

**Міністерство освіти і науки України**

**ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ:  
ПРОБЛЕМИ ТА ПЕРСПЕКТИВИ**

*Монографія*

*За загальною редакцією*

*д-ра екон. наук, професора В. С. Пономаренка*

**Харків, 2017**

УДК 681.518.54

ББК 32.966

И 74

*Рекомендовано до видання рішенням ученої ради Харківського національного економічного університету імені Семена Кузнеця  
(протокол № 6 від 06.03.2017 р.)*

**Рецензенти:**

**Корабльов Микола Михайлович** – доктор технічних наук, професор, професор кафедри ЕОМ Харківського національного університету радіоелектроніки (ХНУРЕ);

**Семенов Сергій Генадійович** – доктор технічних наук, завідувач кафедри “Обчислювальної техніки і програмування” НТУ “ХПІ”;

**Остапов Сергій Едуардович** – доктор фізико-математичних наук, завідувач кафедри програмного забезпечення комп’ютерних систем Чернівецького національного університету ім. Ю. Федьковича.

**И 74** Інформаційні технології: проблеми та перспективи : монографія / за заг. ред. В. С. Пономаренка. – Х. : Вид. Рожко С. Г., 2017. – 447 с.  
ISBN 978-966-97498-5-7

Розглянуто й обґрунтовано результати наукових досліджень в галузі розробки і практичного застосування сучасних інформаційних технологій.

Монографія представляє інтерес як для фахівців, сфера діяльності яких безпосередньо пов’язана з розробкою ІТ-технологій, способів забезпечення безпеки і передачі в комунікаційних системах, так і для більш широкого кола фахівців. Вона буде корисною викладачам, аспірантам і студентам, що спеціалізуються в області ІТ-технологій, і всім, хто серйозно цікавиться проблемами взаємодії інформаційних технологій і суспільства.

Рассмотрены и обоснованы результаты научных исследований в области разработки и практического применения современных информационных технологий.

Монография представляет интерес как для специалистов, сфера деятельности которых непосредственно связана с разработкой ИТ-технологий, способов обеспечения безопасности и передачи в коммуникационных системах, так и для более широкого круга специалистов. Она будет полезна преподавателям, аспирантам и студентам, специализирующимся в области ИТ-технологий, и всем, кто серьезно интересуется проблемами взаимодействия информационных технологий и общества.

*За достовірність викладених фактів, цитат та інших відомостей  
відповідальність несе автор.*

ISBN 978-966-97498-5-7

© Коллектив авторов, 2017

## ЗМІСТ

Вступ	5
<b><i>Інформаційні технології в технічних системах</i></b>	
Розділ 1. Альошин Г. В., Коломійцев О. В. Проблеми створення багатофункціональної лазерної системи контролю і управління літальним апаратом	16
Розділ 2. Алексієв В. О., Алексієв О. П. Інформаційний розвиток порталу віртуального управління процесами транспортного обслуговування	32
Розділ 3. Іванов В. Г. Параллельные и последовательные структуры Хаара для цифровой обработки сигналов	48
Розділ 4. Коваленко А. А. Синтез інформаційної та технічної структури системи управління об'єктом критичного застосування	64
Розділ 5. Іванов В. Г. Сжатие данных портретных изображений	78
Розділ 6. Руденко О. Г., Безсонов О. О. Многокритериальная нейроидентификация нелинейных объектов на основе коэволюционного подхода	94
Розділ 7. Мінухін С. В. Модель планирования пакетов заданий в распределенной вычислительной системе	110
Розділ 8. Лосев М. Ю. Задачи и алгоритмы управления маршрутизацией пакетов в сетях	120
Розділ 9. Різник О. Я. Фільтри графічних зображень на основі числових в'язанок	137
Розділ 10. Удовенко С. Г., Чала Л. Е. Нейросетевой метод кластеризации электронных текстов в распределенной файловой системе	154
<b><i>Захист інформації в інформаційних та комунікаційних системах</i></b>	
Розділ 11. Дудикевич В. Б., Микитин Г. В. Комплексні системи безпеки кібернетичного простору кіберфізичної системи на основі концепції “об'єкт – загроза – захист”	170
Розділ 12. Хорошко В. О., Хохлачова Ю. Є. Підхід до вибору засобів захисту з урахуванням їх оцінок	187
Розділ 13. Євсєєв С. П., Король О. Г. Синергетические модели оценки безопасности в автоматизированных банковских системах	203

Розділ 14. Мельник М. О., Нікітін Г. Д. Моделі побудови політики інформаційної безпеки підприємства	222
Розділ 15. Маракова І. І., Кузнецова Л. А. Разработка и исследование методов реверсивных цифровых знаков (ЦВЗ) для верификации медицинских изображений	237
Розділ 16. Смірнов О. А., Коваленко О. В. Разработка метода управления рисками разработки программного обеспечения	254
Розділ 17. Петришин Л. Б. Аналіз системних властивостей та обґрунтування ефективності методів перетворення форми інформації	269
Розділ 18. Ровінський В. А., Петришин М. Л. Аналіз ефективності методів перетворення форми інформації в адитивних системах числення	281
<b><i>Інформаційні технології в економіці, екології, медицині та освіті</i></b>	
Розділ 19. Задачин В. М. Методи та моделі прогнозування міського добового водоспоживання	297
Розділ 20. Засядько А. А., Королюк С. С. Моделювання максимізації прибутку на основі виробничої функції Кобба-Дугласа	312
Розділ 21. Аксак Н. Г. Мультиагентная система нейросетевой диагностики и удаленного мониторинга пациента	325
Розділ 22. Коц Г. П., Моїсеєнко А. С. Розвиток ІТ-галузі в Україні: світові рейтинги, тенденції, перспективи	341
Розділ 23. Вільхівська О. В., Бринза Н. А. Определение стоимости разработки внедряемых технологий электронного бизнеса на промышленном предприятии	356
Розділ 24. Москаленко В. В., Фонта Н. Г. Подход к построению информационной системы управления эффективностью предприятия, входящего в холдинг	371
Розділ 25. Затхей В. А., Тесленко О. В. Формалізація знань про процеси підготовки вихідних даних при тестуванні	387
<b><i>Комп'ютеризовані технології та системи видавничо-поліграфічних виробництв та електронних мультимедійних видань</i></b>	
Розділ 26. Ізонін І. В., Лотошинська Н. Д. Дослідження технології побудови тривимірних моделей	402
Розділ 27. Ломоносов Ю. В. Методи підвищення якості класифікації символів при обробці текстових зображень	418
Перелік використаної літератури	434



## ВСТУП

Сучасні інформаційні технології (ІТ) займають провідне місце у рішенні завдань розвитку промисловості та сільського господарства в Україні і світі. Це найбільш стрімкий та прогресивний напрямок, що дозволяє швидко реагувати на зміни у економіці та адаптувати виробничі технології підприємств й бізнес-процеси компаній щодо рівня їх конкурентоспроможності на внутрішньому та світовому ринках. Прогрес у галузі ІТ-індустрії надає можливості створення нових робочих місць, сприяє покращенню існуючих технологічних процесів та забезпечує сучасними інструментальними засобами рішення з автоматизації виробництва.

У свою чергу, розвиток сучасних ІТ базується на дослідженнях та застосуванні засобів обробки великих даних (Big Data), технологіям Інтернету речей (IoT), розвитку теорії та практики штучного інтелекту (Artificial Intelligence) та засобів машинного навчання (Machine Learning). Розвиток цих напрямів обумовлений прогресом у теоретичних дослідженнях у галузі комп'ютерних наук, сучасним практикам та тенденціям у галузі програмної інженерії. Підґрунтям для такого зростання є вдосконалення та експоненційне підвищення продуктивності обчислювальних ресурсів, як окремих комп'ютерних систем, так й сучасних центрів обробки даних (ЦОД), формування послуг у вигляді хмарних обчислень (Cloud Computing), GRID та покращення якості Інтернет-послуг і розвиток мобільних систем.

Слід відзначити, що до переліку пріоритетних напрямів розвитку науки і техніки на період до 2020 року входять інформаційні та комунікаційні технології, відповідно до закону України “Про пріоритетні напрями розвитку науки і техніки”. Це підкреслює значення виконання наукових та практичних досліджень у напрямку визначення проблем та перспектив у ІТ-галузі.

У монографії наведені результати наукових досліджень в галузі інформаційних систем та технологій, що представлені на ІХ Міжнародній науково-практичній конференції “Проблеми та перспективи розвитку ІТ-індустрії”, яка відбулась 20–21 квітня 2017 року на базі Харківського національного економічного університету імені Семена Кузнеця. Монографія містить результати за найсучаснішими науковими напрямками застосування, проектування та визначення проблем і перспектив ІТ в технічних системах, захисту інформації в інформаційних та комунікаційних системах. Також робота містить дослідження з ІТ в економіці, екології, медицині та освіті. Окремі питання у монографії стосуються визначення засобів та розробці новітніх

комп'ютеризованих технологій та систем видавничо-поліграфічних виробництв та електронних мультимедійних видань.

Сучасні ІТ є дуже складними та дозволяють створювати відповідні практичні рішення у будь якій галузі науки та техніки. Одним із перспективних напрямків є застосування методів та засобів на основі штучних нейронних мереж (ШНМ). Наприклад, труднощі, пов'язані з ідентифікацією нелінійних динамічних об'єктів класичними методами, привели до появи і розвитку альтернативного, нейромережевого, підходу до вирішення завдань такої ідентифікації.

У розділі 6 запропоновано використання коеволюційного підходу до побудови математичних моделей нелінійних об'єктів при наявності перешкод вимірювання. Розглянуто загальну структуру еволюційної нейромережі прямого поширення. Проведено імітаційне моделювання різних динамічних об'єктів, яке підтвердило ефективність підходу, що розвивається у роботі.

У розділі 10 запропоновано метод кластеризації та індексації електронних текстів, заснований на застосуванні алгоритму SOINN (Self-Organizing Incremental Neural Network), лінгвістичних дескрипторів і модульної системи обробки файлових масивів. Метод дозволяє формувати динамічну структуру кластерів вхідної інформації і створювати єдині індекси електронної бібліотеки. Результати тестування підтверджують ефективність застосування методу для обробки і класифікації документів в текстових масивах.

Для рішення завдань побудови інформаційних систем на основі ШНМ традиційно залучають надпродуктивні комп'ютерні ресурси та відповідні технології розподілених систем, наприклад, обчислювальні кластери, GRID-системи, засоби хмарних обчислень. Для таких сучасних розподілених систем обробки даних актуальним є планування пакетів завдань.

У розділі 7 розглянуто узагальнення моделі та методів планування пакетів завдань в розподілених обчислювальних системах (РОС), які базуються на вирішенні задачі визначення найменшого покриття. Для узагальнення моделі використана нотація Грехема, яка дозволяє врахувати положення і принципи використання моделі планування, а також динаміку процесів обробки і планування пакетів завдань в РОС в рамках дискретно-подієвої моделі розподілених обчислень.

Слід відзначити, що ефективність роботи сучасних розподілених комп'ютерних ресурсів також залежить від застосування відповідних алгоритмів щодо маршрутизації пакетів у мережі. У розділі 8 аналізуються алгоритми збору та обробки інформації щодо вирішення задачі маршрутизації пакетів в корпоративних комунікаційних мережах обміну даними. Розглядаються шляхи вдосконалення алгоритмічного забезпечення маршрутизації пакетів на основі використання розподіленої стратегії управління процесом інформаційного обміну.

Пропонуються алгоритми, що забезпечують маршрутизацію пакетів в мережах для різних видів з'єднань.

Практичне застосування розподілених комп'ютерних ресурсів та мереж розглянуто у розділі 2 на прикладі розвитку транспортного порталу WEB-рішень в умовах постійного розширення кількості користувачів на рівні міста або регіону. Це відповідає розв'язанню існуючого протиріччя між стрімким розвитком засобів та методів інформатизації складних об'єктів і систем та гетерогенним характером підсистем та ланок транспортного комплексу. Аналізуються здатність системи до збільшення загальної пропускнуєї спроможності відповідно до підвищеного навантаження, визначаються засоби та технології щодо рішення завдання масштабування апаратних ресурсів.

У розділі 4 розглядаються та пропонуються підходи до синтезу інформаційної та технічної структур системи управління об'єктом критичного застосування. Синтез інформаційної структури враховує вибір задач управління компонентами системи, реалізації алгоритмів і методів розв'язання задач і розподіл задач по компонентах в процесі їх розв'язання. Також сформульовано критерій якості і можливі відповідні обмеження. У процесі синтезу технічної структури враховуються вимоги за доступністю компонентів системи управління та процес її функціонування в контексті навколишнього середовища за умови можливих впливів. Окремо порушено питання синтезу організаційної структури системи управління.

Також практичне застосування ІТ та особливості проектування систем керування розглядаються на прикладі технічних рішень у галузі авіабудування. У розділі 1 вирішуються проблеми створення багатофункціональної лазерної системи контролю і управління літальним апаратом. У роботі визначається, що концентрація енергії у лазерному випромінюванні (ЛВ), стабільність частот, що випромінюються дозволяють створити більш багатофункціональну, компактну та з високою точністю вимірювання параметрів руху (ВІР) літального апарату (ЛА) лазерну систему (ЛС), ніж системи радіодіапазону.

У свою чергу, на сьогодні практично всі досягнення сучасних ІТ є результатом активної взаємодії технологій побудови надвеликих інтегральних схем (НВІС), обчислювальної техніки, цифрової обробки сигналів і засобів телекомунікацій. Зараз телекомунікації фактично поєднані із обчислювальною технікою. Це значно посилює вимоги до швидкості процесу обробки сигналів, що визначає потреби у наявності надвеликих обчислювальних ресурсів. Поруч із цим дешеві та швидкодіючі НВІС з високою щільністю компоновки елементів дозволяють здійснити відповідні суперобчислення, якщо оцінювати їх за розмірами похідних пристроїв, продуктивності і вартості. Тому у розділі 3 приділяється увага методам розробки архітектури обчислювачів для проектування високопаралельних НВІС-процесорів обробки сигналів. Отримані результати і

висновки дозволяють на практиці будувати різні архітектури процесорів обробки сигналів в базисі Хаара, надаючи перевагу їх ефективності за тими чи іншими критеріями, визначеними у роботі.

Інформаційні технології в технічних системах – це не тільки засоби та методи обробки даних, технології керування та управління складними об'єктами, сучасні НВІС та розподілені комп'ютерні ресурси і засоби побудови обчислювальних мереж, а й рішення актуальних завдань візуалізації та представлення даних. Візуальне представлення даних завжди більш інформативніше за інші методи отримання та сприйняття інформації.

З поширенням мультимедійних комп'ютерних засобів дуже актуальними стали задачі обробки та покращання якості зображень. Мета роботи, результати якої наведені у розділі 9, полягає в опрацюванні теоретичних положень, щодо завдання обробки зображень; дослідженні вже існуючих методів фільтрації та застосування числової лінійки-в'язанки для розробки фільтрів. У розділі розглянуто теоретичні та практичні проблеми побудови фільтрів графічних зображень за допомогою числових в'язанок. Зроблено аналіз та описано необхідні умови для створення якісних графічних фільтрів розмиття (низькочастотні фільтри), різкості (високочастотні фільтри), тиснення та виділення контурів.

Відповідно до рішення завдань візуалізації даних та обробки зображень, розділ 5 монографії присвячено визначенню та аналізу методів і засобів стиснення портретних зображень на основі виділення об'єктів та фону й кодування їх за різними ступенями якості. Для рішення цього завдання застосовується математичний апарат цифрової обробки даних – вейвлет-перетворення та засоби багатомасштабного аналізу даних із наступною обробкою виділених областей за допомогою JPEG-технологій. Завдяки такому підходу можна зберегти високу якість інформативних областей (об'єктів), а менш інформативні області (фон) подати з більшою компресією, але з гіршою якістю і таким чином підвищити загальний коефіцієнт стиснення для всього зображення.

Наступний напрям монографії присвячено рішенню завдань щодо захисту інформації в інформаційних та комунікаційних системах. Засади та завдання забезпечення кібернетичної безпеки України сьогодні є стратегічним фундаментом для розроблення та впровадження комплексних систем захисту інформаційно-комунікаційних технологій (ІКТ). У розділі 11 розглянуті комплексні системи безпеки кіберфізичної системи (КФС) на рівні кібернетичного простору (КП), “ресурси – системи – процеси” на основі концепції “об'єкт – загроза – захист”. Представлено багаторівневу модель захисту інформації в КП КФС.

Розділ 12 монографії розглядає особливості підходу, що пропонується для вибору засобів захисту інформації з урахуванням їх оцінок. Дається короткий огляд вибору засобів захисту, та визначається як і коли концепція базової безпеки

може бути відповідно використана. У роботі визначається два головних підходи до вибору засобів захисту, а саме: використання базового підходу та проведення детального дослідження ризиків. Наведені основні компоненти для визначення типу інформаційної системи. Проаналізовано поняття фізичної безпеки інформаційної системи. Показано в чому полягає розробка базової безпеки організації, як проводиться вибір засобів захисту відповідно до детальних оцінок, а також що містить у собі процес планування безперебійної роботи.

У розділі 14 розглянуто основні моделі політик та типів політик інформаційної безпеки підприємства. В результаті цього було виявлено, що в залежності від особливостей організації можна реалізувати ту чи іншу модель політики безпеки, яка дозволить захистити дані компанії від несанкціонованого доступу, а також зменшить втрати компанії у випадку дій з боку співробітників компанії, які можуть нанести шкоду організації.

Захист інформації є важливим завданням для будь-якої організації чи підприємства, однак, захист банківських даних є головним завданням інформаційних систем відповідних організацій та установ. У розділі 13 розкрито сутність побудови синергетичної моделі оцінки безпеки банківської інформації (БІн), запропоновані формальний математичний опис синергетичної моделі безпеки БІн на основі методології та синергетичного підходу до забезпечення безпеки БІн і оцінювання безпеки інформаційних технологій автоматизованих банківських систем (АБС) організацій банківського сектору (ОБС) України, а також приватних моделей: інфраструктурної моделі АБС, синергетичної моделі загроз, моделі порушника та моделі проведення оцінки захищеності АБС. Розглянуто модель оцінки економічної доцільності впровадження того чи іншого механізму технічних засобів захисту інформації (ТЗЗІ) в АБС ОБС в залежності від цінності БІн.

Практичне питання забезпечення конфіденційності і цілісності медичної інформації розглянуто у розділі 15. Зараз темпи розвитку інформаційних технологій, медичного діагностичного обладнання значно випереджають розробку відповідних законодавчих норм. Тому актуальним питанням є розвиток технологій захисту медичної інформації, яка подається у цифровому вигляді. У розділі розглянуто метод ідентифікації і верифікації медичних діагностичних зображень, заснований на реверсивних цифрових знаках (ЦВЗ). Метод був реалізований і протестований з використанням реальних даних. На підставі методу розроблено систему, яка дозволяє проводити подальші дослідження.

Поруч із цим у розділі 16 розглянуті завдання управління ризиками розробки програмного забезпечення за умови обмеженості ресурсів (фінансових, технічних та ін.), які призначаються для усунення помилок безпеки. Розглядається у вигляді напівмарковської моделі прийняття рішень для процесу, який управляється у неперервному часі, з критерієм мінімуму витрат на усунення

аномалій. У роботі пропонується метод управління ризиками розробки програмного забезпечення (ПЗ). В якості прикладу розглянуті ситуації виникнення помилок безпеки програмного забезпечення, і визначена оптимальна стратегія управління для усунення зазначеної аномальної ситуації. Слід відзначити, що наведений в розділі метод доцільно використовувати не тільки при управлінні ризиками безпеки ПЗ, але й у разі функціонального, навантаженого, стресового та інших видах тестування для запобігання можливих втрат.

Слід підкреслити важливість завдань перетворення форми інформації у інфокомунікаціях. Реалізація системної функції перетворення форми інформації є вирішальною при визначенні докладності первинного перетворення, достовірності та інтенсивності інфообміну. Тому розділ 17 присвячено рішенням завдань аналізу системних властивостей та обґрунтуванню ефективності методів перетворення форми інформації. Здійснено систематизацію та аналіз базових методів перетворення форми інформації, визначено властивості та обґрунтовано ефективність їх застосування для забезпечення системних функцій інфотехнології. Проаналізовано класи дискретно гармонічних та дискретно нерегулярних систем функцій та базисів.

При перетворенні форми інформації (ПФІ) вирішальним є вибір методу кодування цифрових повідомлень. На стадії аналізу та обґрунтування вибору методу ПФІ ефективним є застосування методології моделювання процесів перетворення повідомлень. В роботі, результати якої наведені у розділі 18, проаналізовано ефективність застосування двійкових методів ПФІ з надлишковими операціями як способу вирішення проблеми невизначених областей в двійкових методах з лінійними операціями перетворення. Побудовано векторно-розгалужені діаграми, які дозволяють здійснити моделювання, візуалізацію та оцінку процесів ПФІ. Визначено кількісні параметри ефективності кожного з методів та вказано оптимальний.

Слід підкреслити, що у монографії приділено достатньо уваги прогресивним засобам та методам забезпечення захисту інформації та збереженню її цілісності, поруч із розділами, де розглянуто вже практичні аспекти застосування теорії та практики ІТ в економіці, екології, медицині та освіті. Автори цих розділів також звертають увагу на проблеми застосування ІТ у визначених галузях.

Так, у розділі 22 розглядаються питання розвитку ІТ-галузі в Україні, як однієї з інвестиційно привабливої галузі, на основі міжнародних рейтингів розвитку інформаційного суспільства. Визначено пріоритетні напрямки розвитку ІТ в Україні. У роботі відзначається, що сучасне інформаційне суспільство базується на розвитку людських ресурсів, а саме їх креативності, яка є основою інноваційних ІТ. Проведено аналіз позицій України в глобальному рейтингу за конкурентоспроможністю талантів.

Поруч із цим ІТ відіграють важливу роль у інфраструктурі сучасного виробничого підприємства. Одним з напрямків максимізації прибутку виробництва є застосування виробничих функцій. Виробнича функція – це економетрична модель, що відображає залежність показників виробничо-господарської діяльності від факторів, що визначають ці показники. Робота, результати якої наведено у розділі 20, присвячена дослідженню екстремуму функції для оптимізації функціонування виробничого підприємства за допомогою виробничої функції Кобба-Дугласа. В результаті отримана залежність максимального значення функції прибутку двохресурсного підприємства від показників цієї функції. Аналіз такої залежності дає можливість оцінити значення цих параметрів, при яких функція прибутку досягне максимуму.

У розділі 24 запропонована структура EPM (Enterprise Performance Management) – системи підприємства, що входить до холдингу. Розглянуто процес розробки ключових показників діяльності холдингу до бюджетних показників підприємств, які входять до визначеного холдингу. Каскадування передбачає декомпозицію ключових показників холдингу на стратегічному періоді на ключові показники діяльності підприємств по роках. Річні показники далі декомпонуються до показників бізнес-процесів операційної, фінансової та інвестиційної діяльності, а потім на їх підставі формуються показники бюджетів підприємства.

Впровадження прогресивних технологій на промислових підприємствах призводить до модернізації елементів виробничої системи іншими, більш сучасними. Проте будь-який процес впровадження вимагає значних витрат коштів, тому керівництву підприємства необхідно з розумінням підійти до вирішення цього питання. У роботі, за результатами якої у розділі 23, на основі авторських методичних рекомендацій, наведені розрахунки вартості розробки впровадження технологій електронного бізнесу на промисловому підприємстві. Для спрощення розрахунків пропонується спеціальне програмне забезпечення, розроблене у середовищі Java.

Іншою актуальною проблемою, що має вплив на економіку й екологічний стан міст та регіонів країни є рішення завдань прогнозування міського добового попиту на водні ресурси, що є основою для прийняття оперативних, тактичних і стратегічних рішень комунальними підприємствами (міськими водоканалами), що надають послуги з постачання води населенню. У розділі 19 розглядаються методи та моделі розрахунку прогнозу міського водоспоживання та запропоновано алгоритм прогнозування міського добового водоспоживання, заснований на понятті однотипних днів за попередні роки. Алгоритм, що пропонується, був реалізований у вигляді програмного комплексу та пройшов багаторічну перевірку в реальних умовах, яка підтвердила його ефективність.

Для галузі надання дистанційних послуг в телемедицині у розділі 21 наведено результати розробки відповідної мультиагентної системи. Одним із ключових компонентів системи, що пропонується, є нейронна мережа, яка використовується для постановки діагнозу на основі обробки медико-реєстраційної інформації та зображень. Розроблений метод адаптації Інтернет-сервісу враховує переваги користувачів і дозволяє автоматично налаштовувати контент медичного порталу.

Поруч із застосуванням теорії та практики ІТ в економіці, екології, медицині слід також звернути увагу на проблему підготовки відповідних фахівців та на загальні проблеми освіти. В роботі, якій присвячено розділ 25, запропоновано підхід до формалізованого опису процесу визначення рівня знань студентів шляхом тестування. Враховуючи складність опису звичайними математичними моделями нечіткої, неповної і різномірної інформації, пропонується для визначення результатів тестування використання методів штучного інтелекту, а саме використання нечітких мір і множин. Еталонні та поточні результати тестування представляються у вигляді нечітких LR-інтервалів, а процес логічного виводу на формалізованих знаннях організовується за допомогою неоднорідної функціональної мережі.

Окремим напрямом монографії є огляд розробок та теоретичних основ, які застосовуються у комп'ютеризованих технологіях та системах видавничо-поліграфічних виробництв та електронних мультимедійних виданнях.

Технологія тривимірного друку поки що є екзотикою не тільки для України, але й для багатьох інших країн. Тим не менш, фахівці запевняють, що адаптація до цих пристроїв пройде дуже швидко. Більше того, вже сьогодні 3D принтери активно використовуються в медицині, авіації, аерокосмонавтиці і, навіть, харчовій індустрії. У розділі 26 розглянуто теоретичні та практичні проблеми побудови тривимірних моделей реальних об'єктів та пропонується оригінальне вирішення науково-технічної задачі вдосконалення доступної технології тривимірного сканування та технології створення тривимірних моделей за допомогою цифрового фотоапарату.

В розділі 27 розглядаються методи класифікації, що застосовуються при стисненні файлу з бітональним зображенням тексту, отриманим скануванням або цифровим фотографуванням. Особлива увага звертається на використання при цьому заходи щодо визначення відмінності двох зображень символів, виділених із зображення тексту. Також розглянуто методи прискорення алгоритмів, що класифікують зображення символів, та покращують читання тексту, відновленого після стиснення зображення.

Таким чином, у монографії відбиваються результати наукових досліджень із застосування та впровадження ІТ у різних галузях промисловості України та зарубіжжя, визначаються напрямки ефективних рішень щодо поставлених



проблем, а також розглядаються перспективи розвитку сучасних інформаційних технологій та систем.

Монографія підготовлена колективом авторів у наступному складі:

1. Аксак Н. Г., Харківський національний економічний університет імені Семена Кузнеця, кандидат технічних наук, доцент – розділ 21;

2. Алексієв В. О., Харківський національний економічний університет імені Семена Кузнеця, доктор технічних наук, професор, – розділ 2 (у співавторстві);

3. Алексієв О. П., Харківський національний автомобільно-дорожній університет, доктор технічних наук, професор – розділ 2 (у співавторстві);

4. Альошин Г. В., Українська державна академія залізничного транспорту, доктор технічних наук, професор кафедри транспортного зв'язку – розділ 1 (у співавторстві);

5. Безсонов О. О., Харківський національний економічний університет імені Семена Кузнеця, кандидат технічних наук, доцент – розділ 6 (у співавторстві);

6. Бринза Н. О., Харківський національний економічний університет імені Семена Кузнеця, кандидат технічних наук, доцент кафедри ІКТ – розділ 23 (у співавторстві);

7. Вільхівська О. В., Харківський національний економічний університет імені Семена Кузнеця – розділ 23 (у співавторстві);

8. Дудикевич В. Б., Національний університет “Львівська політехніка”, доктор технічних наук, професор, завідувач кафедри захисту інформації – розділ 11 (у співавторстві);

9. Євсєєв С. П., Харківський національний економічний університет імені Семена Кузнеця, кандидат технічних наук, старший науковий співробітник, доцент кафедри інформаційних систем – розділ 13 (у співавторстві);

10. Задачин В. М., Харківський національний економічний університет імені Семена Кузнеця, кандидат фізико-математичних наук, доцент кафедри інформаційних систем – розділ 19;

11. Засядько А. А., Черкаський навчально-науковий інститут ДВНЗ “Університет банківської справи”, доктор технічних наук, професор. – розділ 20 (у співавторстві);

12. Затхей В. А., Харківський національний економічний університет імені Семена Кузнеця, кандидат технічних наук, доцент – розділ 25 (у співавторстві);

13. Іванов В. Г., Національний юридичний університет імені Ярослава Мудрого, доктор технічних наук, професор – розділ 3, розділ 5;

14. Ізонін І. В., Національний університет “Львівська політехніка”, кандидат технічних наук – розділ 26 (у співавторстві);

15. Коваленко А. А., Харківський національний економічний університет імені Семена Кузнеця, кандидат технічних наук, доцент – розділ 4;
16. Коваленко О. В., Центральноукраїнський національний технічний університет, кандидат технічних наук, доцент кафедри програмування та захисту інформації – розділ 16 (у співавторстві);
17. Коломійцев О. В., Харківський національний університет Повітряних Сил імені Івана Кожедуба, доктор технічних наук, начальник науково-дослідного відділу наукового центру Повітряних Сил – розділ 1 (у співавторстві);
18. Король О. Г., Харківський національний економічний університет імені Семена Кузнеця, кандидат технічних наук, доцент кафедри інформаційних систем – розділ 13 (у співавторстві);
19. Королюк С. С., Черкаський навчально-науковий інститут ДВНЗ “Університет банківської справи”, кандидат фізико-математичних наук, доцент – розділ 20 (у співавторстві);
20. Коц Г. П., Харківський національний економічний університет імені Семена Кузнеця, кандидат економічних наук, доцент кафедри інформаційних систем – розділ 22 (у співавторстві);
21. Кузнецова Л. А., Одеський національний політехнічний університет, старший викладач кафедри інформаційної безпеки – розділ 15 (у співавторстві);
22. Ломоносов Ю. В., Національний юридичний університет імені Ярослава Мудрого, кандидат технічних наук, доцент кафедри криміналістики – розділ 27;
23. Лосев М. Ю., Харківський національний економічний університет імені Семена Кузнеця, кандидат технічних наук, доцент кафедри інформаційних систем – розділ 8;
24. Лотошинська Н. Д., Національний університет “Львівська політехніка”, кандидат технічних наук, доцент – розділ 26 (у співавторстві);
25. Маракова І. І., Одеський національний політехнічний університет, доктор технічних наук, професор – розділ 15 (у співавторстві);
26. Мельник М. О., Одеський національний політехнічний університет, кандидат технічних наук, доцент кафедри інформаційної безпеки – розділ 14 (у співавторстві);
27. Микитин Г. В., Національний університет “Львівська політехніка”, доктор технічних наук, професор, – розділ 11 (у співавторстві);
28. Мінухін С. В., Харківський національний економічний університет імені Семена Кузнеця, доктор технічних наук, професор кафедри інформаційних систем – розділ 7;

29. Моїсеєнко А. С., Харківський національний економічний університет імені Семена Кузнеця, аспірант кафедри інформаційних систем – розділ 22 (у співавторстві);

30. Москаленко В. В., Національний технічний університет “ХПІ”, кандидат технічних наук, доцент – розділ 24 (у співавторстві);

31. Нікітін Г. Д., Одеський національний політехнічний університет – розділ 14 (у співавторстві);

32. Петришин Л. Б., АГН науково-технологічний університет, Краків, Польща, доктор технічних наук, професор – розділ 17;

33. Петришин М. Л., Прикарпатський національний університет ім. В. Стефаника, аспірант – розділ 18 (у співавторстві);

34. Різник О. Я., Національний університет “Львівська політехніка”, кандидат технічних наук, доцент кафедри інформаційних технологій видавничої справи – розділ 9;

35. Ровінський В. А., Прикарпатський національний університет ім. В. Стефаника, кандидат технічних наук, доцент – розділ 18 (у співавторстві);

36. Руденко О. Г., Харківський національний економічний університет імені Семена Кузнеця, доктор технічних наук, професор, завідувач кафедри інформаційних систем – розділ 6 (у співавторстві);

37. Смірнов О. А., Центральноукраїнський національний технічний університет, доктор технічних наук, професор, завідувач кафедри програмування та захисту інформації – розділ 16 (у співавторстві);

38. Тесленко О. В., Харківський національний економічний університет імені Семена Кузнеця, кандидат технічних наук, доцент – розділ 25 (у співавторстві);

39. Удовенко С. Г., Харківський національний економічний університет імені Семена Кузнеця, доктор технічних наук, професор, завідувач кафедри інформатики та комп’ютерних технологій – розділ 10 (у співавторстві);

40. Фонта Н. Г., Національний технічний університет “ХПІ”, кандидат технічних наук, доцент – розділ 24 (у співавторстві);

41. Хорошко В. О., Національний авіаційний університет, доктор технічних наук, професор – розділ 12 (у співавторстві);

42. Хохлачова Ю. Є., Національний авіаційний університет, кандидат технічних наук, доцент кафедри БІТ – розділ 12 (у співавторстві);

43. Чала Л. Е., Харківський національний університет радіоелектроніки, кандидат технічних наук, доцент кафедри штучного інтелекту – розділ 10 (у співавторстві).

Кафедра інформаційних систем Харківського національного економічного університету імені Семена Кузнеця висловлює подяку всім дослідникам, які взяли участь в підготовці та публікації монографії.

## РОЗДІЛ 1

### ПРОБЛЕМИ СТВОРЕННЯ БАГАТОФУНКЦІОНАЛЬНОЇ ЛАЗЕРНОЇ СИСТЕМИ КОНТРОЛЮ І УПРАВЛІННЯ ЛІТАЛЬНИМ АПАРАТОМ

**Анотація** Запропонована селекція подовжніх мод лазера-передавача для багатофункціональної лазерної системи контролю і управління літальним апаратом (ЛА) та їх використання для суміщення інформаційних і шести вимірювальних каналів, які забезпечують високу точність вимірювання параметрів руху (ВПР) ЛА за рахунок взаємозв'язку параметрів, що вимірюються, нових принципів використання високої стабільності частот міжмодових биттів (МБ) лазерного випромінювання (ЛВ), частотно-часового методу (ЧЧМ) вимірювання, методів «підкрашування» різними частотами МБ і сканування динамічними діаграмами спрямованості (ДС) ЛВ.

