ніж переформулюванням первісної задачі, так як розв'язання системи нелінійних рівнянь за складністю еквівалентне їй. Тому широкого розповсюдження для таких задач набули чисельні методи, важливе місце серед яких посідають методи прямого пошуку. Ці методи, як до речі і всі інші, не гарантують знаходження глобального оптимуму, і підвищення їх надійності досягається вибором декількох достатньо віддалених початкових точок та визначення відповідних інтервалів пошуку.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ

знати: ⇨ умови існування екстремуму функції n змінних з обмеженнями у вигляді рівнянь; ⇨ умови Куна-Такера для пошуку оптимальних рішень функції n змінних з обмеженнями загального вигляду; ⇨ методи прямого пошуку оптимальних рішень, їх переваги та недоліки;

вміти: ⇨ знаходити оптимальні рішення за допомогою методу множників Лагранжа; ⇨ застосовувати методи чисел Фібоначі, золотого перетину та апроксимації для пошуку екстремумів унімодальної функції якості; ⇨ використовувати метод Хука-Дживса для прямого пошуку екстремуму функції n змінних.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ локальний екстремум
- ☑ множники Лаґранжа
- ☑ прямий пошук
- ☑ глобальний екстремум
- ☑ необхідні та достатні умови екстремуму
- ☑ золотий перетин
- ☑ ряд Тейлора
- ☑ умови Куна-Такера
- ☑ унімодальна функція
- ☑ матриця Гессе
- ☑ числа Фібонічі
- ☑ метод покоординатного спуску
- ☑ звуження інтервалу пошуку
- ☑ квадратична інтерполяція
- ☑ лінія сталого рівня
- ☑ базова точка
- ☑ локальна поведінка функції
- ☑ досліджувальний пошук
- ☑ пошук за зразком
- ☑ інтервал невизначеності

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

18.1. Основні визначення та поняття.
18.2. Метод множників Лаґранжа.
18.3. Обмеження у вигляді нерівностей. Умови Куна-Такера.
18.4. Методи прямого пошуку для функції однієї змінної.
18.5. Методи прямого пошуку для функцій n змінних.
18.6. Метод деформованого многогранника.

18.1. ОСНОВНІ ВИЗНАЧЕННЯ ТА ПОНЯТТЯ

Функції однієї змінної.

Функція $f(x)$ має локальний мінімум в точці x_0, якщо існує деяка додатна величина δ, така, що якщо $|x-x_0|<\delta$, то $f(x)\geq f(x_0)$, тобто якщо існує окіл точки x_0, такий, що для всіх значень x в цьому околі $f(x)$ більше $f(x_0)$. Функція $f(x)$ має глобальний мінімум у точці x^*, якщо для усіх x справедлива нерівність $f(x)\geq f(x^*)$.

На рис 18.1. наведене графічне представлення функції $f(x)$, що має локальний мінімум у точці x_0 і глобальний мінімум у точці x^*.

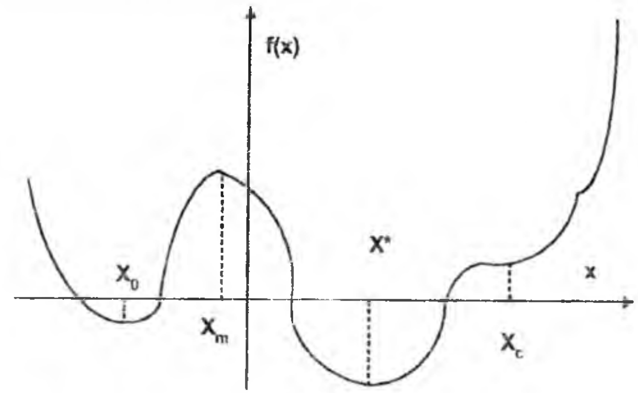


Рис. 18.1. Глобальні та локальні екстремуми

В точках x_0 і x^* похідна $f'(x)$ (градієнт функції) рівна нулю. Отже, x_0 і x^* будуть розв'язками рівняння $f'(x)=0$.

У точках x_m, (досягається локальний максимум), та x_c, (знаходиться точка горизонтального перегину функції), також виконується ця умова, яка є необхідною, але не достатньою умовою існування екстремуму.

Розглянемо розклад функції $f(x)$ у ряд Тейлора в околі точки x_0, (вважаючи, що $f(x)$ і її похідні неперервні):

$$f(x_0+h)-f(x)=hf'(x_0)+\frac{h^2}{2!}f''(x_0)+\frac{h^3}{3!}f'''(x_0)+\frac{h^4}{4!}f''''(x_0)+\ldots \quad (18.1)$$

Якщо в точці x_0 досягається мінімум, то ліва частина співвідношення буде невід'ємною для будь-якого достатньо малого h $(|h|<\delta)$. Отже, перша похідна $f'(x_0)$ повинна бути рівною нулю. Оскільки в наступній складовій $h^2>0$ завжди, коли $f''(x_0)>0$, то у точці x_0 досягається мінімум. Якщо $f'(x_m)=0$ і $f''(x_m)<0$, то з аналогічних міркувань у точці x_m досягається максимум.

Таким чином, у загальному випадку, якщо функція $f(x)$ і її похідні неперервні, то точка x_0 є точкою екстремуму (максимуму або мінімуму) тоді, і лише тоді, коли n парне, де n - порядок найпершої похідної, що не перетворюється в нуль у точці x_0. Якщо $f^n(x_0)<0$, то в точці x_0 досягається максимум, якщо $f^n(x_0)>0$, то в точці x_0 досягається мінімум.

Функції n змінних.

Розглянемо функцію n дійсних змінних $f(x_1, x_2, x_3, ..., x_n) = f(x)$. Точка в n-вимірному евклідовому просторі з координатами $(x_1, x_2, x_3, ..., x_n)$, є вектором-стовпчиком x. Градієнт функції, тобто вектор із компонентами $(\partial f / \partial x_1, \partial f / \partial x_2, ..., \partial f / \partial x_n)$, позначається $\nabla f(x)$ або g(x). Матриця Гессе (гессіан) функції $f(x)$ позначається як $G(x)$ і є симетричною матрицю $n \times n$ елементів виду

$$G_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}.$$

Функція $f(x)$ має локальний мінімум у точці x_0, якщо існує окіл точки x_0, такий, що $f(x)$ більше $f(x_0)$ у всіх точках цього околу, тобто існує додатна величина δ, така, що для $|x - x_0| < \delta$ справедлива нерівність $f(x) \geq f(x_0)$. У випадку глобального мінімуму в точці x^* для всіх x справедлива нерівність $f(x) \geq f(x^*)$. Розкладаючи в ряд, отримаємо:

$$f(x_0 + h) - f(x_0) = \sum_{i=1}^{n} h_i \frac{\partial f}{\partial x_i}(x_1, ..., x_n) + \frac{1}{2!}\sum_{i=1}^{n}\sum_{j=1}^{n} h_i h_j \frac{\partial^2 f}{\partial x_i \partial x_j}(x_1, ..., x_n) + ... \tag{18.2}$$

$$= h^T \nabla f(x_0) + \frac{1}{2} h^T G(x_0) h + ...$$

Якщо x_0 є точкою мінімуму функції $f(x)$, то кожна перша окрема похідна $\partial f / \partial x_i$ $i = (1, ..., n)$ повинна перетворюватись в нуль у точці x_0. Якщо це не так, то відповідним вибором h_i можна добитися того, що різниця $f(x_0 + h) - f(x_0)$ буде від'ємною.

Отже, необхідною умовою мінімуму в точці x_0 є рівняння:

$$\nabla f(x_0) = 0, \text{ тобто } \frac{\partial f(x_0)}{\partial x_i} = 0 \quad (i = 1, ..., n). \tag{18.3}$$

Тоді знак різниці $f(x_0 + h) - f(x_0)$ визначається складовою

$\frac{1}{2} h^T G(x_0) h$.

Якщо матриця $G(x_0)$ додатньо визначена, то ця складова додатна для всіх h. Отже, необхідними і достатніми умовами мінімуму є $\nabla f(x_0) = 0$, $G(x_0)$ позитивно визначена. Відповідно необхідними і достатніми умовами максимуму є $\nabla f(x_m) = 0$, $G(x_m)$ негативно визначена.

18.2. МЕТОД МНОЖНИКІВ ЛАГРАНЖА

Розглянемо задачу мінімізації функції двох змінних $z = f(x, y)$, де на x і y накладене обмеження, що задається рівнянням

$$g(x, y) = 0. \tag{18.4}$$

Взагалі, рівняння $g(x, y) = 0$ можна розв'язати відносно y як функцію від x, тобто $y = h(x)$. Звісно, на практиці може виявитись важко або навіть неможливо знайти явний вигляд функції $h(x)$. За виконання певних умов диференціювання похідна функції $h(x)$

$$\frac{dy}{dx} = \frac{d}{dx}h(x) = -\frac{\partial g}{\partial x}\bigg/\frac{\partial g}{\partial y} \tag{18.5}$$

Тоді функцію $z = f(x, y) = f[x, h(x)]$ (18.6)

можна записати як функцію одної незалежної змінної x. Необхідною умовою мінімуму функції z буде співвідношення

$$\frac{dz}{dx} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}\frac{dy}{dx} = 0, \text{ тобто } \frac{\partial f}{\partial x} + \left(\frac{-\frac{\partial f}{\partial y}}{\frac{\partial g}{\partial y}}\right)\frac{\partial g}{\partial x} = 0. \tag{18.7}$$

Співвідношення (18.4) і (18.5) можуть бути розв'язані з метою отримання значень x^*, y^* у точці мінімуму.

Цей результат може бути представлений в іншій формі. Якщо покласти

$$\lambda = \frac{-\partial f}{\partial y}(x, y)\bigg/\frac{\partial g}{\partial y}(x, y) \tag{18.8}$$

при $x = x^*$, $y = y^*$, то у точці мінімуму виконуються співвідношення:

$g(x, y) = 0$, $\frac{\partial f}{\partial x}(x, y) + \lambda\frac{\partial g}{\partial x}(x, y) = 0$, $\frac{\partial f}{\partial y}(x, y) + \lambda\frac{\partial g}{\partial y}(x, y) = 0$, причому останнє співвідношення випливає безпосередньо з співвідношення (18.8).

Отримати ці три необхідні умови можна, використовуючи функцію Лагранжа

$$F(x, y, \lambda) = f(x, y) + \lambda g(x, y), \tag{18.9}$$

яка є сумою функції мети і добутку множника Лагранжа λ на функції обмеження. Тоді необхідні умови мінімуму функції $f(x, y)$ при наявності обмежень можуть бути записані в наступному вигляді:

$$\left.\begin{aligned} \frac{\partial F}{\partial x}(x,y,\lambda) &= \frac{\partial f}{\partial x}(x,y)+\lambda\frac{\partial g}{\partial x}(x,y)=0,\\ \frac{\partial F}{\partial y}(x,y,\lambda) &= \frac{\partial f}{\partial y}(x,y)+\lambda\frac{\partial g}{\partial y}(x,y)=0,\\ \frac{\partial F}{\partial \lambda}(x,y,\lambda) &= g(x,y)=0. \end{aligned}\right| \qquad (18.10)$$

Це система трьох рівнянь, розв'язком якої є значення x^*, y^* і λ^* - у точці мінімуму.

Необхідні умови мінімуму (18.10) можуть бути узагальнені для функцій n змінних при наявності m обмежень у вигляді рівностей.

Розглянемо задачу мінімізації функції $z=f(x)=f(x_1,x_2,\dots,x_n)$, де на змінну x накладені обмеження

$$g_1(x)=0,\ g_2(x)=0,\dots,g_m(x)=0. \qquad (18.11)$$

Обмеження можна використовувати для того, щоб висловити m змінних (без обмеження загальності їх можна позначити $x_1,x_2,\dots,x_m$) через решту $(n-m)$ змінних, які можна розглядати як незалежні змінні. В точці мінімуму при наявності обмежень $f(x+h)-f(x)\ge 0$ для усіх h, що задовольняють умові $g_i(x+h)=g_i(x)=0$ при $i=1,\dots,m$.

Тоді з точністю до першого порядку h_j будемо мати:

$$\sum_{j=1}^{n} h_j\frac{\partial f}{\partial x_j}=0,\ \text{де}\ \sum_{j=1}^{n} h_k\frac{\partial g_i}{\partial x_j}=0\ \text{їде}\ i=1,2,\dots m.$$

Цю умову можна записати інакше:

$$\sum_{j=1}^{n} h_j\Big(\frac{\partial f}{\partial x_j}+\sum_{i=1}^{m}\lambda_i\frac{\partial g_i}{\partial x_j}\Big)=0, \qquad (18.12)$$

де $\lambda_1,\lambda_2,\dots,\lambda_m$ - множники Лагранжа.

Оскільки $h_{m+1},h_{m+2},\dots,h_n$ є незалежними приростами, коефіцієнти при них повинні бути рівні нулю, тобто

$$\frac{\partial f}{\partial x_j}+\sum_{i=1}^{m}\lambda_i\frac{\partial g_i}{\partial x_j}=0\ \text{при}\quad j=m+1,\dots,n.$$

Прирости $h_1,h_2,\dots,h_m$ не є незалежними, і їх можна покласти рівними

нулю вибором множників Лагранжа в рівнянні (18.12).

Таким чином, ми обираємо $\lambda_1,\lambda_2,\dots,\lambda_m$ такими, щоб

$$\frac{\partial f}{\partial x_j}+\sum_{i=1}^{m}\lambda_i\frac{\partial g_i}{\partial x_j}=0\ \text{при}\ j=1,2,\dots,m.$$

Остаточно отримаємо:

$$\frac{\partial f}{\partial x_j}+\sum_{i=1}^{m}\lambda_i\frac{\partial g_i}{\partial x_j}=0\ \text{при}\ j=1,2,\dots,n. \qquad (18.13)$$

Отже, якщо визначити функцію Лагранжа у вигляді

$$F(x,\lambda)=f(x)+\sum_{i=1}^{m}\lambda_i g_i(x). \qquad (18.14)$$

то необхідні умови мінімуму функції $f(x)$ при наявності обмежень можна записати наступним чином:

$$\frac{\partial F}{\partial x_j}=\frac{\partial f}{\partial x_j}+\sum_{i=1}^{m}\lambda_i\frac{\partial g_i}{\partial x_j}=0\ \text{при}\ j=1,2,\dots,n, \qquad (18.15)$$

$$\frac{\partial F}{\partial \lambda_i}=g_i(x)=0\ \text{при}\ i=1,\dots,m. \qquad (18.16)$$

Зазначимо, що для припустимих значень x (таких, що задовільняють обмеження) справедливе співвідношення:

$$F(x,\lambda)=f(x)+\sum_{i=1}^{m}\lambda_i g_i(x)=f(x).$$

В точці мінімуму при наявності обмежень на значення x^* можна записати, що $f(x^*+h)-f(x^*)\ge 0$, де h задовільняє рівняння $g_i(x^*+h)=0$ для усіх i.

Таким чином,

$$F(x^*+h)-F(x^*)=\sum_{j=1}^{n}\frac{\partial f}{\partial x_j}h_j+\frac{1}{2}\sum_{j=1}^{n}\sum_{i=1}^{n}h_i\frac{\partial^2 F}{\partial x_i\partial x_j}h_j+\ldots\ge 0,$$

де добутки обчислені в точці x^* при $\lambda=\lambda^*$.

З врахуванням рівняння (18.15) отримаємо для усіх h, що задовольняють обмеженням,

$$\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}h_i\frac{\partial^2 F}{\partial x_i \partial x_j}h_j \geq 0.$$

Достатньою умовою мінімуму при наявності обмежень є рівняння (18.15) і (18.16), а також позитивна визначеність квадратичної форми

$$\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}h_i\frac{\partial^2 F}{\partial x_i \partial x_j}h_j \qquad (18.17)$$

для значень h, що задовольняють обмеження.

Не завжди просто звести квадратичну форму до вигляду, придатного до застосування.

18.3. ОБМЕЖЕННЯ У ВИГЛЯДІ НЕРІВНОСТЕЙ. УМОВИ КУНА-ТАКЕРА

Розповсюдимо метод множників Лагранжа на обмеження у вигляді нерівностей. Розглянемо загальну задачу математичного програмування: мінімізувати функцію $f(x)$ при наявності m обмежень

$g_i(x) \leq b_i \quad (i = 1, 2, ..., m)$.