**Ключові слова:** багатофункціональна лазерна система, літальний апарат, суміщення каналів, продовжні моди, точність вимірювання параметрів руху

**Abstract.** The selection of longitudinal fashions of laser-transmitter is offered for the multifunction laser checking and management system by an aircraft (A) and their use for combination of informative and six measuring channels, that provide high exactness of measuring of parameters of motion of A due to intercommunication of measureable parameters, new principles of the use of high stability of frequencies between frequencies of beatings (BB) of laser radiation (LR), frequency-temporal method of measuring, methods of «tinting» different frequencies of BB and scan-out by the dynamic diagrams of orientation LR.

**Keywords:** multifunction laser system, aircraft, combination of channels, longitudinal fashions, exactness of measuring of parameters of motion

Концентрація енергії у лазерному випромінюванні (ЛВ), стабільність частот, що випромінюються дозволяють створити більш багатофункціональну, компактну та з високою точністю вимірювання параметрів руху (ВПР) літального апарату (ЛА) лазерну систему (ЛС), ніж системи радіодіапазону. Однак, для цього треба подолати наступні науково-технічні проблеми:

- 1) електромагнітної сумісності багатьох вимірювальних (за параметрами руху ЛА) та інформаційного каналів;
- 2) підвищення точності вимірювання похилої дальності (відстані) до ЛА;
- 3) підвищення точності вимірювання швидкості (радіальної) ЛА;
- 4) підвищення точності вимірювання кутів азимута і місця ЛА;

5) підвищення точності вимірювання кутових швидкостей ЛА;

6) підвищення стійкості і точності автоматичного супроводження ЛА при сумісній обробці параметрів його руху, а також їх похідних за фільтрацією;

7) підвищення ефективності передачі інформаційних (телеметричних) каналів тощо.

При подоланні цих проблем багатофункціональна (однопунктна) ЛС отримує ряд переваг перед існуючими системами радіо і оптичного діапазонів довжин хвиль:

- висока точність вимірювання шести параметрів руху (траєкторні вимірювання) ЛА (також за рахунок використання їх взаємозв'язку);

- висока стійкість кутового автоматичного супроводження ЛА за напрямком (АСН);

- висока швидкість передачі інформації на ЛА;

- об'єктивний контроль ЛА у денних і нічних умовах та, за необхідністю, його розпізнавання;

- обробка, збереження, відображення та передача інформації, що отримується під час проведення випробувань ЛА;

- висока економічна ефективність;

- висока точність геодезичної прив'язки;

- висока точність прив'язки до системи єдиного часу;

- високі надійність, мобільність тощо.

Основні переваги багатофункціональної ЛС засновані на використанні селекції подовжніх мод лазера-передавача та їх використання для суміщення інформаційних і шести вимірювальних каналів. Висока точність вимірювальних каналів системи забезпечується за рахунок використання нових та вдосконалених методів:

- формування лазерних сигналів, що зондують багатофункціональною ЛС;

- «підкрашування» різними частотами міжмодових биттів динамічних діаграм спрямованості (ДС) ЛВ;

- вимірювання кутів азимута і місця ЛА, похилої дальності, радіальної і кутових (тангенціальних) швидкостей;

- взаємозв'язку вимірювальних параметрів руху ЛА;

- частотно-часового методу вимірювання;

- вузькосмугової фільтрації тощо.

Багатофункціональність радіотехнічних систем (РТС) досягається використанням відповідного числа каналів, які виконують свої функції та застосуванням окремих ортогональних або суміщених сигналів. Завдяки цьому

РТС обходиться тільки однією сукупністю антено-передавальних та приймальних пристроїв, що дуже економічно, за зрівнянням з сукупністю однофункціональних систем, потребує лише однієї геодезичної прив'язки і дає можливість взаємно зв'язувати вимірювальні параметри руху ЛА. Такими якостями частково володіють, наприклад, супутникові системи зв'язку з неекваторіальними орбітами.

На відміну від РТС, лазерні вимірювальні системи, які працюють з динамічними об'єктами (ЛА) мають обмежену можливість передачі декількох різнорідних каналів в одному ЛВ, оскільки можлива лише амплітудна і поляризаційна модуляція сигналу. Навіть для декількох інформаційних каналів можливий лише часовий розподіл сигналів з синхронізацією у загальній смузі в одиницю Гігабайтів, хоча стаціонарний лазер може мати хвильову мультиплікацію до 40 Гб.

Таким чином, пропонується суміщення та селекція каналів (інформаційного і вимірювальних) багатofункціональної ЛС контролю і управління ЛА за рахунок використання подовжніх мод і частот міжмодових биттів ЛВ, а також розширення функціональних можливостей системи у денний і нічний час доби за рахунок використання оптико-електронного модуля (ОЕМ), який складений з телевізійного і інфрачервоного каналів.

Оскільки багатofункціональна ЛС володіє такими якостями як понад вузькі діаграмами спрямованості (ДС) ЛВ, висока стабільність подовжніх мод і частот міжмодових биттів ЛВ, то з'явилися нові можливості більш точного вимірювання шести параметрів руху ЛА і швидкої передачі інформації, навіть кутовою модуляцією. Використання в системі ширококутового телевізійного каналу підвищує стійкість кутового автоматичного супроводження ЛА.

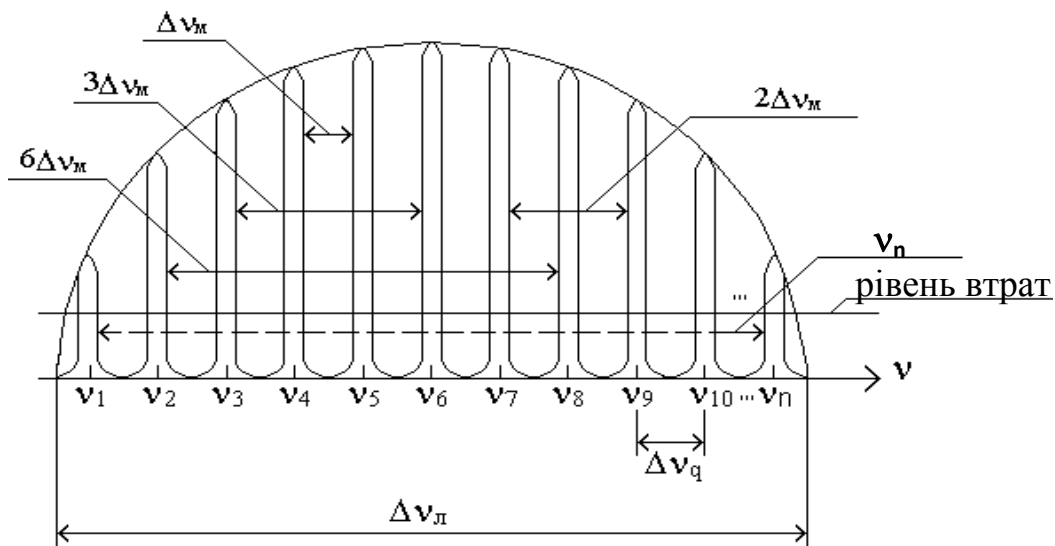
Однак, враховуючи небажаний вплив атмосфери на якість роботи багатofункціональної ЛС в цілому, її застосуванням може бути, як приклади, використання під час проведення випробувань сучасних зразків ЛА, старту ракет при кутовому розносі факела двигуна і променя, морського старту ракетно-носія космічного апарату з екваторіальної зони, в автоматизованій системі зліту і посадки ЛА тощо.

### **Проблема електромагнітної сумісності інформаційних та вимірювальних каналів багатofункціональної лазерної системи**

Проблема електромагнітної сумісності, тобто розв'язки каналів в РТС, звичайно вирішується за рахунок використання таких параметрів селекції, як часова, частотна, кутова, фазова, поляризаційна, структурна (ортогональність) та комплексна селекція. Для цього РТС повинна мати додаткові пристрої:

- систему синхронізації;
- формувач еталонних частот;
- набір кутових каналів;
- поляризатори тощо.

Для багатофункціональної ЛС, що пропонується доцільна частотна селекція каналів, тому що у спектрі одномодового багаточастотного лазера-передавача вже є набір частот (подовжніх мод). Причому, рознесення цих частот, яке зветься міжмодовими биттями, достатньо велике для доброї селекції. Тому, з виходу лазера-передавача системи за допомогою модифікованого селектора подовжніх мод [72] виділяються зі спектра одномодового багаточастотного з синхронізацією подовжніх мод ЛВ (рис. 1.1) необхідні подовжні моди, різниця частоти яких відповідає кожному каналу.

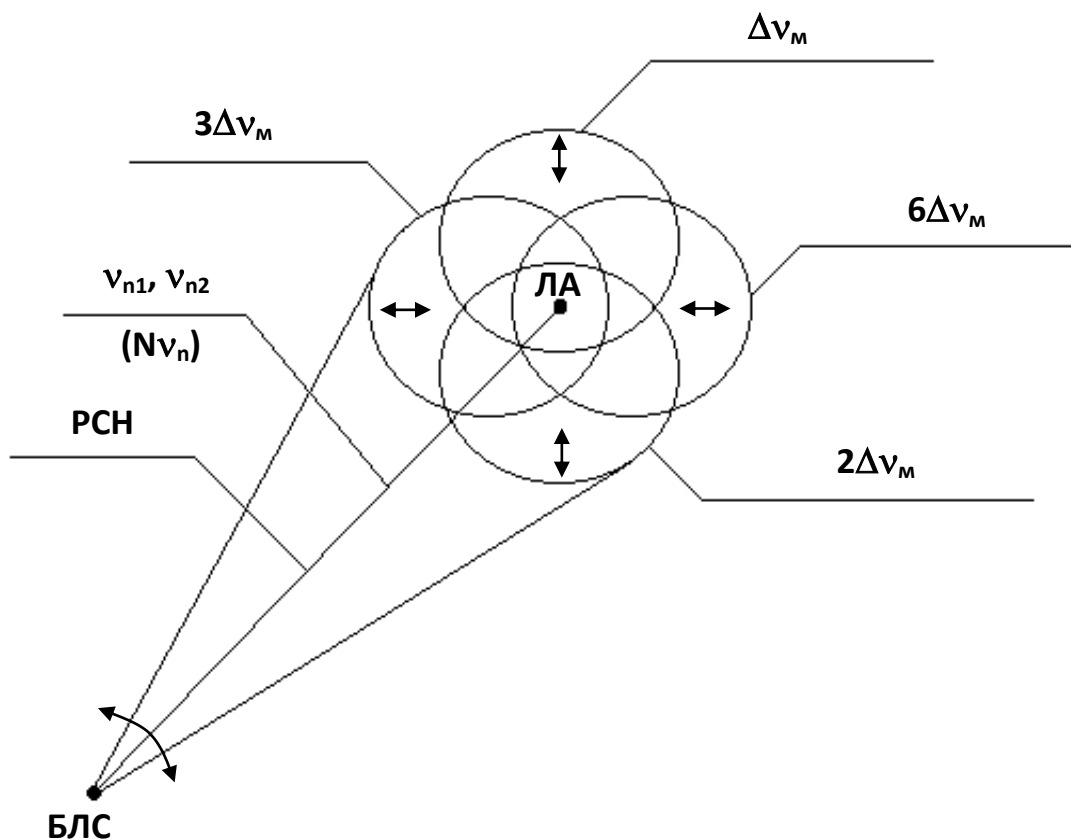


**Рис. 1.1. Виділення зі спектра одномодового багаточастотного із синхронізацією подовжніх мод лазерного випромінювання подовжніх мод**

При цьому розширюються можливості багатофункціональної ЛС для будь-якої модуляції інформаційного (телеметричного) каналу модуляцією подовжньої моди, для «підкрашування» (розпізнавання) каналу з рухомими ДС ЛВ, для зрівняння доплерівської міжмодової частоти з такою ж опорною. Причому набір опорних частот береться з того ж лазера-передавача.

Динамічна картина ДС ЛВ, що сканують у двох ортогональних площинах представлена на рис. 1.2. Дві пари парціальних ДС ЛВ («підкрашені» різницею частотами міжмодових биттів) рухаються у двох ортогональних площинах, для яких використовуються комбінації частот подовжніх мод:

$$\Delta\nu_{54}=\nu_5-\nu_4=\Delta\nu_m, \quad \Delta\nu_{97}=\nu_9-\nu_7=2\Delta\nu_m, \quad \Delta\nu_{63}=\nu_6-\nu_3=3\Delta\nu_m, \quad \Delta\nu_{82}=\nu_8-\nu_2=6\Delta\nu_m.$$



*Рис. 1.2. Сканування чотирма парціальними ДС ЛВ у двох ортогональних площинах та сумарною ДС ЛВ*

Лазерні сигнали  $N$  інформаційних каналів для зв'язку з ЛА використовують окремі несучі частоти (подовжні моди).

Таким чином, кожний просторовий вимірювальний канал багатофункціональної ЛС створюється незалежно парою подовжніх мод та відрізняється різницею оптичних несучих, тобто міжмодовими биттями, якими можна також «підкрашувати» динамічні за кутами канали. Це дозволяє створити селекцію каналів і використати частотно-часовий метод вимірювання кутів та швидкостей ЛА [46].

Діаграми спрямованості ЛВ рухаються в кожній з двох ортогональних площин за кутами азимута  $\alpha$  і місця  $\beta$  (рис. 1.2) періодично назустріч і навпаки та зміщуються на півщини ДС ЛВ. Для їх руху використовується модифікований блок дефлекторів на передавальному кінці оптичної лінії.

Оскільки динамічні ДС ЛВ незалежні, інформаційний сигнал на несучих частотах  $Nv_n$  та лазерні сигнали з модуляцією поляризації  $(v_{n1}, v_{n2})$ , що необхідні для розпізнавання ЛА, проходять у вздовж рівносигнального напрямку (РСН).



### **Проблема підвищення точності вимірювання похилої дальності (відстані) до літального апарату багатофункціональною лазерною системою**

Для рішення проблеми підвищення точності вимірювання похилої дальності до ЛА на старті можуть застосовуватись широкі можливості комбінацій частот міжмодових биттів для формування потрібного набору гармонік, які необхідні для фазового багатошкального методу вимірювання з використанням більш завадостійкої їх вузькосмугової фільтрації і точних цифрових методів вимірювання. У якості опорних частот використовуються ті ж гармоніки безпосередньо з лазера-передавача (або традиційні методи модуляції).

### **Проблема підвищення точності вимірювання радіальної швидкості літального апарату багатофункціональною лазерною системою**

Вимірювання радіальної швидкості ЛА доцільно здійснювати за ефектом Доплера та методом цифрового підрахунку частоти на оптимальному вимірювальному інтервалі часу на великих відстанях. Але здвиг частоти на оптичній несучій може змінюватись у широкому діапазоні. Тому смугу фільтрації після фотодетектора (ФТД) багатофункціональної ЛС треба брати великою, що не покращує завадостійкість, або для вузькосмугового каналу підвищувати час пошуку сигналу фазовою автоматичною підстройкою частоти (ФАПЧ). Підбором оптимальних зондуючих частот міжмодових биттів можна підвищити якість вимірювання радіальної швидкості ЛА.

### **Проблема підвищення точності вимірювання кутів азимута і місця літального апарату багатофункціональною лазерною системою**

Проблему підвищення точності вимірювання кутів азимута і місця ЛА доцільно вирішувати наступним чином. Зустрічне сканування пар парціальних ДС ЛВ в кожній з двох ортогональних площин, забезпечує на приймальній стороні багатофункціональної ЛС можливість вимірювання кутів азимута і місця за рахунок зрівняння здвигів періодів огинаючих різницевої частоти міжмодових биттів за один повний прохід ДС ЛВ у прямому та зворотному напрямку сканування. Часову тривалість різниці здвигів періодів огинаючих сканування ДС ЛВ раціонально вимірювати методом заповнення цього інтервалу еталонними імпульсами, що формовані, наприклад, з першої частоти міжмодових биттів. Це дає можливість обчислювати половину і знак різниці здвигу огинаючих періодів (пульсацій) та формувати сигнали похибки за двома вісями координат (перестроювати РСН за допомогою дефлекторів) і визначати кути азимута  $\alpha$  і місця  $\beta$  ЛА каналом АСН.

Точність оцінки кутового положення ЛА відносно РСН залежить від крутизни пульсацій міжмодових биттів, від завад та від похибки дискретності.

Виконавчі механізми за кутами повертають приймально-передавальну платформу багатофункціональної ЛС таким чином, щоб ЛА находився на РСН чотирьох парціальних ДС ЛВ.

Цифрові дані вимірювань, отримані від всіх параметрів руху ЛА, дають змогу їх обробки у єдиної обчислювальної машині системи.

Відомо, що точність вимірювання кутів азимута і місця ЛА тим вище, чим крутіше дискримінаційна крива. Максимальне значення крутизни дискримінатора обирається з залежності рівня сигналу (ДС ЛВ) від часу:

$$G(\theta) = G_0 \cdot \exp\left\{\frac{-\theta^2(t)}{2\Delta\theta_x^2}\right\} = G_0 \cdot \exp\left[-\frac{k^2 t^2}{2\Delta\theta_x^2}\right], \quad (1)$$

де  $k = \frac{2\Delta\theta_x}{T_{ск}}$  – швидкість сканування ДС ЛВ;  $\Delta\theta_x = \frac{k \cdot T_{ск}}{2}$  – ширина ДС ЛВ;  $T_{ск}$  –

час сканування ДС ЛВ;  $G_0$  – максимальний рівень сигналу;  $\theta_{max} = \frac{\Delta\theta_x}{2}$  – максимальний кут сканування ДС ЛВ (відхилення від РСН).

Перша похідна у часі від виразу (1) відповідає крутизни дискримінатора:

$$G'_t(t) = \frac{dG}{dt} = G_0 \cdot \exp\left(-\frac{k^2 t^2}{2\Delta\theta_x^2}\right) \cdot \left(-\frac{2k^2 t}{2\Delta\theta_x^2}\right). \quad (2)$$

Оптимальне значення розузгодження  $\Delta\theta_{opt}$  за одним з кутів, при якому крутизна найбільша, дорівнює половині ДС ЛВ ( $\Delta\theta_x$ ):

$$\Delta\theta_{opt} = k \cdot t_{max} = \Delta\theta_x \quad \text{за час} \quad t_{maxG} = \frac{\Delta\theta_x}{k} = \frac{\Delta\theta_x \cdot T_{ск}}{2\Delta\theta_x} = \frac{T_{ск}}{2}.$$

Відносний рівень сигналу при такому розузгодженні дорівнює:

$$\frac{G}{G_0} = e^{-\frac{k^2 \cdot T_{ск}^2}{4 \cdot 2\Delta\theta_x^2}} = e^{-\frac{\Delta\theta_x}{2\Delta\theta_x^2}} = e^{-\frac{1}{2}} = \frac{1}{\sqrt{e}}. \quad (3)$$

Максимальна крутизна дискримінаційної характеристики:

$$G' = G_0 \cdot \exp\left(-\frac{k^2 T_{ск}^2}{4 \cdot 2\Delta\theta_x^2}\right) \cdot \left(-\frac{2k^2}{2\Delta\theta_x^2} \cdot \frac{T_{ск}}{2}\right) = -\frac{G_0 k}{\Delta\theta_x} \cdot \exp\left(-\frac{1}{2}\right), \quad (4)$$

де  $U_c = U_m \cdot G_0 \Rightarrow U'_c = U_m \cdot G'_0$ ;  $U_c$  – огибающая сигналу;  $U_m$  – максимальне значення огибающей.

Середньоквадратична похибка (СКП) можна записати, як:

$$\sigma_{t_1}^2 = \frac{\sigma_{ш}^2}{\left(U_{\Delta\theta_x}^1\right)^2} = \frac{\sigma_{ш}^2}{U_{\max}^2 \frac{1}{e} \left(\frac{G_0 k}{\Delta\theta_x}\right)^2} = \frac{e \cdot \Delta\theta_x^2}{q \cdot k^2}, \quad (5)$$

де  $q = \frac{(U_m G_0)^2}{\sigma_{ш}^2}$  – відношення сигнал/шум за потужністю;  $\sigma_{ш}^2$  – шуми

фотодетектора;  $G'(\theta)$  – крутизна дискримінатора.

Оскільки СКП  $\sigma_{t_1}^2 = \sigma_{t_2}^2$ , то  $\sigma_{\Delta T}^2 = 2\sigma_{t_1}^2$ . Тоді:

$$\sigma_{t_1}^2 = \frac{\sigma_{ш}^2}{U_m^2 (G'_t)^2} = e \cdot \frac{\sigma_{ш}^2}{U_m^2 G_0^2 \frac{k^2}{\Delta\theta_x^2}} = \frac{e}{4} \cdot \frac{T_{ск}^2}{q}, \text{ так як } \frac{\Delta\theta_x}{k} = \frac{T_{ск}}{2}.$$

Звідси:

$$\sigma_{\Delta T}^2 = \frac{e}{2} \cdot \frac{T_{ск}^2}{q}. \quad (6)$$

З урахуванням  $\Delta\theta_x = \frac{kT_{ск}}{2}$ , СКП положення 2-х фронтів:

$$\sigma_{\Delta T}^2 = \frac{e}{2q} \cdot \left(\frac{2\Delta\theta_x}{k}\right)^2 = 2e \cdot \frac{\Delta\theta_x^2}{q \cdot k^2}, \text{ отже: } \sigma_{\theta}^2 = \frac{\Delta\theta_x^2}{4 \cdot T_{ск}^2} \cdot \sigma_{\Delta T}^2 = \frac{k^2 \cdot \sigma_{\Delta T}^2}{16}.$$

Можна показати, що СКП кута відхилення:

$$\sigma_{\theta}^2 = \frac{\Delta\theta_x^2}{4 \cdot T_{ск}^2} \cdot \frac{e}{2} \cdot \frac{T_{ск}^2}{q} = \frac{e}{8} \cdot \frac{\Delta\theta_x^2}{q}. \quad (7)$$

Для випадків, коли потребується пошук сигналу в широкому апріорному діапазоні кутів, він реалізується перестройкою або багатошкальним методом.

### **Проблема підвищення точності вимірювання кутової швидкості літального апарату багатофункціональною лазерною системою**

Оскільки високі кутові швидкості при супроводженні сучасних ЛА потребують високих темпів оновлення інформації, які становляться недосяжними для вузької ДС ЛВ багатофункціональної ЛС за причиною зриву супроводження, то проблема вирішується підвищеними вимогами до високій ймовірності утримання РСН на ЛА, за рахунок підвищення поля зору ЛВ, або підвищенням точності супроводження ЛА за рахунок використання вимірювання його кутової швидкості. Підвищення надійності і точності супроводження ЛА доцільно здійснювати за двома факторами: звуженням ДС ЛВ, тобто підвищенням енергетичного потенціалу та використанням інформації

про кутову швидкість. Для цього потрібно застосувати поданий прямий метод вимірювання кутових швидкостей ЛА [37].

Треба відмітити, що в звичайних вимірювальних системах радіо і оптичного діапазонів хвиль безпосереднє вимірювання кутових швидкостей ЛА невідоме. Однак, далі поданий такий можливий принцип [37] за рахунок зустрічного сканування парами ДС ЛВ в двох ортогональних площинах.

Головна ідея полягає у тому, що напівперіоди огинаючої ДС ЛВ відбитого сигналу міжмодових биттів від ЛА, що співпадає з напрямком руху ДС ЛВ, мають більшу довжину, а при зворотному русі (не співпадаючому з рухом ЛА) – більш коротше. Різниця довжини напівперіодів огинаючої пропорційна тангенціальній швидкості руху ЛА:

$$V_{\tau} = \frac{V_{\text{л}}}{2} \cdot \frac{\Delta T_{\text{р}}}{T_{\text{ск}}}, \quad (8)$$

де  $\Delta T_{\text{р}}$  – різниця часових інтервалів (результуючий час);  $V_{\text{л}}$  – швидкість сканування (руху) ДС ЛВ.

Залежність тангенціальної швидкості від  $V_{\text{л}}$  має вигляд:

$$V_{\tau} = \frac{V_{\text{л}}}{2} \cdot \frac{\gamma}{1 - \gamma^2}, \quad \text{де } \gamma = \frac{\Delta T_{\text{р}}}{T_{\text{ск}}} < 1.$$

В меншому діапазоні  $0 < \gamma < 0,1$ :  $V_{\tau} \approx \frac{V_{\text{л}}}{2} \Delta T$ .

Для стабільних  $V_{\text{л}}$  та  $T_{\text{ск}}$  СКП тангенціальної швидкості:

$$\sigma_{\tau}^2 = \frac{V_{\text{л}}^2}{4T_{\text{ск}}^2} \sigma_{\Delta T}^2. \quad (9)$$

Оцінки вимірювань тангенціальної швидкості ЛА та дисперсії похибки у двох площинах ідентичні, але вимірювання здійснюється на своїх частотах міжмодових биттів. Для того, щоб виключити залежність від дальності, можна взяти від автоматичного регулювання підсилення у якості порога половину пікової напруги та використовувати її для порога виявлення міжмодових биттів.

### **Проблема підвищення стійкості і точності супроводження літального апарата при сумісній обробці його параметрів руху та додаткової фільтрації**

В багатофункціональній ЛС можливо врахувати не тільки прямі оцінки шістьох параметрів руху ЛА, але і їх уточнення за рахунок використання взаємозв'язків параметрів руху та їх похідних. Більше того, при звуженні ДС ЛВ та при відповідному збільшенню енергетичного потенціалу, визначення кутових швидкостей є пріоритетним, для чого потрібне підвищення стійкості та

точності супроводження ЛА за напрямком. Крім того, підвищення енергетичного потенціалу і забезпечення потрібної точності супроводження ЛА можливо за допомогою звуження смуги пропускання за рахунок застосування відповідних фільтрів, наприклад, фільтром Калмана-Бьюсі.

Звужуючи таким чином смугу пропускання каналу АСН за рахунок фільтрації і за рахунок слідкування, можна підвищити точність і надійність слідкування каналу АСН, тим самим підняти енергетичний потенціал системи, тобто підвищити дальність дії і точність вимірювання за іншими параметрами.

Оцінка стійкості каналу АСН здійснюється ймовірністю зриву супроводження ЛА за напрямком, або підвищенням відповідного квантіля для випадку простішої одномірної фільтрації для центрованого процесу за методом Калмана-Бьюсі, який враховує результати вимірювань похідних параметру. Застосування вузькосмугових фільтрів призводить до підвищення часу кореляції і дає змогу точніше прогнозувати процес.

Таким чином, застосування методу вимірювання тангенціальної (кутової) швидкості, дозволяє використовувати результати вимірювань в каналі АСН, практично вилучити динамічну і флуктуаційну похибки, а також підвищити відношення сигнал/шум. У зв'язку з цим доповнений одномірний фільтр Калмана-Бьюсі для центрованих процесів [19] введеним С з різницеvim блоком – для віднімання прогнозованого значення динамічної похибки; блоком формування результуючої оцінки кутової швидкості А зі входним зв'язком В – введеним вимірювальної інформації від каналу кутової швидкості (рис. 1.3).

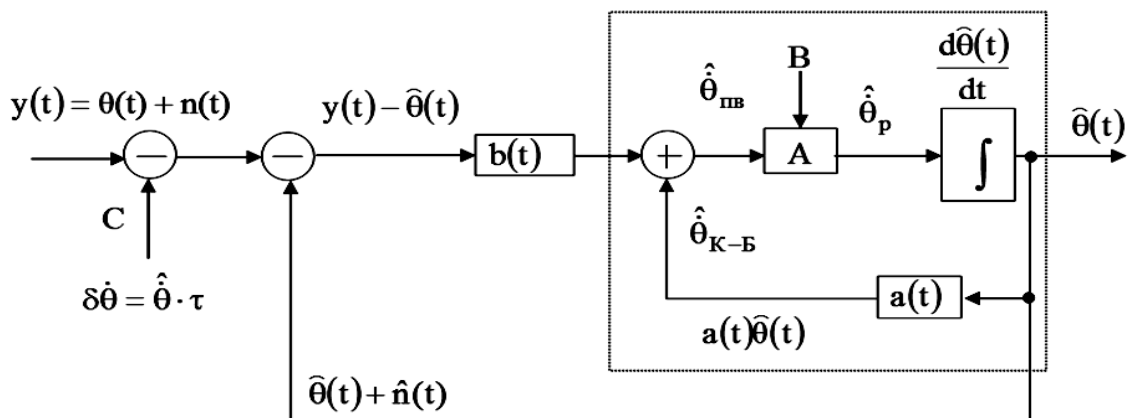


Рис. 1.3. Функціональна схема модернізованого фільтра

Модернізований одномірний фільтр Калмана-Бьюсі для центрованих процесів, що пропонується для структури каналу АСН, може бути

використаний у якості фільтра нижніх частот (ФНЧ). Диференціальне рівняння фільтра для оцінки сигналу на виході модернізованого фільтра Калмана-Бьюсі:

$$\frac{d\hat{\theta}_p(t)}{dt} = a(t)\hat{\theta}(t) + \hat{\theta}_{n\theta} + b(t)[(y(t) - \delta\theta) - \hat{\theta}(t)], \quad (10)$$

де інтенсивність білого шуму на вході формуючого фільтра, який входить у фільтр каналу АСН, визначається змінним ваговим коефіцієнтом передачі вхідного каскаду підсилювача  $b(t)$ ;  $a(t)$  – змінний коефіцієнт передачі підсилювача формуючого фільтра;  $\hat{\theta}_{n\theta}$  – оцінка прямих вимірювань кутової швидкості ЛА від каналу вимірювання кутової швидкості, яка необхідна для уточнення вимірювання  $\theta$ ;  $\varepsilon(t) = \hat{\theta}(t) - \theta(t)$  – похибка фільтрації.

Результуюча зважена оцінка  $\hat{\theta}_p$  знаходиться з розподілу ймовірності  $p\left(\begin{matrix} \hat{\theta} \\ \hat{\theta}_p \end{matrix}\right)$  зваженої оцінки:

$$p\left(\begin{matrix} \hat{\theta} \\ \hat{\theta}_p \end{matrix}\right) = p\left(\begin{matrix} \hat{\theta} \\ \theta_{K-B} \end{matrix}\right) \cdot p\left(\begin{matrix} \hat{\theta} \\ \theta_{n\theta} \end{matrix}\right), \quad (11)$$

де  $\hat{\theta}_{K-B}$  – оцінка кутової швидкості ЛА фільтром Калмана-Бьюсі.

З рівності коефіцієнтів при  $\hat{\theta}_p^2$  результуюча СКП:

$$\sigma_p^{-2} = \sigma_{K-B}^{-2} + \sigma_{n\theta}^{-2}. \quad (12)$$

Таким чином, рівняння коефіцієнтів при  $\hat{\theta}_p$  (зважена оцінка):

$$\sigma_p^{-2} \cdot \hat{\theta}_p = \sigma_{K-B}^{-2} \cdot \hat{\theta}_{K-B} + \sigma_{n\theta}^{-2} \cdot \hat{\theta}_{n\theta}. \quad (13)$$

Отже, результуюча усереднена оцінки кутової швидкості ЛА дорівнює:

$$\hat{\theta}_p = \frac{\sigma_{K-B}^{-2}}{\sigma_p^{-2}} \cdot \hat{\theta}_{K-B} + \frac{\sigma_{n\theta}^{-2}}{\sigma_p^{-2}} \cdot \hat{\theta}_{n\theta}, \quad \text{або} \quad \hat{\theta}_p = (1 - \alpha) \cdot \hat{\theta}_{K-B} + \alpha \cdot \hat{\theta}_{n\theta},$$

де  $\alpha = \frac{\sigma_{n\theta}^{-2}}{\sigma_p^{-2}}$ .

Оцінка стійкості каналу АСН визначається передаточною функцією:

$$W(p) = \frac{bW_1(p)}{1 + bW_1(p)}, \quad (14)$$

де  $W_1(p) = \frac{b}{p-a}$  – передаточна функція формуючого фільтра.

$$\text{Тоді:} \quad W(p) = \frac{b/(p-a)}{1 + b/(p-a)} = \frac{b}{p+b-a}.$$

Модернізований одномірний фільтр Калмана-Бьюсі для центрованого процесу буде стійкий за критерієм Раussa-Гурвіця, якщо характеристичний поліном матиме позитивні коефіцієнти, при яких перехідні процеси затухають. Тобто, фільтр стійкий завжди, якщо виконується умова:  $1 > b-a > 0$ .

Зважаючи на стаціонарність та незалежність сигналу і фонових шумів ФТД багатофункціональної ЛС, результуюча СКП  $\sigma_p^2$  оцінки сигналу дорівнює сумі СКП  $\sigma_\theta^2$  вихідного сигналу (оцінки), обумовленої неточним відпрацюванням каналом АСН та СКП  $\sigma_n^2$  згладжених фонових шумів на виході самого модернізованого фільтра:

$$\sigma_p^2 = \sigma_\theta^2 + \sigma_n^2. \quad (15)$$

Фонові завади на виході модернізованого фільтра:

$$\sigma_n^2 = N_0 \cdot \Delta\omega_{ef} = N_0 \frac{b^2}{2(b-a)}, \quad (16)$$

де  $\Delta\omega_{ef} = \frac{b^2}{2(b-a)}$ ;  $N_0$  – згладжений фонний шум ФТД.

Результуюча СКП оцінки сигналу дорівнює:

$$\sigma_p^2 = \frac{\sigma_s^2 a + N_0 b^2}{2(b-a)}. \quad (17)$$

Умова стійкості каналу АСН доповнена умовою незриву автоматичного супроводження ЛА для його нерухомого стану:

$$\beta_x \sigma_\theta = \beta_x \sigma_p \cdot G'_\theta \leq \frac{\Delta\theta_x}{2}, \quad (18)$$

де  $\beta_x$  – квантіль, відповідний заданій довірчій ймовірності;  $\sigma_p \cdot G'_\theta$  – перерахунок СКП вимірювань фронтів огинаючих сигналу сканування ДС ЛВ у прямому і зворотному напрямках з СКП оцінки кутової розбіжності.

Отже, показником якості незриву (стійкості) автоматичного супроводження ЛА каналом АСН у кожній площині сканування ДС ЛВ може служити квантіль за умовою:

$$\beta_x \geq \frac{\Delta\theta_x \cdot 2(b-a)}{2(\sigma_s^2 \cdot a + N_0 \cdot b^2) \cdot G_\theta'} \quad (19)$$

Вплив руху ЛА на стійкість процесу слідкування за кутами ЛА полягає у наступному. Кутове відхилення ЛА від РСН пропорційне різниці напівперіодів огинаючої лазерного сигналу. Ця різниця вимірюється з точністю визначення положення двох фронтів огинаючої. Вона може бути визначена як у цифровому, так і в аналоговому вигляді. Оскільки положення фронтів залежить не тільки від відхилення ЛА від РСН, але і від шумів ФТД та шумів активних елементів, то на виході вимірювача буде спостерігатися додатний процес.

Якщо ЛА нерухомий, то  $\hat{\theta}(t) + n_1(t)$ , де  $\hat{\theta}(t)$  – оцінка кутового відхилення ЛА від РСН в однієї з площин, а  $n_1(t)$  – кутове відхилення, обумовлене шумами ФТД. Тому, що дискримінаційна характеристика  $U_d(\theta)$  практично лінійна, цифрові (чи аналогові) шуми кутового каналу  $n_1(t)$  також є стаціонарним процесом з гаусовою щільністю розподілу ймовірності. Тоді:

$$n_1(t) = \frac{n(t)}{U_d'(\theta)}, \quad (20)$$

де  $n(t)$  – шуми ФТД,  $U_d'(\theta)$  – чутливість вимірювача, або крутизна дискримінатора.

Якщо ЛА рухається, то в залежності від ступеня астатизму слідкуючої системи каналу АСН може з'явитися динамічна похибка слідкування за кутами  $\theta_d(t)$  ЛА. Хай динаміка ЛА за одним з кутів наступна:

$$\theta_{об}(t) = \sum_{n=0}^{\infty} \theta_{об}^{(n)}(t_0) \frac{\Delta t^n}{n!}, \quad (21)$$

де  $t_0$  – час відліку;  $\Delta t = t - t_0$ ;  $\theta_{об}^{(n)}(t_0)$  – n-я похідна від  $\theta_{об}(t)$  за часом у точці  $t_0$ ;

при цьому  $\theta_{об}^{(0)}(t_0) = \theta_{об}(t_0)$ .

Звичайно для ЛА з обмеженою енергією (диспативні системи) всі одночлени з підвищенням n зменшуються, особливо при малих  $\Delta t$ . Оскільки час кореляції процесу  $\theta(t)$  в каналі АСН дорівнює  $t_{кор} = 1/\alpha$ , де  $\alpha$  – ширина енергетичного спектра процесу  $\theta(t)$ , то за час  $t_{кор}$  кутове положення  $\theta(t_0)$  каналу АСН без астатизму не може суттєво змінитись. Виходячи з цього, кутове розузгодження за рахунок руху ЛА в каналі АСН:

$$\Delta\theta_d(t) = \left[ \theta_{об}(t) - \theta(t_0) \right] = \sum_{n=1}^{\infty} \theta_{об}^{(n)}(t_0) \frac{\Delta t^n}{n!}. \quad (22)$$



Якщо  $\dot{\theta}_{об} \gg \theta_{об} \frac{t_{кор}}{2}$ ,

тоді  $\Delta\theta_{\partial}(t) \approx \dot{\theta}_{об} / \alpha$ ,

де  $\alpha$  – смуга пропускання модернізованого фільтра Калмана-Бьюсі.

Додатне (максимальне) відхилення  $\delta\theta$  ЛА від РСН:

$$\delta\theta = \beta_x \theta + \dot{\theta}_{об} / \alpha. \quad (23)$$

Кутова швидкість  $\dot{\theta}_{об}(t)$  ЛА, або його відхилення від РСН (поправка  $\Delta\dot{\theta}_{об}(t)$ ), величина випадкова. Тоді при нормальному законі розподілу величини  $\dot{\theta}_{об}(t)$  дисперсія  $\delta\theta$  визначається як додатна погрішність:

$$\sigma_{\Sigma} = \sqrt{\sigma_{\theta}^2 + \frac{\sigma_{\dot{\theta}_{об}}^2}{\alpha^2}}. \quad (24)$$

Критерієм незриву автосупроводження ЛА слуге довірчий інтервал:

$$\beta_x \left( \sigma_{\theta} + \frac{\sigma_{\dot{\theta}_{об}}}{\alpha} \right) \leq \frac{\Delta\theta_x}{2}. \quad (25)$$

Так як всі три погрішності наведення РСН на ЛА незалежні та існують одночасно, то додатна погрішність підкоряється композиції законів їх розподілу. Для гаусових законів додатна СКП також розподілена гаусово з

дисперсією:  $\sigma_{\Sigma}^2 = \sigma_p^2 + \sigma_{\partial}^2 = \frac{(G_{\theta}')^2}{2(b-a)} \cdot (\sigma_s^2 \cdot a + N_0 \cdot b^2) + \frac{\sigma_{\dot{\theta}_x}^2}{\alpha^2}$ , (26)

де  $G_{\theta}'$  – крутизна або чутливість вимірювача  $\theta_x$ ;  $\sigma_p^2$  – додатна похибка фільтрації сигналу і шуму, тобто. додаток дисперсій сигналу та сглаженого шуму на виході;  $\sigma_{\partial}^2$  – динамічна похибка фільтрації.

Таким чином, умова незриву автосупроводження:

$$\beta_x \sigma_{\Sigma} = \frac{\Delta\theta_x}{2}, \quad (27)$$

а показник такої якості

$$\beta_x = \frac{\Delta\theta_x}{2 \cdot \sigma_{\Sigma}}.$$

Якщо  $\theta_x$  не змінюється за часом, то оптимальність за методом Калмана-Бьюсі слід доповнити пошуком оптимальної смуги пропускання каналу АСН. Класичний фільтр Калмана-Бьюсі – статична система, тому похибка слідування лише зменшується у  $1+K$  разів, де  $K$  – коефіцієнт передачі розімкнутої системи.

Квантіль  $\beta_x$  (за каналом  $X$ ) описує довірчий інтервал додатної похибки каналу АСН за інтервал часу  $t_0 - t_1$ , де процес практично стаціонарний, а, значить, і довірчу ймовірність  $p_{\text{дов}\tau}(\beta_{x\tau})$  незриву автосупроводження ЛА в інтервалі  $t_0 - t_1$ . За час кореляції стаціонарність процесу (або за час  $\tau_1$ ) значення довірчого інтервалу процесу  $S(t)$  суттєво не зміниться. Через час, рівний  $\tau_1$ , значення довірчого інтервалу процесу  $S(t+\tau)$  може бути іншою. Тому для нього існує своя довірна ймовірність незриву автосупроводження ЛА.

Факторами, які визивають нестационарність слідування, можуть бути: невідповідність астатизму багатофункціональної ЛС реальній динаміці ЛА, зміна умов спостереження ЛА тощо.

Тривалість сеансу спостереження за ЛА часом  $T$  звичайно буває десятки мінут. Довірча ймовірність  $p_{\text{дов}T}(\beta_{xT})$  незриву автосупроводження ЛА за часом  $T$  (або за  $n = T/\tau$  момент часу), якщо вказані явища незалежні і рівнонадійні, дорівнює:

$$p_{\text{дов}T} = p_{\text{дов}\tau}^n(\beta_{x\tau}) = p_{\text{дов}\tau}^{T/\tau}(\beta_{x\tau}). \quad (28)$$

Оскільки в каналі АСН є два ідентичних канала  $X$  та  $Y$ , тоді:

$$p_{\text{дов}T} = p_{\text{дов}\tau}^{2T/\tau}(\beta_{y\tau}). \quad (29)$$

Таким чином, показник якості  $p_{\text{дов}\tau}$  є адаптивним і суттєво впливає на якість інших вимірювальних каналів багатофункціональної ЛС. В цьому відношенні він є одним з основних для всієї системи.

За представленими результатами показана можливість забезпечення необхідної стійкості кутового автоматичного супроводження ЛА каналом АСН за умови використання оптимальних за стійкістю його ланків та модернізованого одномірного фільтра Калмана-Бьюсі для центрованих процесів, а також розроблений ряд оцінок точностних характеристик ЛА з урахуванням [37]. Оскільки такі оцінки залежать від чисельних, у тому числі випадкових факторів, слід чітко обмежувати умови отримання даних оцінок, а також визначати ефективність застосування вказаних методів.

Також, позитивне буде врахування сучасних наукових розробок [116] стосовно визначення кутових координат ЛА з високою точністю.

### **Проблема підвищення ефективності і принципів передачі телеметричних або інформаційних каналів**

Особливістю дії ФТД багатофункціональної ЛС є те, що по суті він перетворює амплітудну модуляцію оптичних коливань у широкосмуговий відеосигнал. Смугу такої модуляції обмежує як модулятор, так і ФТД.

Сучасна смуга модулятора і ФТД поки не перевищує 1,2 ГГц. Тому багатофункціональна ЛС зможе вирішити проблему підвищення швидкості передачі інформації, якщо використовувати для передачі багато пар подовжніх мод ЛВ з несуттєво меншими смугами. При цьому виникає проблема електромагнітної сумісності інформаційних каналів з розподіленими сигналами на частотах міжмодових биттів.

Використання у всіх вимірювальних каналах частот міжмодових биттів можливе, оскільки кожна пара рівних за амплітудою подовжніх мод перетворюється у радіосигнал, який можна модулювати шляхом будь-якої модуляцією одної подовжньої моди, а детектувати звичайним детектором на радіочастоті.

Частотна розв'язка повинна задовольняти необхідну якість, оскільки різниця між частотами міжмодових биттів складає величину порівняну з цими частотами. Для вимірювальних каналів проблеми електромагнітної сумісності каналів майже не існує, тому що смуги каналів після багатократного перетворення несучої (міжмодової) частоти дуже малі. Недоліком багатоканального методу може бути те, що потужність лазера повинна зростати майже пропорційно числу каналів і те, що динамічний діапазон ФТД повинен відповідати додатній потужності сигналу, в залежності від числа каналів.

Таким чином можна зробити наступні висновки.

1. Сучасні лазерні системи можуть почати новий виток розвитку у напрямку їх інтеграції у вимірювальний комплекс, який надасть їм нові властивості при експлуатації.

2. Не можна виключати навіть їх подальшу конкуренцію з радіосистемами за точністю вимірювання параметрів руху ЛА, надійністю роботи та енергетичною економічністю.

3. Завжди доцільне комплексування лазерних систем з радіотехнічними системами, які зменшують недоліки одні до других.

4. Не виключено, що міні ЛС за розглянутими принципами побудови можуть застосовуватися в інших галузях (для діагностики, машинобудуванні).

## РОЗДІЛ 2

# ІНФОРМАЦІЙНИЙ РОЗВИТОК ПОРТАЛУ ВІРТУАЛЬНОГО УПРАВЛІННЯ ПРОЦЕСАМИ ТРАНСПОРТНОГО ОБСЛУГОВУВАННЯ

***Анотація.** Розглянуто проблему розвитку транспортного порталу WEB рішень в умовах постійного розширення кількості користувачів на рівні міста або регіону. Це відповідає розв'язанню існуючого протиріччя між стрімким розвитком засобів та методів інформатизації складних об'єктів і систем та гетерогенним характером підсистем та ланок транспортного комплексу. WEB рішення передбачають віртуальне управління транспортними підприємствами та перевізними процесами. Це має пряме відношення до галузі телекомунікацій, інформаційно-комунікаційним системам та процесами, планування та адаптації програмно-апаратних засобів як серверної так і клієнтської частини порталу до підвищених загрузок. Також є бажаної масштабованість такої інформаційно-комунікаційної системи. Аналізуються здатність системи до збільшення загальної пропускнуєї спроможності відповідно до підвищеного навантаження, коли додані апаратні ресурси.*

***Ключові слова:** транспорт, портал, масштабованість, віртуалізація, кластер.*

***Abstract.** The problem of scalability of transport portal about WEB decisions is considered. This corresponds to the settlement of the conflict between the rapid growth of information tools and methods for complex systems and heterogeneous nature of subsystems and parts of the transport sector. WEB solutions include virtual management of transport companies and shipping processes. This is directly related to telecommunications, information and communication systems and processes, planning and adaptation of software and hardware both server and client side portal to high load. There is desired scalability of such information and communication systems. There is a generalized analyzes of the ability of the systems for increase total throughput under an increased load when added hardware resources.*

***Keywords:** transportation, web, scalability, virtualization, cluster.*

**Вступ та постановка задачі.** Практика досліджень авторів роботи з розвитку віртуального управління транспортних систем [5], дослідження з застосування WEB-технологій на транспорті та практика інтерактивного моніторингу транспортних систем [67] свідчать про актуальність та безумовну ефективність сучасного підходу до WEB як інструментального засобу застосування хмарних обчислень, їх реалізації від рівня звичайного серверу віртуалізації до хмарної інформаційної інфраструктури. Далі у роботі розглянуто з позиції системної інженерії [49] найбільш важливу задачу

розширення клієнтури транспортних систем і відповідні вимоги до забезпечення масштабованості та адаптації інформаційно-комунікаційної технології руху наземного транспорту великих міст. Це є задачею планування та адаптації програмно-апаратних засобів транспортного рішення і завдань відповідного порталу до підвищених навантажень.

В галузі телекомунікацій і програмного забезпечення, масштабованість системи є бажаною властивістю, мережі, або процесу, яка свідчить про здатність системи обробити більший обсяг роботи або бути легко розширеною (за даними <http://wikipedia.org/>). Наприклад, масштабованість може позначати здатність системи до збільшення загальної пропускнуєї спроможності відповідно до підвищеного навантаження, коли додано (здебільшого, апаратні) ресурси. Цей термін має аналогічне значення, коли його вживають в галузі комерції, наприклад, масштабованість компанії припускає, що основна бізнес-модель надає можливості для економічного зростання всередині компанії.

Масштабованість, як властивість системи, як правило, важко визначити, і в кожному конкретному випадку потрібно визначити конкретні вимоги до параметрів, які вважаються важливими. Це є дуже складним питанням у галузі електронних систем, баз даних, маршрутизаторів і мереж. Систему, що підвищує продуктивність роботи після додавання апаратних засобів пропорційне доданим ресурсам, називають масштабованою. Алгоритм, архітектура, мережевий протокол, програма або інша система називається масштабованими, якщо вони ефективні в застосуванні до великих задач (наприклад, великий набір вхідних даних або велика кількість вузлів у випадку розподіленої системи).

**Основна частина.** Транспортний портал є гетерогенною розподіленою системою. Вертикальне масштабування такої системи можна здійснити завдяки збільшенню серверних потужностей базової системи. Робочу версію транспортного порталу (з урахуванням попереднього тестування елементів програмних, технічних та апаратних рішень взаємодії бортового інформаційно-комунікаційного комплексу з транспортним порталом та результатів обчислювального експерименту з визначення інформаційних потоків та обсягу комп'ютерних ресурсів, необхідних для моніторингу транспортних ситуацій) було розгорнуто на сервері на базі двох процесорів Intel Xeon 3.00 GHz, 2 Гбайт оперативної пам'яті, двох жорстких дисків за технологією SCSI 73Gb у апаратному RAID-масиві.

Завдяки експериментальним дослідженням при навантаженні серверу запитами користувачів визначено, що основним недоліком обраної архітектури

був надто малий об'єм оперативної пам'яті, що є критичним у разі серверної віртуалізації. Тому для здійснення масштабування серверних ресурсів було придбано та введено до експериментальної дії сервер самостійної зборки української компанії (рис. 2.1) на базі процесору Quad-Core 3 ГГц INTEL Xeon QC E3-1240V2, материнської плати INTEL S1200BTLR, 4-ох модулів пам'яті Kingston 8Gb із загальним об'ємом 32Гбайт DDR3, 3-ох накопичувачів HDD 1Тбайт Seagate ES ST1000N 3, оптичного пристрою DVD-RW ASUS DRW-24B5ST та корпусу серверного 2U CSV UNI (400Вт).



*Рис. 2.1. Сервер R-Line на базі INTEL Xeon E3*

Така архітектура є масштабованою «вертикально», тобто дозволяє додавати кількість модулів пам'яті, накопичувачі. Однак цей тип масштабування звичайно є обмеженим за «горизонтальне», що передбачає додавання однотипних чи гетерогенних вузлів до ферми або кластеру транспортного порталу. Поруч із цим масштабування на рівні застосування технологій віртуалізації дозволяє виконувати фактичне прозоре збільшення ресурсів як за моделлю «вертикального» масштабування, так й «горизонтального». Для виконання завдань платформи віртуалізації транспортного порталу обрано систему віртуалізації Proxmox VE, яка використовує тільки x86\_64 архітектуру. Proxmox VE базується на дистрибутиві Debian Linux та є WEB-інтерфейсом до застосування технологій віртуалізації типу KVM та LXC.

Для використання KVM-віртуалізації центральний процесор повинен підтримувати апаратну віртуалізацію (Intel VT або AMD-V). Система KVM (Kernel-based Virtual Machine) дозволяє запускати у середовищі віртуальних машин фактично будь-яку операційну систему, наприклад, Linux, Windows чи FreeBSD. Для LXC (Linux Containers) апаратна віртуалізація не потрібна, оскільки контейнери Linux функціонують як запуск одного ядра операційної системи Linux для кожного контейнеру. Контейнери є механізмом захисту веб-додатків та створюють оточення навколо кожного додатку. Це призводить до покращення надійності системи та дозволяє проектувати за модульним принципом вміст віртуальної машини.

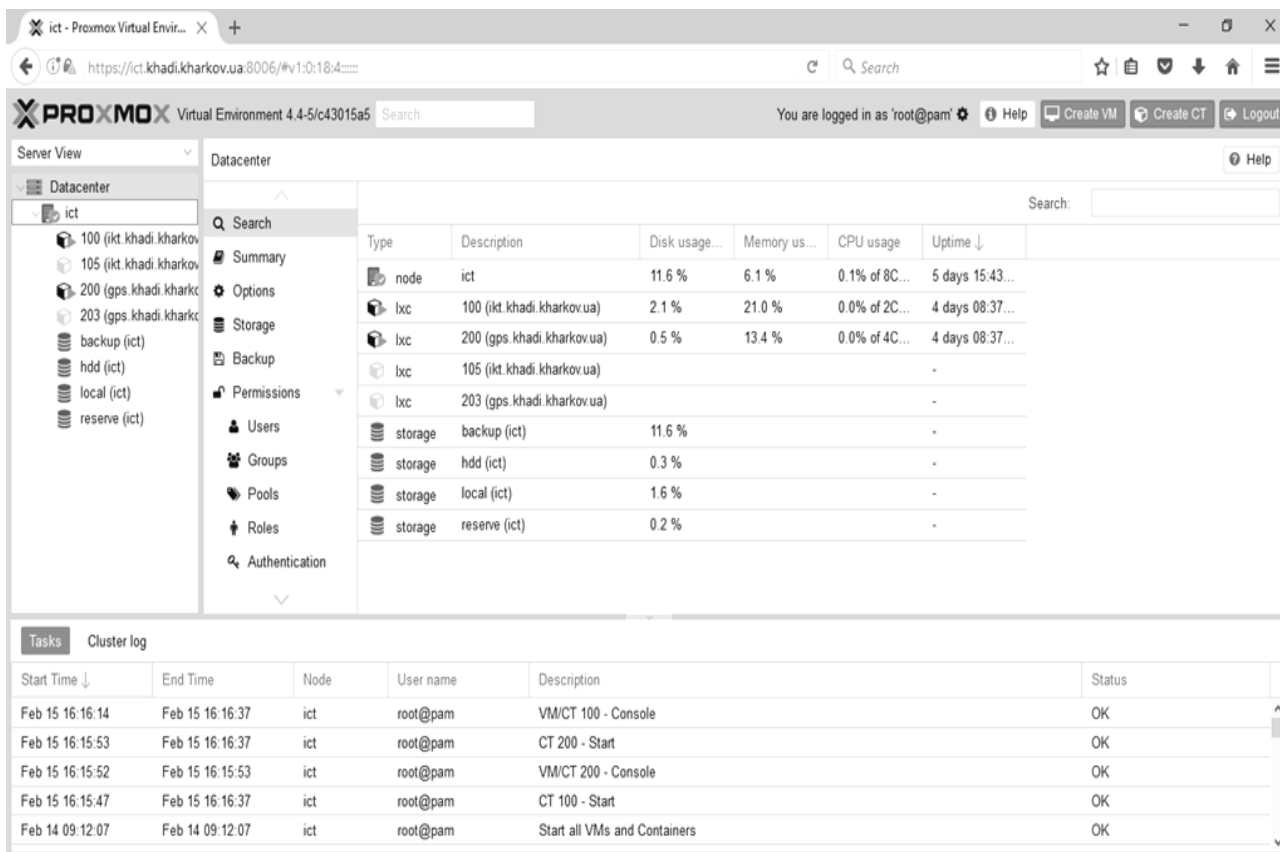
Установка Proxmox VE є дуже простою та швидкою. Упевнившись, що перший сервер працює слід виконати установку додаткового сервера або вузла, процес установки ідентичний процесу установки “першого”, тільки вказуємо відповідне ім'я сервера та його IP адресу. У самому кластері можна заходити на кожен вузол та отримати ідентичний веб-інтерфейс керування системою.

Таким чином можна створити на базі системи віртуалізації кластер високої доступності. Позначаються аббревіатурою HA (англ. High Availability - висока доступність). Створюються для забезпечення високої доступності сервісу, що надається кластером. Надмірна кількість вузлів, що входять в кластер, гарантує надання сервісу у разі відмови одного або декількох серверів. Типове число вузлів – два, це мінімальна кількість, що приводить до підвищення доступності. У разі кластеру високої доступності слід застосовувати декілька фізичних систем з Proxmox VE (за даними: <http://habrahabr.ru/post/122338>).

Принцип дії кластеру розподілу навантаження будується на розподілі запитів через один або кілька вхідних вузлів, які перенаправляють їх на обробку в інші, обчислювальні вузли. Початкова мета такого кластера – продуктивність, однак, у них часто використовуються також і методи, що підвищують надійність. Подібні конструкції називаються серверними фермами. Для такого типу кластеру можна визначити ряд віртуальних машин на одному фізичному сервері з Proxmox VE, однак, у такому разі слід пам'ятати про загрозу відмови системи в цілому.

Комбінація кластерів на базі віртуальних машин дозволить значно підвищити надійність роботи транспортного порталу, а також його продуктивність та готовність до навантажень. На рис. 2.2 наведено зовнішній вигляд панелі управління Proxmox VE. Відповідна система дозволяє уявити всі інформаційні ресурси та віртуальні мережі у якості певного віртуалізованого

центру обробки даних. Перевагою застосування кластеру на базі Proxmox VE є можливість швидкої міграції віртуальних машин у різні обчислювальні середовища та розташування їх накопичувачів у певних мережеих сховищах даних.



**Рис. 2.2. Середовище адміністрування Proxmox VE**

Слід розділяти поняття: масштабованість – здатність своєчасно реагувати на безперервне зростання навантаження і непередбачені напливи користувачів; доступність – надання доступу до додатку навіть у разі надзвичайних обставин; продуктивність – навіть найменша затримка в завантаженні сторінки може залишити негативне враження у користувача (за матеріалами: <http://www.insight-it.ru/masshtabiruemost/masshtabiruemye-veb-arkhitektury/>).

Розробка фактично кожного сайту (порталу) спрямована на функціонування із максимальною стабільністю, тобто мати доступність для абсолютно всіх потенційних відвідувачів у кожен момент часу, однак завжди трапляються непередбачені ситуації, які можуть стати причиною тимчасової недоступності. Для мінімізації потенційного збитку доступності додатка необхідно уникати наявності компонентів в системі, потенційний збій у яких привів би до недоступності, зменшення функціональності або втрати даних



(сайту в цілому або частини порталу). Таким чином, кожен сервер або будь-який інший компонент системи повинен мати хоча одного дублера (не важливо в якому режимі вони працюватимуть: паралельно або один «страхує» інший, перебуваючи при цьому в пасивному режимі), а дані повинні бути реплікованими як мінімум у двох примірниках (причому бажано не на рівні RAID, а на різних фізичних машинах). Зберігання декількох резервних копій даних десь окремо від основної системи (наприклад, на спеціальних сервісах або на окремому кластері) також допоможе уникнути багатьох проблем. Не варто забувати і про фінансову сторону питання: «страхування» на випадок збоїв вимагає додаткових істотних вкладень в устаткування, які має сенс намагатися мінімізувати.

Масштабованість прийнято розділяти на два напрямки: вертикальна масштабованість – збільшення продуктивності кожного компонента системи з метою підвищення загальної продуктивності та горизонтальна масштабованість – розбиття системи на більш дрібні структурні компоненти та рознесення їх по окремих фізичних машинах (або їх груп) та/чи збільшення кількості серверів, які паралельно виконують одну й ту ж функцію.

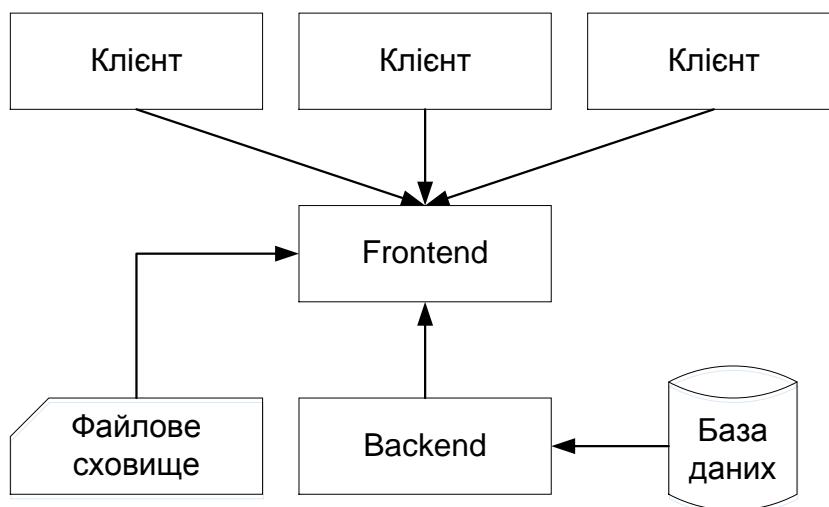
При розробці стратегії росту системи слід шукати компроміс між ціною, часом розробки, підсумковою продуктивністю, стабільністю та багатьма іншими критеріями. З фінансової точки зору вертикальна масштабованість є далеко не самим привабливим рішенням, оскільки ціни на сервера з великою кількістю процесорів завжди ростуть практично експоненціально щодо кількості процесорів. Саме по-цьому найбільш цікавий є горизонтальний підхід. Але вертикальна масштабованість має право на існування, особливо в ситуаціях, коли основну роль відіграє час і швидкість вирішення завдання, а не фінансове питання. Це продиктовано тим, що купити потужній сервер істотно швидше, ніж практично заново розробляти додатки, адаптуючи їх до роботи на великій кількості паралельно працюючих серверів.

Архітектуру транспортного порталу обрано таким чином, що кожен компонент системи є окремим та незалежним від інших. Тому актуальною є задача рівномірного розподілення запитів між доступними серверами додатків. Запуск типового сайту починається з дуже простої архітектури – один web-сервер (звичайно в його ролі виступає Apache), який виконує всю роботу по обслуговуванню HTTP-запитів, що надходять від відвідувачів. Він віддає клієнтам так звану «статіку», тобто файли, що лежать на диску сервера і не потребують обробки: картинки (gif, jpg, png), листи стилів (css), клієнтські скрипти (javascript). Той же сервер відповідає на запити, що вимагають

обчислень – це формування html-сторінок, хоча іноді динамічно створюються і зображення та інші документи. Найчастіше відповіді на такі запити формуються скриптами, написаними на java, php або іншими мовами (за матеріалами: <http://habrahabr.ru/post/15362/>).

Недоліком простої схеми роботи сайту в тому, що різні за характером запити (віддача файлів з диска і обчислювальна робота скриптів) обробляються одним й тим же web-сервером. Обчислювальні запити потребують збереження в пам'яті сервера багато інформації (інтерпретатор скриптової мови, самі скрипти, дані, з якими вони працюють) та можуть займати багато обчислювальних ресурсів. Видача статичної сторінки, навпаки, не вимагає багато ресурсів процесора, але може займати тривалий час, якщо у клієнта низька швидкість зв'язку. Внутрішній устрій сервера Apache припускає, що кожне з'єднання обробляється окремим процесом. Це зручно для роботи скриптів, однак неоптимально для обробки простих запитів. Відповідно “важкі” процеси Apache багато часу проводять в очікуванні (спочатку при отриманні запиту, потім при формуванні відповіді), неоптимально займаючи пам'ять сервера.

Вирішення цієї проблеми – розподіл роботи по обробці запитів між двома різними програмами – тобто поділ на frontend (сторона користувача ресурсу або зовнішнє представлення) та backend (бізнес-логіка роботи додатка). Легкий frontend-сервер виконує завдання по віддачі статичної сторінки, а решта запитів перенаправляє на backend (застосовує режим проксі), де виконується формування сторінок. Очікування повільних клієнтів також бере на себе frontend, і якщо він використовує мультиплексування (коли один процес обслуговує кількох клієнтів, наприклад, nginx або Nginx), то очікування практично не впливає на навантаження обладнання серверу (рис. 2.3).



*Рис. 2.3. Типова архітектура веб-ресурсу*

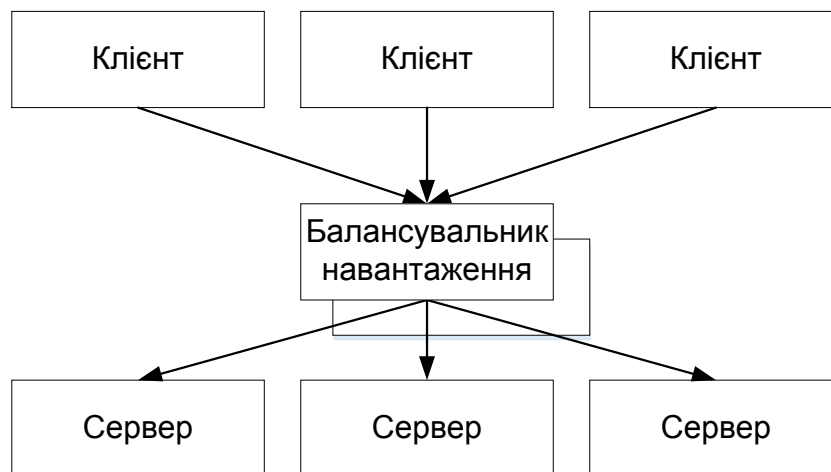
Слід зазначити, що сучасний веб-ресурс фактично завжди застосовує базу даних, в якій зазвичай зберігаються основні дані системи, наприклад, MySQL та PostgreSQL. Часто окремо можна виділити окреме сховище бінарних файлів, де містяться картинки (наприклад, ілюстрації до статей сайту та фотографії) або інші файли.

У більшості випадків на початку життєвого циклу сайту всі компоненти його архітектури розташовуються на одному сервері. Якщо він перестає справлятися з навантаженням, то є просте рішення – винести найбільш легко відокремлювані частини на інший сервер.

Найбільш продуктивним рішенням є застосування статичних сайтів (тільки frontend, що автоматично генерується одноразово на стороні серверу чи засобів розробника), однак, хоча вони є найбільш швидкісними, для рішення порталу такий підхід не задовольнить. У разі рішення масштабу WEB-порталу звичайно будуть застосовані стандартні технології рівня frontend-backend із можливістю кешування статичних сторінок.

Типовою ситуацією для зростаючого сайту є стан, коли база даних вже винесена на окрему машину і виконано поділ на frontend та backend, однак навантаження продовжує збільшуватися і backend не встигає обробляти запити. Це означає, що необхідним є розподіл обчислень на кілька серверів. Зробити це відносно просто – достатньо ввести до експлуатації другий сервер (певну віртуальну машину) і розгорнути на ньому програми та скрипти, що необхідні для роботи backend. Після цього треба зробити так, щоб запити користувачів розподілялися (балансувалися) між наявними серверами.

Мережеве устаткування, що дозволяє розподіляти навантаження між декількома серверами, зазвичай коштує досить значні суми, але серед інших варіантів саме цей підхід пропонує найвищу продуктивність та стабільність (в основному завдяки якості, також таке обладнання іноді поставляється парами, що працюють за принципом HeartBeat або визначення працездатності вузлів). Іншим варіантом є застосування програмного забезпечення, зазвичай це просто HTTP-сервери, що перенаправляють запити іншим WEB-серверам на інших вузлах замість відправки безпосередньо на обробку інтерпретатору мови програмування. Для прикладу можна виділити nginx із mod\_proxy. Крім цього мають місце менш поширені варіанти, засновані на DNS, тобто в процесі визначення клієнтом IP-адреси сервера з необхідним йому Інтернет-ресурсом адреса для нього формується з урахуванням навантаження на доступні сервери, а також з певних географічних міркувань (рис. 2.4).



**Рис. 2.4. Типова архітектура балансування навантаження веб-ресурсу**

Всі запити проходять через балансувальник, який визначає кому з серверів віддати запит на обробку. При отриманні запиту від клієнта балансувальнику потрібно визначити якому з веб-серверів переслати запит. Алгоритм прийняття рішення називається методом або стратегією балансування (за матеріалами: <http://www.phphighload.com/2012/08/blog-post.html>; <http://habrahabr.ru/post/147390/>). Найбільш поширені стратегії:

- Round Robin де з доступних серверів будуються черги і балансувальник вибирає перший в черзі. Після виконання запиту сервер переміщається в кінець черги;

- менша кількість з'єднань. Балансувальник веде облік кількості незакритих з'єднань та обирає той сервер, у якого кількість таких з'єднань менша;

- використання “ваги” серверів. Кожному серверу в залежності від потужності присвоюється вага, яка використовується для ранжирування.

Стратегія балансування, що не включає перевірку стану серверів або хоча б працездатності, не придатна для використання, так як не гарантує обробку запиту. Тому, балансувальнику слід перевіряти стан сервера, його завантаженість і вибирати найбільш здібний.