Не існує методу, який гарантував би існування розв'язку довільної подібної задачі. Обмеження у вигляді нерівностей можуть бути перетворені в обмеження у вигляді рівностей додаванням до кожного з них невід'ємної послаблюючої змінної u_i^2 (зауважимо, що змінна u_i^2 завжди додатна): $g_i(x) + u_i^2 = b_i$ або

$$g_i(x) + u_i^2 - b_i = 0. \qquad (18.18)$$

Таким чином, задача зводиться до мінімізації функції $f(x)$ при наявності m обмежень у вигляді рівності $g_i(x) + u_i^2 - b_i = 0$. Згідно з викладеним у попередньому питанні, сформуємо функцію Лагранжа:

$$F(x, \lambda, u) = f(x) + \sum_{i=1}^{m}\lambda_i[g_i(x) + u_i^2 - b_i]. \qquad (18.19)$$

Необхідними умовами, які повинні виконуватися в стаціонарній точці, є наступні:

$$\frac{\partial F}{\partial x_j} = 0 = \frac{\partial f}{\partial x_j} + \sum_{i=1}^{m}\lambda_i\frac{\partial g_i}{\partial x_j} \quad \text{при} \quad j = 1, 2, ... n, \qquad (18.20)$$

$$\frac{\partial F}{\partial \lambda_i} = 0 = g_i(x) + u_i^2 - b_i \quad \text{при} \quad i = 1, 2, ..., m, \qquad (18.21)$$

$$\frac{\partial F}{\partial u_i} = 0 = 2\lambda_i u_i \quad \text{при} \quad i = 1, 2, ..., m. \qquad (18.22)$$

Помноживши останнє рівняння на $u_i/2$, отримаємо $\lambda_i u_i^2 = 0$, тобто

$$\lambda_i[b_i - g_i(x)] = 0 \quad \text{при} \quad i = 1, 2, ..., m. \qquad (18.23)$$

Рівняння (18.20), (18.21) і (18.23) є необхідними умовами мінімуму в точці x^* при наявності обмежень. Рівняння (18.21) є повторним записом обмежень $g_i(x) \leq 0$. Рівняння (18.23) означає, що або $\lambda_i = 0$, або $b_i - g_i(x^*) = 0$. Якщо $\lambda_i \neq 0$, то $g_i(x^*) = b_i$ і обмеження є активним і являє собою обмеження у вигляді рівності. З іншого боку, якщо обмеження є обмеженням у вигляді строгої нерівності, $g_i(x^*) < b_i$, то відповідний множник Лагранжа $\lambda_i = 0$. Насправді, якщо $g_i(x^*) < b_i$, то розглядається мінімум, який задовольняє обмеженню, що є неактивним, і яким можна знехтувати, а відповідні множники $\lambda_i = 0$. Звісно, попередньо невідомо, якими обмеженнями можна знехтувати.

Є також додаткова умова, яка повинна бути виконана у точці мінімуму при наявності обмежень, а саме $\lambda_i \geq 0$.

Нехай рівняння (18.20), (18.21) і (18.23) справедливі в точці $(x^*; \lambda^*; u^*)$. Якщо фактичний мінімум функції при наявності обмежень $z = f(x^*)$, то можна розглядати z як функцію від b_i і зміна b_i буде міняти обмеження і, таким чином, змінювати саму функцію z. Покажемо, що

$$\frac{\partial z}{\partial b_i} = -\lambda_i^*, \quad \frac{\partial z}{\partial b_i} = \sum_{j=1}^{n}\frac{\partial f}{\partial x_j}\frac{\partial x_j}{\partial b_i} \qquad (18.24)$$

де часткові похідні обчислюються в точці x^*.

Оскільки $g_k(x) + u_k^2 = b_k$, то

$$\frac{\partial g_k}{\partial b_i} = \sum_{j=1}^{n}\frac{\partial g_k}{\partial x_j}\frac{\partial x_j}{\partial b_i} = \begin{cases} 0, & \text{якщо } i \neq k, \\ 1, & \text{якщо } i = k. \end{cases}$$

Тоді

$$\frac{\partial z}{\partial b_i} + \sum_{k=1}^{m}\lambda_i^*\frac{\partial g_k}{\partial b_i} = \frac{\partial z}{\partial b_i} + \lambda_i^* = \sum_{j=1}^{n}\left(\frac{\partial f}{\partial x_j} + \sum_{k=1}^{m}\lambda_k^*\frac{\partial g_k}{\partial x_j}\right)\frac{\partial x_j}{\partial b_i}.$$

Але цей вираз рівний нулю згідно з рівнянням (18.20). Таким чином,

$$\frac{\partial z}{\partial h_i} = -\lambda_i^*$$

Зі зростанням b_i область обмежень розшириться, що не може призвести до збільшення значення z - мінімуму функції $f(x)$, що знаходиться всередині області обмежень, а може лише зменшити його.

Таким чином,

$$\frac{\partial z}{\partial b_i} \le 0 \text{, тобто } \lambda_i^* \ge 0 . \quad (18.25)$$

Необхідні умови мінімуму функції $f(x)$ при наявності обмежень $g_i(x) \le b_i \quad (i = 1, 2, ..., m)$ мають такий вид, що можна знайти x і λ, для яких

$$\frac{\partial f}{\partial x_j} + \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j} = 0 \text{ при } j = 1, ..., n,$$
$$g_i(x) \le b_j \text{ при } i = 1, 2, ..., m;$$
$$\lambda[g_i(x) - b_i] = 0 \text{ при } i = 1, 2, ..., m, \quad (18.26)$$
$$\lambda_i \ge 0 \text{ при } i = 1, ..., m.$$

(Знак λ_i змінюється на протилежний, якщо розглядається максимум.)

Ці умови відомі як умови Куна – Такера.

18.4. МЕТОДИ ПРЯМОГО ПОШУКУ ДЛЯ ФУНКЦІЇ ОДНІЄЇ ЗМІННОЇ. МЕТОДИ АПРОКСИМАЦІЇ

Методи прямого пошуку є методами, у яких використовуються лише значення функції. Ідеї та складові методів прямого пошуку використовуються і при побудові складніших методів пошуку оптимальних рішень.

У більшості випадків рівняння $f'(x) = 0$ не вирішується простим способом, внаслідок своєї нелінійності. Тому розглянемо декілька простих числових процедур, що дозволяють знайти екстремум (мінімум) функції $f(x)$. За допомогою числових методів ми безпосередньо шукаємо мінімум функції $f(x)$ у деякому інтервалі $a < x < b$, у якому, як передбачається, він знаходиться, розраховуючи значення функції в обраних точках даного інтервалу.

Іноді це є єдиною можливою стратегією пошуку. Наприклад, вартість проведення хімічного процесу може залежати від температури процесу. Відомо, що вартість є функцією від T, хоча й не відомий явний вид функції. Проте можна поставити експеримент і провести процес при різноманітних температурах, визначивши ту, для якої вартість мінімальна.

Можна спробувати знайти положення мінімуму в точці, апроксимуючій його з потрібною точністю, або визначити малий інтервал, у якому знаходиться мінімум. Спробуємо досягти поставлену мету найефективнішим способом, тобто здійснюючи найменшу кількість обчислень функції. У наведеному вище прикладі, неможливо точно регулювати температуру процесу, тому точність у $1°C$ або навіть у $10°C$ є цілком задовільною. Проте, оскільки проведення експериментів потребує певних витрат, їхня кількість повинна бути мінімальною.

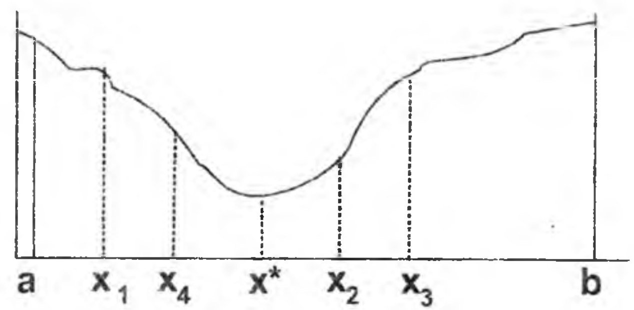


Рис. 18.2 Ілюстрація унімодальної функції

Припустимо, що точки a та b визначають (можливо, дуже грубо) інтервал, що містить дійсну точку мінімуму, і усередині цього інтервалу функція унімодальна, тобто має один мінімум у точці x^*. Отже, функція має форму, близьку до тої, що наведена на рис.18.2. Якщо відомі значення функції такого виду в трьох точках x_1, x_2, x_3, таких, що

$$a < x_1 < x_2 < x_3 < b \text{ , а } f(x_2) < f(x_1) \text{ і } f(x_2) < f(x_3) \text{, то } x_1 < x^* < x_3 .$$

Тоді точка x^* буде лежати усередині інтервалу $(x_1; x_3)$, меншого за розміром, ніж інтервал $(a; b)$.

Пошук методом Фібоначі.

Припустимо, що потрібно визначити мінімум як можна точніше, тобто з найменшим можливим інтервалом невизначеності, але при цьому можна виконати тільки n обчислень функції. Яким чином потрібно обрати n точок, у яких обчислюється функція? З першого погляду зрозуміло, що не варто шукати рішення для всіх точок, одержалих у результаті експерименту. Навпаки, треба спробувати зробити так, щоб значення функції, отримані в попередніх експериментах, визначали положення наступних точок. Дійсно, знаючи значення функції, ми тим самим маємо інформацію про саму функцію і положення її мінімуму і використовуємо цю інформацію у подальшому пошуку.

Припустимо, що є інтервал невизначеності (x_1, x_3) і відомо значення функції $f(x_2)$ усередині цього інтервалу (рис 18.2.). Якщо можна обчислити функцію усього один раз у точці x_4, тоді де потрібно помістити точку x_4, для того, щоб одержати найменший можливий інтервал невизначеності?

Нехай $x_2 - x_1 = L$ і $x_3 - x_2 = R$, причому $L > R$, як показано на рис.18.2. і ці значення будуть фіксовані, якщо відомі x_1, x_2 та x_3. Якщо x_4 знаходиться в інтервалі, то:

1. Якщо $f(x_4) < f(x_2)$, то новим інтервалом невизначеності буде $[x_1, x_2]$ довжиною $x_2 - x_1 = L$;
2. Якщо $f(x_4) > f(x_2)$, то новим інтервалом невизначеності буде $[x_4; x_3]$ довжиною $x_3 - x_4$.

Оскільки не відомо, яка з цих ситуацій буде насправді, оберемо x_4 таким чином, щоб мінімізувати найбільшу з довжин $x_3 - x_4$ і $x_2 - x_1$. Досягнути цього можна, зробивши довжини $x_3 - x_4$ і $x_2 - x_1$ рівними, тобто помістивши x_4 усередині інтервалу симетрично щодо точки x_2, що вже лежить усередині інтервалу. Будь-яке інше положення точки x_4 може призвести до того, що отриманий інтервал буде більше L. Переміщуючи x_4 симетрично щодо x_2, ми в будь-якому випадку нічим не ризикуємо.

Якщо виявиться, що можна виконати ще одне обчислення функції, то варто застосувати описану процедуру до інтервалу (x_1, x_2), у якому вже є значення функції, обчислене в точці x_4, або до інтервалу (x_4, x_3), у якому вже є значення функції, обчислене в точці x_2. Отже, стратегія зрозуміла із самого початку: потрібно розмістити таку точку у середині інтервалу невизначеності симетрично щодо точки, яка там вже знаходиться. Парадоксально, але, щоб зрозуміти, як потрібно починати обчислення, необхідно зрозуміти те, як його варто закінчувати.

На n-му обчисленні n-ну точку варто розташувати симетрично відносно $(n-1)$-ї точки. Положення цієї останньої точки в принципі залежить від нас. Для того щоб одержати найбільше зменшення інтервалу на певному

етапі, слід розділити навпіл попередній інтервал. Тоді точка x_n буде збігатися з точкою x_{n-1}. Однак при цьому ми не одержуємо ніякої нової інформації. Звичайно точки x_{n-1} і x_n розміщуються на достатній відстані одна від одної, щоб визначити, у якій половині, лівій або правій, знаходиться інтервал невизначеності. Вони розташовуються на відстані $\varepsilon/2$ з обох боків від середини відрізка L_{n-1}; можна самим задати величину ε або обрати цю величину рівною мінімально можливій відстані між двома точками. (Припустимо, що в нашому прикладі регулювання температури можливе з інтервалом у $1^\circ C$, тому $\varepsilon = 1$.)

Інтервал невизначеності буде мати довжину L_n отже, $L_{n-1} = 2L_n - \varepsilon$ (рис.18.3).

На попередньому етапі точки x_{n-1}, x_{n-2} повинні бути поміщені симетрично усередині інтервалу L_{n-2} на відстані L_{n-1} від кінців цього інтервалу. Отже, $L_{n-2} = L_{n-1} + L_n$.

З рис.18.3 ясно, що на передостанньому етапі x_{n-2} залишається в якості внутрішньої точки.

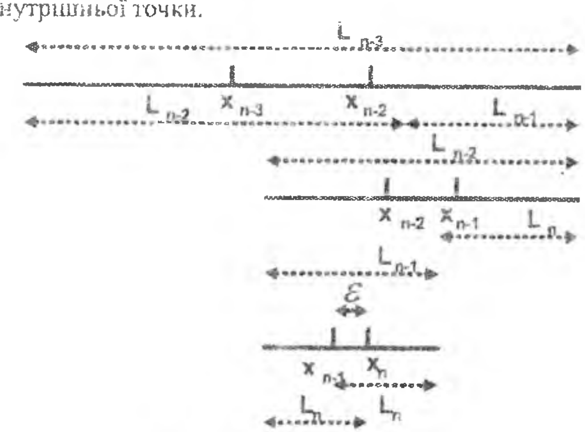


Рис. 18.3. Прямий пошук методом Фібоначі

Аналогічно $L_{n-3} = L_n + L_{n-1}$. У загальному випадку $L_{j-1} = L_j + L_{j+1}$ при $1 < j < n$.

Таким чином,

$$
\begin{aligned}
L_{n-1} &= 2L_n - \varepsilon, \\
L_{n-2} &= L_{n-1} + L_n = 3L_n - \varepsilon, \\
L_{n-3} &= L_{n-2} + L_{n-1} = 5L_n - 2\varepsilon, \\
L_{n-4} &= L_{n-3} + L_{n-2} = 8L_n - 3\varepsilon
\end{aligned}
\tag{18.27}
$$

і т.д. Якщо визначити послідовність чисел Фібоначчі наступним чином: і

$F_o = 1,\ F_1 = 1,\ F_k = F_{k-1} + F_{k-2}\ , k = 2,3,\ldots,$ то

$$L_{n-j} = F_{j+1}L_n - F_{j-1}\varepsilon\ ,\ j = 1,2,\ldots,n-1. \qquad (18.28)$$

Якщо початковий інтервал (a,b) має довжину $L_1 = (b-a)$, то

$$L_1 = F_nL_n - \varepsilon F_{n-2}\text{, тобто } L_n = \frac{L_1}{F_n} + \varepsilon\frac{F_{n-2}}{F_n}. \qquad (18.29)$$

Отже, провівши n обчислень функції, ми зменшимо початковий інтервал невизначеності в $1/F_n$ разів у порівнянні з його початковою довжиною (незважаючи на ε), і це – найкращий результат.

Якщо пошук почався, то його нескладно продовжити, використовуючи описане вище правило симетрії. Отже, необхідно знайти положення першої точки, що на відстані L_2 від одного з кінців початкового інтервалу, причому не важливо, від якого кінця, оскільки друга точка поміщається відповідно до правила симетрії на відстані L_2 від другого кінця інтервалу: (18.30)

$$L_2 = F_{n-1}L_n - \varepsilon F_{n-3} = F_{n-1}\frac{L_1}{F_n} + \varepsilon\frac{(F_{n-1}F_{n-2} - F_nF_{n-3})}{F_n} = \frac{F_{n-1}}{F_n}L_1 + \frac{(-1)^n\varepsilon}{F_n}.$$

Після того як знайдене положення першої точки, числа Фібоначчі більше не потрібні. Використовуване нами значення ε може визначатися з практичних міркувань. Воно повинно бути менше L_1/F_{n+1}, у іншому випадку ми будемо задарма витрачати час на обчислення функції.

Таким чином, пошук методом Фібоначчі є ітераційною процедурою. В процесі пошуку інтервалу (x_1, x_3) з точкою x_2, що знаходиться в цьому інтервалі, наступна точка x_4 завжди обирається такою, що $x_3 - x_4 = x_2 - x_1$ або $x_4 - x_1 = x_3 - x_2$, тобто $x_4 = x_1 - x_2 + x_3$.

Якщо $f(x_2) = f_2$ і $f(x_4) = f_4$, то існують чотири випадки (рис. 18.4).

Метод "золотого перетину".

Не завжди можна заздалегідь визначити, скільки разів потрібно обчислювати функцію. У методі Фібоначчі це потрібно знати для визначення L_2, тобто положення початкової точки .

Метод "золотого перетину" майже настільки ж ефективний, як і метод Фібоначчі, проте при цьому не потрібно знати n - кількість обчислень функції, обумовлену спочатку. Після того як виконано j обчислень,

виходячи з тих же міркувань, що і раніше, записуємо $L_{j-1} = L_j + L_{j+1}$.

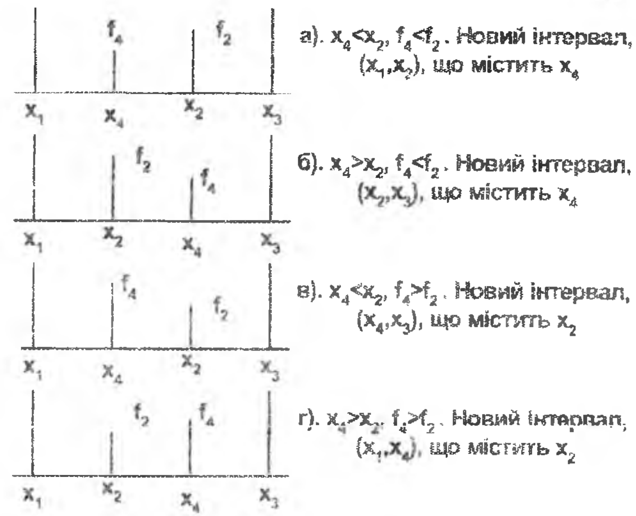


Рис. 18.4. Порядок звуження інтервалу пошуку

Проте якщо n не відомо, те ми не можемо використовувати умову $L_{n-1} = 2L_n - \varepsilon$. Якщо відношення довжин наступних інтервалів буде постійним,

$$\frac{L_{j-1}}{L_j} = \frac{L_j}{L_{j+1}} = \frac{L_{j+1}}{L_{j+2}} = \ldots = \tau\text{, то } \frac{L_{j-1}}{L_j} = 1 + \frac{L_{j+1}}{L_j}\text{, тобто } \tau = 1 + 1/\tau\ . \qquad (18.31)$$

Таким чином, $\tau^2 - \tau - 1 = 0$, звідки

$$\tau = (1+\sqrt{5})/2 \approx 1{,}618033989\,. \qquad (18.32)$$

Тоді $\dfrac{L_{j-1}}{L_{j+1}} = \tau^2$, $\dfrac{L_{j-2}}{L_{j+1}} = \tau^3$ і т.д. Отже, $\dfrac{L_1}{L_n} = \tau^{n-1}$, тобто $L_n = \dfrac{L_1}{\tau^{n-1}}$. (18.33)

В результаті аналізу двох розглянутих значень функції буде визначений той інтервал, який повинен досліджуватись надалі. Цей інтервал буде містити одну з попередніх точок і наступну точку, яку розташовуємо симетрично до неї. Перша точка знаходиться на відстані L_1/τ від одного кінця інтервалу, інша – на такій-же відстані від іншого. Оскільки $\lim_{n\to\infty} F_{n-1}/F_n = 1/n$, то з рівняння видно, що пошук методом "золотого "перетину" є граничною формою пошуку методом Фібоначчі. Назва "золотий перетин" пішла від назви відношення в рівнянні. L_{j-1} ділиться на дві частини так, що відношення цілого до більшої частини рівне відношенню більшої частини до меншої, тобто дорівнює так званому "золотому відношенню".

Таким чином, якщо шукається мінімум в інтервалі (x_0, x_3) і є два значення функції f_1 і f_2 в точках x_1 і x_2, то слід розглянути два випадки.

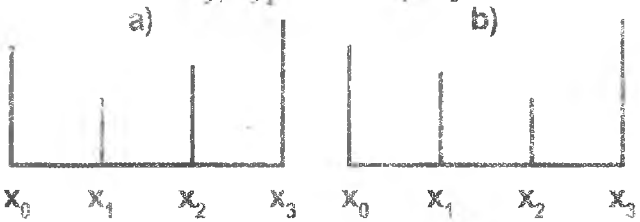


Рис. 18.5. Звуження інтервалу пошуку в методі "золотого перетину"

У випадку а) $f_1 < f_2$ новим інтервалом буде (x_0, x_2), а у випадку b) $f_1 > f_2$ - (x_1, x_3).

Методи апроксимації.

Попередні методи побудовані на ідеї пошуку малого інтервалу, у якому знаходиться екстремум (мінімум) функції. Тепер застосуємо інший підхід, що ґрунтується на використанні декількох значень функції в певних точках для апроксимації її звичайним поліномом хоча б у невеличкій області значень. Після цього положення мінімуму функції апроксимується положенням мінімуму полінома, оскільки останній обчислити простіше.

Квадратична інтерполяція.

Якщо відомі значення функції $f(x)$ у трьох різних точках α, β, γ, рівні відповідно $f_\alpha, f_\beta, f_\gamma$, то функція $f(x)$ може бути апроксимована квадратичною функцією $\varphi(x) = Ax^2 + Bx + C$, де A, B і C визначаються з рівнянь $A\alpha^2 + B\alpha + C = f_\alpha$, $A\beta^2 + B\beta + C = f_\beta$, $A\gamma^2 + B\gamma + C = f_\gamma$.

Після перетворень цих рівнянь отримаємо:

$$A = [(\gamma - \beta)f_\alpha + (\alpha - \gamma)f_\beta + (\beta - \alpha)f_\gamma]/\Delta,$$

$$B = [(\beta^2 - \gamma^2)f_\alpha + (\gamma^2 - \alpha^2)f_\beta + (\alpha^2 - \beta^2)f_\gamma]/\Delta \qquad (18.34)$$

$$C = [\beta\gamma(\gamma - \beta)f_\alpha + \gamma\alpha(\alpha - \gamma)f_\beta + \alpha\beta(\beta - \alpha)f_\gamma]/\Delta,$$

де $\Delta = (\alpha - \beta)(\beta - \gamma)(\gamma - \alpha)$. Ясно, що $\varphi(x)$ буде мати мінімум у точці

$x = -B/2A$, якщо A>0. Отже, можна апроксимувати точку мінімуму функції $f(x)$ значенням:

$$\delta = \frac{1}{2}\left[\frac{(\beta^2 - \gamma^2)f_\alpha + (\gamma^2 - \alpha^2)f_\beta + (\alpha^2 - \beta^2)f_\gamma}{(\beta - \gamma)f_\alpha + (\gamma - \alpha)f_\beta + (\alpha - \beta)f_\gamma}\right] \qquad (18.35)$$

Цей метод може безпосередньо застосовуватись до функцій однієї змінної. Він може бути дуже корисний для здійснення лінійного пошуку в процедурах. В цих процедурах потрібно знайти мінімум функції $f(x)$ в точках прямої $x_0 + \lambda d$, де x_0 - задана точка, а d визначає заданий напрямок. Значення функції $f(x_0 + \lambda d)$ на цій прямій є значеннями функції однієї змінної λ:

$$\varphi(\lambda) = f(x_0 + \lambda d).$$

Ідеї і результати, викладені вище, перетворюються в обчислювальні процедури, що описані далі. Припустимо, що задані унімодальна функція однієї змінної $f(x)$, початкова апроксимація положення мінімуму і довжина кроку H, що є величиною того ж порядку, що і відстань від точки A до точки істинного мінімуму x^* (умова, яку не завжди легко задовольнити).

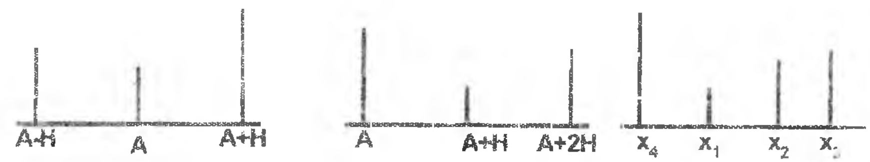


Рис 18.6. Ілюстрація обчислювальної процедури квадратичної інтерполяції

Обчислювальна процедура містить наступні кроки:

1. Обчислити $f(A)$ і $f(A+H)$.
2. Якщо $f(A) < f(A+H)$, то взяти в якості третьої точки $A-H$ і обчислити $f(A-H)$. У протилежному випадку в якості третьої точки взяти $A+2H$ і знайти $f(A+2H)$ (рис. 18.6).
3. Використовуючи координату цих трьох точки, знайти δ із рівняння і обчислити $f(\delta)$.
4. Якщо різниця між найменшим значенням функції і наступним найменшим значенням функції менше заданої точності, то процедура завершується.

Якщо процедура не завершилася на кроці 4, то точка з найбільшим значенням, звичайно, відкидається, і ми повертаємося на крок 3. Але якщо, залишивши точку з найбільшим значенням функції, ми визначимо кінцеві границі інтервалу, у якому лежить мінімум, то слід дійсно залишити це значення і потім повернутися на крок 3.

Наприклад, на рис.18.6 залишені точки x_1, x_2 і x_4, а не точки x_1, x_2 і x_3.

Зауважимо, що якщо точність Е задана занадто малою, то α, β, γ а також $f_\alpha, f_\beta, f_\gamma$ будуть дуже близькі один до одного і значення δ (див. рівняння) може стати взагалі безмежним. Щоб перебороти цю проблему, перепишемо рівняння для другої і наступних інтерполяцій:

$$\delta = \frac{1}{2}(\alpha+\beta) + \frac{\frac{1}{2}(f_\alpha - f_\beta)(\beta-\gamma)(\gamma-\alpha)}{(\beta-\gamma)f_\alpha + (\gamma-\alpha)f_\beta + (\alpha-\beta)f_\gamma}. \qquad (18.36)$$

18.5. МЕТОДИ ПРЯМОГО ПОШУКУ ДЛЯ ФУНКЦІЙ N ЗМІННИХ

На розробку методів прямого пошуку для визначення мінімуму функцій n змінних було витрачено багато зусиль. Методи прямого пошуку є методами, у яких використовуються лише значення функції.

Ми розглянемо докладно два з них, які ефективні і застосовуються в багатьох програмах.

Розглянемо функцію двох змінних. Її лінії постійного рівня подані на рис. 18.7, а мінімум знаходиться у точці (x_1^*, x_2^*). Лінією сталого рівня називається крива в двовимірному січенні простору параметрів (в даному випадку - в площині (x_1, x_2)), значення функції на якій - константа.

Для перевірки працездатності методів запропоновано декілька функцій, що внаслідок своїх властивостей є достатньо "незручними", а тому тестовими для методів оптимізації. Нижче наведено декілька таких функцій.

Функція Розенброка:

$$f(x_1, x_2) = 100\ (x_2 - x_1^2)^2 + (1-x_1)^2;\ x^* = (1;1). \qquad (18.37)$$

Функція Павела:

$$f(x) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4;$$

$$x^* = (0;0;0;0). \qquad (18.38)$$

Двовимірна експоненційна функція:

$$f(x_1, x_2) = \sum_a [(e^{-ax_1} - e^{-ax_2}) - (e^{-a} - e^{-10a})]^2, \qquad (18.39)$$

де $a = 0,1\ (0,1)\ 1^*;\ x^* = (1;10)$.

Будь-яка серйозна оптимізаційна процедура повинна ефективно вирішувати ці та інші тестові задачі.

Метод покоординатного спуску.

Найпростішим методом пошуку є метод покоординатного спуску. З точки A ми здійснюємо пошук мінімуму уздовж напрямку осі x_1 і, таким чином, знаходимо точку В, у якої дотична до лінії постійного рівня паралельна осі x_1. Потім, здійснюючи пошук із точки B в напрямку осі x_2, отримуємо точку C, здійснюючи пошук паралельно осі x_1, одержуємо точку D, і т. д.

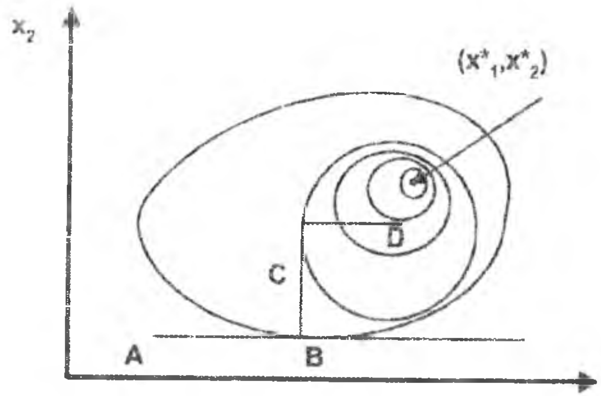


Рис. 18.7. Траєкторія руху при реалізації методу покоординатного спуску

Таким чином, ми приходимо до оптимальної точки. Будь-який з одновимірних методів, описаних у попередньому розділі, може бути використаний для пошуку екстремуму вздовж осі. Очевидно, що цю ідею можна застосувати для функцій n змінних.

Теоретично даний метод ефективний у випадку єдиного мінімуму функції. Але на практиці він здається занадто повільним. Тому були розроблені складніші методи, що використовують більше інформації на підставі вже отриманих значень функції.

Метод Хука – Дживса.

Цей метод був розроблений у 1961 році, але і дотепер він є дуже ефективним. Пошук складається з послідовності кроків дослідження навколо базової точки, за яким у випадку успіху реалізується пошук за зразком.

Алгоритм методу Хука – Дживса утворює наступну послідовність кроків:

Крок 1. Обрати початкову базову точку b_1 і крок довжиною h_j для кожної змінної x_j, $j = 1, 2, ..., n$. У деяких програмах для кожної змінної використовується крок свій h, проте зазначена вище модифікація теж може виявитися корисною.

Крок 2. Обчислити $f(x)$ у базовій точці b_1 з метою одержання даних про локальну поведінку функції $f(x)$. Ці дані будуть використані для знаходження найвідповіднішого напрямку пошуку за зразком, за допомогою якого можна сподіватися досягти найзначнішого зменшення значення функції.

Функція $f(x)$ у базовій точці b_1 знаходиться наступним чином:

2.1. Обчислюється значення функції $f(b_1)$ у базовій точці b_1.

2.2. Значення кожної змінної за чергою змінюється додаванням довжини кроку.

Таким чином, ми обчислюємо значення функції $f(b_1 + h_1e_1)$, де e_1 - одиничний вектор у напрямку осі x_1. Якщо це призводить до зменшення значення функції, то b_1 заміняється на $b_1 + h_1e_1$. У протилежному випадку обчислюється значення функції $f(b_1 - h_1e_1)$, і якщо її значення зменшилось, то b_1 заміняємо на $b_1 - h_1e_1$. Якщо ні один із пророблених кроків не викликає зменшення значення функції, то точка b_1 залишається незмінною і розглядаються зміни в напрямку осі x_2, тобто знаходиться значення функції $f(b_1 + h_2e_2)$ і т.д. Коли будуть розглянуті усі n змінних, отримається нова базова точка b_2.

2.3. Якщо $b_2 = b_1$, тобто зменшення функції не було досягнуто, таке ж дослідження повторюється навколо тієї ж базової точки b_1, але зі зменшеною довжиною кроку. На практиці задовільним є зменшення кроку (кроків) у десять разів від початкової довжини.

2.4. Якщо $b_2 \neq b_1$ то виконується пошук за зразком.

Крок 3. При пошуку за зразком використовується інформація, отримана в процесі дослідження, і мінімізація функції завершується пошуком у напрямку, заданому зразком. Ця процедура здійснюється в такий спосіб:

3.1. Доцільно рухатися з базової точки b_2 в напрямку $b_2 - b_1$, оскільки пошук у цьому напрямку вже привів до зменшення значення функції. Тому обчислимо функцію в точці зразка $P_1 = b_1 + 2(b_2 - b_1)$. У загальному випадку $P_i = b_i + 2(b_{i+1} - b_i)$.

3.2. Далі дослідження продовжуємо навколо точки $P_1(P_i)$.

3.3. Якщо найменше значення на кроці 3.2 менше значення в базовій точці b_2 (у загальному випадку b_{i+1}), то одержуємо нову базову точку b_3, (b_{i+2}), після чого повторюємо крок 3.1. У протилежному випадку пошук за зразком із точки $b_2(b_{i+1})$ не здійснюємо, а продовжуємо дослідження в точці $b_2(b_{i+1})$.

Крок 4. Завершуємо процес, коли довжина кроку (довжини кроків) буде зменшена до заданого малого значення.

Таким чином, використання ідеї прямого пошуку дозволяє конструювати достатньо ефективні алгоритми пошуку оптимальних рішень за допомогою числових методів.