При балансуванні часто виникає проблема зберігання сесій, адже сесія доступна тільки на тому сервері, який створив її. Це слід враховувати в алгоритмі перенаправлення запиту або додаток повинен зберігати сесії на окремому сервері або в базі даних.

Для реалізації програмного балансувальника є багато рішень, наприклад: nginx – вільний веб-сервер і проксі-сервер, який має версії для сімейства Unix-подібних операційних систем (FreeBSD, Linux та ін.) й Microsoft Windows.

Основні функції – це HTTP-сервер для обслуговування статичних запитів, акселерування проксіювання з підтримкою кешування, акселерована підтримка FastCGI та memcached серверів, має простий розподіл навантаження і відмовостійкість, модульність. Apache HTTP-сервер – відкритий веб-сервер для UNIX-подібних, Microsoft Windows та ін. операційних систем. На сьогодні є найуживанішим Веб сервером мережі Інтернет.

Більш простим рішенням є застосування проксі-серверу, наприклад, HAProxy. Для його конфігурування потрібно вказати IP-адреси веб-серверів. Налаштування балансування зводиться до запису в конфігурації проксі-сервера `/etc/haproxy/haproxy.cfg` – “mode http” режиму роботи проксі, який стосується обробки пакетів, “balance leastconn” – алгоритм вибору сервера (перенаправляємо запит серверу з найменшою кількістю з’єднань). Параметр “cookie serv insert” застосовується для забезпечення роботи сесій. Таким чином клієнт завжди буде потрапляти на один сервер і мати доступ до даних сесії. Параметр “option httpclose” автоматично закривати з’єднання після завершення передачі даних. Запускається проксі сервер командою: `sudo haproxy-f / etc / haproxy / haproxy.cfg`. Конфігурація HAProxy звичайно сильно залежить від завдання, наприклад, для роботи із статичним контентом або роботи сайту без сесій немає потреби прив’язувати конкретний сервер до клієнта.

Обробка запитів (особливо динамічних), наприклад, таким веб-сервером як Apache може бути досить вимогливим до системних ресурсів. Тому не випадково все більшою популярністю користуються веб-сервери, наприклад, nginx, які можуть працювати автономно або в зв’язці з Apache в якості кешу для статичних запитів. Однак можна розглянути дещо інший підхід, пов’язаний з використанням HTTP-прискорювача Varnish, який дозволяє помітно розвантажити веб-сервери і зменшити загальний час відгуку (згідно електронному ресурсу: <http://webperformance.ru/2011/06/03/varnish-speed-up/>).

Varnish є безкоштовним рішенням для кешування як статичного, так і динамічного контенту. Працює він як балансувальник до будь-якого веб-сервера або сервера додатків і повністю орієнтований на високу продуктивність, багатопоточність та максимально ефективне використання можливостей операційних систем сімейства Linux.

Альтернативою до визначених підходів виступають так звані Content Delivery Network (CDN) – зовнішні сервіси, що забезпечують доступність контенту користувачам. Перевага очевидна – немає необхідності організувати власну інфраструктуру для вирішення цього завдання, але з’являється інша додаткова стаття витрат.

Таким чином, для рішення завдань балансування навантаження на транспортний портал оптимальним вибором є система Varnish, що забезпечує як балансування, так й кеш даних. Поставивши між користувачем і веб-сервером прозорий проксі-сервер, можна надавати користувачу дані з кеша проксі (який може бути як в оперативній пам'яті, так і дисковим), не доводячи запити навіть до HTTP-серверів. У більшості випадків цей підхід актуальний тільки для статичного контенту, в основному різних форм медіа-даних: зображень, відео і тому подібного. Це дозволяє веб-серверам зосередитися тільки на роботі з самими сторінками.

У разі здійснення заходів оптимізації додаток все одно може не впоратись з навантаженням. У такому випадку рішенням проблеми, очевидно, може послужити рознесення його по декільком вузлам, з метою збільшення загальної продуктивності додатка за рахунок збільшення доступних ресурсів.

Більшість web-додатків апріорі є розподіленими, оскільки в їх архітектурі можна виділити мінімум три шари: web-сервер, бізнес-логіка (додаток), дані (база даних, статика). Кожен із цих шарів можна масштабувати. Тому, якщо у системі додаток і база даних розміщені на одному хості – першим кроком, безсумнівно, має стати рознесення їх по різних хостах (за матеріалами: <http://habrahabr.ru/post/113992/>, <http://www.phphighload.com/2012/10/mysql-scaling-strategies.html>).

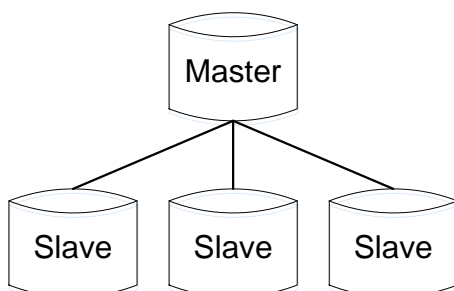
Починаючи масштабування системи, варто визначити, який із шарів є “вузьким місцем” – тобто працює повільніше за решту системи. Для цього можна скористатися простими утилітами типу top (htop) для оцінки завантаження процесора/пам'яті та df, iostat – для оцінки продуктивності дискової підсистеми серверу. Однак, бажано виділити окремий хост, з емуляцією промислового навантаження (наприклад, за допомогою Apache JMeter), на якому можна буде профілювати роботу додатка. Для виявлення вузьких запитів до бази даних можна скористатися утилітами на основі логів з журналу сервера, що знаходиться у промислової експлуатації.

Більшість залежить від архітектури веб-додатку, але найбільш вірогідними кандидатами на “вузьке місце” в загальному випадку є бази даних та код. Якщо додаток працює з великим об'ємом даних користувача, то “вузьким місцем”, відповідно, швидше за все буде зберігання статички.

Часто вузьким місцем в сучасних додатках є бази даних. Проблем базами даних діляться, як правило, на два класи: продуктивність і необхідність зберігання великої кількості даних. Знизити навантаження на базу даних можна

за умови поширення її на декілька вузлів. При цьому гостро постає проблема синхронізації між вузлами.

У масштабуванні бази даних є свої нюанси через складність підтримки актуальності даних на багатьох серверах, у зв'язку з чим стратегія залежить від структури даних та обраної архітектури [118]. У веб-додатках зазвичай домінують запити на отримання інформації – читання коментарів, постів, призначених для користувача даних та ін. Таким чином, слабким місцем є операція читання, тому саме її потрібно масштабувати. Для вирішення цього завдання використовується механізм реплікації – один сервер призначається головним (майстер-сервером), який виконує всі запити модифікації даних, а інші сервера (підлеглі) обробляють тільки запити отримання даних. При зміні або додаванні даних, майстер (Master) сервер повідомляє всі підлеглі (Slave) сервера, таким чином підтримуючи актуальність даних. Крім того, реплікація використовується для географічного розподілу серверів. Майстер сервер при виконанні модифікацій записує всі зроблені зміни до журналу, а Slave-сервера з деякою періодичністю перевіряють лог (журнал) на предмет появи нових даних і якщо вони є – виконують аналогічні дії зі своїми даними (рис. 2.5).

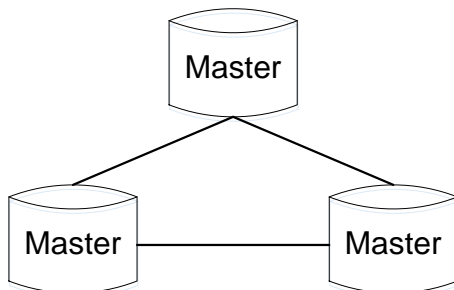


**Рис. 2.5. Масштабування бази даних «Master-Slave»**

Така схема зазвичай використовується в додатках з домінуючими запитами на читання – всі запити на зміну бази направляються майстер серверу, тоді як запити на читання розподіляються між підлеглими. Для транспортного порталу такий метод масштабування бази даних задовольняє у рішенні завдань оптимізації інформаційної складової. Зазвичай в додатку вказується список серверів (пул) призначених для читання, з якого MySQL клієнт деяким чином (випадковим або за вказаною алгоритмом) вибирає сервер для виконання запиту, таким чином, балансує навантаження між усіма серверами.

Недолік схеми – при виході з ладу майстер сервера, всі запити на модифікацію і додавання даних не будуть виконуватися.

Дуже зручна для географічного розподілу даних схема ланцюжка майстер серверів (рис. 2.6). Таким чином можна отримати прискорення виконання запитів за рахунок зменшення відстані запиту до сервера. Слід відзначити, що «Master-Master» реплікація дозволяє отримати надійний кластер для збереження даних.



*Рис. 2.6. Масштабування бази даних «Master- Master»*

Така схема є зручною при масштабуванні бази даних транспортного порталу, однак, потребує застосування спеціалізованих рішень систем керування базами даних.

Схема ланцюжка з двох майстер серверів та багатьох підлеглих серверів дозволяє майстрам виконувати запити на модифікацію даних та мають рівну кількість підлеглих серверів. Таким чином при виході з ладу одного майстра, додаток продовжить працювати з другими підлеглими серверами.

Перспективним напрямом у разі рішення завдань інформаційного розвитку порталу віртуального управління процесами транспортного обслуговування є масштабування самої бази даних із застосуванням сегментування чи шардінгу. Такий тип масштабування не передбачає резервування даних (ця задача вирішується реплікацією), однак, дозволяє рознести реляційну базу даних на різні сервера за відповідними таблицями. Тобто різні таблиці даних будуть фізично розташовані на різних серверах. У більшості випадків такий підхід буде вимагати певної зміни коду веб-додатку.

Для рішення масштабування баз даних доцільним є огляд рішень нереляційних баз даних, наприклад, MongoDB. Такі бази даних оброблюють не пов'язані між собою таблиці, як це виконують SQL-рішення, а документи в довільному вигляді. Звичайно формат представлення документу в нереляційній базі даних – це JSON формат для так званих колекцій. Системи керування нереляційними базами даних вже містять необхідні компоненти

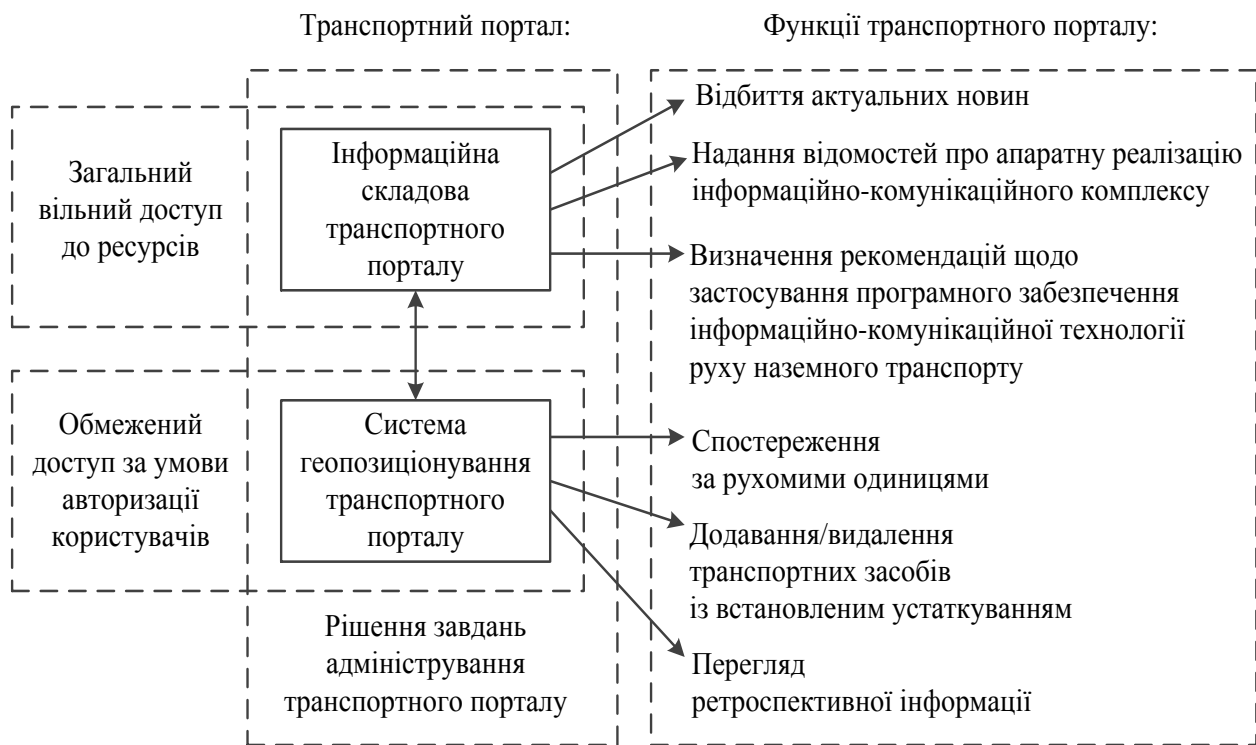


масштабування та резервування даних, що обумовлюється специфікою документ-орієнтованих рішень. На сьогодні можна визначити, що для транспортного порталу доцільно все ж таки застосовувати SQL-рішення, відповідно до більш традиційного застосування у розробці програмного забезпечення, що впроваджується вже як рішення та виконується його інтеграція у робочу систему.

Базовий варіант транспортного порталу призначено для надання інформації щодо впровадження інформаційно-комунікаційної технології руху наземного транспорту великих міст (інформаційна складова). У якості фізичної реалізації єдиного інформаційного простору на транспорті виступає розподілена система обчислювальних ресурсів. Джерелом інформації для цього порталу слугуватиме розподілена мережа інформаційно-комунікаційних комплексів, що встановлюються на борту транспортних засобів. Вхідними даними для інформаційно-комунікаційного комплексу є показники давачів прискорень транспортного засобу, його поточні координати в просторі та швидкість. Ці дані оброблюються та з них формується пакет, який із застосуванням засобів безпроводного зв'язку передається до транспортного порталу (розділ геопозиціонування).

Портал містить основний сайт проекту та засоби функціонування системи геопозиціонування (рис. 2.7). Масштабування визначеного рішення ґрунтується на додаванні рішення віртуального управління процесами транспортного обслуговування. Фактично віртуальне управління ґрунтується на залученні WEB-рішень, які дозволять, завдяки формуванню та сприянню попиту на виконання завдань перевезень вантажів та пасажирів, отримати віртуальні інструменти щодо покращення транспортного обслуговування міст та регіонів. Можна стверджувати, що таким оптимальним рішенням повинен стати агрегатор ресурсів, які надають інформаційні, пошукові, логістичні та ін. послуги та мають відкритий програмний інтерфейс (API – Application programming interface). Відповідно сайт-агрегатор доцільно розроблювати із застосуванням практики масштабування веб-рішень та баз даних.

Технічно транспортний портал складається з окремих компонентів, пов'язаних між собою посиланнями. Сервер порталу відповідає вимогам необхідності і достатності для виконання поставлених завдань з урахуванням зростання обсягів обчислень в міру збільшення потоку інформації.



**Рис. 2.7. Структура складових транспортного порталу**

Базовий транспортний портал має готовий набір WEB 2.0 сервісів в єдиній компонентній архітектурі. Застосування у проекті відповідної технології обумовлене наданням сервісів та послуг Інтернет, які дозволяють користувачам ресурсу брати активну участь в його функціонуванні та розвитку.

Слід відзначити, що у разі масштабування WEB рішень слід приділяти увагу не тільки стороні серверу (backend), а й клієнтській частині (frontend). Наприклад, є можливим застосування одно-сторінкових сайтів, тобто (SPA – Single-page Application). У такому рішенні кожна сторінка сайту не завантажується окремо, а виконується завантаження тільки потрібних даних за технологією AJAX. Розвитком такого підходу стає застосування технології ізоморфних веб-додатків, які застосовують код на JavaScript, як на стороні клієнту, так й серверу.

**Висновки.** Таким чином, шлях практично будь-якого веб-проекту з точки зору баз даних починається з одного виділеного чи віртуального сервера, на якому працює весь проект цілком. Потім настає необхідність винести базу даних на окремий сервер, але і він з часом починає не справлятися з навантаженням. Наступним кроком зазвичай буває рішення завдань реплікації даних. Наступний крок скоріше буде у напрямку рішення завдань рефакторингу

коду самого веб-додатку та визначенні більш ефективної архітектури. Наприклад, перехід від монолітної архітектури веб-рішення до мікро-сервісів.

З часом витрати на операції масштабування стають більш затратними, тому доцільно застосовувати ресурси хмарних обчислень та виконувати перехід до гібридного хмарного рішення, що передбачає розумний компроміс між використанням наявних комп'ютерних ресурсів підприємства та ресурсів хмарного сервісу [79].

Рішення задач масштабування та адаптації порталу віртуального управління процесами транспортного обслуговування до збільшення кількості користувачів зводиться до вертикального та, згодом, горизонтального нарощування серверної потужності транспортного порталу, поруч із визначенням оптимальної структури масштабування баз даних додатків веб-порталу.

### РОЗДІЛ 3

## ПАРАЛЛЕЛЬНЫЕ И ПОСЛЕДОВАТЕЛЬНЫЕ СТРУКТУРЫ ХААРА ДЛЯ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ

*Аннотация.* Приводятся результаты построения параллельных и последовательных структур Хаара для цифровой обработки сигналов, а также результаты их сравнения по выбранным и определенным в работе критериям: систоличность, производительность и объем оборудования, эффективность процессорного поля.

*Ключевые слова:* цифровая обработка сигналов; параллельные и последовательные структуры Хаара.

*Abstract.* Results of construction of parallel and sequential structures Haara for digital processing signals, and also results of their matching by the criteria selected and defined in operation are resulted: systolation, productivity and size of the equipment, efficiency of a processor field.

*Keywords:* digital signal processing; parallel and serial Haar structure.

**Введение и постановка задачи.** Сегодня практически все достижения и сложности современной информатики являются результатом активного взаимодействия технологий построения СБИС, вычислительной техники, цифровой обработки сигналов и средств телекоммуникаций [9; 88; 48]. Связь, телекоммуникации и вычислительная техника становятся неразделимы, образуя единую систему функционирования. Это усиливающееся взаимодействие обусловлено возрастающими требованиями к скорости процесса обработки сигналов, что приводит к необходимости наличия огромных вычислительных ресурсов [48].

Вместе с тем дешевые и быстродействующие СБИС с высокой плотностью компоновки элементов делают реальными такие супервычисления, если оценивать их по размерам реализующих их устройств, производительности и стоимости [88].

Хотя при высокой степени интеграции возможно обеспечить максимальный параллелизм, и тем самым почти неограниченное аппаратное обеспечение при очень низкой стоимости, имеются определенные ограничения, обусловленные технологией реализации связей, сложностью и временем разработки, тестированием, сохранением высокой реальной производительности [88]. Это означает, что существует необходимость в новой методологии проектирования систем на СБИС. С целью увеличения

потенциальных возможностей СБИС в области обработки сигналов, системное проектирование должно охватывать широкий спектр дисциплин, координировать использование достижений теории обработки сигналов и теории разработки и создания современных вычислителей [48].

С одной стороны, наличие дешевых, компактных и быстродействующих СБИС делают высокоскоростную параллельную обработку данных большого объема практичной и экономичной. С другой стороны, совершенно очевидно, что потенциальные возможности СБИС могут быть использованы только при условии четкого представления области их применения. Поэтому следует отметить, что традиционные методы разработки архитектуры вычислителей не совсем подходят для проектирования высокопараллельных СБИС-процессоров обработки сигналов.

Во-первых, характеристики алгоритмов обработки сигналов существенно отличаются от характеристик обработки данных по следующим причинам. Операции обработки сигналов обычно проводятся над массивами, содержащими очень много элементов. Отдельные элементы этих массивов имеют сравнительно невысокую точность и представлены в виде операндов фиксированной разрядности (8–12 разрядов), а в ряде случаев в виде комплексных величин фиксированной разрядности. Эти операции характеризуются регулярностью элементарных арифметических операций (сложение и вычитание), часто связаны друг с другом и образующих цепочку, что позволяет использовать регулярные аппаратные поточные средства, в которых практически отсутствуют ветвления, циклы или прямая и обратная связи по операндам.

Даже в высокоавтоматизированных системах на программируемой логике выбор оптимальной структурной ПЛИС – реализации алгоритма цифровой обработки сигналов не может быть определен строгими алгоритмическими целевыми функциями. Кроме того, необходимо учитывать довольно мощного ряда кристаллов ПЛИС, отличающихся быстродействием, потребляемой мощностью, логической емкостью, типом корпуса и другими важными параметрами [71].

В этой связи весьма актуальной видится задача анализа и выбора совокупности перспективных характеристик и критериев, которым должны отвечать структуры и алгоритмы цифровых процессоров обработки сигналов, синтез таких структур.

**Основная часть** В области цифровой обработки сигналов предпочтение отдается специализированным устройствам ввиду жестких требований,

связанных с обработкой в реальном масштабе времени. Для осуществления такой обработки обычно требуется выполнение многих тысяч миллионов операций в секунду, а универсальные системы с такой производительностью стоят миллионы долларов каждая.

Подход, основанный на использовании специализированной аппаратуры, может быть успешным только при выполнении определенных условий. Укажем некоторые из условий, которые, как мы полагаем, являются наиболее важными.

Различают два типа специализированных вычислителей. Первый характеризуется негибкой и жестко специализированной структурой. Второй допускает программирование и реконфигурацию. “Жесткие” процессоры применимы при обработке сигналов в реальном масштабе времени, так как они обеспечивают высокую скорость обработки. Однако специализация аппаратных средств часто приводит к длительным циклам проектирования и высокой стоимости. С появлением современного алгоритмического (архитектурного) анализа программируемые (реконфигурируемые) параллельные процессоры цифровой обработки сигналов стали не только практически реализуемыми, но и во многих случаях более экономичными [88; 71].

Одной из основных проблем технологии СБИС является реализация внутренних соединений. Поэтому локальность и рекурсивность алгоритмов будут иметь решающее значение, и роста эффективности обработки можно ожидать тогда, когда алгоритм подготовлен с учетом сбалансированного распределения работ и с соблюдением требования локальности, т.е. коротких линий связи. Впервые эти понятия для СБИС-алгоритмов появились при разработке систолических матриц [88]. В этих работах рассмотрены параллельные архитектуры и сделан вывод, что систолические и волновые матричные архитектуры обеспечивают наиболее перспективную совокупность характеристик при использовании технологии СБИС для обработки сигналов в реальном масштабе времени.

Систолическая система представляет собой сеть процессоров, которые ритмически обрабатывают и пересылают данные через систему. Информация в систолической системе передается между ячейками в конвейерном режиме, а связь с внешней средой осуществляется только через “граничные ячейки” [88].

Поэтому систолическую архитектуру можно рассматривать как результат реализации последовательности рекурсивных алгоритмов с помощью сети идентичных (или большей частью идентичных) вычислительных ячеек.

Важнейшими свойствами систолической матрицы являются модульность, регулярность, локальность связей, высокая степень конвейеризации

вычислений и высокая синхронность параллельной обработки. Такая матрица не требует управления, совмещает операции ввода-вывода с вычислениями и, следовательно, ускорение вычислений может быть достигнуто без увеличения требований к пропускной способности ввода-вывода.

Алгоритм называется рекурсивным, если в нем повторяются операции одного и того же типа над последовательно поступающими входными данными. В параллельном рекурсивном алгоритме входные и выходные данные снабжаются как пространственными, так и временными индексами. Индекс времени выходных данных всегда на единицу больше индекса входных данных. Пространственные индексы выходных и входных данных связаны регулярным образом. Говорят, что рекурсивный алгоритм является локальным, если разность пространственных индексов в двух последовательных рекурсиях находится в заданных пределах. В противном случае говорят, что алгоритм глобальный, т.е. для такого алгоритма будет всегда требоваться вычислительная структура с глобальными связями. Типичным примером рекурсивного алгоритма локального типа является алгоритм вычисления быстрого преобразования Фурье (БПФ). При такой классификации подавляющее большинство алгоритмов обработки сигналов принадлежит к группе локально-рекурсивных.

Важными факторами при разработке любой СБИС являются простота и эффективность проектирования. Так как площадь кремния, необходимая для реализации алгоритма сильно зависит от его степени регулярности, и по мере увеличения плотности логических элементов на кристалле до миллионов, то время проектирования и сложность топологии кристалла становятся доминирующими условиями, определяющими возможности создания СБИС цифровой обработки сигналов. Поэтому модульное проектирование высокорегулярных структур является основным аргументом в пользу внедрения таких СБИС.

В общем случае период времени для выполнения специализированной системы и ее применение для получения значительных результатов должен быть коротким, иначе имеется риск, что система никогда не будет использована. Во-первых, ее компоненты могут устареть до того, как система станет действующей и, таким образом, ее преимущества перед универсальными системами потеряются. Во-вторых, требования к специализированной системе могут измениться в течение длительного этапа разработки. Поэтому создание простых правил построения и методологии структурного проектирования позволит разработчикам быстро и уверенно проектировать свои собственные

кристаллы СБИС обработки сигналов. Повышенное внимание должно быть уделено обеспечению оптимального параллелизма и поддержанию высокого процента активности всех элементов процессорных матриц (эффективность процессорного поля).

Также системы цифровой обработки сигналов на СБИС благодаря высокой степени параллелизма и массовости используемых в них процессоров, могут обеспечить определенную отказоустойчивость, в случае дефектов в пластине кремния.

Таким образом, с учетом приведенных выше рассуждений определим основные характеристики и показатели, которым должны отвечать проектируемые современные цифровые процессоры обработки сигналов. Это: однородность, регулярность, локальность и рекурсивность (систолическая архитектура); производительность и объем оборудования, избыточность или эффективность процессорного поля; надежность и время проектирования.

Базовой составляющей практически всех методов обработки сигналов (фильтрация, сжатие, распознавание и т.д.) является получение отсчетов спектра в том или ином базисе ортогональных функций [9; 88; 48].

Поэтому очень актуальной является задача синтеза цифровых устройств спектрального анализа сигналов с использованием таких базисных функций, которые позволяют минимизировать как количество вычислительных операций, так и объем аппаратуры.

В [34] показано, что анализ и синтез сигналов в базисе обобщенных функций Хаара можно проводить с минимальными вычислительными затратами. Так, например, из анализа любого алгоритма быстрого преобразования Фурье следует, что для вычисления спектра по дискретному массиву из  $N$  чисел необходимо осуществить  $\frac{N}{2} \log N$  операций сложения и умножения. Быстрое преобразование Уолша-Адамара с упорядочением по Уолшу осуществляется за  $N \log N$  операций сложения и вычитания, что существенно больше  $2(N-1)$  операций, которое дает быстрое преобразование Хаара.

Также следует обратить внимание на фундаментальные свойства локальности системы Хаара, которые лежат в основе современных математических методов Wavelet-преобразований.

Покажем, что алгоритмы и структуры получения коэффициентов Хаара отвечают выделенным нами основным характеристикам цифровых процессоров обработки сигналов.



Если на вход специализированного вычислителя спектра Хаара подается  $N$  отсчетов дискретного сигнала  $X_i$ , то начальный процесс обработки включает в себя получение обобщенных промежуточных сумм Хаара в виде [34]:

$$X_i^{(n)} = \sum_{k=2i-1}^{2i} X_k^{(n-1)}, \quad (1)$$

где  $n=1, 2, \dots, (\log_2 N-1)$ ,  $i=1, 2, \dots, \frac{N}{2^n}$ , а  $X_k^{(0)}$  являются исходными отсчетами входного сигнала.

Коэффициенты Хаара на выходе анализатора определяются через (1) в

виде:

$$C_{mj} = \frac{1}{N} 2^{\frac{m-1}{2}} \left[ X_k^{(\log_2 N-1)-m} - X_{k+1}^{(\log_2 N-1)-m} \right], \quad (2)$$

где  $m=1, 2, \dots, \log_2 N$ ;  $j=1, 2, \dots, 2^{m-1}$ , а для выражения, стоящего в квадратных скобках  $m=m-1$  и  $k=2j-1$ .

Коэффициент Хаара с минимальным индексом (свободный член) будет определяться как

$$C_{01} = \frac{1}{N} \left[ X_k^{\log_2 N-1} + X_{k+1}^{\log_2 N-1} \right]. \quad (3)$$

Выражения (1-3) являются рекуррентными и дают возможность построения схем локально-рекурсивных процессоров на принципах систологической архитектуры.

Если  $N=8$ , то процесс отображения исходных отсчетов сигнала в пространство коэффициентов Хаара по выражениям (2-4) можно представить в виде следующей последовательности операций.

Первый шаг: определяем по (2) промежуточные суммы Хаара.

$$n=1, i=1, 2, \dots, \frac{N}{2^1} = 1, 2, \dots, 4 \text{ и соответственно}$$

$$X_1^1 = X_1^0 + X_2^0; X_2^1 = X_3^0 + X_4^0; X_3^1 = X_5^0 + X_6^0; X_4^1 = X_7^0 + X_8^0$$

Второй шаг:  $n=2, i=1, 2$ .

$$X_1^2 = X_1^1 + X_2^1; X_2^2 = X_3^1 + X_4^1$$

Третий шаг:  $m=1, j=1$ .

$$C_{11} = [X_1^2 - X_2^2] = (X_1^1 + X_2^1) - (X_3^1 + X_4^1) = (X_1^0 + X_2^0 + X_3^0 + X_4^0) - (X_5^0 + X_6^0 + X_7^0 + X_8^0) = C_2$$

$m=2, j=1, 2.$

$$C_{21} = [X_1^1 - X_2^1] = [(X_1^0 + X_2^0) - (X_3^0 + X_4^0)] = C_3.$$

$$C_{22} = [X_3^1 - X_4^1] = (X_5^0 + X_6^0) - (X_7^0 + X_8^0) = C_4.$$

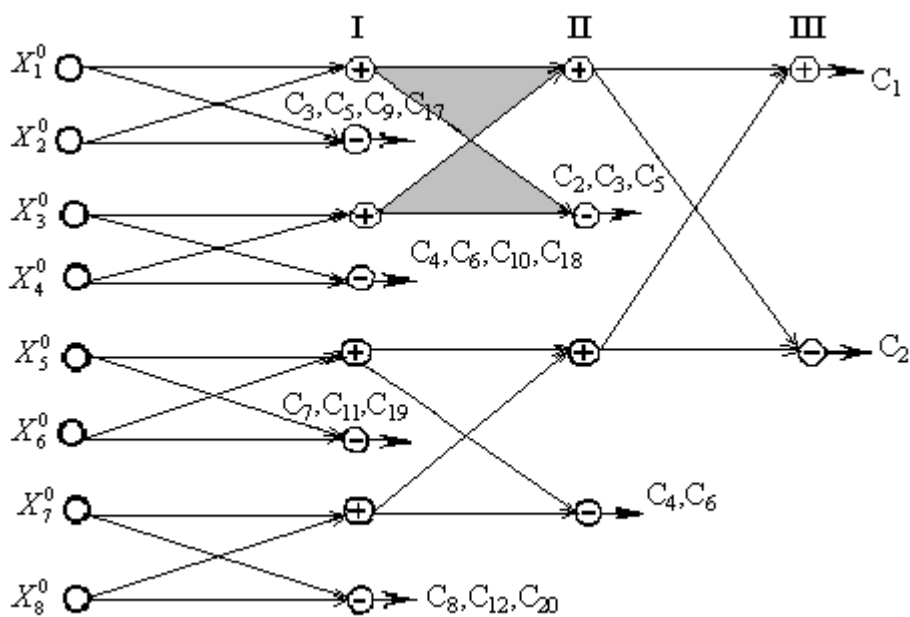
$m=3, j=1, 2, 3.$

$$C_{31} = [X_1^0 - X_2^0] = C_5; C_{32} = [X_3^0 - X_4^0] = C_6; C_{33} = [X_5^0 - X_6^0] = C_7;$$

$$C_{34} = [X_7^0 - X_8^0] = C_8$$

Мощным инструментом прямого представления последовательности полученных рекурсий является граф потока сигналов. Точки вход-выход в графе указывает на порядок вычисления, а ребра, соответствующие задержке, обозначают разделение и упорядочение двух последовательных рекурсий. Граф потока сигналов во первых, обеспечивает мощный абстрактный аппарат для выражения параллелизма и во-вторых – переход от графов к реальным систолическим матрицам Хаара достаточно прост. Необходимо только в схеме на рис. 3.1 вместо вершин графа поставить соответствующие элементарные процессоры сложения или вычитания.

Полученная рекурсивная вычислительная схема приводит к локально связанным структурным примитивам (заштрихованные участки), дающим высокую степень однородности аппаратуры, модульность и регулярность потоков данных. Используя эти примитивы в качестве элементарных модулей, можно легко наращивать структуры Хаара для вычисления коэффициентов при любом значении  $N$  входных данных. Эти структуры обладают также свойством совместимости сверху вниз, которого нет в структурах алгоритмов БПФ. То есть специализированный вычислитель Хаара для получения, например, 32 коэффициентов позволяет также получить 16, 8 и т.д. коэффициентов без перестройки его конфигурации. Это свойство отображено в табл.3.1.



**Рис. 3.1. Граф-схема получения коэффициентов Хаара.  $N=8$**

## Совместимость структур Хаара сверху вниз

Коэффициенты	N	Ярусы (такты)				
		I	II	III	IV	V
	4	3, 4	1, 2			
	8	5, 6, 7, 8	3, 4	1, 2		
	16	9, 10, 11, 12, 13, 14, 15, 16	5, 6, 7, 8	3, 4	1, 2	
	32	17, 18, 19, 20, ... 31, 32	9, 10, 11, 12, 13, 14, 15, 16	5, 6, 7, 8	3, 4	1, 2

Если например,  $N=16$ , то соответствующие  $C_9, C_{10}$ , и т.д. будут находиться на втором ярусе 32 точечной структуры, коэффициенты  $C_5, C_6, C_7$  и  $C_8$  на третьем ярусе и т.д.

В свою очередь, коэффициенты  $C_5, C_6, C_7, C_8$  являются результатом работы процессоров первого яруса при  $N=8$ . Исключения составляет только коэффициенты  $C_1$  для различных  $N$ , которые всегда будут находиться на соответствующем первом сумматоре соответствующего яруса. Результаты вычислений первого яруса процессоров последовательно и локально передаются (“проталкиваются”) процессорам второго яруса и т.д., пока не будут получены значения всех коэффициентов Хаара.