18.6. МЕТОД ДЕФОРМОВАНОГО МНОГОГРАННИКА

Метод Нелдера - Міда (або пошук по деформованому многограннику) є розвитком симплексного методу Спендлі, Хекста і Хімсворта. Як відомо, множина $(n+1)$ рівновіддалених точок в n-вимірному просторі називається регулярним симплексом. Отже, у двовимірному просторі симплексом є рівносторонній трикутник, а в тривимірному просторі— правильний тетраедр. Ідея методу полягає в порівнянні значень функції в $(n+1)$ вершинах симплекса і переміщенні симплекса в напрямку оптимальної точки за допомогою ітераційної процедури. У симплексному методі, запропонованому спочатку, регулярний симплекс використовувався на кожному етапі. Нелдер і Мід запропонували декілька модифікацій цього методу з неправильними симплексами. У результаті утворився дуже надійний метод прямого пошуку, що є одним із найефективніших, якщо $n < 6$.

У методі Спендлі, Хекста і Хімсворта симплекс переміщається за допомогою трьох основних операцій: відбивання, розтягання і стискання. Зміст цих операцій стане зрозумілим при розгляді кроків процедури.

Алгоритм методу Нелдера-Міда.

Крок 1. Обчислюємо значення функції:

$f_1 = f(x_1), f_2 = f(x_2) ... f_{n+1} = f(x_{n+1})$ у вершинах симплекса.

Крок 2. Шукаємо найбільше значення функції f_h, наступне за найбільшим значенням функції f_g, найменше значення функції f_l і відповідні їм точки x_h, x_g і x_l.

Крок 3. Розраховуємо координати центру ваги всіх точок, за винятком точки x_h. Нехай центром ваги буде $x_0 = \frac{1}{n}\sum_{i \neq h} x_i$ і обчислюємо $f(x_0) = f_0$.

Крок 4. Початок переміщення. Зручніше всього почати переміщення від точки x_h. Відбиваючи точку x_h відносно точки x_0, одержимо точку x_r і знайдемо $f(x_r) = f_r$.

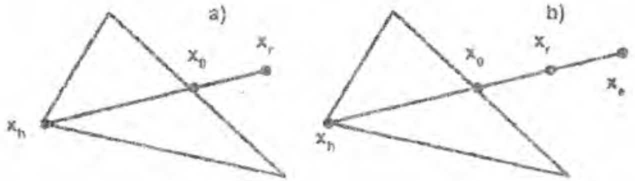


Рис. 18.8. Операції відбивання (a) та розтягання (b)

Операція відбивання ілюструється рис.18.8а. Якщо $\alpha > 0$ - коефіцієнт відбивання, то положення точки x_r визначається в такий спосіб: $x_r - x_0 = \alpha(x_0 - x_h)$, тобто $x_r = (1+\alpha)x_0 - \alpha x_h$.

Крок 5. Порівнюємо значення функцій f_r і f_l: $f_r < f_l$?

5.1. Якщо $f_r < f_l$, то ми одержали найменше значення функції. Напрям з точки x_0 в точку x_r є найзручнішим для переміщення. Таким чином, ми робимо розтяг у цьому напрямку, знаходимо точку x_e і значення функції $f_e = f(x_e)$. Рисунок рис.18.8b ілюструє операцію розтягання симплексу. Коефіцієнт розтягання $\gamma > 1$ визначається з наступних співвідношень:

$x_e - x_0 = \gamma(x_r - x_0)$,тобто $x_e = \gamma x_r + (1-\gamma)x_0$,де

$\gamma = |x_e - x_0| / |x_r - x_0|$

Перевірка умови: $f_e < f_l$?

а) Якщо $f_e < f_l$ то заміняємо точку x_h на точку x_e і перевіряємо $(n+1)$ -у точку симплекса на збіжність до мінімуму (19.40). Якщо збіжність досягнута, то процес зупиняється; у протилежному випадку повертаємося на крок 2.

б) Якщо $f_e \geq f_l$ то відкидаємо точку x_e. Очевидно, ми перемістилися занадто далеко від точки x_0 до точки x_e. Тому варто замінити точку x_h на точку x_r, у якій було отримане покращення (крок 5.1), перевірити збіжність і, якщо вона не досягнута, повернутися на крок 2.

5.2. Якщо $f_r > f_l$, але $f_r \leq f_g$, то x_r є кращою точкою в порівнянні з іншими двома точками симплекса, заміняємо точку x_h, на точку x_r і, якщо збіжність не досягнута, повертаємося на крок 2, тобто виконуємо пункт 5.1б, описаний вище.

3. Якщо $f_r > f_l$ і $f_r > f_g$, то перейдемо на крок 6.

Крок 6. Порівняємо значення функцій f_r і f_h.

6.1. Якщо $f_r > f_h$, то переходимо безпосередньо до кроку стиску 6.2.

Якщо $f_r < f_h$, то заміняємо точку x_h на точку x_r і значення функції f_h на значення функції f_r. Запам'ятовуємо значення $f_r > f_g$ із кроку 5.2. наведеного вище. Потім переходимо на крок 6.2.

6.2. У цьому випадку $f_r > f_h$, тому ясно, що ми перемістилися занадто далеко від точки x_h до точки x_0. Спробуємо виправити це, знайшовши точку x_c (а потім f_c) за допомогою кроку стиснення, показаного на рис. 18.9.

Якщо $f_r > f_h$, то відразу переходимо до кроку стиснення і знаходимо точку x_c із співвідношення $x_c - x_0 = \beta(x_h - x_0)$, де β $(0 < \beta < 1)$ - коефіцієнт стискання. Тоді $x_c = \beta x_h + (1-\beta)x_0$.

Якщо $f_r < f_h$, то спочатку замінимо точку x_h на точку x_r, а потім виконаємо стискання. Тоді точку x_c знайдемо із співвідношення $x_c - x_0 = \beta(x_r - x_0)$, тобто $x_c = \beta x_r + (1-\beta)x_0$ рис. 18.9b .

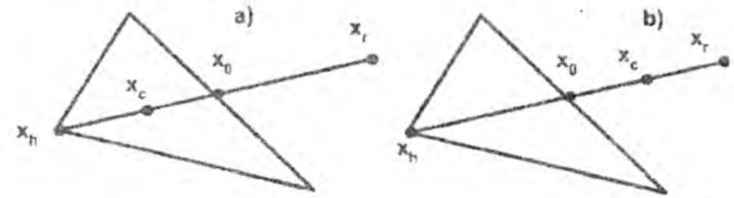


Рис. 18.9. Операції корегування (a) та стиснення (b)

Крок 7. Порівняємо значення функцій f_c і f_h.

1. Якщо $f_c < f_h$, то заміняємо точку x_h на точку x_c, і якщо збіжність не досягнута, то повертаємося на крок 2.

2. Якщо $f_c > f_h$, то всі спроби знайти значення менше f_h закінчилися невдачею, тому ми переходимо до кроку 8

Крок 8. Зменшуємо розмірність симплекса діленням навпіл віддалі від кожної точки симплекса до x_l - точки, що визначає найменше значення функції

Таким чином, точка x_i заміняється точкою $x_i + \frac{1}{2}(x_i - x_l)$, тобто змінюємо точку x_i точкою $\frac{1}{2}(x_i + x_l)$.

Потім обчислюємо f_i для $i = 1, 2, ..., (n+1)$, перевіряємо збіжність і, якщо вона не досягнута, повертаємося на крок 2.

Перевірка збіжності.

Перевірка збіжності ґрунтується на тому, щоб стандартне відхилення $(n+1)$-го значення функції було менше деякого заданого малого значення ε.

У цьому випадку обчислюється: $\sigma^2 = \sum_{i=1}^{n+1}(f_i - \bar{f})^2/(n+1)$, де

$$\bar{f} = \sum_{i=1}^{n+1} f_i / n+1 . \qquad (19.40)$$

Якщо $\sigma < \varepsilon$, то всі значення функції дуже близькі одне до одного, і тому вони, можливо, лежать поблизу точки мінімуму функції x_l. Виходячи з цього, такий критерій збіжності є розумним, хоча існують і інші критерії („безпечна" перевірка Бокса, Девіса і Свена).

Коефіцієнти α, β, γ у вищенаведеній процедурі є відповідно коефіцієнтами відображення, стискання і розтягання. Нелдер і Мід рекомендують брати $\alpha = 1$, $\beta = 0.5$, $\gamma = 2$. Рекомендація ґрунтується на результатах експериментів із різноманітними комбінаціями значень. Ці значення параметрів дозволяють методу бути ефективним і працювати в різноманітних складних ситуаціях.

Початковий симплекс вибирається довільно і може формуватися наступним чином. Наприклад, точка x_1 є початковою точкою, потім формуються точки $x_2 = x_1 + ke_1$, $x_3 = x_1 + ke_2$, $k_{n+1} = x_1 + ke_n$, де k - довільна довжина кроку, а e_i - одиничний вектор.

Характер просування при пошуку оптимуму наведений на рис. 18.10.

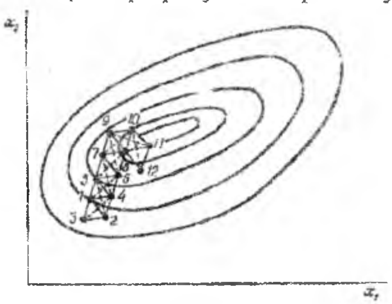


Рис. 18.10. Пошук оптимуму методом деформованого многогранника

Кроки методу Нелдера-Міда подані у вигляді блок-схеми на рис. 18.11.

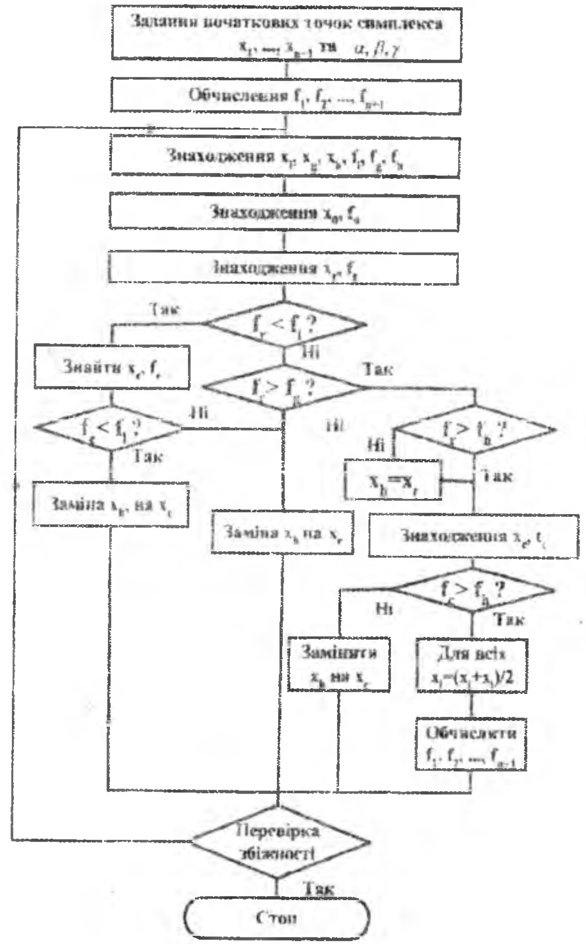


Рис. 18.11. Блок-схема алгоритму Нелдера-Міда

ПРИКЛАДИ

Приклад 18.1. Знаходження оптимуму за допомогою методу множників Лагранжа.

Знайти мінімум функції $f(x,y)=x^2+y^2$ при обмеженні $x+y=4$.

Розв'язання.

Функція Лагранжа має вигляд: $F(x,y,\lambda)=x^2+y^2+\lambda(4-x-y)$.

Відповідні умови мінімуму можна записати наступним чином:

$$\frac{\partial F}{\partial x}=2x-\lambda=0,\ \frac{\partial F}{\partial y}=2y-\lambda=0,\ \frac{\partial F}{\partial \lambda}=4-x-y=0.$$

Розв'язком цієї системи рівнянь є $x=y=2,\ \lambda=4$. Мінімум функції рівний 8.

Приклад 18.2.

Перевірте, що точка $(2,2)$ є мінімумом функції $f(x)=x_1^2+x_2^2$ при обмеженні $x_1+x_2=4$.

Розв'язання.

Для $F(x,\lambda)=x_1^2+x_2^2-\lambda(4-x_1-x_2)$ були отримані результати:

$$\frac{\partial F}{\partial x_1}=0,\ \frac{\partial F}{\partial x_2}=0,\ \frac{\partial F}{\partial \lambda}=0 \text{ при } x_1=x_2=2 \text{ і } \lambda=4.$$

Матриця Гессе функції F має вигляд $\begin{bmatrix}2 & 0\\ 0 & 2\end{bmatrix}$ і, отже, позитивно визначена, а це доводить, що точка $(2;2)$ є точкою мінімуму.

Приклад 18.3. Умови Куна-Такера.

Написати умови Куна – Такера для мінімуму функції $f(x)=3x_1^2+4x_1x_2+5x_2^2$ при обмеженнях $x_1\ge 0,\ x_2\ge 0$ і $x_1+x_2\ge 4$.

Розв'язання.

Цю задачу можна представити наступним чином:

мінімізувати функцію $f(x)=3x_1^2+4x_1x_2+5x_2^2$

при обмеженнях $-x_1\le 0,\ -x_2\le 0,\ -x_1-x_2\le -4$. Функція Лагранжа $F(x,u,\lambda)$ буде мати вигляд:

$$F=3x_1^2+4x_1x_2+5x_2^2+\lambda_1(u_1^2-x_1)+\lambda_2(u_2^2-x_2)+\lambda_3(u_3^2-x_1-x_2+4)$$

Необхідними умовами мінімуму є:

$6x_1+4x_2-\lambda_1-\lambda_3=0,\ 4x_1+10x_2-\lambda_2-\lambda_3=0,$

$-x_1\le 0,\ -x_2\le 0,\ -x_1-x_2\le -4,\ \lambda_1x_1=0,\ \lambda_2x_2=0,\ \lambda_3(4-x_1-x_2)=0,$

$\lambda,\ \lambda_2,\ \lambda_3\ge 0.$

Неважко перевірити, що ці умови виконуються при $x_1=3,\ x_2=1,\ \lambda_1=0,\ \lambda_1=0,\ \lambda_3=22$ і функція має мінімум, рівний 44 в точці А з координатами (3;1) (див.рис. 18.8).

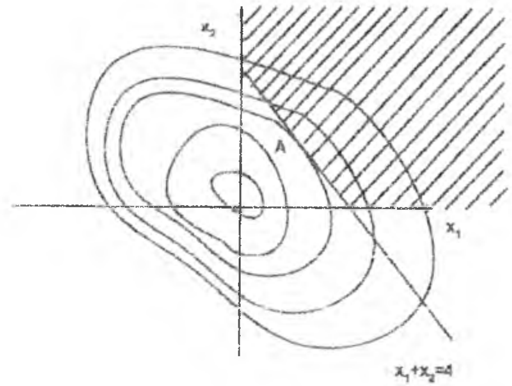


Рис. 18.8. Ілюстрація умов Куна-Такера

Лініями постійного рівня функції $f(x)$ є еліпси $3x_1^2+4x_1x_2+5x_2^2=c$.

Мінімум функції $f(x)$ при відсутності обмежень рівний нулю і знаходиться в початку координат. Область обмежень показана тіньовим контуром на рис. 18.8, що ілюструє задачу.

РЕЗЮМЕ

18.1.Для задач оптимізації функцій однієї змінної без обмежень умови екстремуму задаються у вигляді рівності похідної від функції нулеві, а характер екстремуму (мінімум, максимум, точка перегину) визначається знаком другої похідної. У випадку функцій n змінних у результаті прирівнювання часткових похідних до нуля виникає система нелінійних рівнянь, а характер екстремуму визначається негативною чи позитивною визначеністю матриці Гессе в точці екстремуму. За будь-яких умов аналітичне розв'язання цих задач можливе лише в найпростіших випадках.

18.2. У випадку пошуку екстремумів функції n змінних з обмеженнями *у вигляді рівнянь, первинна функція заміняється функцією Лагранжа, і система рівнянь отримується шляхом знаходження часткових похідних від змінних та множників Лагранжа і прирівнюванням їх до нуля.*

18.3. Умови Куна-Такера є розширенням ідей методу множників Лагранжа на задачі з обмеженнями загального вигляду – як у вигляді рівнянь, так і у вигляді нерівностей. Ці умови є необхідними умовами, які повинні виконуватись в стаціонарній точці. Розв'язання отриманої системи рівнянь та обмежень можливе лише для найпростіших задач, і в загальному для розв'язання використовуються чисельні методи.