Для характеристики свойств параллельного алгоритма (структуры) Хаара введем понятие ширины параллелизма, которое определяет число операций  $\ell$ , которые можно выполнять одновременно, параллельно. В нашем случае параллельная задача получения коэффициентов Хаара выполняется за несколько тактов. Причем величина  $\ell$  меняется от такта к такту и поэтому имеет смысл определить понятие среднего значения ширины параллелизма

$$\ell_{\text{ср}} = \frac{1}{q} \sum_{i=1}^q \ell_i, \quad (4)$$

где  $q$  – число тактов исполнения параллельного алгоритма, которое равно  $\log N$ . Числовое значение  $q$  отражает время исполнения параллельного алгоритма, выраженное в тактах. Оно характеризует глубину параллелизма и алгоритмическую сложность структур Хаара.

Важными характеристиками параллельных структур Хаара для цифровой обработки сигналов являются производительность  $P$  и объем оборудования  $Q$ , которые определим соответственно как

$$P = \frac{2(N-1)}{\log N} \quad \text{и} \quad Q = 2(N-1) \quad (5)$$

Здесь  $P$  определяет условное время, за которое параллельная структура реализует вычисление  $2(N-1)$  арифметических операций того или иного алгоритма,  $Q$  – количество элементарных процессоров этой структуры.

Однако для сравнительной оценки множества вариантов архитектур, предназначенных для цифровой обработки сигналов, этих характеристик явно недостаточно. Поэтому необходимо использовать дополнительные критерии качества. Так в некоторых случаях весьма целесообразно использовать для оценки удельную производительность

$$V = \frac{P}{Q}, \quad (6)$$

где  $P$  и  $Q$  были определены выше. Удельную производительность наряду с производительностью можно использовать как важный критерий сравнения структур Хаара с параллельной и последовательной архитектурой.

Наиболее информативной характеристикой параллельного алгоритма является ускорение  $\lambda$ , показывающее во сколько раз применение параллельного алгоритма уменьшает время исполнения задачи по сравнению с последовательным алгоритмом. Вместе с тем следует ввести и характеристику ( $\alpha$ ), которая показывает, во сколько раз оборудование параллельного процессора проигрывает объему оборудования последовательного:

$$\lambda = \frac{T_{\text{посл.}}}{T_{\text{пар.}}} \quad \text{и} \quad \alpha = \frac{Q_{\text{пар.}}}{Q_{\text{посл.}}}, \quad (7)$$

где  $T$  – количество тактов для получения коэффициентов Хаара в последовательной и параллельной структуре.

Параллельные вычислительные структуры Хаара, которые строятся путем прямого отображения информационного графа, имеют и определенные недостатки, вытекающие из того, что абсолютно параллельных процессов в принципе не существует, а объем структур растет с ростом числа вершин графа и при большом числе  $N$  они становятся труднореализуемыми, а процессоры – малозагруженными. Так из рис. 3.1 следует, что на первом такте вычисления коэффициентов Хаара работают только процессоры первого яруса, на втором –

только вторая линейка (второй ярус) и т.д. Поэтому очень целесообразной в практическом отношении видится такое понятие, как “эффективность процессорного поля”  $E$ :

$$E = \frac{1}{q} \sum_{i=1}^q n_i \quad \text{или} \quad E = \frac{N}{q}, \quad (8)$$

где  $n_i$  – число процессорных элементов (ПЭ), которое необходимо для реализации параллелизма в такте  $i$ . Величина  $E$  указывает среднюю загрузку ПЭ при выполнении конкретного параллельного алгоритма. Можно уточнить это понятие, если ввести такую характеристику, как избыточность структуры или коэффициент загрузки оборудования “ $R$ ”, который показывает отношение суммарного числа работающих процессоров ( $M$ ) к числу неработающих ( $M^*$ ) при каждом такте решения задачи и в целом после ее завершения:

$$M_i = \sum_{i=1}^{\log N} \frac{N}{2^{i-1}} \quad \text{и} \quad M_i^* = \sum_{i=1}^{\log N} [2(N-1) - 2^i], \quad (9)$$

$$\text{а} \quad R_i = \left( \sum_{i=\log N}^1 2^i \right) / \sum_{i=1}^{\log N} [2(N-1) - 2^i], \quad (10)$$

где  $i$  в числителе формулы (10) меняется от  $\log N$  до 1 с шагом минус единица.

По выражениям (9) и (10) для различных  $N$  построим таблицы значений коэффициентов  $R$ ,  $M$  и  $M^*$  и проведем их сравнительный анализ (табл. 3.2).

Таблица 3.2

**Значения коэффициентов  $R$ ,  $M$  и  $M^*$ .  $N=64$ ,  $i=1,2,3,4,5,6$**

Такты	1	2	3	4	5	6	$\Sigma$
$M$	64	32	16	8	4	2	126
$M^*$	62	94	110	118	122	124	630
$R$	1,03	0,34	0,145	0,067	0,032	0,016	0,2
$R^*$	0,507	0,253	0,127	0,063	0,031	0,0158	1

Здесь и далее  $R^*$  определяет отношение числа работающих процессоров на каждом такте решения задачи к их общему числу в схеме, т.е.  $Q$  (табл. 3.3–3.4).

Таблица 3.3

**Значения коэффициентов  
при N=16**

Такты	1	2	3	4	$\Sigma$
М	16	8	4	2	30
М*	14	22	26	28	90
R	1,14	0,363	0,153	0,07	0,33
R*	0,533	0,266	0,133	0,066	1

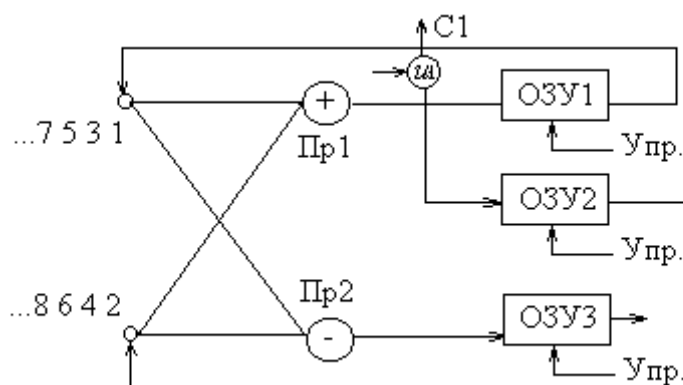
Таблица 3.4

**Значения коэффициентов при  
N=8**

Такты	1	2	3	$\Sigma$
М	8	4	2	126
М*	6	10	12	28
R	1,33	0,4	0,166	0,5
R*	0,57	0,285	0,142	1

Анализ этих таблиц показывает очень низкий коэффициент загрузки оборудования в параллельных структурах Хаара, независимо от длины обрабатываемой реализации. Так уже на втором такте работы схемы при N=64 (табл. 3.2.) число работающих процессорных элементов по отношению к не работающим составляет всего 34 %, а на третьем такте эта цифра опускается до 14,5%. Чем меньше длина обрабатываемой реализации N, тем выше коэффициент загрузки оборудования параллельной структуры. Избыточность структур связана с коэффициентом загрузки оборудования R обратной зависимостью, т.е., чем меньше R, тем больше избыточность.

Эти результаты дают основание говорить о необходимости построения базовых параллельно-последовательных структур, которые дают возможность синтезировать безизбыточные схемы быстрых преобразований Хаара при любом значении N длины обрабатываемой реализации сигналов путем наращивания таких структур и простым их соединением. На рис. 3.2 изображена такая базовая структура (модуль) с указанием необходимой последовательности поступления номеров отсчетов сигнала, а в табл. 3.5 – выполняемые действия на каждом такте работы при N=4.



**Рис. 3.2. Базовая безизбыточная структура Хаара**

**Результаты потактовой работы в схеме на рис. 3.2**

Такты					
1	2	3	4	5	6
(1+2)→Пр1→ОЗУ1 (1- 2)→Пр2→С <sub>3</sub> →ОЗУ2	(3+4)→Пр1 (3- 4)→Пр2→С <sub>4</sub>	Пр1→ОЗУ3 С <sub>4</sub> →ОЗУ2	ОЗУ1→(1+2 ) ОЗУ3→(3+4 )	(1+2)+(3+4) →Пр1 (1+2)- (3+4)→Пр2	u1→С <sub>1</sub> Пр2→ С <sub>2</sub>

В случае соединения двух базовых модулей мы приходим к четырехпроцессорной структуре, четырех – к восьмипроцессорной и т.д. Один из вариантов схемы с использованием двух базовых модулей приведен на рис. 3.3, а в табл. 3.6 определены соответствующие ей последовательности выполняемых элементарных операций при N=4.

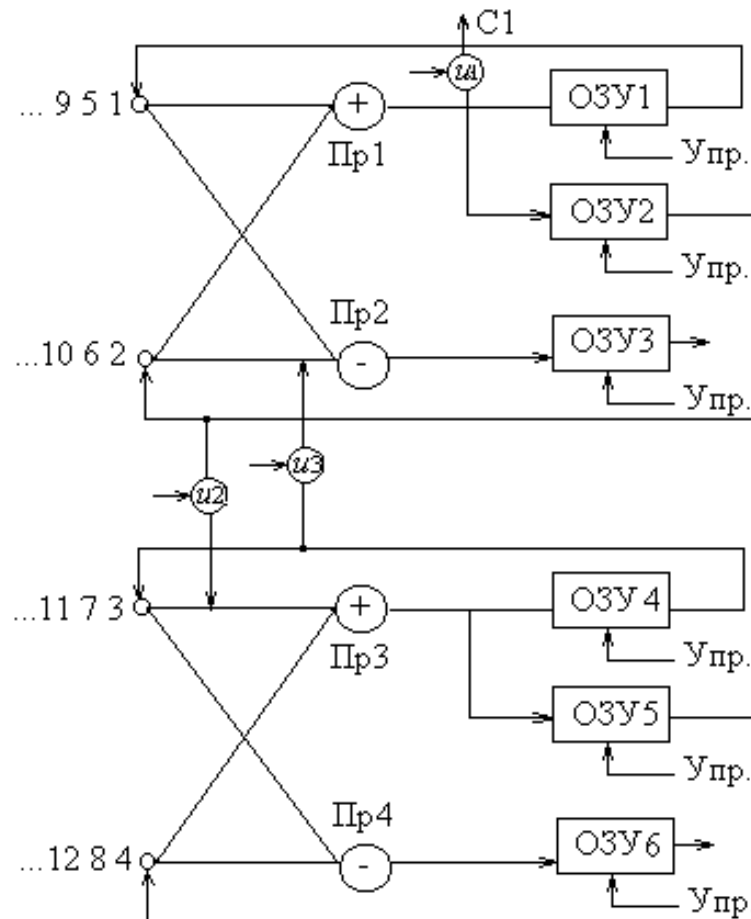
**Результаты потактовой работы в схеме на рис. 3.3**

Такты				
1	2	3	4	5
(1+2)→Пр1 (3+4)→Пр3 (1-2)→Пр2 (3-4)→Пр4	Пр1→ОЗУ1 Пр3→ОЗУ4 Пр2→С <sub>3</sub> →ОЗУ2 Пр4→С <sub>4</sub> →ОЗУ6	ОЗУ1→(1+2)→Пр1 ОЗУ4→(3+4)→u3→Пр2	(1+2)+(3+4)→Пр1→С <sub>1</sub> (1+2)-(3+4)→Пр2→С <sub>2</sub>	С <sub>1</sub> →u1→

Также достаточно просто построение схемы из 4 базовых модулей и соответствующей ей таблицы выполняемых потактовых элементарных операций. Проведем анализ эффективности предложенного подхода для двух, четырех и восьмипроцессорных структур в сравнении с параллельными структурами на основании определенных нами в работе характеристик и критериев.

Исходными данными для анализа будут: число арифметических операций  $2(N-1)$ , которое необходимо выполнить для получения коэффициентов Хаара, как в параллельной, так и в последовательной (параллельно-последовательной) структуре; количество тактов T в параллельной и последовательной структуре для решения задачи получения коэффициентов; объем оборудования параллельной Q и последовательных структур Q<sub>2</sub>, Q<sub>4</sub>, Q<sub>8</sub>, где Q<sub>2</sub> – двухпроцессорная, Q<sub>4</sub> – четырехпроцессорная и Q<sub>8</sub> – восьмипроцессорная

структура. Причем оборудование базового модуля (рис. 3.3) состоит из двух процессоров и двух блоков памяти ОЗУ1 и ОЗУ3, которые работают на частоте процессоров, как это реализовано в современных персональных ЭВМ. Блок памяти коэффициентов ОЗУ2 не входит в состав оборудования базового модуля, а технологическую сложность процессора и блока памяти будем считать одинаковой. Таким образом, можно считать что аппаратные затраты базового модуля составляют четыре процессорных элемента. Полученные численные значения основных характеристик по формулам (5), (6), (7), (9) и (10) сведены в соответствующие табл. 3.7 и табл. 3.8 и по этим значениям, с учетом данных таблицы тактов 9, построены семейства графиков (рис. 3.4 и рис. 3.5), позволяющих судить об эффективности той или иной структуры.



**Рис. 3.3. Четырехпроцессорная структура на двух базовых модулях**



Таблица 3.7

**Характеристики последовательных и параллельных структур Хаара**

logN	2	3	4	5	6
$\alpha_2$	1,5	3,5	7,5	15,5	31,5
$\lambda_2$	3	4,33	7,5	12,4	21
$\alpha_4$	0,75	1,75	3,75	7,75	15,75
$\lambda_4$	2,5	3	4,5	6,6	10,6
$\alpha_8$	0,375	0,875	1,875	3,875	7,875
$\lambda_8$	2,5	2,66	2,75	4	6,33

$\alpha_2, \lambda_2$  – характеристики 2-х процессорной структуры,  $\alpha_4, \lambda_4$  – 4-х процессорной структуры и т.д.

Таблица 3.8

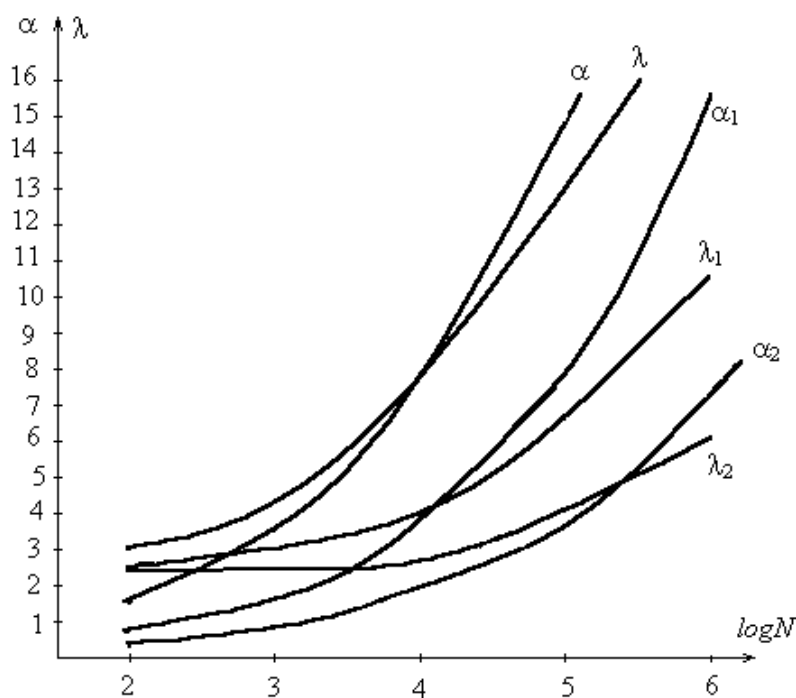
**Характеристики**

logN	2	3	4	5	6
$P_{\text{пар.}}$	3	4,67	7,5	12,4	21
$P_2$	1	1	1	1	1
$P_4$	1,2	1,55	1,66	1,87	1,9
$P_8$	1,2	1,75	2,72	3,1	3,15
$V_{\text{пар.}}$	0,5	0,333	0,25	0,2	0,166
$V_2$	0,25	0,25	0,25	0,25	0,25
$V_4$	0,15	0,193	0,2	0,234	0,246
$V_8$	0,075	0,109	0,17	0,193	0,2

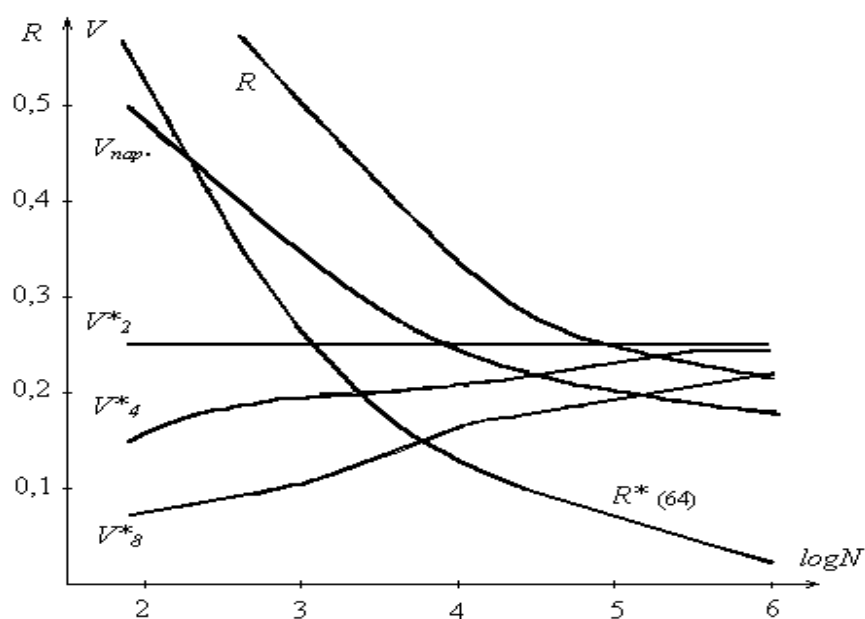
Таблица 3.9

**Значения количества необходимых тактов**

logN		2	3	4	5	6
Такты	Пр <sub>пар.</sub>	2	3	4	5	6
	Пр <sub>2</sub>	6	13	30	62	126
	Пр <sub>4</sub>	5	9	18	33	64
	Пр <sub>8</sub>	5	8	11	20	38



**Рис. 3.4. Характеристики параллельных и последовательных структур**



**Рис. 3.5. Характеристики параллельных и последовательных структур**

Из графиков на рис. 3.4 и рис. 3.5 следует, что для всех последовательных структур Хаара существуют определенные значения размерности векторов обрабатываемых данных, когда их применение становится более

предпочтительным, то есть когда затраты на оборудование параллельного процессора начинают опережать его выигрыш в скорости вычислений.

По показателю удельной производительности  $V$  двухпроцессорная система всегда остается лучшей по сравнению с остальными последовательными структурами, а при  $N > 16$  она начинает выигрывать и по сравнению с параллельной. При больших значениях  $N$  удельная производительность всех последовательных структур не сильно отличается и стремится к значению 2,5. График коэффициента загрузки оборудования ( $R$ ) показывает, что с ростом длины массива ( $N$ ), число работающих процессоров в параллельных структурах по отношению к числу не работающих резко падает. Причем наибольшая крутизна графика наблюдается при малых значениях  $N = 2$  и  $3$ . Поведение графика  $R^*$  демонстрирует, как изменяется отношение числа работающих процессоров на каждом такте решения задачи к их общему числу при  $N = 64$  и свидетельствует о резком уменьшении эффективности загрузки процессорного поля в параллельных структурах Хаара с ростом  $N$ .

**Заключение.** Полученные в работе результаты и выводы позволяют на практике строить различные архитектуры процессоров обработки сигналов в базисе Хаара, ставя во главу угла их эффективность по тем или иным критериям, определенных в работе. Так как ортогональные преобразования Хаара являются обратимыми, то приведенные результаты остаются в силе и при выборе схем, реализующих алгоритмы быстрых обратных преобразований. Также предложенная методика может быть использована при анализе и синтезе схем обработки сигналов в других ортогональных базисах.

## РОЗДІЛ 4

# СИНТЕЗ ІНФОРМАЦІЙНОЇ ТА ТЕХНІЧНОЇ СТРУКТУРИ СИСТЕМИ УПРАВЛІННЯ ОБ'ЄКТОМ КРИТИЧНОГО ЗАСТОСУВАННЯ

***Анотація.** У главі розглянуто підходи до синтезу інформаційної структури систем управління, а також здійснено поставку задач синтезу інформаційної структури системи управління об'єктом критичного застосування. Запропоновано таксономічну схему взаємозв'язку формуючих інформаційну структуру системи управління та концептуально розглянуто послідовність етапів синтезу структури системи управління, так і етапів синтезу логіки такої системи. Також розглянуто підходи до синтезу технічної структури комп'ютерної системи, що лежить в основі сучасних систем управління об'єктами критичного застосування. Виділено та описано основні етапи синтезу, а також характерні задачі, що розв'язуються на них. Зроблено постановку та формалізацію відповідних задач синтезу при урахуванні вимог до доступності компонентів системи і вимог до процесу її функціонування в контексті оточуючого середовища при умові можливих зовнішніх впливів.*

***Ключові слова:** синтез, управління, структура, система, мережа, компонент, формалізація, зв'язок.*

***Abstract.** This chapter discusses approaches to synthesis of informational structure of control systems, as well as statements a problem of critical application object control system's informational structure synthesis. Taxonomy for interrelations between elements that form control system's informational structure is proposed; the concepts for sequences of control system's structure synthesis stages and control system's logic synthesis stages are presented. Also discussed approaches to synthesis of technical structure of a computer system, which is underlying for modern critical application objects' control systems. The main stages of the synthesis are marked out and described, as well as appropriate problems that should be resolved. Moreover, appropriate synthesis tasks were raised and formalized subject to requirements to the system's components availability and requirements to operation process under environment conditions and possible impact.*

***Keywords:** synthesis, control, structure, system, network, component, formalization, connection.*

Задача синтезу сучасних систем управління (СУ) різними об'єктами є комплексною проблемою, вирішення якої неможливо без попередньої декомпозиції, структуризації і формалізації.

До теперішнього часу багатьма авторами описаний ряд можливих підходів і рішень як до синтезу структур обчислювальних мереж для

ієрархічних систем управління, так і до синтезу структур інформаційно-телекомунікаційних мереж, а також багаторівневих інформаційних структур інтеграційних компонентів їх гетерогенних складових [51; 52; 43; 44; 42].

До основних етапів синтезу СУ можна віднести наступні:

- синтез організаційної структури СУ об'єктом;
- синтез (включаючи рішення задач стратифікації) інформаційної структури комп'ютерної системи (КС), що забезпечує функціонування СУ;
- синтез технічної структури КС.

Основною задачею синтезу організаційної структури СУ об'єктом є розробка логічної структури організації такої СУ, включаючи її основні компоненти і взаємозв'язки між ними.

Результатом синтезу інформаційної структури є структура КС, за допомогою якої реалізується управління процесом функціонування СУ, спрямованої на розв'язання задач певних класів. Така структура являє собою логічну сукупність необхідних управляючих компонентів, стратифікована за відповідними рівнями, а також зав'язків між компонентами і рівнями, які забезпечують обмін, як службовою інформацією, так і даними, що особливо важливо для СУ об'єктом критичного застосування.

Синтез технічної структури КС включає розробку фізичної структури для отриманої на попередньому етапі інформаційної структури КС.

Крім того, в процесі синтезу СУ, на кожному з етапів, розподіл сукупності задач, що розв'язуються, за сукупністю використовуваних компонентів КС має прагнути до екстремуму по заданому критерію якості з обов'язковим урахуванням усіх обмежень, що застосовуються.

Метою даного підрозділу є розробка формалізованих описів задач, що виникають при синтезі організаційної структури СУ, задач вибору оптимальної структури та логіки КС, що забезпечує функціонування СУ (включаючи її підсистеми) в тому числі, які можна застосувати до об'єктів критичного застосування (КЗ), а також розробка формалізованих описів задач, що виникають при синтезі технічної структури КС СУ об'єктом КЗ.

### **Підходи до синтезу організаційної структури СУ**

У процесі синтезу структури системи управління об'єктом критичного застосування основними високорівневими задачами є наступні:

- формалізація структури і компонентів СУ, включаючи визначення необхідної кількості компонентів і необхідної кількості рівнів, за якими вони повинні бути розподілені;

– формалізація взаємозв'язків між компонентами СУ, включаючи визначення принципів управління і відтворення необхідних зав'язків в загальній ієрархії рівнів управління;

– формалізація задач, що реалізуються такими СУ, включаючи їх оптимальний розподіл по компонентах.

Організаційна структура СУ повинна давати однозначну відповідь на нерозривні питання розподілу процесів по підсистемах різних рівнів, а також розподілу всієї сукупності можливих функцій управління і методів їх реалізації по необхідним підсистем.

При розв'язанні задачі синтезу організаційної структури СУ потрібно визначення:

– множини принципів і методів управління, які повинні реалізовуватися СУ, що синтезується;

– достатньої множини взаємопов'язаних функцій, що реалізуються СУ;

– множини компонентів СУ, взаємопов'язаних між собою і стратифікованим за рівнями;

– відображення елементів множини взаємопов'язаних функцій, що реалізуються СУ на множину компонентів СУ.

Важливою задачею початкового етапу синтезу є визначення та формалізація вимог до СУ. Реалізація вимог системою управління можлива багатьма способами, що найбільш зручно формалізується апаратом графів. Таким чином, стає можливою формалізація взаємозв'язків для СУ, що синтезується, логічного і часового типів, а також опис збору, перетворення, обробки і виведення інформації.

При побудові таких графів можливе розв'язання задач розподілу певних елементів по множині доступних ресурсів. При відображенні множини взаємопов'язаних функцій і задач СУ об'єктом КЗ, які задані у вигляді відповідних графів, на множину компонентів СУ, можна отримати орієнтований мультиграф, окремі частини якого відповідають можливим варіантам розподілу функцій по компонентах СУ, а дуги відображають взаємозв'язки між компонентами.

Оптимальним варіантом реалізації організаційної структури СУ об'єктом КЗ буде підграф орієнтованого мультиграфа, який одночасно є допустимим варіантом реалізації організаційної структури СУ і реалізована на його основі СУ задовольняє специфікації до характеристик якості.

Таким чином, модель структури СУ може бути представлена орієнтованим графом:

$$G_s = (V_D, R_F, \Theta),$$

де  $V_D = \{v_d; d = \overline{1, d_0}\}$  – множина можливих варіантів організації даних;  $R_F = \{r_f; f = \overline{1, f_0}\}$  – множина можливих варіантів реалізації функцій;  $\Theta = (\theta_{df})$ ,  $d = \overline{1, d_0}; f = \overline{1, f_0}; \dim \Theta = d_0 \times f_0$  – матриця, що відображує взаємозв'язок даних і функцій. При цьому  $\theta_{df} = 1$ , якщо для формування множини даних  $v_d$  використовується функція RF та  $\theta_{df} = 0$  інакше.

На графі  $G$  можливе завдання варіантів реалізації даних і способів їх формування, за допомогою функцій, з інших даних СУ, що дозволяє відображати варіанти перетворення вхідних даних у вихідні дані з метою вибору оптимального варіанту при заданих критеріях оптимальності і обмеженнях, що існують.

### **Постановка задач синтезу інформаційної структури КС, що забезпечує функціонування СУ об'єктом КЗ**

Вихідна інформація для синтезу інформаційної структури КС представлена вимогами до характеристик її якості. Послідовність окремих задач, що розв'язуються при синтезі СУ об'єктом КЗ, виглядає наступним чином:

- розробка моделі інформаційної структури КС з описом входів і виходів, а також відповідними процедурами обробки даних;
- аналіз варіантів обробки і представлення даних стосовно реалізації компонентів КС, що використовується, на основі апарату графів;
- детальний аналіз, включаючи аналіз допустимих варіантів побудови, синтезованого графа в аспекті реалізації входів-виходів і обробки даних КС;
- визначення оптимального варіанту інформаційної структури КС.

Однією з найбільш актуальних наукових задач етапу синтезу СУ є формалізація задачі вибору оптимального варіанту інформаційної структури. Розглянемо загальний підхід, коли множина задач  $i = \overline{1, I}$ , що має множину варіантів розв'язання  $v_i = \overline{1, V_i}$ , можливо розбитих на множину етапів  $\ell_{iv_i} = \overline{1, L_{iv_i}}$ , які можуть виконуватися множиною можливих варіантів  $v_{\ell_{iv_i}} = \overline{1, V_{\ell_{iv_i}}}$  на множині доступних компонентів  $j = \overline{1, J}$  системи, що мають множину варіантів побудови  $w_j = \overline{1, W_j}$ . Окрім того, за допомогою  $p_{ij}$  позначимо максимальне число етапів обраного варіанту  $i$ -ї задачі, що розв'язується в  $j$ -м компоненті КС. При такій нотації стає можливим сформулювати задачу синтезу інформаційної структури КС наступним чином:

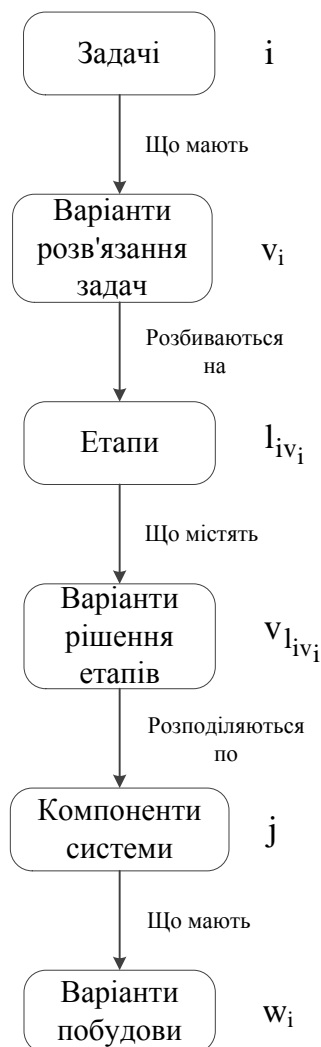
$$P_{iv_{\ell_{iv_i}}} \left( x_{iv_{\ell_{iv_i}}}, x_{i\ell_{iv_i}}, x_{i\ell_{iv_i}j}, x_{jw_j} \right) \rightarrow \text{opt},$$

де  $P$  – показники якості, що оптимізуються;  $x_{iv\ell_{iv_i}}$ ,  $x_{i\ell_{iv_i}}$ ,  $x_{i\ell_{iv_i}j}$  та  $x_{jw_j}$  є булеві змінні, що приймають одиничні значення тільки в разі вибору відповідних етапів або варіантів; при наступних  $\eta_0$  обмеженнях на шукане відображення множини взаємопов'язаних функцій на множину компонентів системи:

$$P_{\eta} \left( x_{iv\ell_{iv_i}}, x_{i\ell_{iv_i}}, x_{i\ell_{iv_i}j}, x_{jw_j} \right) \leq P_{\eta}^{(\text{треб})},$$

$$\eta = \overline{1, \eta_0}.$$

Таксономія введених нотацій представлена на рис. 4.1.



**Рис. 4.1. Таксономічна схема взаємозв'язку утворюючих інформаційної структури КС**



З точки зору деталізації можливих обмежень задачі синтезу інформаційної структури системи, кожна задача з множини задач може мати низку обмежень різних типів, основні з яких наведено нижче.

Обмеження на однозначність вибору з представленого числа можливих варіантів розв'язання задач може бути представлено у вигляді наступного виразу:

$$\sum_{v_{\ell_{iv_i}}=1}^{V_{\ell_{iv_i}}} x_{iv_{\ell_{iv_i}}} = 1,$$

де  $i = \overline{1, I}$  – множина задач.

Обмеження на відсутність дублювання етапів безпосереднього розв'язання задачі, кожен з яких може бути реалізований тільки одним певним компонентом системи, може бути представлено як

$$\sum_{j=1}^J x_{\ell_{iv_i} ij} = 1,$$

де  $i = \overline{1, I}$  – множина задач;  $\ell_{iv_i} = \overline{1, L_{iv_i}}$  – допустима множина етапів розв'язання кожної із задач.

Обмеження на максимальну кількість задач  $M$ , яке допустимо реалізацією системи, має вигляд:

$$\sum_{i=1}^I \sum_{v_{\ell_{iv_i}}=1}^{V_{\ell_{iv_i}}} x_{iv_{\ell_{iv_i}}} \leq M.$$

Таким чином, декомпозиція етапів синтезу СУ об'єктом КЗ може бути представлена у вигляді рис. 4.2 – 4.3, для ієрархії СУ і її логіки відповідно.