18.4. Методи прямого пошуку є методами, у яких використовуються лише значення функції і орієнтовані на пошук екстремумів унімодальних функцій. Ідеї та складові методів прямого пошуку використовуються і при побудові складніших методів пошуку оптимальних рішень. Найефективнішим є метод, що використовує числа Фібоначі, однак з точки зору зручності реалізації доцільно використовувати метод золотого перетину, який незначно поступається методу Фібоначі. Окрім цих методів, використовуються також апроксимаційні, що використовують ідею квадратичної інтерполяції.

18.5. Методи прямого пошуку застосовуються також і для пошуку екстремумів функцій n змінних. Найпростішим з них є метод покоординатного спуску (у випадку пошуку мінімуму), який почергово по кожній зі змінних реалізує одновимірний прямий пошук, і в якості процедури одновимірного пошуку найчастіше використовується метод золотого перетину. Однак з практичної точки зору цей метод є доволі повільним, а тому були розроблені ефективніші методи, одним з найвідоміших серед яких є метод Хука-Джнівса. Цей метод використовує інформацію про локальну поведінку функції для обрання найвідповіднішого напрямку пошуку за зразком.

18.6. Метод Нелдера - Міда (або пошук по деформованому многограннику) є розвитком симплексного методу. Ідея методу полягає в порівнянні значень функції в $(n+1)$ вершинах симплекса і переміщенні симплекса в напрямку оптимальної точки за допомогою ітераційної процедури. Нелдер і Мід запропонували декілька модифікацій цього методу з неправильними симплексами. У результаті утворився дуже надійний метод прямого пошуку, що є одним із найефективніших, якщо $n<6$.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 18.1. Метод множників Лагранжа.

Знайти найменше та найбільше значення функції:

$$f(x)=9x_1^2+4x_2^2+x_3^2-\left(3x_1^2+2x_2^2+x_3^2\right)^2$$

при обмеженнях $x_1^2+x_2^2+x_3^2=1$.

Завдання 18.2. Умови Куна – Такера.

Випишіть умови Куна – Такера для задачі:

$$f(x)=2x_1+4x_2-x_1^2-2x_2^2 \Rightarrow Max$$

$$\begin{cases} x_1+2x_2\le 8, \\ 2x_1-x_2\le 12, \end{cases}$$

$x_1, x_2 \ge 0$.

Завдання 18.3. Прямі методи одновимірного пошуку.

Знайдіть координати інтервалу невизначеності за умови виконання 5-ти кроків за методом Фібоначі та методом "золотого перетину" і порівняйте ці інтервали між собою для наступної задачі:

$f(x)=x^2+2x$ за умови $-3\le x\le 5$.

Завдання 18.4. Прямі методи багатовимірного пошуку.

Розв'язати за методом Хука-Джівса з дискретним кроком задачу $(x_1-2)+(x_1-2x_2)^2 \Rightarrow Min$.

Значення координат початкової точки $x_1=2,\ x_2=3$, параметрів методу становлять $\alpha=1.0,\ h=0.2$. Виконати 4 ітерації методу та визначити координати точки оптимуму після цих ітерацій.

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Сформулюйте необхідні умови екстремуму для функції n змінних без обмежень.
2. З якою метою використовується матриця Гессе?
3. Для розв'язання яких задач використовується метод множників Лагранжа?
4. Запишіть вигляд функції Лагранжа.
5. Сформулюйте необхідні умови існування екстремуму для задачі з обмеженнями у вигляді рівнянь.
6. Який сенс мають умови Куна-Такера?
7. Яку інформацію використовують методи прямого пошуку?
8. Яка функція називається унімодальною?
9. В чому суть методу Фібоначі?
10. Які переваги методу "золотого перетину" порівняно з методом Фібоначі?
11. З якою метою використується квадратична інтерполяція?
12. Яким чином працює метод покоординатного спуску?
13. За рахунок чого підвищується ефективність методу Хука-Дживса?
14. Сформулюйте послідовність кроків методу Хука-Дживса.

ТЕМА 19. МЕТОДИ, ЩО ВИКОРИСТОВУЮТЬ ІНФОРМАЦІЮ ПРО НАПРЯМОК ПОШУКУ

Ідея градієнтних методів базується на тому, що напрямок градієнта є напрямком найшвидшого зростання функції. Отже, протилежний напрямок є напрямком найшвидшого спадання функції. Напрямок градієнта перпендикулярний в будь-якій точці лінії постійного рівня, оскільки уздовж цієї лінії значення функції є сталим. В методі найшвидшого спуску пошук мінімуму функції здійснюється вздовж напрямку, оберненого до напрямку градієнту. Як критерій завершення обчислень, найчастіше використовується близькість до нуля градієнту. Основною складністю застосування методу найшвидшого спуску є залежність його від вибору масштабу змінних, що оптимізуються. В методах, що використовують другі похідні, використовується матриця Гессе, яку необхідно інвертувати на кожному кроці, як в методі Ньютона-Рафсона. Оскільки ця процедура може займати основний час обчислень, використовуються наближення до неї, і позитивно визначена симетрична матриця у остаточному рахунку при виконанні певних умов стає рівною асимптотично матриці Гессе, як у методі Давідсона – Флетчера – Павела. Основна ідея методу штрафів для розв'язання задачі оптимізації з обмеженнями полягає у перетворенні задачі мінімізації функції з обмеженнями у задачу пошуку мінімуму без обмежень модифікованої функції, в якій при порушенні обмежень додаткова складова, сформована на основі обмежень, "штрафувала" б модифіковану функцію, збільшуючи її значення. На цій ідеї побудовані метод Фіако-Маккорміка, який у якості кроку алгоритму використовує метод Давідсона-Флетчера-Павела, та метод Бокса, який, по суті, є модифікацією симплексного методу Нелдера-Міда, що дозволяє враховувати обмеження.

ПІСЛЯ ВИВЧЕННЯ ТЕМИ ВИ ПОВИННІ:

знати: ⇨суть методів, що використовують інформацію про градієнт функції; ⇨переваги та недоліки використання квадратичних функцій; ⇨принципи оптимізації з обмеженнями; ⇨алгоритми методів Давідсона - Флетчера – Павела, Фіако-Маккорміка, Бокса;

вміти: ⇨розв'язувати задачі чисельної оптимізації без обмежень за допомогою методу найшвидшого спуску, Давідсона - Флетчера – Павела; ⇨здійснювати формальну постановку задачі оптимізації з обмеженнями:

⇨ застосовувати практично методи пошуку оптимальних рішень для задач з обмеженнями.

КЛЮЧОВІ ПОНЯТТЯ ТА ТЕРМІНИ

- ☑ градієнт
- ☑ матриця Гессе
- ☑ комплексний метод
- ☑ лінія постійного рівня
- ☑ спряжені напрямки
- ☑ симплекс
- ☑ віддаль між точками комплексу
- ☑ метод Давідсона - Флетчера – Павела
- ☑ припустима початкова точка
- ☑ умови збіжності
- ☑ штрафна функція
- ☑ комплекс
- ☑ гіперпростір
- ☑ спадна послідовність
- ☑ ітерація
- ☑ квадратична функція
- ☑ метод Фіако-Маккорміка
- ☑ центр ваги множини точок

ПЛАН ВИКЛАДЕННЯ ТА ЗАСВОЄННЯ МАТЕРІАЛУ

19.1. Метод найшвидшого спуску.
19.2. Методи, що використовують другі похідні.
19.2.1. Квадратичні функції.
19.2.2. Метод Давідсона - Флетчера - Павела.
19.3. Оптимізація з обмеженнями.
19.3.1. Постановка задачі та штрафні функції.
19.3.2. Метод Фіако і Маккорміка.
19.3.3. Метод Бокса.

19.1. МЕТОД НАЙШВИДШОГО СПУСКУ

Розглянемо методи пошуку, у яких поряд із значеннями функції використовується і її градієнт, тобто інформація про напрямок пошуку. За допомогою методу покоординатного спуску здійснюється пошук з заданої точки в напрямку, паралельному одній з осей, до точки мінімуму в цьому напрямку. Потім пошук проводиться в напрямку, паралельному іншій осі і т.д. Напрямки, звичайно, фіксовані. Здається розумним спробувати модифікувати цей метод таким чином, щоб на кожному етапі пошук точки

мінімуму здійснювався вздовж „найкращого" напрямку. Не зрозуміло, який напрямок є „найкращим", але відомо, що напрямок градієнту є напрямком найшвидшого зростання функції. Отже, протилежний напрямок є напрямком найшвидшого спадання функції.

Ця властивість обгрунтовується наступним чином. Припустимо, що здійснюється переміщення з точки x у наступну точку $x+hd$, де d - певний напрямок, а h - крок певної довжини. Отже, переміщення здійснюється з точки $(x_1, x_2, ..., x_n)$ у точку

$$(x_1 + \delta x_1, x_2 + \delta x_2, ..., x_n + \delta x_n), \text{ де } \delta x_i = hd_i, \tag{19.1}$$

а d_i - косинуси напрямку d , такі, що

$$\sum_{i=1}^{n} d_i^2 = 1. \tag{19.2}$$

Зміна значень функції визначається співвідношеннями:

$$df = f(x_1 + \delta x_1, x_2 + \delta x_2, ..., x_n + \delta x_n) - f(x_1, x_2, ..., x_n) =$$
$$= \frac{\partial f}{\partial x_1}\delta x_1 + \frac{\partial f}{\partial x_2}\delta x_2 + ... + \frac{\partial f}{\partial x_n}\delta x_n \tag{19.3}$$

з точністю до першого порядку δx_i , а часткові похідні обчислюються в точці x. Яким чином слід обрати напрямки d_i , що відповідають умові (19.2), щоб одержати найбільше значення зміни функції df ?

У цьому випадку виникає задача максимізації з обмеженням, і для її розв'язання використаємо метод множників Лагранжа, за допомогою якого визначимо функцію

$$\varphi(d_1, d_2, ..., d_n) = df + \lambda(\sum d_i^2 - 1). \tag{19.4}$$

Величина df , що задовільняє обмеження (19.2), досягає максимуму, коли функція

$$\varphi(d_1, d_2, ..., d_n) = h(\frac{\partial f}{\partial x_1}d_1 + \frac{\partial f}{\partial x_2}d_2 + ... + \frac{\partial f}{\partial x_n}d_n) + \lambda(d_1^2 + d_2^2 + ... + d_n^2 - 1)$$

досягає максимуму. Її похідна

$$\frac{\partial \varphi}{\partial d_j} = h\frac{\partial f}{\partial x_j} + 2\lambda d_j, \text{ при } j = 1, 2, ..., n. \tag{19.5}$$

Якщо $\frac{\partial \varphi}{\partial d_j} = 0$, то $d_j = -\frac{h}{2\lambda}\frac{\partial f}{\partial x_j}$. (19.6)

Отже, $\frac{d_1}{\frac{\partial f}{\partial x_1}} = \frac{d_2}{\frac{\partial f}{\partial x_2}} = ... = \frac{d_n}{\frac{\partial f}{\partial x_n}}$. (19.7)

Тоді $d_i \sim \partial f / \partial x_i$ і напрямок d паралельний напрямку $\nabla f(x)$ у точці x.

Таким чином, найбільше локальне зростання функції для заданого малого кроку h буде тоді, коли d є напрямком $\nabla f(x)$ або $g(x)$. Тому напрямком найшвидшого спуску є напрямок

$$-\nabla f(x) \text{ або } -g(x). \qquad (19.8)$$

У простішому вигляді рівняння (19.3) записується наступним чином: $df = |\nabla f(x)|\,|dx|\cos\theta$, де θ - кут між векторами $\nabla f(x)$ і dx. Для заданого значення dx ми мінімізуємо df, обираючи $\theta = 180^\circ$, щоб напрямок dx збігався з напрямком $-\nabla f(x)$.

Напрямок градієнту перпендикулярний в будь-якій точці лінії постійного рівня, оскільки уздовж цієї лінії значення функції є сталим. Таким чином, якщо $(d_1, d_2, \dots, d_n)$ - малий крок уздовж лінії рівня, то

$$f(x_1 + d_1, x_2 + d_2, \dots, x_n + d_n) = f(x_1, x_2, \dots, x_n)$$

і, отже, $df = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} d_i = [\nabla f(x)]^T d = 0$ (рис. 19.1). (19.9)

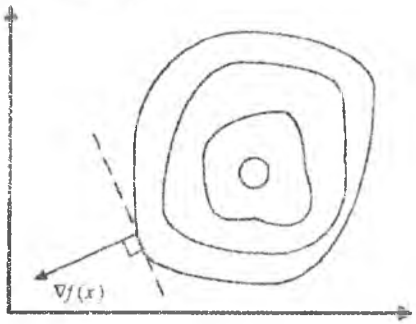


Рис. 19.1. Напрямок градієнта

В методі найшвидшого спуску використовується ця властивість напрямку градієнта. Тому, якщо ми знаходимося в точці $x^{(i)}$ на деякому кроці процесу оптимізації, то пошук мінімуму функції здійснюється вздовж напрямку $-\nabla f(x^{(i)})$.

Метод є ітераційним. На кроці i точка мінімуму апроксимується точкою $x^{(i)}$. Наступною апроксимацією є точка

$$x^{(i+1)} = x^{(i)} - \lambda_i \nabla f(x^{(i)}), \qquad (19.10)$$

де λ_i - значення λ, що мінімізує функцію

$$\varphi(\lambda) = f[x^{(i)} - \lambda \nabla f(x^{(i)})]. \qquad (19.11)$$

Значення λ_i може бути знайдене за допомогою одного з методів одновимірного пошуку, що описані у попередній темі.

Окрім того, в простішому випадку крок λ_i може обиратися достатньо великим, і якщо при здійсненні переходу до наступного значення функції порушуватиметься умова

$$f\left(x^{(i+1)}\right) < f\left(x^{(i)}\right), \qquad (19.12)$$

величина кроку зменшується (наприклад в два рази – $\lambda_i = \lambda_i / 2$).

При обранні закону зміни λ_i слід дотримуватися виконання наступних умов: $\lim_{k\to\infty} \sum_{i=0}^{k} \lambda_i = \infty$ - ця умова забезпечує збіжність власне до точки оптимуму, тому що в іншому випадку ітераційний процес може не досягнути оптимуму;

$\lim_{i\to\infty} \lambda_i = \infty$ - ця умова сприяє зменшенню коливань біля точки оптимуму і швидшому досягненню оптимуму з заданою точністю.

Як критерій завершення обчислень, найчастіше використовується близькість до нуля градієнту $\nabla f(x_{(i)})$, $\left|\frac{df\left(x^{(k)}\right)}{dx_i}\right| \le \varepsilon$, $i = 1, \dots, n$, або

$\left\|f'\left(x^{(k)}\right)\right\| = \sqrt{\sum_{i=1}^{n} \left(\frac{df\left(x^{(k)}\right)}{dx_i}\right)^2} \le \varepsilon$, де ε - достатньо мале задане додатне число, після чого вважають, що оптимум досягнутий в точці $x^{(k)}$, $f^* = f\left(x^{(k)}\right)$.

Основною складністю застосування методу найшвидшого спуску є залежність його від вибору масштабу змінних, що оптимізуються. Якщо гіперпростір дуже витягнутий, так що утворює „яр" з повільним пониженням до мінімуму, процедура найшвидшого спуску сходитиметься дуже повільно, тому що напрям найшвидшого спуску – з боків яру – виявляється майже ортогональним до найкращого напрямку досягнення мінімуму.

В цьому випадку доцільно використовувати інформацію другого порядку, що дається частковими похідними другого порядку.

19.2. МЕТОДИ, ЩО ВИКОРИСТОВУЮТЬ ДРУГІ ПОХІДНІ

19.2.1. Квадратичні функції

Квадратична функція $F(x)=a+x^{T}b+\frac{1}{2}x^{T}Gx,$ (19.13)

де a - константа; b - постійний вектор і G - позитивно визначена симетрична матриця - має мінімум у точці x^{*}, причому x^{*} визначається наступним співвідношенням:

$$\nabla F(x^{*})=b+Gx^{*}=0, \qquad (19.14)$$

звідки $x^{*}=-G^{-1}b$. При виконанні умов неперервності будь-яку функцію можна апроксимувати в околі точки x_0 функцією

$$\varphi(x)=f(x_0)+(x-x_0)^{T}\nabla f(x_0)+\frac{1}{2}(x-x_0)^{T}G(x_0)(x-x_0), \qquad (19.15)$$

де $G(x_0)$ - матриця Гессе, обчислена в точці x_0. Для спрощення індексації ввважатимемо $x^{(k)}=x_k$.