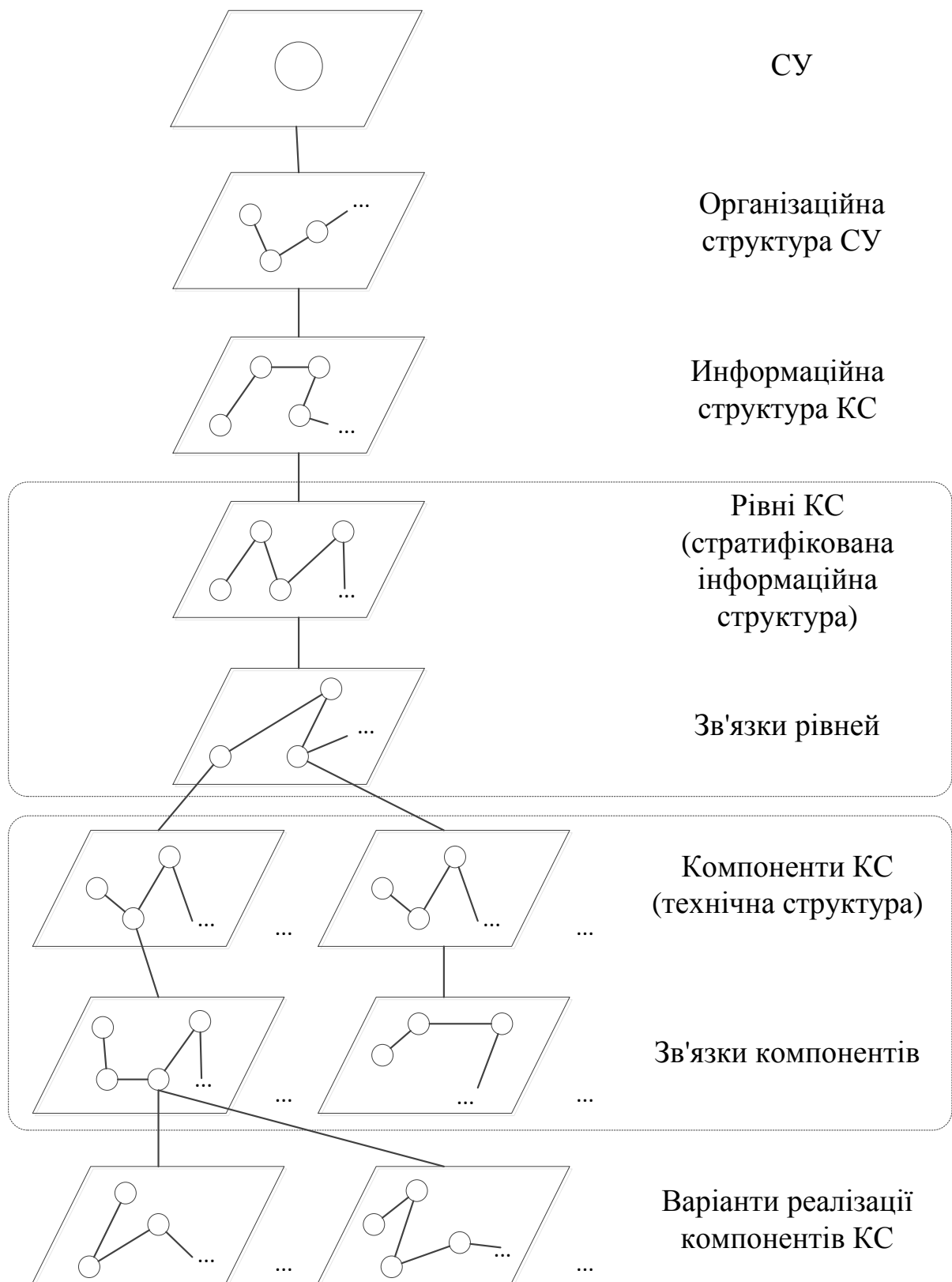
Обмеження на мінімально необхідну кількість компонентів системи, що реалізують її функціональність, в процесі вибору компонентів СУ для розв'язання етапів задачі, можна записати у вигляді наступної нерівності:

$$\sum_{j=1}^J \sum_{w_j=1}^{W_j} x_{jw_j} \geq \overline{M}.$$

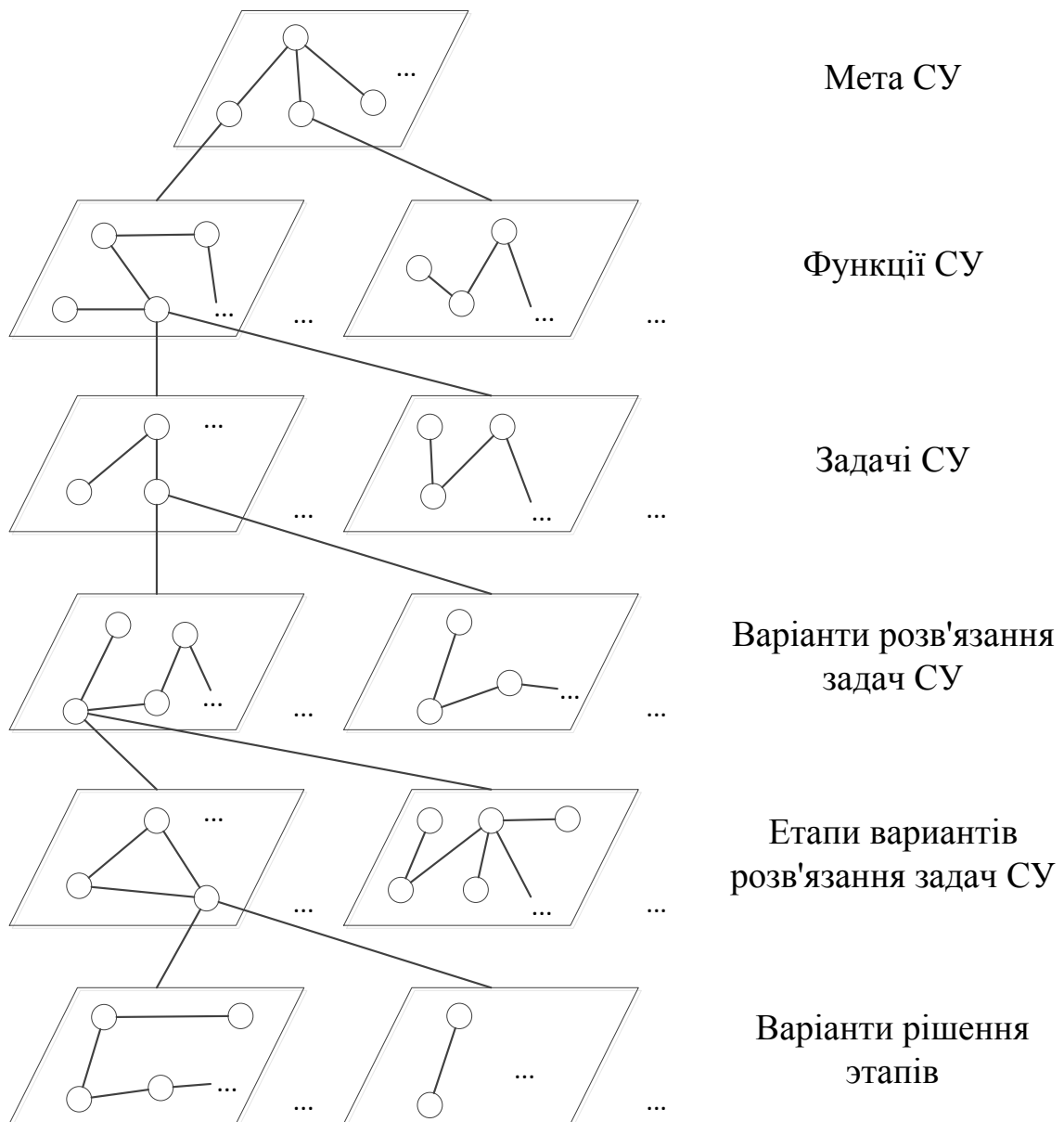
При цьому виникають два обмеження. Перше з них є обмеженням на максимальну кількість етапів однієї задачі ( $n_{ij}$ ), що розв'язується в будь-якому з компонентів системи, яке представляється як

$$\sum_{\ell_{iv_i}=1}^{L_{iv_i}} x_{i\ell_{iv_i} j} \leq n_{ij},$$

де  $j = \overline{1, J}$  – вся множина компонентів системи.



**Рис. 4.2. Ієрархічна схема етапів синтезу структури СУ**



**Рис. 4.3. Ієрархічна схема етапів синтезу логіки СУ**

Друге обмеження є обмеженням на максимальну кількість етапів різних задач ( $n_j$ ), що розв'язуються будь-яким з компонентів системи:

$$\sum_{i=1}^I \sum_{\ell_{iv_i}=1}^{L_{iv_i}} x_{i\ell_{iv_i}j} \leq n_j.$$

Оскільки, в загальному випадку, кожна задача з множини задач, що реалізуються системою, що синтезується, може реалізовуватися деякою сукупністю компонент такої системи, є справедливим наступне обмеження:

$$\sum_{j=1}^J x_{i\ell_{iv_i}j} = 1.$$

Таким чином, стає можливим завдання умови типу логічного “І” між деякими компонентами  $j$ , що реалізують певні задачі або етапи  $j \in N_{jw_j} \left( N_{i'v_i} \ell_{i'v_i j}, N_{i \ell_{i'v_i j}} \right)$ , і пов'язаним з ними множиною компонентів, задач або етапів: якщо  $x_{i'w_j} = 1$ , то  $x_{i'w'_j} = 1$  для усіх  $j'w'_j \in N_{jw_j}$ . Така умова має наступний вигляд:

$$\left( \prod_{j'w'_j \in N_{jw_j}} x_{j'w'_j} - 1 \right) x_{jw_j} = 0.$$

Як слідство з вищенаведеної умови, можлива умова типу логічного “АБО”, яка формулюється для випадку, коли необхідно виконання рівності  $x_{j'w'_j} = 1$  тільки для єдиного  $j'w'_j$  з заданої множини  $M_{jw_j}$  при  $x_{jw_j} = 1$ :

$$\left( \sum_{j'w'_j \in M_{jw_j}} x_{j'w'_j} - 1 \right) x_{jw_j} = 0.$$

Обмеження на недопущення розв'язання в безпосередньо не пов'язаних компонентах взаємопов'язаних етапів  $\ell_{i'v_i}$  та  $\ell'_{i'v_i}$  однієї задачі:

$$\sum_{i, \ell_{i'v_i}, j} \theta_{jj'} x_{i \ell_{i'v_i j}} x_{i' \ell'_{i'v_i j'}} = 1,$$

де  $\theta_{jj'} = 1$  при наявності безпосереднього зв'язку між компонентами  $j$  та  $j'$  КС.

### **Підходи до синтезу технічної структури КС СУ**

У процесі синтезу СУ об'єктом КЗ одними з виникаючих окремих задач є оптимальне відображення множини взаємопов'язаних функцій управління на множину взаємопов'язаних компонентів КС, представлених певними технічними засобами (ТЗ).

До таких ТЗ існує ряд вимог, що пред'являються, включаючи економічні вимоги на їх створення, експлуатацію та модифікацію, вимоги до оперативності реалізації функцій, а також вимоги до надійності.

Під час синтезу СУ об'єктом КЗ, крім задач розподілу функцій СУ між множиною компонентів системи відповідно до визначених критеріїв, виникають задачі вибору ТЗ і каналів зв'язку в контексті гарантування певних характеристик СУ.

В даний час існують різні підходи і моделі, що використовуються для розв'язання задач синтезу СУ; вони передбачають послідовність характерних етапів [42]:

- визначення числа, розташування і складу компонентів КС СУ для кожного з можливих класів управління, що дозволяє виконати мінімізацію витрат на реалізацію таких компонентів;

- визначення варіантів реалізації та складу компонентів і їх взаємозв'язків на основі вимог до СУ, що дозволяє виконати мінімізацію витрат на забезпечення необхідного рівня надійності при певних умовах експлуатації;

- розробка топологічної структури СУ на основі рішень двох попередніх етапів, включаючи визначення функціональної структури, розподіл функцій СУ по множині її компонентів і оптимізацію такого процесу, а також безпосередній вибір комплексу ТЗ для реалізації компонентів КС СУ, що дозволяє виконати мінімізацію витрат на реалізацію і експлуатацію компонентів;

- аналіз процесу функціонування компонентів для обраного варіанту реалізації КС СУ за допомогою імітаційного моделювання.

Таким чином, наступний матеріал даного підрозділу буде покривати перші два з розглянутих етапів.

### **Синтез технічної структури КС СУ ОКЗ**

Перш за все, при синтезі технічної структури КС СУ об'єктом КЗ на рівні компонентів і відповідних зв'язків, виникає ряд задач, що неухильно підлягають оптимізації. У загальному випадку, процес синтезу технічної структури можна розбити на два складових етапи, які можна застосувати до формалізації за допомогою апарату графів [51]:

- синтез найменшій можливій сукупності компонентів для реалізації СУ, їх розташування в рамках відповідної КС, а також кількості та характеру взаємозв'язків між компонентами; вхідною інформацією для такого етапу є вимоги доступності компонентів СУ;

- синтез прийнятних варіантів реалізації кожного з компонентів і взаємозв'язків; вхідною інформацією є вимоги до надійності СУ об'єктом КЗ,

в контексті як навколишнього середовища, так і можливих впливів на СУ під час її функціонування.

Сутність першого з етапів полягає у розв'язанні задачі знаходження мінімальної кількості компонентів КС при дотриманні вимог застосовності СУ для заданої множини задач управління, включаючи, можливо, також специфічні вимоги (наприклад, вимоги щодо диверсності компонентів для певного типу задач). Таку задачу можна зручно сформулювати у вигляді лінійної цілочисельної задачі математичного програмування, що показано нижче.

Нехай,  $i = \overline{1, I}$  – сукупність задач управління, що підлягають розв'язанню СУ,  $j = \overline{1, J}$  – сукупність компонентів КС СУ, а  $\ell = \overline{1, L}$  – фізичні області КС СУ, в яких реалізується управління за допомогою певних компонентів в залежності від типу задач управління. Тоді  $L_i$  є області КС, в яких проводиться розв'язання задач  $i$ -го типу. За допомогою  $m_i$  позначимо сукупність компонентів КС, доступних для розв'язання задач  $i$ -го типу, а за допомогою  $\tau_\ell$  – час, за який потрібно розв'язати задачу компонентами КС в її  $\ell$ -й області. Крім того, необхідно ввести наступні булеві змінні, що відображають використання компонента  $i$  області КС СУ при розв'язанні задач управління:

$$x_j = \begin{cases} 1, & \text{якщо використовується компонент } j, \\ 0 & \text{– інакше;} \end{cases}$$

$$y_\ell = \begin{cases} 1, & \text{якщо використовується область } \ell, \\ 0 & \text{– інакше.} \end{cases}$$

Тепер можна стверджувати, що основна задача, що підлягає розв'язанню, полягає в мінімізації витрат на реалізацію таких компонентів:

$$\min \sum_{j=1}^J c_j x_j,$$

де  $c_j$  – витрати на створення  $j$ -го компонента КС СУ.

Якщо задані вимоги до застосування СУ для заданої множини задач управління, то необхідно враховувати що

$$\sum_{\ell \in L_i} \tau_{\ell} y_{\ell} \leq \tau_i,$$

де  $\tau_i$  – обмеження на час розв'язання задачі управління і-го типу; а при наявності ще й спеціальних обмежень (наприклад, по диверсності компонентів для і-го типу задач), що

$$\sum_{j \in m_i} x_j \geq k_i.$$

Крім того, необхідно задати наступний взаємозв'язок між введеними змінними, враховуючи що  $n_{\ell}$  – сукупність компонентів, які можна застосувати для розв'язання задачі відповідного типу в  $\ell$ -й області КС:

$$\sum_{j \in n_{\ell}} x_j \geq y_{\ell} \geq \frac{1}{|n_{\ell}|} \sum_{j \in n_{\ell}} x_j.$$

В процесі реалізації другого етапу, для кожного з можливих у конкретній КС СУ типів задач управління задається мінімальна сукупність компонентів і взаємозв'язків між ними, що забезпечує коректне функціонування КС СУ.

Множини варіантів реалізації компонентів КС і їх взаємозв'язків визначаються як в контексті мінімізації відповідних витрат, так і в контексті надійності, що забезпечується.

Введемо наступні позначення, стосовно до часового інтервалу функціонування СУ об'єктом КЗ, що розглядається, і наявності допустимих впливів на СУ під час функціонування:  $P_{ik}(k = \overline{1, K_i})$  – ймовірність безвідмовного функціонування частини КС, що реалізує розв'язання k-го варіанту і-й задачі управління;  $P_{jk}(k = \overline{1, K_j})$  – ймовірність безвідмовного функціонування k-го варіанту реалізації j-го компонента КС СУ;  $P_{Rk}(k = \overline{1, K_r})$  – ймовірність безвідмовного функціонування k-го варіанту реалізації R-го мережевого компонента, що забезпечує функціонування КС СУ;  $P_{dk}$  – ймовірність безвідмовного функціонування складеного каналу зв'язку, що має k-й варіант реалізації, і утворює певний маршрут (d) з'єднання при розв'язанні задачі управління;  $c_{ik}, c_{jk}, c_{Rk}, c_{dk}$  – витрати на фізичну і логічну реалізацію відповідних компонентів, включаючи вузли обробки і передачі інформації, а

також канали зв'язку (що утворюють складовий канал зв'язку маршруту з'єднання для розв'язання задачі).

Таким чином, стає можливим виділення агрегованих варіантів реалізації певних областей КС СУ, з урахуванням множини і характеристик необхідних компонентів. Введемо наступні булеві змінні, що приймають одиничні значення тільки в разі вибору k-тих варіантів побудови:  $\theta_{ik}$  - і-й задачі,  $\theta_{jk}$  - j-го компонента КС,  $\theta_{Rk}$  - R-го проміжного вузла мережі КС,  $\theta_{\ell dk}$  - каналів зв'язку на маршруті з'єднання для розв'язання задачі. Таким чином, для знаходження оптимального розподілу необхідно знайти матрицю  $\Theta$ , при якій витрати мінімізуються.

Тепер можна стверджувати, що основна задача, що підлягає розв'язанню на другому етапі, полягає в мінімізації витрат на побудову компонентів, яка описується наступним виразом

$$C = \sum_{\alpha} \sum_k c_{\alpha k} \theta_{\alpha k} + \sum_{\beta} \sum_k c_{\beta k} \theta_{\beta k} \rightarrow \min$$

при обмеженнях, що описуються наступним виразом:

$$P_{i0}(\theta_{ik}, \theta_{0k}) \left[ \begin{array}{l} P_{Rk}(\theta_{Rk}, \theta_{iRk})(1 - P_{ijk}(\theta_{ijk}, \theta_{Rjk}, \theta_{jk})) + \\ + (1 - P_{Rk}(\theta_{Rk}, \theta_{iRk}))(1 - P_{ij}(\theta_{ijk}, \theta_{jk})) \end{array} \right] \geq P_i^*,$$

$$\sum_k \theta_{\alpha k} = 1, \alpha = i, j, R,$$

$$\sum_k \theta_{\beta k} = 1, \beta = \{iR, ij, jR, j0\},$$

де  $P_{Rk}(\theta_{Rk}, \theta_{iRk}) = \left( \sum_k P_{Rk} \theta_{Rk} \right) \left( \sum_k P_{iRk} \theta_{iRk} \right)$ ;

$$\begin{aligned} P_{ijk}(\theta_{ijk}, \theta_{Rjk}, \theta_{jk}) &= \\ &= \prod_j \left[ 1 - \left( 1 - \sum_k P_{ijk} \theta_{ijk} \right) \left( 1 - \sum_k P_{Rjk} \theta_{Rjk} \right) \left( \sum_k P_{jk} \theta_{jk} \right) \right], \end{aligned}$$



$$P_{ik}(\theta_{ik}, \theta_{0k}) = \left( \sum_k P_{ik}, \theta_{ik} \right) \left( \sum_k P_{0k}, \theta_{0k} \right),$$

$$P_{ij}(\theta_{ijk}, \theta_{jk}) = \prod_j \left[ 1 - \left( \sum_k P_{ijk} \theta_{ijk} \right) \left( \sum_k P_{jk} \theta_{jk} \right) \right].$$

**Висновки.** В главі запропоновано підходи до синтезу інформаційної структури системи управління об'єктом критичного застосування, що враховують вибір задач управління компонентами системи управління, реалізації алгоритмів і методів розв'язання задач і розподіл задач по компонентах в процесі їх розв'язання. Сформульовано критерій якості і можливі відповідні обмеження. Окремо порушено питання синтезу організаційної структури системи управління, яке є одним з основоположних.

Також запропоновано підхід до синтезу технічної структури системи управління об'єктом критичного застосування. Розв'язання такої задачі має містити два етапи, що включають синтез найменшої можливої сукупності компонентів для реалізації системи управління, їх розташування в рамках відповідної комп'ютерної системи, кількості і характеру взаємозв'язків між компонентами, а також синтез прийнятних варіантів реалізації кожного з компонентів і взаємозв'язків. У процесі синтезу враховуються вимоги за доступністю компонентів системи управління та процес її функціонування в контексті навколишнього середовища за умови можливих впливів.

## РОЗДІЛ 5

### СЖАТИЕ ДАННЫХ ПОРТРЕТНЫХ ИЗОБРАЖЕНИЙ

***Аннотация.** Предложены и проанализированы методы и средства сжатия портретных изображений на основе выделения объектов и фона и кодирования их с разной степенью качества. Предложенный метод разделяет элементы изображения на два класса при помощи вейвлет-преобразования изображения и выделения контуров в пространстве коэффициентов высокочастотных областей многомасштабного анализа. Обработка выделенных областей осуществляется при помощи JPEG-технологий на базе дискретного косинусного преобразования, а также с помощью многомасштабного вейвлет-анализа.*

***Ключевые слова:** сжатие изображений, выделение объектов и фона, вейвлет-преобразования, битовая плоскость, JPEG-технологии.*

***Abstract.** Proposed and analyzed the methods and means of the portrait image compression based on the selection of objects and background, and encoding them with varying degrees of quality. The proposed method separates the pixels into two classes using wavelet transform and image edge enhancement in areas of high coefficient space multiresolution analysis. Processing selections made using JPEG-based technologies discrete cosine transform and using multiresolution wavelet analysis.*

***Keywords:** image compression, the allocation of objects and the background, the wavelet transformation, the bit plane, JPEG-technology.*

**Введение и постановка задачи.** Термин “сжатие данных” означает уменьшение объема данных, используемого для представления определенного количества информации. При этом между понятиями данные и информация должны быть проведены четкие различия. Они не являются синонимами. Данные фактически являются тем средством, с помощью которых информация передается или хранится, и для представления одного и того же количества информации может быть использовано различное количество данных. Если данные несут несущественную либо повторяющуюся информацию, то говорят, что эти данные избыточны. Избыточность данных является центральным понятием цифрового сжатия изображений, которые используются практически во всех технологических и культурных областях человеческой деятельности.

Методы цифровой обработки изображений решают две основные задачи. Это повышение качества изображения для улучшения его визуального восприятия человеком и обработка изображений для их эффективного хранения, передачи, анализа и поиска в глобальной сети интернет, а также в системах компьютерного зрения, в том числе биометрических системах идентификации личности по изображениям лица и другим параметрам.

Изображения наряду с большой информационной емкостью (информативностью) обладают и большой информационной избыточностью. Поэтому одна из основных проблем обработки заключается в извлечении из исходных изображений лишь необходимой информации, т.е. в ее сжатии путем устранения избыточности. Определим следующие виды избыточности изображений: кодовая избыточность, межэлементная избыточность (статистическая избыточность), психовизуальная избыточность, структурная избыточность. Сжатие данных достигается в том случае, когда сокращается или устраняется избыточность одного или нескольких из вышеуказанных видов. Данные, в которых нет избыточности – сжать нельзя.

Возникновение кодовой избыточности можно пояснить на основе теоретико-информационного понятия энтропии источника, которое ввел Клод Шеннон в созданной им теории. Согласно этой теории среднее число бит(энтропия), необходимое для кодирования символов источника сообщений, должно быть согласовано с вероятностями появления этих символов. Более вероятным значениям символов присваиваются кодовые слова с меньшим числом бит. Такой подход называют неравномерным кодированием. Физической природой возникновения кодовой избыточности в этом случае является то, что изображения, как правило, состоят из объектов, имеющих регулярную, в некотором смысле предсказываемую морфологию (форму) и отражательные свойства поверхности. Прямым следствием этого является тот факт, что на большинстве изображений определенные значения яркости оказываются более вероятными, чем другие (т.е. гистограммы большинства изображений не являются равномерными). Обычное двоичное кодирование значений яркости таких изображений присваивает кодовые слова одинаковой длины как более вероятным, так и менее вероятным значениям. В результате и появляется кодовая избыточность. Важным следствием теории информации является тот факт, что энтропия сжатых данных совпадает с энтропией исходного источника. При этом предполагается, что по сжатым данным можно полностью восстановить исходную информацию. Такой подход принято называть сжатием без потерь (энтропийным сжатием).

Причиной возникновения межэлементной избыточности изображений является высокая разрешающая способность дискретного поля изображения], которая реализуется только вдоль контуров(границы резкого изменения цвета); на всех гладких участках изображения она расходуется впустую, т.е. возникает межэлементная избыточность(соседние пиксели окрашены в близкие цвета и сильно коррелированы), что приводит к увеличению объема информации.

При сжатии видеоинформации кроме вероятностно-статистических свойств изображения (кодовая и межэлементная избыточность) весьма важно учитывать и особенности получателя изображений. Зрение человека обладает ограниченными возможностями и характеризуется некоторыми известными (отчасти) отличительными особенностями. Использование конкретных особенностей зрения для сокращения избыточности изображений называется психофизической обработкой, заключающейся в том, что часть информации, переносимой изображением, может быть исключена (потеряна) без соответствующих негативных последствий для восприятия этого изображения человеком. Так высокочастотные и низкочастотные шумы и искажения воспринимаются слабее среднечастотных. Также искажения проявляются в меньшей степени вблизи перепадов яркости на границах объектов (пространственное маскирование).

Процесс зрительного восприятия той или иной сцены меньше всего напоминает количественный анализ яркости или координат цвета отдельных элементов изображения. Наблюдатель скорее стремится отыскать в изображении наиболее важные отличительные характеристики такого типа, как контуры или текстурные области, и образовать из них комбинации, поддающиеся распознаванию. Наличие таких областей определяет структурную избыточность изображений, которая является следствием упорядоченности реальных сцен, состоящих из более или менее однородных областей, которые в конечном счете образуют предметы той или иной сцены. Опыт человеческого восприятия одной и той же сцены в различных графических, черно-белых и цветных вариантах показал, что для сохранения семантики изображения более важным оказывается не тип поля, а форма и взаимное расположение в нем целостных структурных единиц (объектов).

Следовательно, при структурированном (контурно-текстурном) представлении цвет и яркость элементов играют по всей видимости вспомогательную роль, а успех восприятия определяется организацией экономного описания таких элементов, как контур или область. Поэтому есть все основания ожидать, что перспективные более емкие алгоритмы кодирования будут основываться главным образом на структурах изображения [50; 17].

Существует и другой вид избыточности структурного типа, которая определяется семантической природой изображения на основании цели обработки и анализа изображений – содержательная избыточность.

В задачах сжатия изображений устранение содержательной избыточности заключается в выделении объектов и фона и кодировании их с различным визуальным качеством. Это самый сложный и эффективный подход к решению задачи сжатия изображений, где в максимальной степени объединяются и используются методы распознавания образов и эффективного кодирования данных. Сложность такого подхода обусловлена недостаточной формализацией такого понятия как “объект” и отсутствием возможности иметь обязательный список обрабатываемых объектов, каждый из которых задан детерминировано, например, уравнением границ и распределением отсчетов сигнала внутри границ, что является характерным для задач распознавания и классификации.

Содержательная (семантическая) избыточность тесно связана с конкретной решаемой задачей. Для одного и того же изображения она может быть различной в зависимости от целей обработки или получателя информации, поэтому формальное ее измерение затруднительно. В частности, два изображения можно считать семантически (содержательно) эквивалентными, если они позволяют выработать идентичные решения после их анализа человеком или ЭВМ.

Тогда один из возможных подходов к передаче данных с устранением содержательной избыточности может быть такая организация последовательности передаваемых данных, при которой у получателя информации обеспечивается наиболее быстрое принятие решения. Когда решение принято, передачу (или выборку данных из памяти) можно прекратить, что эквивалентно сокращению объема передачи или сжатию данных. Следовательно, первой должна выбираться, передаваться, анализироваться и качественно кодироваться семантически наиболее насыщенная часть данных. Такой подход к кодированию на основе сокращения содержательной избыточности применяется во всех новейших информационных технологиях сжатия изображений [17].

В течение многих лет доминирующей тенденцией в исследованиях проблемы сжатия данных и изображений являются методы обработки сигналов и методы теории информации, которые используют декоррелирующие свойства различных линейных преобразований и возможности квантования и энтропийного кодирования коэффициентов этих преобразований.

Развитие этих методов привело к созданию известных JPEG-форматов, которые используют субоптимальные косинусные преобразования Фурье и методы вейвлет-преобразований, позволяющие в 20 и более раз сократить объем графических данных, сохранив при этом хорошее визуальное качество

[2; 35; 80]. Повысить количественные и качественные характеристики этих форматов можно в результате сокращения содержательной избыточности изображений. Суть такого подхода заключается в том, что в памяти с высоким качеством хранится только часть изображения, представляющая интерес в данном конкретном применении.

Основной проблемой данного подхода к сжатию изображений остается выделение значимых (информативных) участков изображений (объектов) и участков менее информативных (фона). И хотя решению этой проблемы посвящено большое количество работ [50; 17], в которых получены интересные результаты, в целом задача кодирования изображений на основе этих принципов еще далека от своего окончательного решения. Такой же вывод можно сделать и в отношении систем видеоаналитики и портретной идентификации, в которых первым и основным этапом обработки является поиск лица на изображениях в условиях изменения освещенности, ракурса или мимики лица [47].

В целях получения приемлемой сложности решения задачи автоматического выделения объектов часто используют поиск на изображении окон (шаблонов), имеющих простую форму и стилизованное содержание. Такой подход использован в системе для нахождения лиц [47]. Модель системы предполагает, что все лица имеют определенный размер и занимают область известной формы, а для подходящей маркировки используется алгоритм кластеризации и большое число изображений-образцов, который требует одинаковые условия освещенности на исходном изображении.

В той же работе [47] описано средство поиска лиц с помощью нейронных сетей, преобразующих окно шаблона во фронтальную проекцию. Однако если внешний вид объекта изменяется, то сравнение с шаблоном становится более сложным, поскольку требуется применить гораздо большее их количество. Системы, которые используют методы корреляционного поиска на мультимасштабной пирамиде сжатых изображений, требуют анализа очень большого объема данных, что приводит к чрезвычайно большим вычислительным затратам.

В цифровых библиотеках изображений средства поиска пытаются оценить семантику на уровне объектов. Такие системы вначале сегментируют изображение, а затем фокусируются на некоторых сегментах. Сегментация здесь является ключевым вопросом, поскольку она определяет какую информацию при сравнении изображений можно отбросить как несущественную. Система BlobWorld построена на этих принципах.

Изображения в ней представляются как наборы областей когерентного цвета и текстуры. Однако форма этих областей представлена грубо и дальнейший выбор объекта осуществляется в диалоговом режиме с пользователем в полуавтоматическом режиме. Наиболее продвинутое и эффективное решение в области распознавания лиц и биометрической идентификации опубликовано Национальным институтом стандартов и технологий США (<http://www.frvt.org/FRVT2006/>).

Кодирование изображений с различной степенью качества реализовано в графическом формате DjVu [121]. Основная идея этого формата заключается в отделении на изображении текста (передний план) от фона (задний план) и использовании для каждого из них наиболее подходящего метода сжатия.

Однако применяемый в формате метод мультимасштабной кластеризации не всегда дает ожидаемые результаты при определении, что есть передний план (объект), а что есть фон, а при кодировании полутоновых изображений типа портрет метод DjVu вообще перестает работать и сводится к традиционному алгоритму вейвлет-компрессии, как в JPEG 2000.

**Основная часть.** Предлагаемый ниже метод устраняет эти недостатки и разделяет элементы изображения на два класса (объект и фон) при помощи вейвлет-преобразования изображения и выделения контуров в пространстве коэффициентов высокочастотных областей многомасштабного анализа. Конечным результатом выделения объектов на изображении является построение битовой плоскости, которая содержит информацию о расположении объекта и фона. Дальнейшая обработка выделенных областей осуществляется при помощи JPEG-технологий на базе дискретного косинусного преобразования, а также с помощью многомасштабного вейвлет-анализа.

Для анализа алгоритма из библиотеки стандартных изображений <http://www.icsl.ucla.edu>, был выбран файл zelda.bmp, который представлен на рис. 5.1. Параметры исходного изображения: размер 256x256, глубина цвета в градации серого 8 бит. Вейвлет коэффициенты исходного изображения, после одного шага преобразования, представлены на рис. 5.2.

На практике вейвлет-преобразование проводится с помощью биортогональных вейвлет-базисов по формулам разложения (1) и синтеза (2). Если  $\{a_i\}$  – коэффициенты разложения остаточного члена, то  $\{b_i\}$  и  $\{c_i\}$  – коэффициенты, соответственно нового остаточного члена в следующем масштабе представления

$$b_i = \sum_s h_s a_{2i+s} \text{ и } c_i = \sum_s g_s a_{2i+s} . \quad (1)$$

Аналогичным образом, если известны коэффициенты  $\{b_i\}$  и  $\{c_i\}$ , то коэффициенты  $\{a_i\}$  можно восстановить в соответствии с выражением

$$a_i = \sum_s h_s b_{i-s} + \sum_s g_s c_{i-s} . \quad (2)$$

Для разложения и восстановления сигнала используют две разные пары квадратурных зеркальных фильтров: пара  $\{\tilde{h}, \tilde{g}\}$  при разложении и пара  $\{h, g\}$  при восстановлении. Основное уравнение, в данном случае имеет вид

$$\overline{h(\omega)\tilde{h}(\omega)} + \overline{h(\omega + \pi)\tilde{h}(\omega + \pi)} = 2 .$$

Для этого равенства существует семейство решений:

$$\tilde{h}_{n,m}(\omega) = \sqrt{2} \left( \frac{1 + e^{i\omega}}{2} \right)^n ; h_{n,m}(\omega) = \sqrt{2} \left( \frac{1 + e^{i\omega}}{2} \right)^2 P_m \left( \sin^2 \frac{\omega}{2} \right) e^{-im\omega} , \quad (3)$$

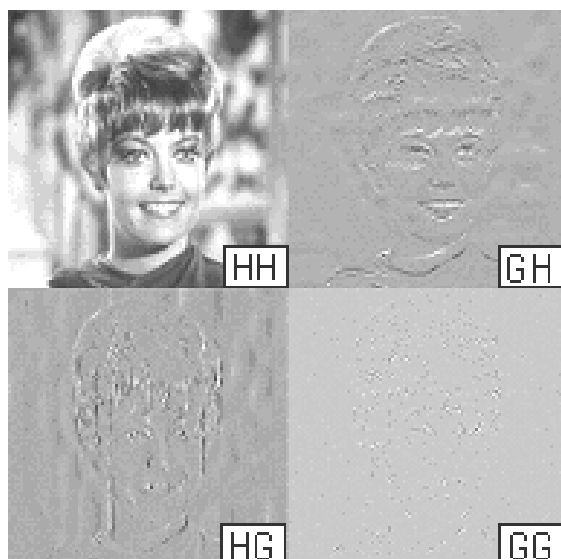
где  $P_m(x) = \sum_{s=0}^{m-1} C_{m-1+s}^s x^s$  – многочлен Дебеша.

Для вейвлет преобразования исходного изображения, рис. 5.1, использовалась пара квадратурных зеркальных фильтров  $h$  и  $\tilde{h}$  с параметрами  $n=1$  и  $m=2$ , выражение (3), общий вид, которых представлен на рис. 5.3.

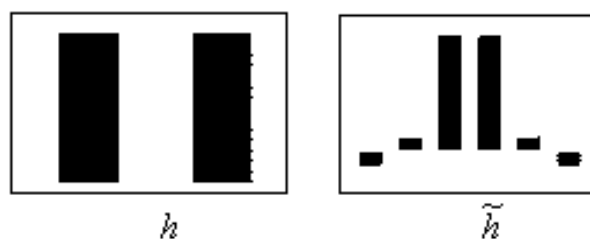


*Рис. 5.1. Исходное изображение*





*Рис. 5.2. Вейвлет коэффициенты после одного шага преобразования*



*Рис. 5.3. Пара квадратурных зеркальных фильтров с параметрами  $n=1$  и  $m=2$*

При построении разделяющей битовой плоскости, использовались вейвлет коэффициенты  $HG$  и  $GH$  областей (рис. 5.2), которые были получены в соответствии с выражением (1). После усредненного суммирования выбранных вейвлет-коэффициентов  $(HG+GH)/2$  получим изображение, которое представлено на рис. 5.4.



*Рис. 5.4. Изображение полученное в результате усредненного суммирования вейвлет коэффициентов  $HG$  и  $GH$  областей*

Полученное изображение имеет более равномерное распределение яркости по сравнению с оригиналом (рис. 5.1), или остаточным членом разложения НН (рис. 5.2). Это качество достигается за счет применения соответствующей пары фильтров  $h$  и  $\tilde{h}$  при вейвлет преобразовании исследуемого изображения. Исходное изображение раскладывается на “грубую” (НН) и ”тонкую”(GG) структуру, (рис. 5.2). Области НГ и ГН имеют более гладкий рельеф яркости на всей плоскости, но при этом сохраняют наиболее существенные ее перепады. Выбор именно этих составляющих (НГ и ГН) обусловлен тем, что они получены в результате обработки исходного изображения одной парой фильтров  $h$  и  $\tilde{h}$ , но только в обратной последовательности их применения. При этом, область НГ содержит наиболее выраженные перепады яркости в вертикальном направлении, что обусловлено последовательностью операций свертки исходного изображения с соответствующими фильтрами  $h$  и  $\tilde{h}$ . Область ГН наоборот, имеет резкие скачки яркости в горизонтальной плоскости.

Усредненное суммирование НГ и ГН плоскостей позволяет сохранить резкие изменения яркости, которые обязательно присутствуют в обоих направлениях. Усредненное суммирование НГ и ГН областей формирует плоскость изображения для пороговой обработки (рис. 5.4) по формуле:

$$X_{i,j} = (HG_{i,j} + GH_{i,j})/2, \quad (4)$$

где  $X_{i,j}$  – точка усредненного изображения,  $i,j=1..n$ ,  $n$ -количество строк и столбцов в матрице изображения размерностью  $n \times n$ .

Применение пороговой обработки к изображению представленного на рис. 5.4, позволяет получить совокупность точек, между которыми значения разности яркости превышает пороговое значение ( $|X_i - X_{i+1}| \geq KX_{cp}$ ), (рис. 5.5 (а)). Формирование плоскости  $Z_{i,j}$ , которая содержит точки превышения порогового значения, (рис. 5.5 (а)) согласно правилу

$$Z_{i,j} = \begin{cases} 1, & \text{если } |X_{i,j} - X_{i,j+1}| \geq KX_{cp}, \text{ или } |X_{i,j} - X_{i+1,j}| \geq KX_{cp}; \\ 0, & \text{если } |X_{i,j} - X_{i,j+1}| < KX_{cp}, \text{ или } |X_{i,j} - X_{i+1,j}| < KX_{cp}. \end{cases} \quad (5)$$

где  $i,j=1 \dots n-1$ ,  $n$ -количество строк и столбцов в матрице изображения.

Пороговое значение привязывалось к среднему значению яркости  $X_{cp}$  на всей плоскости изображения (рис. 5.4), и выбиралось таким образом, чтобы исключить наличие случайных точек, т.е. тех точек, которые не имеют в своем ближайшем окружении себе подобных. Таким образом, производится

фильтрация одиночных точек преодолевших пороговое значение. На рис.5.5 (а) наглядно видны область с высокой плотностью перепадов яркости и область, которая практически не содержит смежных точек превышения порога. Пороговая обработка изображения представленного, на рис. 5.4 проводилась по строкам и столбцам матрицы изображения.

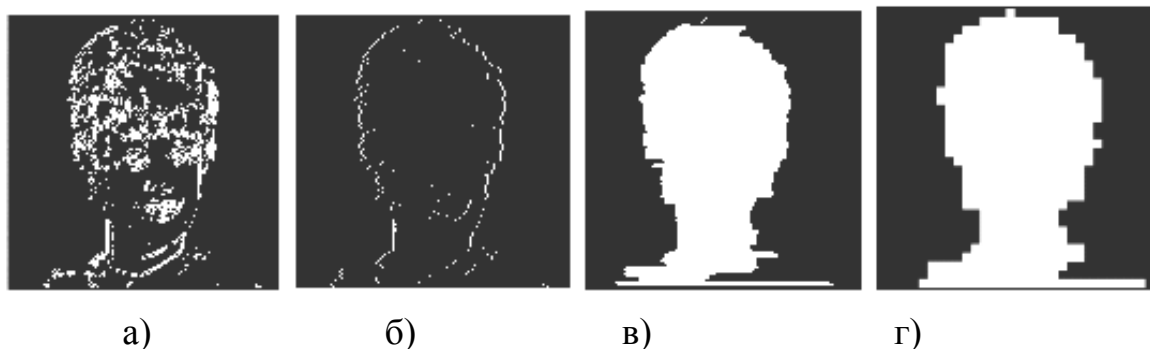
Затем, для отделения объекта от фона, производится последовательное сканирование изображения по строкам и столбцам до выявления первой и последней не нулевой точки. После разделения изображения на области объекта и фона, необходимо создать непрерывную и замкнутую границу между этими областями. На рис. 5.5 (б), представлены граничные точки, которые разделяют области объекта и фона. Эта граница раздела имеет разрывы и не является замкнутой линией. Для формирования непрерывного и замкнутого контура требуется соединить точки разрыва границы между связными фрагментами контура (см. рис. 5.6). Точкой разрыва линии границы считается любая точка имеющая только одну смежную точку, т. е. одного "соседа". Для соединения точек разрыва линии границы  $X_j$  необходимо найти минимальное расстояния между точками разрыва принадлежащим разным подмножествам  $\{R_i\}$ , т.е. концом связного фрагмента границы и началом другого. После соединения граничных точек между собой получим битовую плоскость разделяющую объект и фон, рис. 5.5(в). Построение битовой плоскости является завершающим этапом выделения объекта на плоскости изображения. Точки внутри контура считаются принадлежащими объекту и им присваивается значение 1, точки лежащие вне контура считаются фоном и им присваивается значение 0, рис. 5.5 (в).

В случае, когда компрессия изображений осуществляется при помощи JPEG-технологии на основе дискретного косинусного преобразования, то предварительно осуществляется разбиение плоскости изображения на фрагменты размером  $8 \times 8$  [JPEG]. После одного шага вейвлет преобразования плоскость изображения представленная на рис. 5.4 уменьшилась в 4 раза по сравнению с плоскостью исходного изображения, рис. 5.1. Следовательно, размер фрагмента, для битовой плоскости, должен быть  $4 \times 4$ , что полностью согласуется с разбиением исходного изображения (рис. 5.1) на фрагменты  $8 \times 8$ . Каждый фрагмент может принадлежать трем областям: область объекта, область фона, область границы битовой плоскости. Таким образом, фрагментами области объекта можно считать те фрагменты, которые принадлежат, как области самого объекта, так и области границы битовой

плоскости. Все остальные сегменты плоскости изображения принадлежат области фона.

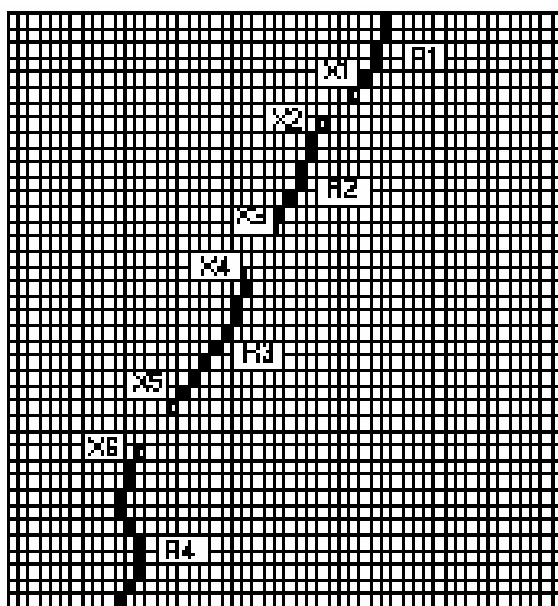
На рис. 5.5 (г) представлена битовая плоскость разделения изображения после сегментного разбиения на фрагменты 4x4.

В том случае, когда сжатие исходного изображения (рис. 5.1) производится с использованием многомасштабного вейвлет преобразования, то сегментация битовой плоскости (рис. 5.5 (в)) не производится.



**Рис. 5.5 а) Совокупность точек превысивших разностный порог; б) граничные точки разделения объекта и фона; в) битовая плоскость разделения; г) битовая плоскость после сегментации**

Для кодирования изображений с различной степенью точности при использовании JPEG-технологий необходимо исходное изображение (рис. 5.1) разбить на сегменты размером 8x8 и осуществить кодирование объекта и фона изображения по методу JPEG-технологии.



**Рис. 5.6. Связные фрагменты  $R_i$  и точки разрыва линии границы  $X_j$  между областями объекта и фона.**

Согласно метода JPEG-компрессии ядро одномерного косинусного преобразования имеет вид:

$$L_x(0) = \frac{1}{\sqrt{N}} \sum X(m); \quad L_x(k) = \sqrt{\frac{2}{N} \sum_{m=0}^{N-1} \left( X(m) \cos \frac{(2m+1)k\pi}{2N} \right)}. \quad (7)$$

В этом выражении  $X(m)$  – значение отчетов в строках соответствующих рабочих матриц, полученных на втором шаге JPEG-технологий,  $L_x(k)$  – коэффициенты дискретного косинусного преобразования, а значения  $m$  и  $k$  изменяются соответственно от 0 до  $N-1$  и от 1 до  $N-1$  с шагом один.

Различная степень точности кодирования объекта и фона определяется количеством косинусных коэффициентов, которые используются в рабочих матрицах объекта и фона. Для фона выберем минимальное значение  $-1$  (имеется ввиду первый коэффициент рабочей матрицы  $8 \times 8$  в каждом сегменте изображения), а для объекта максимальное – 64 коэффициента. Таким образом, сжатие изображения будет осуществляться только за счет сегментов, которые входят в область объекта на битовой плоскости. Область фона будет представлена только одним коэффициентом косинусного преобразования в каждом сегменте области фона. Такой подход позволяет обеспечить максимальное сжатие области фона изображения, а в области объекта обеспечить такую степень компрессии и качества изображения, которая являлась бы приемлемой для решения поставленной задачи.

На рис. 5.7 представлена зависимость общего коэффициента сжатия всей плоскости изображения ( $K_{сж}$ ) от среднеквадратической ошибки (СКО) на плоскости объекта.

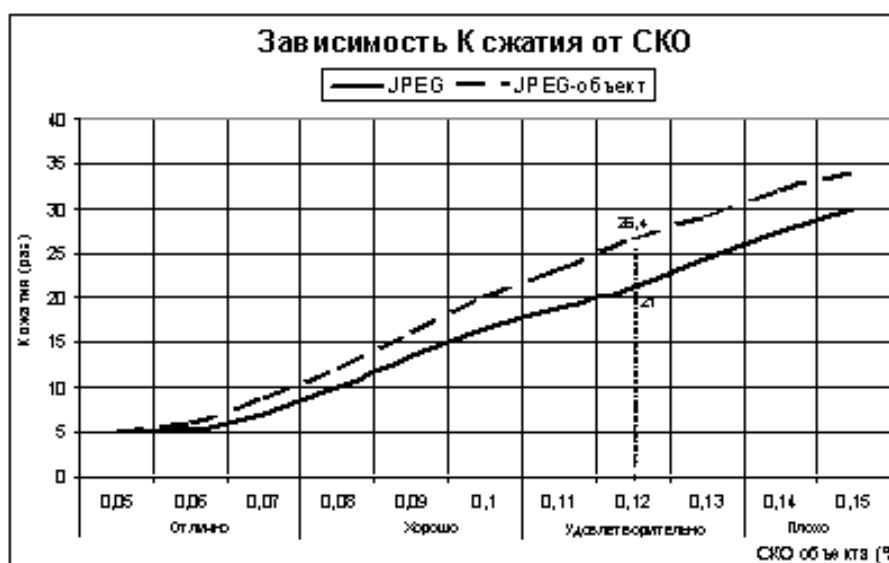


Рис. 5.7. Зависимость  $K_{сж}$  всей плоскости изображения от СКО объекта

На рис. 5.8, представлено исходное изображение после кодирования методом JPEG компрессии. Рис. 5.8(а) показывает восстановленное изображение после раздельного кодирования объекта и фона с различной степенью качества, а рис. 5.8(б) после кодирования всей плоскости изображения с одинаковой степенью качества. СКО на плоскости объекта в обоих вариантах составляет 12%, а увеличение коэффициента сжатия с 21 раза до 26,4 обусловлено большей степенью компрессии области фона (в данном случае К сжатия фона составляет 64 раза, так как только один косинусный коэффициент рабочей матрицы 8x8 представляет каждый сегмент области фона). При JPEG кодировании изображений битовую плоскость разделения необходимо кодировать энтропийными методами и хранить или передавать вместе с самим изображением в сжатом виде, что в незначительной мере влияет на степень компрессии самого изображения.

Кодирование изображений на основе вейвлет преобразований с предварительным выделением объекта имеет некоторые отличия от JPEG кодирования области объекта и фона.



а) К сжатия =26,4 СКО объекта=12%.      б) К сжатия 21, СКО объекта=12%.

**Рис. 5.8. Изображение после раздельного кодирования методом JPEG-технологий**

Выделение объекта на плоскости изображения производится таким же образом, как и в случае преобразования в JPEG формат. Однако, после формирования битовой плоскости разделения изображения на объект и фон, (рис. 5.5 (г)) применить её необходимо только к вейвлет-коэффициентам представленных в областях HG, GH и GG на каждом шаге разложения.

Сама битовая плоскость, при каждом шаге разложения изображения, подвергается соответствующему масштабированию, которое должно быть согласовано с каждым шагом вейвлет преобразования. Таким образом, битовая плоскость в соответствующем масштабе представления применяется к областям HG, GH и GG на каждом шаге вейвлет преобразования, что позволяет обеспечить сохранность вейвлет-коэффициентов области объекта на участках HG, GH и GG, которые в свою очередь отображают высокочастотную составляющую исходного изображения. В результате, все вейвлет-коэффициенты, которые принадлежат области объекта на всех этапах разложения сохраняются, а коэффициенты, которые принадлежат области фона в соответствующих масштабах игнорируются. Необходимо заметить, что использование битовой плоскости на остаточном члене разложения (область HH) приведет к потере низкочастотной составляющей области фона, что в свою очередь полностью удалит из исходного изображения текстуру фона. При таком подходе, общий коэффициент сжатия будет более высоким, но на восстановленном изображении текстура фона будет полностью отсутствовать или заменена на абстрактную текстуру, которая не имеет никакого отношения к обрабатываемому изображению. Использование вейвлет преобразования при кодировании изображений с разной степенью точности не требует отдельного кодирования и хранения битовой плоскости вместе с изображением. Битовая плоскость используется только на этапе разложения изображения, при восстановлении вейвлет-коэффициенты будут распределены в обратном порядке Z-сканирования плоскости разложения. Этот факт можно отнести к преимуществу вейвлет кодирования изображений по сравнению с методом JPEG.

На рис. 5.9 представлена зависимость коэффициента сжатия от СКО при кодировании изображения с использованием вейвлет преобразования.

Для отметки СКО =12 % степень сжатия при использовании вейвлет преобразования соответствует 24,5 раза (кривая – Вейвлет). При использовании тех же фильтров разложения и восстановления, но с разделением изображения на объект и фон степень сжатия увеличилась до 28,5 раза при том же значении СКО=12 % (кривая – Вейвлет-объект). Следует заметить, что при аналогичном значении ошибки восстановления (СКО=12 %) степень сжатия при JPEG кодировании изображения с выделением объекта равна 26,4 раза (рис. 5.7), что превышает соответствующий показатель при обычном вейвлет анализе на 8 % (рис. 5.9, кривая – Вейвлет).



**Рис. 5.9. Зависимость коэффициента сжатия ( $K_{сж}$ ) от среднеквадратической ошибки (СКО) при вейвлет анализе изображения**

На рис. 5.10(а, б) представлены восстановленные изображения полученные после вейвлет кодирования всей плоскости изображения, (рис. 5.10(а)), а так же после отдельного кодирования области фона и объекта, (рис. 5.10(б)), при одинаковом значении ошибки восстановления, СКО=12 %.



а)  $K_{сж}$  =24,5, СКО=12%  
(без выделения объекта)



б)  $K_{сж}$  =28,5, СКО=12%,  
(с выделением объекта)

**Рис. 5.10. Восстановленные изображения после выделения объекта и вейвлет-кодирования**



**Заключение.** Предложенный метод разделения изображения на объект и фон, позволяет кодировать их отдельно, как с различной степенью качества, так и с использованием различных методов. Это дает возможность представлять наиболее информативные области изображения с более высоким качеством, чем остальные его составляющие. Благодаря такому подходу можно сохранить высокое качество информативных областей (объектов), менее информативные области (фон) представить с большей компрессией, но с худшим качеством и таким образом повысить общий коэффициент сжатия для всего изображения.

Предложенный подход позволил выявить резервы классической схемы JPEG-кодирования и получить лучшие количественные характеристики по сравнению с вейвлет-кодированием на основе формата JPEG 2000.

Использование метода выделения объекта в схеме классического JPEG-формата дает более высокое повышение эффективности кодирования, чем выделение объекта в схеме многомасштабного вейвлет преобразования изображения (JPEG 2000). Это объясняется естественной локальной природой многомасштабного анализа, схема преобразований которого уже априори приспособлена для разделения изображений. Метод выделения объекта(лица) на изображении устойчив к изменениям внешности, степени освещенности, мимики лица и может с успехом использоваться в криминалистических системах портретной идентификации.

## РОЗДІЛ 6

# МНОГОКРИТЕРИАЛЬНАЯ НЕЙРОИДЕНТИФИКАЦИЯ НЕЛИНЕЙНЫХ ОБЪЕКТОВ НА ОСНОВЕ КОЭВОЛЮЦИОННОГО ПОДХОДА

***Аннотация.** Предложено использование коэволюционного подхода к построению математических моделей нелинейных объектов при наличии помех измерений. Рассмотрена общая структура эволюционной нейросети прямого распространения. Проведено имитационное моделирование различных динамических объектов, подтвердившее эффективность развиваемого подхода.*

***Ключевые слова:** многокритериальная оптимизация, нейронная сеть, генетический алгоритм, нелинейный объект, множество Парето, оптимальное решение.*

***Abstract.** In this paper it is proposed to utilize coevolutional approach to constructing mathematical models of nonlinear objects in the presence of non-gaussian noise. The general structure of evolutionary feed-forward neural network is considered. The modeling of the proposed approach for various cases of non-linear objects is performed. Simulation proved the effectiveness of the developed algorithms.*

***Key words:** multiobjective optimization, neural network, genetic algorithm, nonlinear object, Pareto set, optimal solution.*

**Введение.** Задача идентификации нелинейного нестационарного динамического объекта, описываемого моделью:

$$y(k) = f(\mathbf{x}(k), k) + \xi(k), \quad (1)$$

где  $\mathbf{x}(k) = [y(k-1), \dots, y(k-l), u(k-1), \dots, u(k-m)]^T$  –  $N \times 1$  вектор обобщенного входного сигнала ( $N = l + m$ );  $y(i)$ ,  $u(i)$  – выходной и входной сигналы объекта в момент времени  $i$  соответственно;  $l$  и  $m$  – порядки запаздывания по выходному и входному каналам соответственно;  $f(\bullet)$  – неизвестная нелинейная функция;  $\xi(k)$  – помеха, состоит в получении оценки функции  $f(\bullet)$  по измерениям входных и выходных переменных.

Трудности, связанные с идентификацией нелинейных динамических объектов классическими методами, привели к появлению и развитию альтернативного, нейросетевого, подхода к решению этой задачи.

Для обучения сетей прямого распространения с учителем применяются, как правило, алгоритмы, оптимизирующие некоторую целевую функцию. Однако традиционно во внимание принимается лишь одна цель в качестве

стоимостной функции либо несколько целей объединяются в одну скалярную функцию. Это главным образом обусловлено тем, что большинство обычных алгоритмов обучения являются одноцелевыми, т.е. могут работать только со скалярными стоимостными функциями. Обычно в таких алгоритмах осуществляется поиск минимума функционала на некоторой выборке обучающих данных

$$F(e) = \frac{1}{k} \sum_{i=1}^k \rho(e(i, \theta)), \quad (2)$$

т.е. являющегося решением системы уравнений

$$\frac{\partial F(e)}{\partial \theta_j} = \sum_{i=1}^k \rho'(e(i, \theta)) \frac{\partial e(i, \theta)}{\partial \theta_j} = 0, \quad (3)$$

где  $\rho(e(i, \theta))$  – некоторая функция потерь, зависящая от вида закона распределения помехи  $\xi$ ;  $e(i) = y(i) - \hat{y}(i)$ ;  $\hat{y}(i)$  – выходной сигнал модели;

$$\rho'(e(i, \theta)) = \frac{\partial \rho(e(i, \theta))}{\partial e(i, \theta)} \text{ – функция влияния.}$$

Более мощным по сравнению с обучением на основе скалярной стоимостной функции является многоцелевое обучение на основе подхода Парето, когда минимизируется векторная целевая функция, учитывающая сложность модели, время ее обучения, реакции на входной сигнал и т.д.

При использовании ИНС, как и при традиционном подходе, также возникают задачи структурной и параметрической оптимизации, соответствующие выбору оптимальной топологии сети и ее обучению (настройке параметров). Если задача определения структуры является дискретной оптимизационной (комбинаторной), то поиск оптимальных параметров осуществляется в непрерывном пространстве с помощью классических методов оптимизации.

Традиционные методы определения структуры сети заключаются либо в последовательном ее усложнении путем ввода новых нейронов и новых связей между ними, либо в последовательном ее упрощении, начиная с некоторой достаточно сложной топологии.

Для обучения (оценивания параметров) сети применяются, как правило, методы, требующие вычисления градиента используемого функционала  $\rho'(e(i, \theta))$  и обладающие в связи с этим рядом существенных недостатков (зависимость получаемого решения от формы минимизируемой функции и требование ее дифференцируемости, застревание в локальных экстремумах, и т.д.).

Попытки устранить недостатки традиционных методов синтеза и функционирования ИНС привели к появлению нового класса сетей – эволюционирующих ИНС (ЭИНС), в которых помимо обучения используется другая фундаментальная форма адаптации – эволюция, что позволяет изменять структуру сети, ее параметры и алгоритмы обучения без внешнего вмешательства.

Основным преимуществом использования эволюционных алгоритмов (ЭА) для оптимизации ИНС является то, что структура и параметры сети могут быть закодированы в геноме и определяться параллельно. Более того, в отличие от большинства алгоритмов оптимизации, предназначенных для потактового решения задачи, ЭА оперируют с множеством решений – популяцией, что позволяет достичь глобального экстремума, не застревая в локальных.

Среди ЭА, являющихся стохастическими и включающих эволюционное программирование, эволюционные стратегии, генетические алгоритмы, генетическое программирование, в частности, программирование с экспрессией генов, одними из наиболее распространенных являются генетические алгоритмы (ГА) [111].

Так как получаемая математическая модель, с одной стороны, должна быть достаточно простой и удобной для использования ее в задачах прогнозирования, управления и т.д., а с другой – наиболее полно отражать свойства исследуемого объекта, ее качество определяется некоторым набором критериев, т.е. задача идентификации является многокритериальной. Следует отметить, что в последнее время задачу идентификации все чаще стремятся решать с помощью коэволюционирующих адаптивных систем, т.е. систем, состоящих из разнообразных эволюционирующих групп особей, которые действуют совместно для выполнения сложных вычислений или выработки совместного эффективного поведения.

В работе предлагается многокритериальный коэволюционный алгоритм определения архитектуры нейронных сетей прямого распространения и их обучения, основанный на стратегиях кооперации и конкуренции. Алгоритм обучения реализует среду, способствующую сотрудничеству и конкуренции популяций, в которых каждая особь представляет собой ИНС прямого распространения, а вся совокупность популяций несет ответственность за окончательное решение поставленной задачи.

Получаемая нейросетевая модель, с одной стороны, должна быть достаточно простой и удобной для использования ее в прикладных задачах, а с другой – наиболее полно отражать свойства исследуемого объекта, ее качество

определяется некоторым набором критериев, которые помимо всего прочего, должны контролировать сложность модели. Таким образом, задача построения нейромодели является многокритериальной.

**Задача многокритериальной оптимизации (МО)**, которую также часто называют мультикритериальной или векторной оптимизацией, заключается в нахождении такого вектора решений, удовлетворяющего определенным ограничениям, который давал бы приемлемые значения для всех целевых функций [105]. Следовательно, существует множество целевых функций (вектор целей), которые оптимизируются (минимизируются или максимизируются) одновременно. Эти цели часто вступают в противоречие друг с другом таким образом, что улучшение одной из них приводит к ухудшению другой, т.е. не существует единого оптимального решения, наилучшего по отношению ко всем целевым функциям. Однако есть множество оптимальных решений задачи многоцелевой оптимизации, известное как Парето оптимальные решения или фронт Парето [128]. Понятие фронта Парето в области значений целевых функций в задаче МО обозначает набор таких решений, которые являются недоминируемыми по отношению друг к другу, но в то же время доминирующими над всеми остальными решениями в пространстве поиска. Это означает, что невозможно найти единое решение, которое бы превосходило все другие по отношению ко всем целям, т.е. переход между решениями, принадлежащими фронту Парето, не может привести к улучшению всех целей одновременно – такой переход приведет к ухудшению не менее одной цели. Таким образом, каждое решение множества Парето включает в себе по крайней мере одну цель хуже, чем у другого решения из данного множества, хотя они оба превосходят все другие в оставшейся части пространства поиска. Математически, данная задача может быть сформулирована следующим образом.

Требуется найти такой вектор  $X^* = [x_1^*, x_2^*, \dots, x_n^*]^T$ , который бы оптимизировал вектор целевых функций

$$F(X) = [f_1(X), f_2(X), \dots, f_k(X)]^T \quad (4)$$

при наличии  $m$  ограничений в виде неравенств

$$g_i(X) \leq 0, i = \overline{1, m}, \quad (5)$$

и  $p$  ограничений в виде равенств

$$h_j(X) = 0, j = \overline{1, p}, \quad (6)$$

где  $X^* \in \mathfrak{R}^n$  – вектор решений,  $F(X) \in \mathfrak{R}^k$  – вектор целевых функций, каждая из которых должна быть оптимизирована. Обычно полагают, что все целевые функции должны быть минимизированы. В данном подходе используются следующие определения.

**Доминирование по Парето.** Вектор  $U = [u_1, u_2, \dots, u_n]^T \in \mathfrak{R}^k$  доминирует над вектором  $V = [v_1, v_2, \dots, v_n]^T \in \mathfrak{R}^k$  (обозначается  $U < V$ ) тогда и только тогда, когда  $\forall i \in \{1, 2, \dots, k\}, u_i \leq v_i \wedge \exists j \in \{1, 2, \dots, k\}: u_j < v_j$ . Другими словами, существует как минимум одна компонента вектора  $u$  ( $u_j$ ), которая меньше, чем  $v_j$ , в то время как остальные компоненты вектора  $u$  меньше либо равны соответствующим компонентам вектора  $v$ .

**Оптимальность по Парето.** Точка  $X^* \in \Omega$  ( $\Omega$  некоторая область пространства  $\mathfrak{R}^n$ , удовлетворяющая условиям (5)-(6)), является Парето-оптимальной по отношению ко всем  $X \in \Omega$  тогда и только тогда, когда  $F(X^*) \not\prec F(X)$ , т.е. решение  $X^*$  является Парето-оптимальным если не может быть найдено никакое другое решение, которое бы доминировало над  $X^*$ .

**Множество Парето.** Множеством Парето  $P^*$  называется такой набор векторов  $X \in \Omega$ , что не существует такого вектора  $X' \in \Omega$ , для которого бы выполнялось условие  $F(X') \prec F(X) \forall X \in P^*$ .

**Фронт Парето.** Фронтом Парето  $PF^*$  называется такой набор векторов значений целевых функций  $F(X) \in \mathfrak{R}^k$ , который получен с помощью векторов из множества Парето, т.е.

$$PF^* = \{F(X) = (f_1(X), f_2(X), \dots, f_k(X)): X \in P^*\}.$$

Для нахождения фронта Парето в задачах МО широко используются эволюционные алгоритмы (ЭА), так как их свойства подходят для таких типов задач. Это обусловлено главным образом их параллельным или популяционным подходом к поиску решений, что позволяет устранить большинство трудностей и недостатков классических методов решения задач МО.

В общем виде алгоритм поиска фронта Парето с помощью ЭА может быть записан следующим образом.

1. Сгенерировать начальную обучающую выборку, состоящую из векторов входных переменных  $X \in \Omega$ . Вычислить векторы значений целевых функций  $F(X)$  для всех  $X$ .

2. На основе обучающей выборки  $X \in \Omega$  и соответствующих значений  $F(X)$  построить модели всех целевых функций  $f_1(X), f_2(X), \dots, f_k(X)$ .

3. На основе полученных моделей  $f'_1(X), f'_2(X), \dots, f'_k(X)$  с помощью выбранного алгоритма определяется Парето-оптимальное множество  $PF^*$  решений задачи (4) – (6).

4. В точках полученного множества решений  $PF^*$  вычисляются точные значения функций  $f_1(X), f_2(X), \dots, f_k(X)$ . Если критерий останова не выполняется (достигнута требуемая точность моделей и построен фронт Парето либо осуществлено максимально допустимое число итераций), то все полученные значения добавляются в обучающую выборку и осуществляется возврат к шагу 2, на котором уточняются модели целевых функций.

Важным этапом данного алгоритма является получение моделей целевых функций. Наличие данных моделей обусловлено тем, что в реальных системах получение значений целевых функций является весьма трудоемкой задачей. При решении же задачи МО подобные вычисления необходимо проводить многократно. В связи с этим, точные и быстродействующие модели целевых функций в значительной мере ускоряют задачу получения Парето-оптимальных решений. Получить такие модели можно с помощью радиально базисных сетей (РБС), которые являются универсальными аппроксиматорами, обладающими простой структурой и отличающиеся высокой скоростью обучения. Среди эволюционирующих алгоритмов оптимизации при решении данной задачи наибольшее распространение получили генетические алгоритмы (ГА).

#### **Козволюционные ГА.**

ГА являются быстрым, эффективным методом оптимизации, который способен решать широкий спектр задач. Однако, основным недостатком ГА является постоянное стремление к популяции, содержащей один локальный оптимум, что уменьшает генетическое разнообразие популяции, приводящее к снижению способности ГА поиска глобального оптимума и/или адаптации к изменяющимся параметрам целевых функций. Результаты последних исследований свидетельствуют о том, что существенно замедлить или полностью устранить вероятную потерю разнообразия можно, используя идеи коэволюции [87].

В природе конкуренция и сотрудничество играют решающую роль выживания для различных видов в условиях ограниченных ресурсов. Сотрудничество позволяет выжить слабым видам в среде, в которой различные виды конкурируют за ограниченные ресурсы. Различные виды, имеющие

различные уровни приспособленности в соответствии с их генами, разнообразием популяций, а также в зависимости от условий окружающей среды, конкурируют и сотрудничают с другими видами с целью обмена ресурсами.