Непоганою апроксимацією мінімуму функції $f(x)$ може бути мінімум функції $\varphi(x)$. Якщо він знаходиться в точці x_m, то

$\nabla f(x_0)+G(x_0)(x_m-x_0)=0,$ звідки $x_m=x_0-G^{-1}(x_0)\nabla f(x_0),$ або

$$x_m=x_0-G^{-1}(x_0)g(x_0). \qquad (19.16)$$

Таким чином, слід модифікувати ітераційне рівняння (19.10), і точкою x_i наступної апроксимації мінімуму буде:

$$x_{i+1}=x_i-G^{-1}(x_i)g(x_i) \qquad (19.17)$$

або, в зручнішому вигляді,

$$x_{i+1}=x_i-\lambda_i G^{-1}(x_i)g(x_i), \qquad (19.18)$$

де довжина кроку λ_i визначається одномірним пошуком в напрямку $G^{-1}(x_i)g(x_i)$.

Метод Ньютона - Рафсона ґрунтується на останньому рівнянні. Розглянемо деякі його особливості. Рівняння (19.17) та (19.18) в тому вигляді, в якому вони записані, потребують обчислення і інвертування матриці Гессе на кожному кроці, що нерідко є основною частиною обчислень. Якщо точка x_i розташована близько до точки x^{*}, то збіжність буде швидкою, оскільки в загальному випадку функція $\varphi(x)$ буде добре апроксимувати функцію $f(x)$ у цьому околі. Як норму градієнта $\|g(x_{i+1})\|$, так і відстань між точками $|x_{i+1}-x_i|$ слід перевірити на виконання критерію завершення. Цікавим є те, що в порівнянні з простим методом найшвидшого спуску напрямком спуску в даному випадку буде не $-g(x_i)$,

а $-G^{-1}(x_i)g(x_i)$, де враховуються і другі похідні. За допомогою методу Давідсона - Флетчера - Павела можна одержати найкращий результат, здійснюючи пошук на i – му етапі в напрямку $-H_i g(x_i)$, де H_i - позитивно визначена симетрична матриця, що у остаточному рахунку гранично при виконанні певних умов стає рівною $G^{-1}(x^{*})$. Таким чином, цей метод обходить як обчислення, так і інвертування матриці $G(x_i)$ при кожній ітерації.

Отже, напрямок пошуку при кожній ітерації є вирішальним чинником з точки зору ефективності ітераційних методів пошуку. При кожній ітерації бажано зробити одновимірний пошук у найкращому напрямку. Для квадратичної функції n змінних найкращим напрямком є напрямок, спряжений із попереднім напрямком пошуку.

Вважають, що два напрямки p і q спряжені відносно симетричної позитивно визначеної матриці G, якщо

$$p^{T}Gq=0. \qquad (19.19)$$

Якщо $p_0, p_1, \dots, p_{n-1}$ є n взаємно спряжених напрямків у n - вимірному просторі, то вони лінійно незалежні. Якщо це не так, то існують сталі $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$, не всі рівні нулю і такі, що

$$\alpha_0 p_0+\alpha_1 p_1+\dots+\alpha_{n-1}p_{n-1}=0.$$

У цьому випадку для будь-якого $k_{n-1}(0\le k\le n-1)$ справедлива рівність $p_k^{T}G\sum_{j=0}^{n-1}\alpha_j p_j+0,$ звідки $\alpha_k p_k^{T}Gp_k=0,$ тому що через спряженість всі інші члени зникають ($p_k^{T}Gp_j=0, k\neq j$).

Оскільки $p_k\neq 0$ і матриця G позитивно визначена, $\alpha_k=0$. Звідси випливає, що вектори $p_0, p_1, \dots, p_{n-1}$ лінійно незалежні.

Перед подальшими обчисленнями функцію (19.13) зручно переписати у вигляді $F(x)=a+x^{T}b+\frac{1}{2}x^{T}Gx,$ причому її мінімум знаходиться в точці $x^{*}=-G^{-1}b$ та

$$F(x)=F(x^{*})+(x-x^{*})^{T}\nabla F(x^{*})+\frac{1}{2}(x-x^{*})^{T}G(x-x^{*})=$$

$$=F(x^{*})+\frac{1}{2}(x-x^{*})^{T}G(x-x^{*})(\text{òàê ÿê}\ \ \nabla F(x^{*})=0)=$$

$$=a+x^{*T}b+\frac{1}{2}x^{*T}Gx^{*}+\frac{1}{2}(x-x^{*})^{T}G(x-x^{*})=$$

$$=a-\frac{1}{2}b^{T}G^{-1}b+\frac{1}{2}(x-x^{*})G(x-x^{*})$$

або $F(x) = c + \frac{1}{2}(x - x^*)^T = G(x - x^*)$, (19.20)

де $c = a - \frac{1}{2}b^T G^{-1} b -$ константа.

Припустимо, що для пошуку мінімуму функції (19.20) використовується ітераційна процедура.

Зрозуміло, що не варто відразу приймати рішення про напрямки пошуку (як, наприклад, при пошуку методом покоординатного спуску), а краще накопичувати інформацію, отриману на попередніх етапах пошуку, для того, щоб визначати подальші напрямки пошуку.

Почнемо з точки x_0 і проведемо пошук у напрямку p_0 з метою знаходження мінімуму в точці $x_1 = x_0 + \lambda_0 p_0$, (19.21)
де λ_0 – деяка скалярна величина.

Відзначимо, що в точці x_1 напрямок $g(x_1) = \nabla F(x_1)$ ортогональний до напрямку p_0 і $g(x_1)^T p_0 = 0$. (19.22)

У загальному випадку на кроці i здійснюється пошук із точки x_i у напрямку p_i із метою знаходження мінімуму в точці

$$x_{i+1} = x_i + \lambda_i p_i, \tag{19.23}$$

де для $F(x)$ справедливі співвідношення

$$g(x_{i+1})^T p_i = 0; \tag{19.24}$$

$$g(x_i) = G(x_i - x^*). \tag{19.25}$$

Повторним застосуванням рівняння (19.22) після n кроків отримаємо:

$$x_n = x_{n-1} + \lambda_{n-1}p_{n-1} = x_{n-2} + \lambda_{n-2}p_{n-2} + \lambda_{n-1}p_{n-1} = x_{j+1} + \sum_{i=j+1}^{n-1} \lambda_i p_i \tag{19.26}$$

для всіх j в інтервалі $0 \le j < n-1$.

Таким чином, згідно з рівнянням. (19.25)

$$G(x_n - x^*) = G(x_{j+1} - x^*) + \sum_{i=j+1}^{n-1} \lambda_i G p_i. \tag{19.27}$$

Отже, $g(x_n)^T p_j = g(x_{j+1})^T p_j + \sum_{i=j+1}^{n-1} \lambda_i p_i^T G p_j$ (19.28)

і з рівняння (19.27) отримуємо:

$$g(x_n)^T p_j = \sum_{i=j+1}^{n-1} \lambda_i p_i^T G p_j. \tag{19.29}$$

Тепер, якщо усі вектори $p_0, p_1, \ldots, p_{n-1}$ взаємно спряжені так, що

$p_i^T G p_j = 0$ і де $i \ne j$, (19.29)

то з співвідношення (19.24) випливає:

$g(x_n)^T p_j = 0$ і де $j = 0, 1, \ldots, n-1$. (19.30)

Але оскільки в цьому випадку вектори $p_0, p_1, \ldots, p_{n-1}$ лінійно незалежні і, таким чином, утворюють базу, то $g(x_n) = 0$, (19.31)

Звідки $G(x_n - x^*) = 0$ (19.32)

і $x_n = x^*$.

Отже, якщо пошук здійснюється за взаємно спряженими напрямками, то мінімум квадратичної функції n змінних буде знайдений не більш ніж за n кроків. На цій ідеї ґрунтується багато методів, а саме: метод Флетчера - Рівса, метод Девідсона - Флетчера - Павела та ін.

19.2.2. Метод Девідсона - Флетчера - Павела

Метод Девідсона - Флетчера - Павела (ДФП) ґрунтується на використанні співвідношень (19.16) і (19.18), але в ньому не потрібно на кожному кроці обчислювати обернений гессіан $G^{-1}(x_i)$, тому що напрямок пошуку на кроці i є напрямком $-H_i g(x_i)$, де H_i - позитивно визначена симетрична матриця, що поновлюється на кожному кроці, яка гранично стає рівною оберненому гессіану. Пошук у методі Девідсона - Флетчера - Павела починається із початкової точки x_0, в якості початкової матриці H_0 звичайно береться одинична матриця, хоча може бути будь-яка симетрична позитивно визначена матриця.

Ітераційна процедура може бути представлена наступною послідовністю кроків (замість $g(x_i)$ в алгоритмі вживатимемо позначення g_i):

Крок 1. На кроці i є точка x_i, і позитивно визначена симетрична матриця H_i. (Для $i = 0$ задаємо початкове наближення x_0 та обираємо в якості H_0 одиничну матрицю).

Крок 2. Як напрямок пошуку обираємо

$$d_i = -H_i g_i. \tag{19.33}$$

Крок 3. Щоб знайти λ_i, що мінімізує функцію $f(x_i + \lambda_i d_i)$, здійснюємо одновимірний пошук вздовж прямої $x_i + \lambda d_i$.

Крок 4. Покладемо $v_i = \lambda_i d_i$. (19.34)

Крок 5. Покладемо $x_{i+1} = x_i + v_i$. (19.35)

Крок 6. Знаходимо $f(x_{i+1})$ і g_{i+1}. Завершуємо процедуру, якщо величини $|g_{i+1}|$ або $|v_i|$ достатньо малі. В протилежному випадку продовжуємо.

Зауваження. З співвідношення (19.24) наслідком є, що

$g_{i+1}^T v_i = 0$. (19.36)

Крок 7. Покладемо $u_i = g_{i+1} - g_i$. (19.37)

Крок 8. Оновлюємо матрицю H наступним чином:

$$H_{i+1} = H_i + A_i + B_i, \tag{19.38}$$

де $A_i = v_i v_i^T / (v_i^T u_i)$; (19.39)

$$B_i = -H_i u_i u_i^T H_i / (u_i^T H_i u_i). \tag{19.40}$$

Крок 9. Збільшуємо i на одиницю і повертаємося на крок 2.

Метод ДФП обґрунтований теоретично і використовує як ідеї метода Ньютона-Рафсона, так і властивість спряжених напрямків, та при застосуванні для мінімізації квадратичної функції n змінних він сходиться не більше, ніж за n ітерацій. Це досить потужна оптимізаційна процедура, дуже ефективна при оптимізації більшості функцій незалежно від того, квадратичні вони чи ні.

19.3. ОПТИМІЗАЦІЯ З ОБМЕЖЕННЯМИ

19.3.1. Постановка задачі та штрафні функції

В загальному випадку задача мінімізації з обмеженнями формулюється наступним чином:

$$f(x) \Rightarrow Min \tag{19.41}$$

при обмеженнях $c_j(x) \geq 0, \ j = 1, 2, ..., m$. (19.42)

Основна ідея методу полягає у перетворенні задачі мінімізації функції $f(x)$ з відповідними обмеженнями, накладеними на x, у задачу пошуку мінімуму без обмежень функції $f(x) + P(x)$.

Функція $P(x)$ є штрафною. Необхідно, щоб при порушенні обмежень вона "штрафувала" функцію Z, тобто збільшувала її значення. В цьому випадку мінімум Z буде знаходитися всередині області обмежень. Функція $P(x)$, що задовольняє цій умові, є не єдиною.

Функцію $P(x)$ представимо у наступному вигляді:

$$P(x) = r\sum_{j=1}^{m} \frac{1}{c_j(x)} \tag{19.43}$$

де r - додатнє значення. Тоді функція $\varphi(x, r)$ буде мати вигляд:

$$\varphi(x, r) = f(x) + r\sum_{j=1}^{m} \frac{1}{c_j(x)}. \tag{19.44}$$

Якщо x має припустимі значення, тобто значення, для яких $c_j(x) \geq 0$, то $\varphi(x, r)$ приймає значення, які більше відповідних значень $f(x)$ (справжньої функції мети задачі), і різницю можна зменшити за рахунок

того, що r може бути дуже малою величиною. Але якщо x приймає значення, які хоч і являються припустимими, але близькі до границі області обмежень, і щонайменш одна з функцій $c_j(x)$ близька до нуля, тоді значення функції $P(x)$ і, відповідно, значення функції $\varphi(x, r)$, стануть дуже великими. Таким чином, вплив функції $P(x)$ полягає в створенні "гребеня з крутими межами" вздовж кожної границі області обмежень. Отже, якщо пошук починається з припустимої точки і здійснюється пошук мінімуму функції $\varphi(x, r)$ без обмежень, то мінімум буде досягатись всередині припустимої області для задачі з обмеженнями. Приймаючи r достатньо малою величиною, для того щоб вплив $P(x)$ був малим у точці мінімуму, ми можемо досягнути співпадання точки мінімуму функції $\varphi(x, r)$ без обмежень з точкою мінімуму функції $f(x)$ з обмеженнями.

В загальному випадку неможливо аналітично визначити положення мінімуму функції $\varphi(x, r)$, якщо розглядати її як звичайну функцію від r. Для його визначення необхідно використати чисельні методи.

Слід відзначити, що якщо функція мети $f(x)$ опукла, а функція $c_j(x)$ увігнута, то функція $\varphi(x, r)$, задана співвідношенням (19.46), також є опуклою функцією у опуклій області обмежень. Отже, $\varphi(x, r)$ має для даного значення r єдиний мінімум.

Якщо x_1 і x_2 - точки, що належать допустимій області, тобто $c_j(x_1) \geq 0$ і $c_j(x_2) \geq 0$ для $j = 1, 2, ..., m$, то при $0 < \theta < 1$ справедлива нерівність $c_j(\theta x_2 + (1-\theta)x_1) \geq \theta c_j(x_2) + (1-\theta)c_j(x_1) \geq 0$, так як функція $c_j(x)$ опукла. Отже, припустима область є опуклою.

Таким чином, точка $x_2 + (1-\theta)x_1$ при $0 < \theta < 1$ також є припустимою.

Крім того, функція $1/c_j(x)$ є опуклою для усіх x, які задовольняють нерівність $c_j(x) \geq 0$. Якщо $h(x) = 1/c_j(x)$, то $\nabla h(x) = \dfrac{-\nabla c_j(x)}{[c_j(x)]^2}$. Отже, гессіан функції $h(x)$ має вигляд $H(x) = -\dfrac{C(x)}{[c_j(x)]^2} + \dfrac{2\nabla c(x) \nabla c(x)^T}{[c_j(x)]^3}$, де $C(x)_{ik} = \partial^2 c_j(x)/\partial x_i \partial x_k$ є гессіан функції $c_j(x)$. Тоді, якщо p - довільний вектор, то справедлива рівність $p^T H(x) p = -\dfrac{p^T C(x) p}{[c_j(x)]^2} + \dfrac{2[p^T \nabla c_j(x)]^2}{[c_j(x)]^3}$, де завжди $p^T H(x) p > 0$, так як $C(x)$ - негативно визначена матриця внаслідок того, що $c_j(x)$ - опукла функція і $c_j(x) \geq 0$. Тоді матриця $H(x)$ позитивно визначена і $1/c_j(x)$ опукла по всій області. Якщо $r > 0$, то функція $P(x)$ і функція $\varphi(x,r)$ також опуклі.

Узагальнимо результати на випадок загальної задачі з обмеженнями (19.41) - (19.42).

Нехай, $x_1^*, x_2^*, ..., x_k^*$ - мінімальні точки функції $\varphi(x,r)$ для спадної послідовності значень $r_1, r_2, ..., r_k, ...$, що прямує до нуля. Тоді послідовність точок $x_1^*, x_2^*, ..., x_k^*, ...$ сходиться до оптимального розв'язку задачі з обмеженнями (19.41) - (19.42) при $r_k \to 0$.

Отже, $\lim\limits_{k\to+\infty} x_k^* = x^*$ (19.45)

та $\lim\limits_{r_k\to 0}[\min \varphi(x,r_k)] = f(x^*)$, (19.46)

де x^* - мінімальна точка функції $f(x)$ при наявності обмежень.

Отриманий результат можна довести наступним чином. Оскільки $f(x)$ - неперервна функція і $f(x^*) \le f(x)$ для усіх припустимих точок, то, задав-ши довільне достатньо мале значення ε, можна знайти припустиму точку x', таку, що

$$f(x') < f(x^*) + \varepsilon/2. \quad (19.47)$$

Оскільки r_k - спадна послідовність, що прямує до нуля, можна знайти таке значення K, що для $k \ge K$ справедлива нерівність

$$r_k \le \frac{\varepsilon}{2m} \min_j \left[\frac{1}{c_j(x')}\right]. \quad (19.48)$$

Оскільки $P(x) > 0$ з визначення функції $\varphi(x,r)$, маємо

$$f(x^*) \le \min \varphi(x,r_k) = \varphi(x^*_k, r_k), \quad (19.49)$$

де x^*_k - мінімальна точка функції $\varphi(x,r_k)$ для задачі без обмежень.

Крім того, якщо $k > K$, то $r_k < r_K$ і справедлива нерівність

$$\varphi(x^*_k, r_k) \le \varphi(x_K^*, r_k). \quad (19.50)$$

Це випливає з того, що оскільки x^*_k мінімізує функцію $\varphi(x,r_k)$ і в будь-якій іншій точці області x, зокрема в точці x^*_K, функція буде приймати значення, більше ніж $\varphi(x_K^*, r_k)$.

Тому $\varphi(x^*_K, r_k) = f(x^*_K) + r_K \sum\limits_{j=1}^{m} \frac{1}{c_j(x^*_K)} > f(x^*_K) + r_k \sum\limits_{j=1}^{m} \frac{1}{c_j(x^*_K)}$, оскільки $r_k < r_K$.

Отже, $\varphi(x^*_K, r_K) > \varphi(x^*_K, r_k)$. Тоді

$$f(x^*) \le \varphi(x^*_k, r_k) \le \varphi(x^*_K, r_k) < \varphi(x^*_K, r_K). \quad (19.51)$$

Але оскільки значення x^*_K мінімізує функцію $\varphi(x, r_K)$, то

$$\varphi(x_K^*, r_K) \le \varphi(x', r_K) = f(x') + r_K \sum_{j=1}^{m} \frac{1}{c_j(x')}. \quad (19.52)$$

Отже, зі співвідношень (19.51) – (19.52) отримаємо:

$$f(x^*) \le \varphi(x^*_k, r_k) \le f(x') + r_k \sum_{j=1}^{m} \frac{1}{c_j(x')}. \quad (19.53)$$

З (19.50) випливає, що $f(x') + r_k \sum\limits_{j=1}^{m} \frac{1}{c_j(x')} \le f(x') + \frac{\varepsilon}{2}$. (19.54)

Тоді з (19.47) отримуємо, що $f(x^*) \le \varphi(x_k^*, r_k) < f(x^*) + \frac{\varepsilon}{2} + \frac{\varepsilon}{2}$, тобто $\varphi(x^*_k, r_k) - f(x^*) < \varepsilon$. (19.55)

Оскільки ε може бути обране як завгодно малим, завжди можна знайти таке значення k, при якому $f(x^*) < \varphi(x^*_k, r_k) < f(x^*) + \varepsilon$.

Таким чином, при $k \to \infty$ $(r_k \to 0)$

$$\lim_{r_k\to 0} \varphi(x^*_k, r_k) = f(x^*). \quad (19.56)$$

З наведеного вище доведення випливає, що при $r_k \to 0$

$$\lim_{r_k\to 0} f(x^*_k) = f(x^*) \text{ і } \lim_{r_k\to 0} r_k \sum_{j=1}^{m} \frac{1}{c_j(x^*_k)} = 0. \quad (19.57)$$

Таким чином $f(x^*_1), f(x^*_2), ..., f(x^*_k)$ утворюють спадну послідовність, таку, що $f(x^*_{k+1}) < f(x^*_k)$. (19.58)

Очевидно, що якщо функція $f(x)$ опукла, а функція $c_j(x)$ при $j = 1, ..., n$ увігнута, то функція $f(x)$ при наявності обмежень має єдиний мінімум.

19.3.2. Метод Фіако і Маккорміка

Результати попереднього параграфа показують, що задачу мінімізації з обмеженнями (мінімізувати функцію $f(x)$ при обмеженнях $c_j(x) \ge 0$) можливо розв'язати, розв'язуючи для послідовності значень r, що прямують до нуля, послідовності задач без обмежень наступного вигляду:

$$\varphi(x,r) = f(x) + r \sum_{j=1}^{m} \frac{1}{c_j(x)} \Rightarrow Min. \quad (19.59)$$

Метод SUMT (sequential unconstrained minimisation technique) був вперше запропонований Керолом у 1961 році. Його ідеї були досліджені Фіако і Маккорміком, які не лише розглянули теоретичні питання і збіжність методу, але і створили практичну систему для його реалізації.

Цей метод рідко можна використовувати "напряму" (див. Приклади 19.2, 19.3), оскільки далеко не завжди можна знайти оптимальну точку для функції $\varphi(x,r)$ у вигляді функції $x^*(r)$, границю якої при $r \to 0$ можна дослідити.

Тому для того, щоб можна було застосувати цей метод на практиці, необхідно побудувати обчислювальний метод, який використовує теоретичну властивість збіжності, що була розглянута у попередньому питанні. Теоретично тут не виникає ніяких труднощів. Для заданих функцією $f(x)$ обмежень $c_j(x) \geq 0$, $j=1,...,n$, необхідно обрати початкове значення $r=r_0$, щоб сформувати функцію $\varphi(x,r_0)$, яка мінімізується без обмежень методом ДФП (Давідсона-Флетчера-Павела). Знайшовши мінімум функції $\varphi(x,r_0)$, необхідно зменшити значення r. Це можна здійснити ефективно і просто, якщо знайти

$$r_1 = r_0 / c, \tag{19.60}$$

де константа $c > 1$ ($c = 10$). Потім потрібно мінімізувати функцію $\varphi(x,r_1)$, знову застосовуючи метод ДФП. Таким чином, повинна працювати ітераційна процедура. На k-му кроці мінімізується функція $\varphi(x,r_k)$, мінімум якої знаходиться в точці x^*_k. Важливо, що її можна використовувати і надалі в якості першої точки в ітераційній процедурі оптимізації функції $\varphi(x,r_{k+1})$, де $r_{k+1} = r_k / c$. Тепер зрозуміло, що послідовність r_k спадає і прямує до нуля, а отже, послідовність точок мінімумів буде зводитись до рішення задачі з обмеженнями.

Вважається, що на початку процедури ми маємо припустиму початкову точку. Важливо, щоб у процесі наступних обчислень отримані точки належали припустимій області. Метод ДФП є градієнтним методом мінімізації, що може використовувати при одновимірному пошуку кубічну інтерполяцію. Тоді, по мірі наближення точки x до межі всередині припустимої області $\varphi(x,r) \to \infty$, а по мірі наближення точки x до границі ззовні припустимої області $\varphi(x,r) \to -\infty$. Таким чином, якщо пошук здійснюється вздовж прямої, що з'єднує дві точки, одна з яких лежить всередині, а інша за межами області обмежень, то кубічна інтерполяція стає неприйнятною, оскільки функція має розрив вздовж даної прямої. Дійсно, якщо мінімум буде знайдений за межами припустимої області, то метод ДФП не дозволить знову ввійти в область обмежень. Це слід мати на увазі, використовуючи для одновимірного пошуку кубічну інтерполяцію.

Вибір початкового значення r може стати важливим з точки зору скорочення числа ітерацій при мінімізації функції $\varphi(x,r)$. Якщо спочатку

r обране дуже мале, для того щоб функція $\varphi(x,r)$ мало відрізнялась від функції $f(x)$, то метод буде сходитись дуже швидко. Однак такий вибір може призвести до серйозних ускладнень при обчисленнях. Для малих r функція $\varphi(x,r)$ буде швидко змінюватись в околі мінімуму, що може викликати ускладнення при використанні градієнтного методу. Але занадто велике значення r може привести до того, що штрафна функція $P(X)$ стане домінуючою. Тому "розумний" вибір початкової точки є дуже важливим.

Для багатьох задач "розумним" значенням для початкової точки є значення $r_0 = 1$. Раціональніший підхід полягає в тому, щоб зрозуміти, що якщо початкова точка x буде знаходитися поблизу мінімуму функції

$$\varphi(x,r) = f(x) + r\sum_{j=1}^{m} \frac{1}{c_j(x)} = f(x) + rP(x),$$ то градієнт функції

$\varphi(x,r)$ буде замалий: $$\nabla\varphi(x,r) = \nabla f(x) + r\nabla P(x) \tag{19.61}$$

Квадрат норми цього вектору

$$\nabla f(x)^T \nabla f(x) + 2r\nabla f(x)^T \nabla P(x) + r^2 \nabla P(x)^T \nabla P(x) \tag{19.62}$$

і мінімум буде досягнутий при $$r = \frac{-\nabla f(x)^T \nabla P(x)}{\nabla P(x)^T \nabla P(x)}. \tag{19.63}$$

Це початкове значення r, як припускають Фіако і Маккормік, повинно давати добрі результати в загальному випадку. Зменшити значення r дуже просто: $r_{k+1} = r_k / c$, де $c = 10$.

Для мінімізації функції $\varphi(x,r_{k+1})$ використовується метод ДФП. В якості початкової точки використовується оптимальна точка функції $\varphi(x,r_k)$, і це виявляється дуже ефективним.

При програмній реалізації потрібно звернути увагу на те, щоб в процесі одновимірного пошуку не вийти за межі області обмежень.

19.3.3. Метод Бокса

Метод Бокса, по суті, є модифікацією симплексного метода Нелдера - Міда, однак дозволяє враховувати обмеження. Бокс назвав його комплексним методом.

Необхідно розв'язати наступну задачу:

$$f(x) = f(x_1, x_2, ..., x_n) \Rightarrow Min,$$

де x визначається явними обмеженнями

$$l_j \leq x_j \leq u_j \text{ при } j = 1,2,...,n. \tag{19.64}$$

а також неявними обмеженнями

$$g_i(x) \le b_i \text{ при } i = 1, 2, \ldots, m. \qquad (19.65)$$

Якщо функція мети $f(x)$ опукла і функції $g_i(x)$ також опуклі, то задача буде мати єдиний розв'язок. Значення l_j і u_j є нижніми і верхніми границями змінних. Якщо в конкретній задачі задані змінні теоретично не мають обмежень, то припущення про наявність у них "безпечних" границь, тобто границь, що включають оптимум, дозволить застосувати комплексний метод.

Припускається, що відомі значення n і m, l_j і u_j і початкова точка x_1, яка задовільняє всі обмеження (19.64)-(19.65). В першу чергу необхідно обрати k точок, які задовільняють обмеження, а також обчислити значення функції мети у всіх k точках. Множина цих точок називається комплексом. Бокс встановив, що k повинно бути більшим за $(n+1)$ - числа точок, які використовуються в симплексному методі Нелдера - Міда і показав $k = 2n$.

Як згадувалось вище, припускається, що точка x_1, що задовільняє усі обмеження, задана. Решта точок, які задовільняють нерівність (19.64), можуть бути обрані наступним чином:

$$x_{ij} = l_i + r(u_j - l_j) \qquad (19.66)$$

для $j = 1, 2, \ldots, n$ і $i = 2, 3, \ldots, k$, де r - псевдовипадкова рівномірно розподілена змінна в інтервалі $(0,1)$.

Точки, які обираються у відповідності до (19.66) для певного j, будуть автоматично задовільняти нерівність (19.64). Якщо ці точки одразу задовільняють також нерівність (19.65), то вони приймаються в якості початкових точок комплексу. Якщо точка, обрана у відповідності до рівняння (19.66), не задовільняє нерівність (19.65), то вона зміщується на половину відстані до центра ваги множини вже прийнятих точок, тобто формується точка

$$x'_i = \frac{(x_i + x_c)}{2}, \qquad (19.67)$$

де $$x_c = \frac{1}{i-1} \sum_{e=1}^{i-1} x_e. \qquad (19.68)$$

Якщо точка у співвідношенні (19.67) все ще не є припустимою, то описана співвідношенням (19.66) процедура повторюється до тих пір, поки точка не стане припустимою. Якщо функція $g_i(x)$ опукла, то врешті-решт обмеження будуть виконуватись. Звісно, оскільки точка x_1 знаходиться

всередині області обмежень, то комплекс буде складатись з припустимих точок.

Точки комплексу зручно впорядкувати у відповідності до значень функції. Процедура ініціалізації (пошуку початкового) комплексу представлена за допомогою блок-схеми рис.19.2.

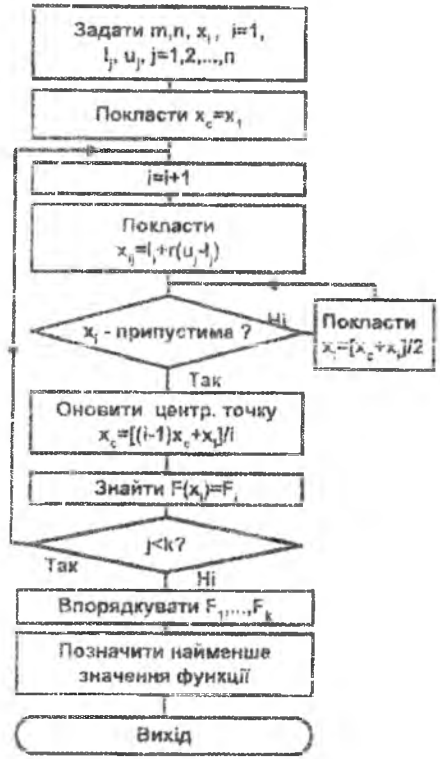


Рис. 19.2. Блок-схема алгоритму пошуку початкового комплексу

Розглянемо ітераційну процедуру комплексного методу, в якій здійснюється пошук мінімуму переміщенням в напрямку до мінімуму всередині області обмежень. Процедура складається з наступних кроків:

Крок 1. Знайти точку з найбільшим значенням функції x_h і знайти центр x_0 решти $(k-1)$ точок.

Крок 2. Спробуємо зміститися від точки x_h і отримати при цьому точку x_r, шляхом відображення точки x_h відносно точки x_0, використовуючи коефіцієнт відображення $\alpha > 1$.

$$x_r = (1+\alpha)x_0 - \alpha x_h . \qquad (19.69)$$

Крок 3. Перевіряємо, чи точка x_r є припустимою.

а) Якщо точка x_r не є припустимою і не виконується обмеження для l_i, то приймаємо $x_{ri} = l_i + 10^{-6}$; якщо не виконується обмеження для u_i, то приймаємо $x_{ri} = u_i + 10^{-6}$.

б) Якщо не виконуються обмеження, то точку x_r переміщають на половину відстані між x_r і центром x_0, тобто

$$x_r = (x_r + x_0)/2 \qquad (19.70)$$

Потім здійснюється повторна перевірка на припустимість і крок 3 повторюється до тих пір, поки не буде отримана припустима точка.

Крок 4. Якщо точка x_r є припустимою, то обчислюється значення функції $f(x_r)$ і порівнюється з $f(x_k)$ - найбільшим біжучим значенням функції.

Якщо $f(x_r) > f(x_k)$, тобто "гірше", ніж найбільше значення, отримане раніше, то точка x_r зміщається до центра x_0 на половину відстані між ними, тобто $x_r = (x_r + x_0)/2$ і процес повертається на крок 3.

Крок 5. Якщо $f(x_r) < f(x_k)$, то точка x замінюється на точку x_r, а потім точки і значення функції комплексу знову впорядковуються.

Крок 6. Обчислюються дві величини, які використовуються при перевірці збіжності методу: середнє квадратичне відхилення δ для k значень функції і максимальна відстань d_m між двома точками комплексу.

Перша величина обчислюється як $\sigma = \left\{ \sum_{e=1}^{k} [f(x_e) - f]^2 / k \right\}^{1/2}$, (19.71)

Де $f = \frac{1}{k}\sum_{e=1}^{k} f(x_e)$ (19.72)

(для обчислення σ^2 алгоритмічно краще використовувати формулу

$$\sigma^2 = \left| \sum_{e=1}^{k} f(x_e)^2 - \frac{\left[\sum f(x)\right]^2}{k} \right| / k . \qquad (19.73)$$

Крок 7. Величини σ^2 і d_m перевіряються на збіжність. Якщо обидві ці величини достатньо малі, то процедура пошуку мінімуму закінчується. У протилежному випадку потрібно повернутись на крок 1 і повторити процедуру.

На блок-схемі (рис. 19.3) наведений алгоритм Бокса.

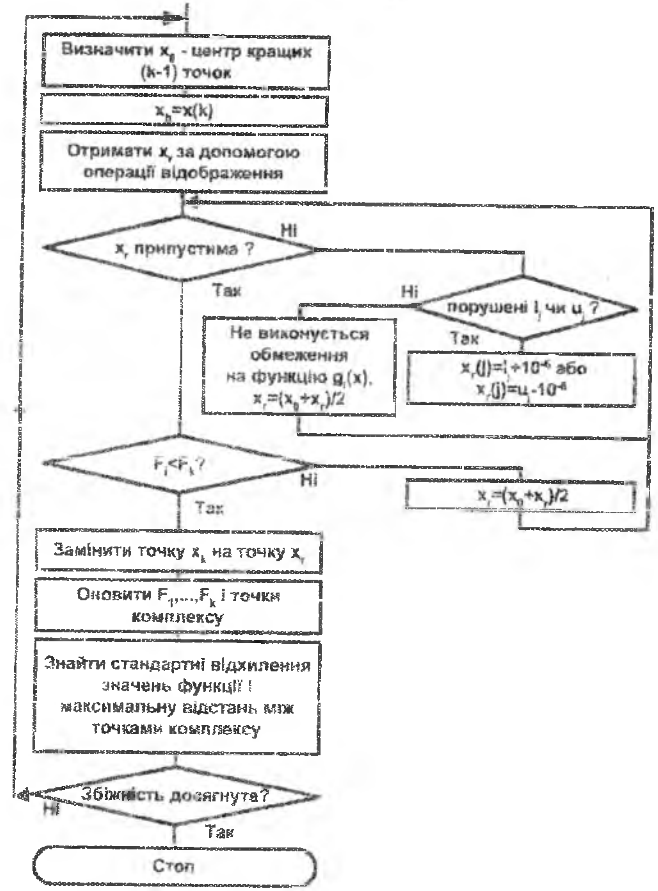


Рис 19.3. Блок-схема ітераційної процедури методу Бокса

Однак слід відзначити, що ідеального методу пошуку оптимуму для багатоекстремальних функцій навіть без обмежень, не кажучи про задачі з обмеженнями, не існує. Тому з метою підвищення достовірності отриманих результатів процедуру слід повторити декілька разів з різними початковими значеннями, а також порівняти результати роботи декількох методів.

ПРИКЛАДИ

Приклад 19.1. Метод найшвидшого спуску.

Необхідно знайти мінімальне значення функції $f(x) = f(x_1, x_2) = x_1^2 + 2x_2^2 + e^{x_1+x_2}$ за допомогою методу найшвидшого спуску, критерій зупинки $\left|\dfrac{df\left(x^{(k)}\right)}{dx_i}\right| \le 0.05$, $i = 1,2$, початкове наближення $x^{(0)} = (0,0)$, початкова довжина кроку $\lambda_0 = 1$, якщо при здійсненні кроку в напрямі градієнту значення функції погіршується, крок зменшується вдвічі.

Розв'язання.

Знаходимо вирази для часткових похідних,

$\dfrac{df(x)}{dx_1} = 2x_1 + e^{x_1+x_2}$, $\dfrac{df(x)}{dx_2} = 4x_2 + e^{x_1+x_2}$, і обчислюємо наступні точки (результати зведені в таблицю).

k	$x_1^{(k)}$	$x_2^{(k)}$	$f\left(x^{(k)}\right)$	$\partial f\left(x^{(k)}\right)/\partial x_1$	$\partial f\left(x^{(k)}\right)/\partial x_2$	λ_k	Примітка
0	0	0	1	1	1	1	
	1	-1	3.145	-	-		Порушена умова (19.12). $\lambda_0 = \lambda_0/2$
	0	0	1	1	1	0,5	
	-0,5	-0,5	1,118		-		Порушена умова (19.12). $\lambda_0 = \lambda_0/2$
	0	0	1	1	1	0,25	
1	-0,25	-0,25	0,794	0,106	-0,393	0,25	Умова виконана
2	-0,2766326	-0,1516326	0,774	0,0983	0,0451	0,25	Умова виконана
3	-0,3012259	-0,1629096	0,772	0,0262	-0,23	-	Точність досягнута

Таким чином, $x^* \approx (-0,301;-0,163)$, $f^* \approx 0,772$.

Приклад 19.2. Штрафні функції

Використовуючи штрафну функцію, яка задана рівнянням (19.43), мінімізувати функцію $f(x) = x$ при обмеженнях $x \ge 2$, тобто $x - 2 \ge 0$. Мінімальним значенням функції 2 є при $x = 2$.

Розв'язання.

Як за допомогою штрафної функції можна знайти розв'язок? Розглянемо

функцію $\varphi(x,r) = x + \dfrac{r}{x-2}$.

Відобразимо графік функції $\varphi(x,r)$ і покажемо положення точок її мінімуму для різноманітних значень r (1; 0,25; і 0,01).

Область обмежень знаходиться справа від вертикальної прямої $x = 2$. Неважко помітити, що послідовність точок Q_1, Q_2, Q_3 прямує до точки Q - мінімуму функції при наявності обмежень. Дійсно, знайдемо мінімум функції $\varphi(x,r)$:

$\dfrac{d\varphi}{dx} = 1 - \dfrac{r}{(x-2)^2}$. Відповідно, якщо $d_\varphi/dx = 0$, $(x-2)^2 = r$,

то $x = 2 \pm \sqrt{r}$. Тоді $\dfrac{d^2\varphi}{dx^2} = \dfrac{2r}{(x-2)^3}$ і мінімум досягається при $x = 2 + \sqrt{r}$ всередині області обмежень.

Отже, функція $\varphi(x,r)$ має мінімум, що рівний $2 + 2\sqrt{r}$ при $x = 2 + \sqrt{r}$. Тоді Q_1 є точкою з координатами (3;4), Q_2 - точка з координатами (2,5;3), Q_3 - точка з координатами (2,1;2,2). Ясно, що при $r \to 0$ мінімум без обмежень функції $\varphi(x,r)$ наближається до значення 2 і мінімальною точкою є точка $x = 2$.

Приклад 19.3. Штрафні функції.

Розглянемо наступну задачу:

мінімізувати функцію $f(x_1, x_2) = \dfrac{1}{3}(x_1 + 1)^3 + x_2$ при обмеженнях $x_1 - 1 \ge 0, x_2 \ge 0$.

За аналогією з рівнянням (19.43) запишемо:

$$\varphi(x,r) = \frac{1}{3}(x_1+1)^3 + x_2 + r\left(\frac{1}{x_1 - 1} + \frac{1}{x_2}\right).$$

Необхідні умови мінімуму функції φ записуються у вигляді рівнянь:

$$(x_1+1)^2 - \frac{r}{(x_1-1)^2} = 0, \quad 1 - \frac{r}{x_2^2} = 0,$$

що мають наступний розв'язок:

$x_1(r) = (1 + \sqrt{r})^{1/2}$, $x_2(r) = \sqrt{r}$.

Тоді мінімальним значенням функції $\varphi(x,r)$ буде:

$$\varphi^*(r)=\left\{\frac{1}{3}[(1+\sqrt{r})^{1/2}+1]^3+\sqrt{r}+r\left[\frac{1}{\sqrt{r}}+\frac{1}{(1+\sqrt{r})^{1/2}-1}\right]\right\}=$$

$$=\left\{\frac{1}{3}[(1+\sqrt{r})^{1/2}+1]^3+\sqrt{r}+r\left[\frac{1}{\sqrt{r}}+\frac{(1+\sqrt{r})^{1/2}+1}{\sqrt{r}}\right]\right\}=$$

$$=\left\{\frac{1}{3}[(1+\sqrt{r})^{1/2}+1]^3+\sqrt{r}+\sqrt{r}[1+1+(1+\sqrt{r})^{1/2}]\right\}.$$

Таким чином, при $r\to 0$ $x_1(r)\to 1, x_2(r)\to 0$ $\varphi^*(r)\to f(1;0)=8/3$.

РЕЗЮМЕ

19.1. Метод найшвидшого спуску ґрунтується на використанні поряд із значеннями функції і її градієнта, тобто інформації про напрямок пошуку. Напрямок градієнту перпендикулярний в будь-якій точці до лінії постійного рівня, оскільки уздовж цієї лінії значення функції є сталим. В методі найшвидшого спуску використовується ця властивість напрямку градієнта. Основною складністю застосування методу найшвидшого спуску є залежність його від вибору масштабу змінних, що оптимізуються.

19.2. Метод Ньютона – Рафсона ґрунтується на використанні квадратичних функцій і інвертує матрицю Гессе на кожному кроці, що є працемісткою процедурою. З метою уникнення цієї складності у методі Давідсона – Флетчера – Павела пошук здійснюється в напрямку позитивно визначеної симетричної матриці, що гранично наближається до матриці Гессе. Пошук здійснюється за взаємно спряженими напрямками з накопиченням попередньої інформації. За цих умов мінімум квадратичної функції n змінних буде знайдений не більш ніж за n кроків. Метод ДФП є досить потужною оптимізаційною процедурою, дуже ефективною при оптимізації більшості функцій незалежно від того, квадратичні вони чи ні.

19.3. Основна ідея методу розв'язання задач з обмеженнями полягає у перетворенні задачі мінімізації функції з обмеженнями у задачу пошуку мінімуму без обмежень функції з „штрафним" додатком, що формується на основі обмежень і різко збільшує значення функції мети при наближенні до межі. Вплив штрафної функції полягає в створенні "гребеня з крутими межами" вздовж кожної границі області обмежень. Метод Фіако-Маккорміка у якості ітераційного кроку використовує метод Давідсона – Флетчера – Павела. При програмній реалізації методу Фіако – Маккорміка потрібно звернути увагу на те, щоб в процесі одновимірного пошуку не вийти за межі області обмежень. Метод Бокса, по суті, є модифікацією симплексного методу Нелдера – Міда, що дозволяє враховувати обмеження. Ідеального методу пошуку оптимуму для багатоекстремальних функцій навіть без обмежень, не кажучи про задачі з обмеженнями, не існує. Тому з метою підвищення достовірності отриманих результатів процедуру слід повторити декілька разів з різними початковими значеннями, а також порівняти результати роботи декількох методів.

ЗАВДАННЯ ДЛЯ РОЗВ'ЯЗУВАННЯ

Завдання 19.1. Метод найшвидшого спуску.

Здійснити один крок градієнтного методу з точки $x^{(0)}=(0,1,0)$ з кроками $\lambda_0=0,1$, $\lambda_0=0,638$, $\lambda_0=10$ та порівняти отримані результати для функції $f(x)=x_1^4+x_2^2+x_3^2+x_1x_2+x_2x_3$. Визначити за допомогою методу "золотого перетину" оптимальне значення довжини першого кроку.

Завдання 19.2. Метод деформованого многогранника.

Здійснити 5 ітерацій методу Нелдера-Міда при мінімізації функції Розенброка $f(x_1,x_2)=100\ (x_2-x_1^2)^2+(1-x_1)^2$, обравши $k=0.5$, $\alpha=1$, $\beta=0.5$, $\gamma=2$, і початкову точку $(1.5;2)$. Оцінити близькість до оптимуму, якщо відомо, що він досягається в точці $x^*=(1;1)$.

Завдання 19.3. Метод Давідсона-Флетчера-Павела.

Розв'яжіть завдання 19.2 (теж 5 ітерацій) за допомогою методу Давідсона – Флетчера – Павела. Порівняйте отримані результати з методом деформованого многогранника.

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ТА ПОВТОРЕННЯ

1. Що таке градієнт?
2. Обґрунтуйте, що є напрямком найшвидшого зростання функції?

3. Що таке лінія рівня та які її властивості?

4. Сформулюйте критерії завершення обчислень в методі найшвидшого спуску.

5. Які недоліки методу найшвидшого спуску?

6. З якою метою використовується квадратична функція в методах пошуку оптимальних рішень?

7. Які особливості методу Давідсона – Флетчера – Павела порівняно з методом Ньютона – Рафсона?

8. Які напрямки називаються спряженими?

9. За скільки кроків знаходиться мінімум квадратичної функції при пошуку за спряженими напрямками?

10. На чому ґрунтується метод Давідсона – Флетчера – Павела?

11. В чому полягає ідея оптимізації функцій з обмеженнями?

12. Які властивості штрафної функції?

13. Сформулюйте основні кроки методу Фіакко – Маккорміка та дайте їх інтерпретацію.

14. Яким чином враховуються обмеження в методі Бокса?

15. Як здійснюється пошук початкового комплексу в методі Бокса?

16. Розкрийте зміст основних кроків методу Бокса.

17. Що слід робити, щоб підвищити достовірність розв'язання задач оптимізації з обмеженнями?

ДОДАТОК 1

ПРЕДМЕТНИЙ ВКАЗІВНИК

А

В

Г

Д

Е

З

І

К

М

О

П

Р

С

Т

У

Ф

Ц

Ч

ДОДАТОК 2

ЛІТЕРАТУРА

1. Акулич И.Л. Математическое программирование в примерах и задачах. - М.: Высшая школа, 1986.

2. Акоф Р., Сасиени М. Основы исследования операций.-М.:Мир, 1971.

3. Ахьюджа Х. Сетевые методы управления и проектирования в производстве. -М.: Мир, 1979.

4. Ашманов С.А. Линейное программирование. - М.: Наука, 1981.

5. Ашманов С.А., Тимохов А.В. Теория оптимизации в задачах и упражнениях. - М.: Наука, 1991.

6. Базара М., Шетти К. Нелинейное программирование. Теория и алгоритмы. - М.: Мир, 1982.

7. Банди Б. Методы оптимизации. Вводный курс. - М.: Мир, 1988.

8. Вагнер Г. Основы исследования операций.Т.1-3.- М.: Мир,1972.

9. Вентцель Е.С. Исследование операций. - Сов.радио, 1972.

10. Вентцель Е.С. Исследование операций. Задачи, принципы, методология. - М.: Наука, 1980.

11. Гудман С., Хидетниеми С. Введение в разработку и анализ алгоритмов. - М.: Мир, 1981.

12. Гуревич Т.Ф., Лущук В.О. Сборник задач по математическому программированию. - М.: Колос, 1977.

13. Дегтяров Ю.И. Исследование операций. - М.: Высшая школа, 1986.

14. Деннис Дж., Шнабель Р. Численные методы безусловной оптимизации и решения нелинейных уравнений. -М.: Мир, 1988.

15. Зайченко Ю.П. Исследование операций. - К.: Вища школа,1986.

16. Иенсен П., Барнес Д. Потоковое программирование. - М.: Мир 1984.

17. Исследование операций. /Под ред. Дж Моудера. Т.1,2. М.: Мир, 1981.

18. Интриллигатор М. Математические методы оптимизации и экономическая теория. - М.: Прогресс, 1975.

19. Калихман И.Л., Войтенко М.А. Динамическое программирование в примерах и задачах. - М.: Высшая школа, 1973.

20. Калихман И.Л. Сборник задач по математическому программированию. - М.: Высшая школа, 1975.

21. Катренко А.В. Системний аналіз об'єктів та процесів комп'ютеризації. – Львів: „Новий світ - 2000”, 2003.

22. Коваленко А.А. Сборник задач по теории игр. -Львов: Высшая школа, 1974.

23. Конвей Р.В., Максвелл, Миллер Л.В. Теория расписаний.- М.: Наука, 1975.

24. Крушевский А.В. Теория игр. -К.: Вища школа, 1977.

25.Кузнецов Ю.Н. и др. Математическое программирование. - М.: Высшая школа, 1976.

26. Липский В. Комбинаторика для программистов. - М.: Мир, 1988.

27. Майника Э. Алгоритмы оптимизации на сетях и графах. - М.: Мир, 1981.

28. Муртаф Б. Современное линейное программирование. - М.: Мир, 1984.

29. Мину М. Математическое программирование.- М.: Наука,1990

30. Оуен Г. Теория игр. - М.: Мир, 1971.

31. Пападимитриу Х., Стайглиц И. Комбинаторная оптимизация. Алгоритмы и сложность. - М.: Мир, 1985.

32. Таха Х.Введение в исследование операций.- М.: Вильямс, 2001.

33. Филипс Д., Гарсия-Диас А. Методы анализа сетей.-М.: Мир 1984.

34. Химмельблау Д. Прикладное нелинейное программирование. - М.: Мир, 1975.

35. Ху Т. Целочисленное программирование и потоки в сетях.- М.: Мир, 1974.

Підручник

Капщенко Анатолій Васильович

ДОСЛІДЖЕННЯ ОПЕРАЦІЙ

Відповідальний за випуск *В.М.Піча*
Дизайн та комп'ютерна верстка *О.М.Безотосний*

Підписано до друку з оригінал-макета 26.04.2004 р.
Формат 60 × 84/16. Папір офсетний. Гарнітура Таймс Нью Роман
Друк офсетний.Умовн. друк. арк. 30,5.

Видавнича організація "Магнолія плюс"
а/с 2623,м.Львів-60,79060 Україна
E-mail:magnolia@org.lviv.net
Свідоцтво про видавничу діяльність і розповсюдження видавничої продукції
серія ДК № 1213 від 28.01.2003 р.,
видане Державним комітетом інформаційної політики,
телебачення та радіомовлення України

Автор оригінал-макета видавець В.М.Піча

Надруковано в поліграфічному центрі ТзОВ НВФ "Магнолія плюс".
79031, м.Львів-31, вул.Стрийська, 202