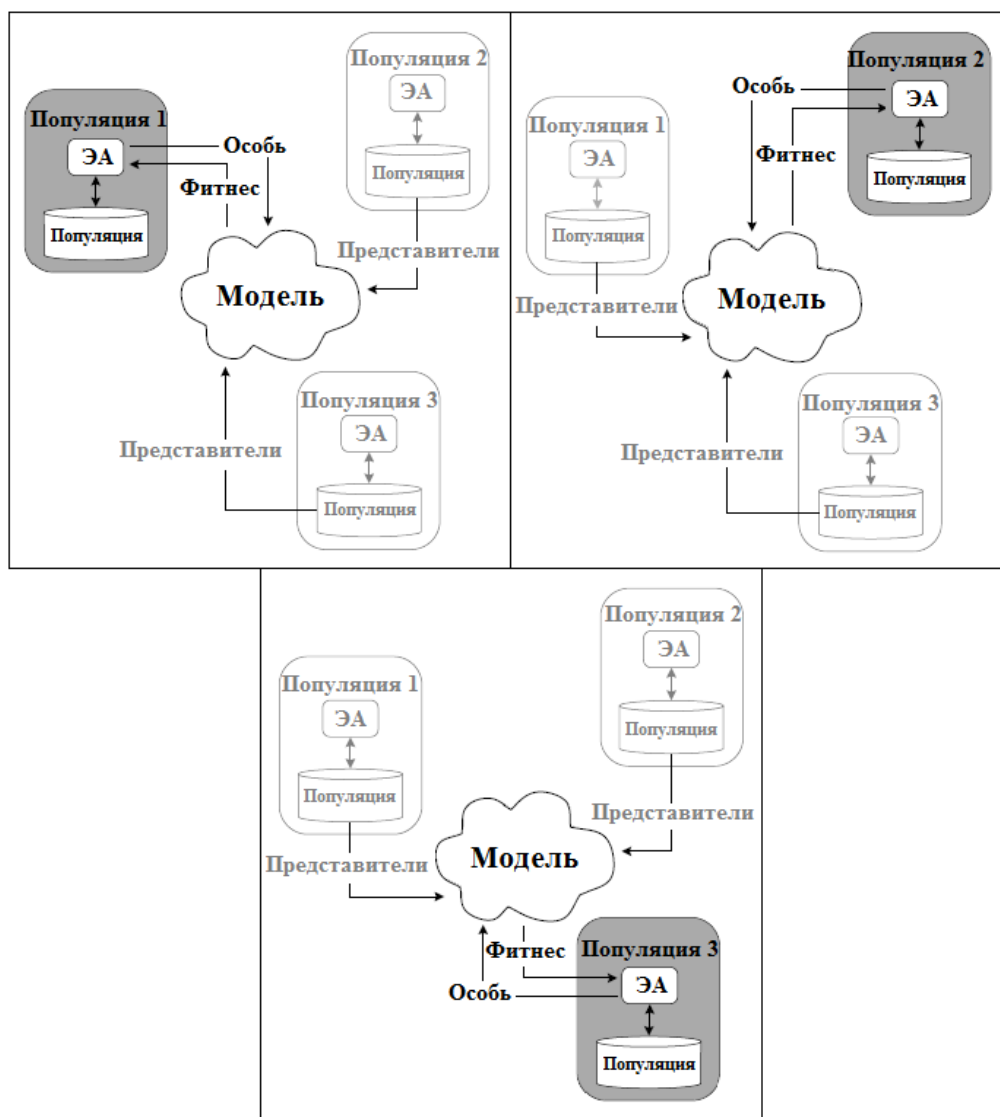
Коэволюционные ГА (КГА) заимствуют у природы механизмы кооперации и конкуренции с целью поддержания разнообразия популяций и, как следствие, получения лучших решений. Обычно в КГА ограничиваются использованием двух подпопуляций, но какого-либо ограничения на их число не существует. КГА разделяют на два основных класса: кооперативных и конкурирующих ГА, каждый из которых в свою очередь разбивается на ряд подклассов. Так конкурирующая коэволюция состоит из следующих подклассов: 1) конкуренция (виды соперничают друг с другом и успех одного вида является неудачей для другого); 2) аменсализм (неудача одного вида не оказывает влияния на другие виды, участвующие в коэволюции).

В кооперативной же эволюции выделяют: 1) мутуализм (виды сотрудничают с выгодой для себя); 2) комменсализм (выигрывает только один вид, а на остальные виды этот процесс не оказывает влияния); 3) хищничество (паразитизм) (только хищники (паразиты) выигрывают, в то время как остальным видам причиняется вред).

### **Многокритериальный метод обучения ИНС на основе кооперативной коэволюции.**

Базовая модель коэволюции на основе кооперации показана на рис. 6.1. Хотя на данном рисунке показаны три популяции, фактическое их число в экосистеме может быть иным. При таком подходе каждый вид эволюционирует в своей популяции и адаптируется к окружающей среде путем многократного применения ЭА. На рисунке показан этап оценки пригодности ЭА с точки зрения каждого из этих трех видов. Для реализаций такой модели может быть использована как последовательная схема оценки популяций, так и параллельная. Оценка приспособленности особей одного вида проводится с участием представителей и других видов. Существует большое количество способов выбора представителей популяций для сотрудничества. В некоторых случаях достаточно просто выбрать текущую лучшую особь от каждого вида в качестве представителя всей популяции. В других случаях такой подход может быть слишком “жадным” и альтернативные стратегии являются более предпочтительными. Так, например, особь от каждого вида может быть выбрана случайным образом или может быть применен более естественный подход, при котором представители выбираются недетерминированно на основе их пригодности.





*Рис. 6.1. Коэволюция на основе кооперации*

В работе [110] предложен несколько иной подход к применению механизма коэволюции к настройке ИНС по получивший название COVNET. COVNET является кооперативной коэволюционирующей моделью, то есть, несколько видов совместно коэволюционируют. Каждый вид при этом является подсетью и представляет собой частичное решение проблемы. Комбинация нескольких особей из разных видов составляет ИНС, которая должна быть применена к конкретной проблеме. Популяция подсетей, называемых нодулями, состоит из нескольких суб-популяций, которые эволюционируют независимо друг от друга. Каждая из таких суб-популяций представляет собой отдельный вид, а ключевым элементом предложенной модели является комбинация особей из этих различных суб-популяций, коэволюционирующих вместе.

Модуль является подсетью, образованной с помощью набора нейронов с свободными связями между ними, соединений этих нейронов с нейронами входного и выходного слоев и не имеет соединений с каким-либо нейроном, принадлежащем другому нодулю. Входные и выходные слои для всех нодулей являются общими, они являются входными и выходными слоями результирующей сети. Следует отметить, что генотип нодуля имеет взаимно-однозначное отображение в фенотип. ИНС при этом можно определить как совокупность нодулей, т.е. ИНС является сочетанием конечного числа нодулей, а выходом сети является сумма выходов всех входящих в нее нодулей. На практике для упрощения алгоритма обычно полагают, что все сети популяции должны иметь одинаковое фиксированное на протяжении всего процесса эволюции количество нодулей.

Процесс эволюции субкомпонент должен решать четыре основные задачи: разложение основной задачи на подзадачи, обеспечение взаимозависимости между субкомпонентами, оценку суб-популяций (присвоение коэффициентов доверия) и обеспечение разнообразия популяций.

Кооперативная коэволюция позволяет решить данные задачи естественным образом. Так как каждая популяция будет производить особи определенного вида, которые должны сотрудничать с целью получения вознаграждения высокими значениями пригодности, то нет необходимости в каких-либо априорных знаниях для разложения проблемы вручную. Взаимозависимость среди суб-компонент обусловлена тем, что фитнес каждой особи зависит от того, насколько хорошо она сотрудничает с членами других видов.

Разнообразие поддерживается в процессе эволюции благодаря тому, что каждый вид развивается без обмена генетическим материалом с другими видами. Это является важным аспектом данной коэволюционной модели. Обмен генетическим материалом между двумя различными видами (субпопуляциями), как правило, производит нежизнеспособное потомство. Кроме того, смешение генетического материала может привести к уменьшению разнообразия популяций.

Алгоритм настройки ИНС с помощью подхода COVNET показан на рис. 6.2.

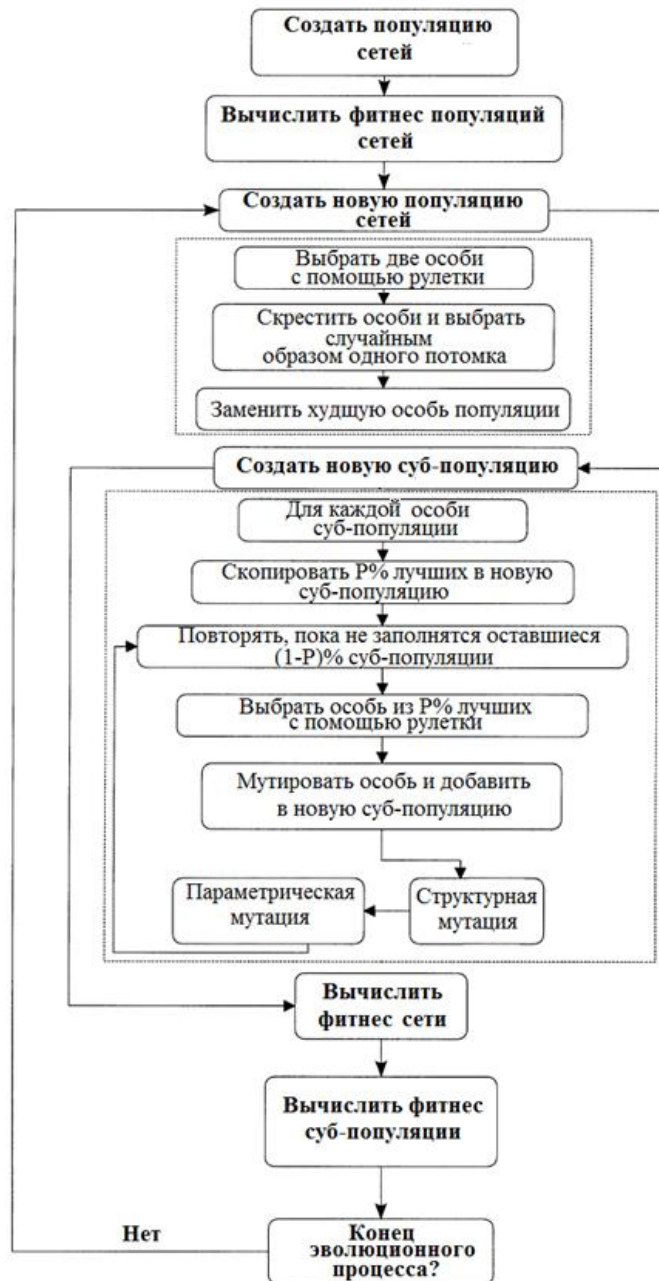


Рис. 6.2. Алгоритм настройки ИНС с помощью подхода COVNET

### Многокритериальный метод обучения ИНС на основе конкурентной коэволюции.

Рассмотрим алгоритм обучения ИНС на основе конкурентной коэволюции. В нем можно выделить следующие основные шаги.

1. Инициализация. Создается  $N$  популяций, состоящих из  $m$  особей (ИНС) каждая. Каждая популяция  $P_i = \{H_1, H_2, \dots, H_m\}$  инициализируется случайным образом. Каждая особь в популяции при этом получает свое уникальное описание, закодированное в хромосоме  $H_j = \{h_{1j}, h_{2j}, \dots, h_{Lj}\}$ , которая состоит из  $L$

генов, где  $h_{ij} \in [w_{\min} w_{\max}]$  – значение  $i$ -го гена  $j$ -ой хромосомы ( $w_{\min}$  – минимальное, и  $w_{\max}$  – максимальное допустимые значения соответственно). Следует отметить, что длина хромосомы зависит от размерности аппроксимируемой функции и максимально допустимого количества нейронов.

2. Первый шаг эволюции. Все  $N$  популяций обучаются на полном наборе входных сигналов.

3. Ранжирование. Все  $N \cdot m$  особей ранжируются (от 0 до  $N \cdot m - 1$ ) в зависимости от их фитнеса  $F_i$ ,  $i = 0, \dots, N \cdot m - 1$ , вычисленного во время обучения. Ранжирование популяций выполняется на основе фитнеса первых  $M$  особей после индивидуального ранжирования при  $M \geq m$ .  $M$  будет отлично от  $m$  только тогда, когда  $F_m = F_{m-1}$ , в этом случае при ранжировании необходимо учесть все особи, имеющие одинаковый фитнес  $F_{m-1} = F_m = \dots = F_{M-1}$ .

4. Присвоение коэффициентов доверия. Каждой из  $M$  особей присваивается коэффициент, вычисляемый следующим образом:  $\Gamma_i = e^{(M-l)/M}$ ,  $i = 0, \dots, m-1$ . Если  $k$  особей имеют одинаковый фитнес  $F_j, F_{j+1}, \dots, F_{j+k-1}$  с  $0 \leq j, j + k - 1 \leq M - 1$ , им будет присвоен одинаковый коэффициент доверия

$$\Gamma = \frac{1}{k} \left( \sum_{l=0}^{k-1} e^{(M-l)/M} \right).$$

Каждой из  $N$  популяций присваивается коэффициент доверия, эквивалентный сумме коэффициентов доверия, присвоенных ее особям. Следует отметить, что если популяция не имеет своих представителей среди  $M$  лучших особей, ей присваивается коэффициент доверия равный 0.

5. Проверка критерия исключения популяции. Если ранг популяции не изменялся в течение предыдущих  $D$  эпох, то популяция исключается на 6 шаге, в противном случае осуществляется переход к шагу 2. Для ускорения процесса обучения  $D$  уменьшается на 1 каждые  $P$  итераций (параметры  $D$  и  $P$  задаются на этапе инициализации алгоритма). Возможен также подход, при котором через каждые  $D$  шагов исключается популяция с наименьшим рейтингом независимо от того, менялся он на последних итерациях или нет.

6. Исключение популяции. Если выполняется критерий исключения популяции, то геномы особей исключаемой популяции распределяются между оставшимися  $N-1$  популяциями. После уменьшения  $N$  на единицу, алгоритм продолжает работу с шага 2, если  $N > 1$ , до тех пор, пока не останется лишь одна

популяция, состоящая из  $N \cdot m$  особей. Оставшаяся популяция обучается до выполнения условия останова.

### **Многокритериальный метод обучения ИНС на основе комбинированного коэволюционного подхода.**

Исследованный в работе [87] алгоритм CCC-GA (Competitive-cooperative coevolutionary – GA) объединяет в себе два рассмотренных выше коэволюционных метода. При этом, фаза кооперации на основе конкуренции начинается с  $N$  популяций, содержащих по  $m$  особей каждая, каждая популяция эволюционирует с помощью ГА с перекрывающимися популяциями и заканчивается с 1 популяцией, содержащей  $N \cdot m$ . Таким образом, менее приспособленные популяции постепенно поглощаются более приспособленными. При этом для каждой популяции применяется ГА с различными параметрами мутаций и вероятностями скрещивания. Алгоритм CCC-GA можно записать следующим образом:

1) Фаза конкуренции.

1.1) Инициализация

1.2) Первый шаг эволюции.

1.3) Ранжирование.

1.4) Присвоение коэффициентов доверия.

1.5) Проверка критерия исключения популяции.

1.6) Исключение популяции.

2) Фаза сотрудничества.

2.1) Интеграция победителя. Победивший в конкуренции вид кооперативно коэволюционирует с лучшими представителями предыдущих выигравших видов, пока СКО не уменьшается менее чем на  $\delta$  в  $I$  последовательных итерациях. В конце итерационного процесса геном с самой высокой пригодностью используется для замещения худшей особи популяции.

2.2) Переобучение. После того, как новая сеть добавляется в популяцию, предпринимается попытка переобучить сразу все виды, созданные до этого момента. Во время переобучения, каждый вид вносит свой вклад в общее решение. Виды переобучаются в случайном порядке, кроме новейших видов, которые всегда переобучаются последними. Переобучение осуществляется с использованием того же критерия останова, как в пункте 2.1. Если вид не может найти более подходящее решение в процессе переобучения, то предыдущее будет сохранено.

## Моделирование.

**Эксперимент 1.** Решалась задача многокритериальной идентификации нелинейного стационарного объекта

$$y(k) = 0.725 \sin \left( \frac{16u(k-1) + 8y(k-1)}{(3 + 4u^2(k-1) + 4y^2(k-1))} \right) + 0.2u(k-1) + 0.2y(k-1), \quad (7)$$

где  $u(k)$  - входной сигнал, представляющий собой стационарную случайную последовательность с равномерным законом распределения в интервале  $[-1, 1]$ , генерируемую датчиком случайных чисел.

Для решения данной задачи использовалась популяция РБС, состоящая из 250 особей (сетей). Максимально возможное количество нейронов в каждой сети было ограничено 15. Таким образом, каждая хромосома состояла из 106 генов. В скрытом слое сети были использованы гауссовские активационные функции. В качестве минимизируемых критериев были выбраны: время симуляции сети (всем сетям в популяции для вычисления приспособленности предъявлялась одинаковая обучающая выборка, состоящая из 2500 тестовых входных сигналов), значение фитнес-функции, которая имела вид

$$f_i(x_j) = \frac{1}{K} \sum_{j=1}^K \left( y_j^*(x_j) - \hat{y}_j(x_j) \right)^2, \quad (8)$$

где  $K=2500$  – размер выборки;

и количество настраиваемых параметров сети - критерий, характеризующий сложность модели.

Первые 100 шагов селекция особей для скрещивания производилась лишь на основе значений фитнес-функции (8). Такое предварительное обучение обусловлено тем, что на начальном этапе эволюции популяции не имеет смысла применение алгоритмов МО, так как ошибки всех моделей независимо от числа параметров являются достаточно большими.

На этапе селекции строился фронт Парето из которого затем выбирались особи для скрещивания и получения нового поколения.

Результирующий фронт Парето и его проекции после 1000 эпох обучения представлены на рис. 6.3. Квадратиком обозначена особь, для которой критерий Акаике является минимальным.

**Эксперимент 2.** Рассматривалась задача идентификации нелинейных многомерных объектов: четырехмерного  $F(x_1, x_2, x_3, x_4)$

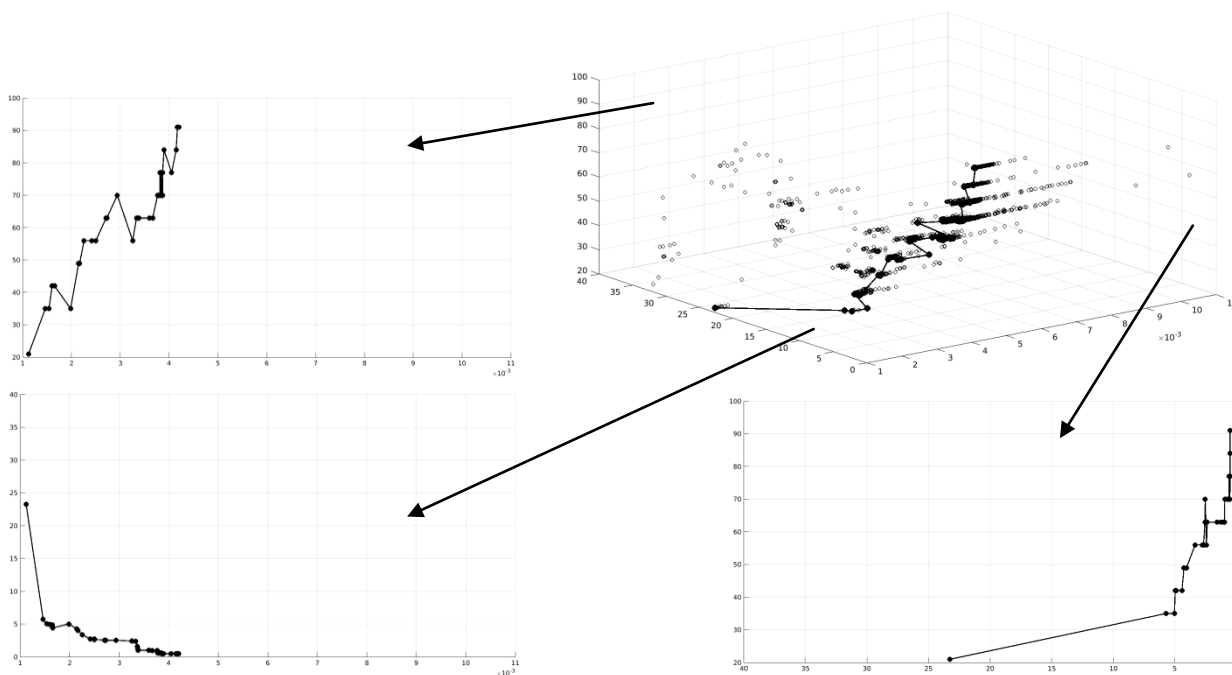
$$F(x_1, x_2, x_3, x_4) = x_1 + \sin(\pi x_1) \cdot \cos(\pi x_2) \cdot \sin(\pi x_3) \cdot \left( \sin(\pi x_4)^2 - 1 \right) \quad (9)$$

и шестимерного  $F(x_1, x_2, x_3, x_4, x_5, x_6)$

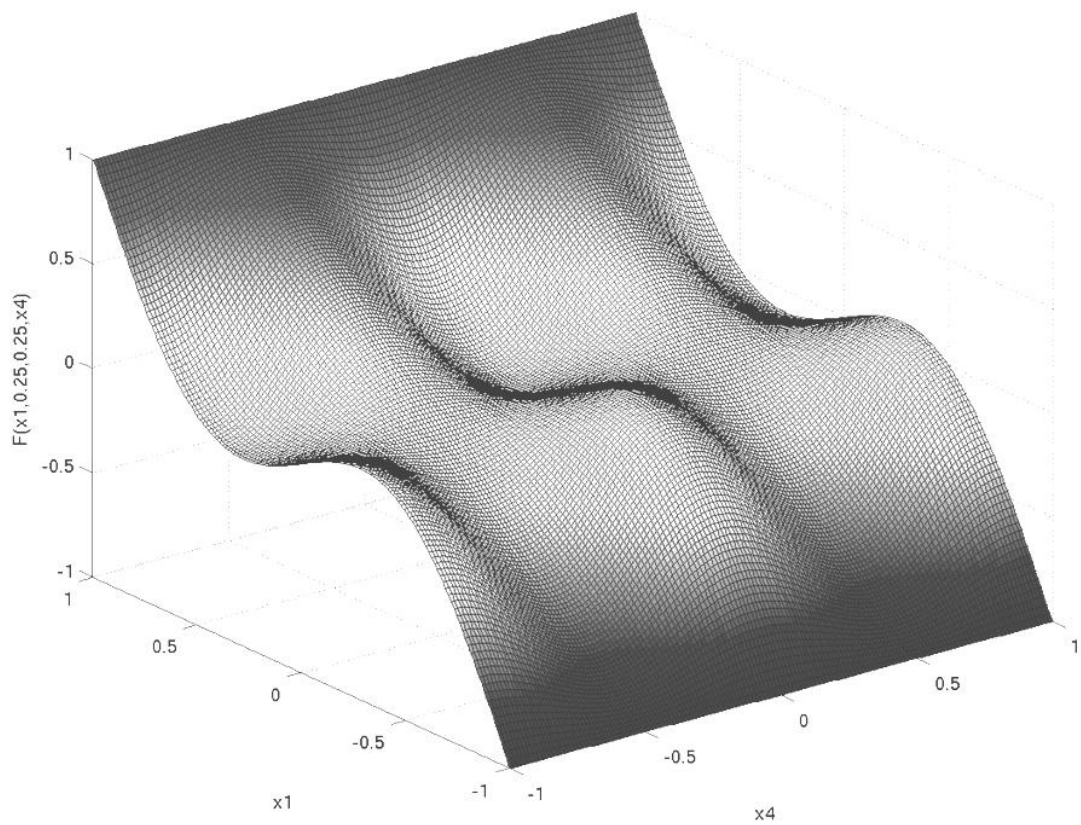
$$\begin{aligned}
 F(x_1, x_2, x_3, x_4) = & \sin(\pi x_1) \cdot \sin(\pi x_2) \cdot \sin(\pi x_3) + \\
 & + \sin(\pi x_4) \cdot \sin(\pi x_5) \cdot \sin(\pi x_6) + \\
 & + \exp(-(x_1^2 + x_2^2)) + \exp(-(x_3^2 + 6x_3x_4 + 10x_4^2))
 \end{aligned}
 \tag{10}$$

Сечение функции (1) при  $x_2 = x_3 = 0.25$  показано на рис. 6.4.

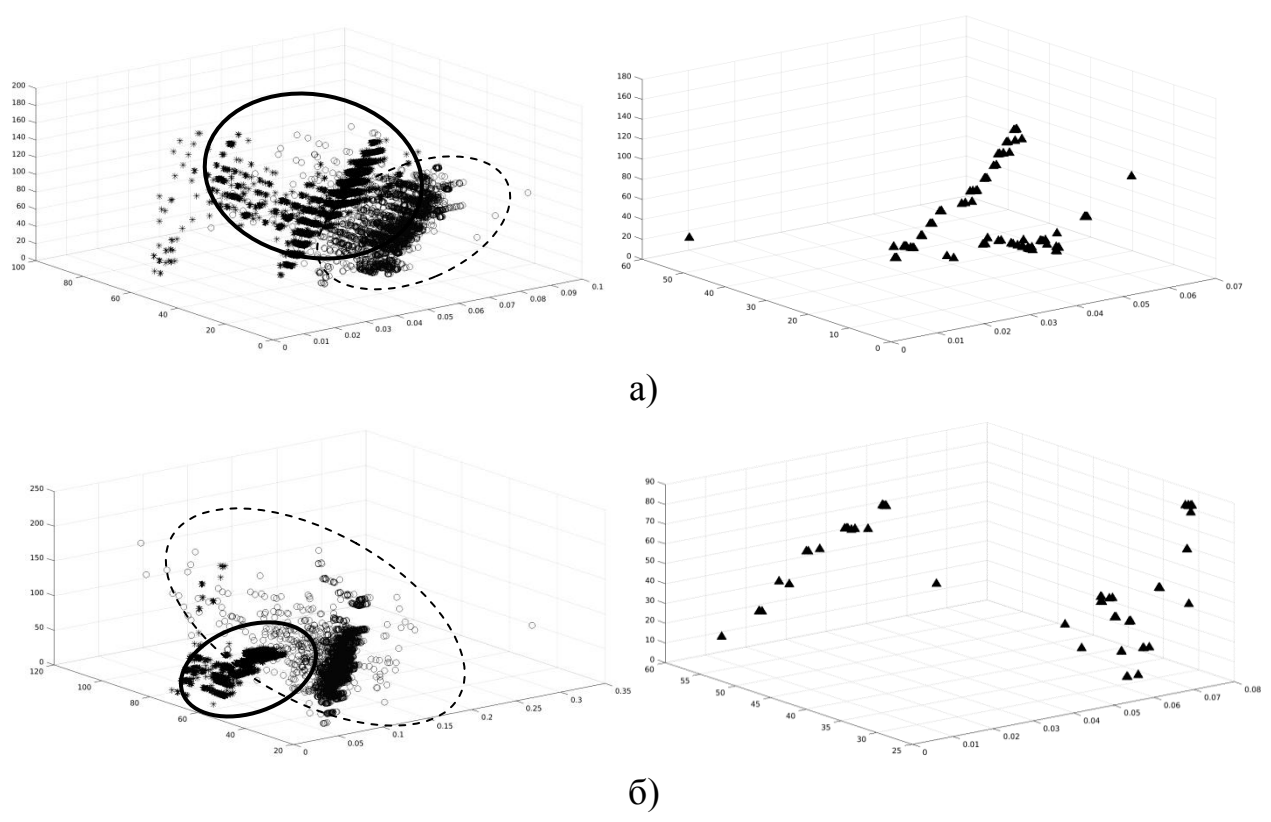
Данная задача решалась с использованием коэволюционного подхода на основе конкуренции. За получение права предоставления наилучшего решения задачи идентификации соревновались две популяции эволюционирующих ИНС: состоящих из РБС и МП. Обе популяции содержали по 150 особей и после каждой эпохи обучения по результатам вычисления значений фитнес-функции (8), измерения времени симуляции и определения количества настраиваемых параметров каждой особи, строился результирующий фронт Парето для обеих популяций. Все возможные решения и результирующий фронт Парето после 1000 эпох обучения представлены на рис. 6.5 (а, б). Решения, полученные с помощью МП, обозначены звездочками, а место их сосредоточения обведено сплошной жирной линией, решения же, полученные с помощью РБС показаны на рисунках кружками, а область их скопления обведена пунктирной линией. Фронт Парето строился общий для обеих популяций и из него выбиралось лучшее решение в соответствии с критерием Акайке. Лучшее решение могло принадлежать как популяции, состоящей из МП, так и из РБС.



**Рис. 6.3. Результирующий фронт Парето и его проекции**



*Рис. 6.4. Вид исходной функции  $F(x_1, 0.25, 0.25, x_4)$*



*Рис. 6.5. Возможные решения и результирующий фронт Парето*



Как видно из результатов моделирования, для 4-х мерного объекта популяция РБС дает лучшие результаты по всем минимизируемым критериям, а для 6-ти мерного объекта популяция МП дает лучшие решения по сложности модели и времени симуляции, несколько проигрывая, однако, по точности идентификации.

**Заключение.** Как свидетельствуют результаты приведенного исследования, применение коэволюционного подхода в сетях прямого распространения является достаточно эффективным инструментом, позволяющим, во-первых, решать более сложные задачи, во-вторых, получать более простые структуры ИНС за счет уменьшения количества нейронов, в-третьих, значительно повысить робастность получаемой при этом сети по сравнению с решениями, основанными на одиночной ИНС. Также следует отметить, что для выбора оптимального решения из получаемого фронта Парето следует использовать какой-либо робастный информационный критерий.

В заключение следует отметить, что при разработке коэволюционирующих систем возникает вопрос анализа их локальной и глобальной робастности. Обычно ограничиваются исследованием проблем глобальной робастности разрабатываемой системы, опуская влияние локальных возмущений, так как после получения окончательного решения все промежуточные популяции теряют свое значение. При разработке коэволюционирующей системы обучения ИНС разработчики обычно заинтересованы не в поиске лишь некоторой оптимальной популяции, а в поиске популяции, не только хорошо решающей поставленную задачу, но и устойчивой к отклонениям в поведении отдельных ее членов. Эта форма устойчивости является важной, поскольку на практике возможности членов популяции, и даже состав самой популяции, часто изменяются с течением времени. Таким образом, например, в рамках кооперативной коэволюции, робастность системы будет определяться возможностью входящих в нее суб-популяций осуществлять поиск таких решений, которые смогут эффективно сотрудничать с широким диапазоном решений, получаемых в других суб-популяциях.

## РОЗДІЛ 7

# МОДЕЛЬ ПЛАНИРОВАНИЯ ПАКЕТОВ ЗАДАНИЙ В РАСПРЕДЕЛЕННОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЕ

*Аннотация.* Рассмотрено обобщение модели и методов планирования пакетов заданий в распределенных вычислительных системах, базирующихся на решении задачи о наименьшем покрытии. Для обобщения модели использована нотация Грэхема, позволяющая учесть положения и принципы использования модели планирования, а также динамику процессов обработки и планирования пакетов заданий в РВС в рамках дискретно-событийной модели распределенных вычислений.

**Ключевые слова:** задача о наименьшем покрытии, распределенная вычислительная система, задание, пакетный режим, локальный ресурс, нотация Грэхема.

*Abstract.* A generalization of the model and batch mode scheduling methods in distributed computing systems based on solving the problem of the minimal cover is developed. To summarize the model used Graham notation, allows to take into account the provisions and principles of the use of the proposed scheduling models and dynamics processing, and scheduling batch jobs within discrete-event model of distributed computing.

**Keywords:** the minimal cover problem, a distributed computing system, a job, the batch mode, a local resource, Graham notation.

**Введение.** Одной из важнейших задач обработки сложноструктурированных и интерактивных одиночных заданий в распределенных вычислительных системах (РВС) является повышение эффективности процессов планирования заданий. Одним из наиболее распространенных режимов планирования является пакетный, модели и методы реализации которого рассмотрены в [54; 61; 60; 62], в частности, для уровня грид-сегмента предложена модель планирования на основе решения задачи о наименьшем покрытии (ЗНП).

Основной проблемой формирования процессов планирования в РВС является их формализация, использующая известные модели и методы, позволяющие учесть не только особенности модели, но и свести их к известным решениям и таким образом обобщить их на частные случаи.

Целью данного исследования является разработка формальной математической модели планирования пакетов заданий на основе решения ЗНП с использованием модели Грэхема на основе предложенных положений и принципов построения модели для пакетного планирования пакетов заданий в двухуровневой РВС.

## Разработка модели планирования пакетов заданий

В качестве базовой архитектуры для реализации модели процесса планирования заданий используется двухуровневая архитектура РВС: уровень грид-сегмента (планировщик грид-сегмента) и уровень локального ресурса, на котором осуществляется планирование заданий, обеспечивающих качество обслуживания пользователей средствами ЛСУР [62].

Для реализации модели планирования заданий на основе решения задачи линейного целочисленного программирования (задачи о наименьшем покрытии) разработаны следующие концептуальные положения и принципы.

### *Общие положения.*

1. Применение централизованной двухуровневой архитектуры РВС, использующей концепцию "планировщик грид-сегмента – локальная система управления ресурсами и планирования".

2. Использование централизованной организации процессов получения, хранения и доступа к базе данных состояния ресурсов и заданий.

3. Независимое планирование заданий на различных уровнях РВС.

4. Применение стратегий планирования заданий, направленных на достижение целей планирования на разных уровнях РВС.

### *Специальные положения.*

1. Использование режима пакетной обработки заданий, позволяющего обеспечить их эффективную обработку в условиях высокой эффективности их поступления в распределенную систему.

2. Соответствие математических моделей планирования заданий стратегиям планирования, выбираемых менеджерами виртуальных организаций (ВО) и системными администраторами вычислительных кластеров путем решения задач управления вычислениями на различных уровнях РВС.

3. Применение единого теоретического подхода для решения задач планирования в РВС, которые на уровне грид-сегмента обеспечивают максимизацию загрузки ресурсов и на уровне локального ресурса (вычислительного кластера) РВС – минимизацию времени выполнения заданий [62].

Предлагаемые этапы планирования заданий реализованы на основе таких принципов.

1. Принципы перепланирования.

1.1. В случае превышения количества заданий, спланированных в локальные очереди на ресурсы, некоторого порогового значения, определяемого производительностью ресурсов и задаваемого администратором

локального ресурса, задание помещается в локальную очередь на другой локальный ресурс, который на этой итерации планирования также вошел в наименьшее покрытие. Это позволяет динамически перепланировать задания без привлечения дополнительных механизмов их миграции между ресурсами и балансировки загрузки ресурсов.

1.2. Текущая итерация планирования реализуется путем решения ЗНП для неспланированных заданий на свободных ресурсах РВС. На каждой итерации планирования ЗНП может решаться многократно до тех пор, пока для выполнения заданий текущего пула имеются свободные ресурсы и/или множество неспланированных заданий не является пустым.

1.3. Если на текущей итерации планирования для выполнения задания пула свободных ресурсов нет, то это задание возвращается в пул для последующей обработки. Это позволяет спланировать его на локальные ресурсы, освободившиеся на следующей после текущей итерации планирования.

1.4. Неспланированные на текущей итерации планирования задания вместе с заданиями, загруженными из входной очереди в текущий момент времени, формируют пул для выполнения следующей итерации планирования.

## 2. Принципы периодичности и синхронизации планирования.

2.1. Планирование осуществляется итерационно в соответствии с периодом, который рассчитывается на основе априорной информации о длительности заданий и производительности локальных ресурсов РВС. Это позволяет реализовать *динамическое планирование* заданий в случае изменения параметров этих характеристик.

2.2. Синхронизация заключается в согласованности сроков обработки событий на основе информации о состоянии ресурсов и заданий в соответствии с периодом планирования. Сроки обработки событий на локальных ресурсах определяются соответствующими настройками в программном обеспечении локальных планировщиков и системах управления локальными ресурсами. Величины этих сроков определяются системными администраторами грид-сегментов (вычислительных кластеров): их величины не должны превышать период планирования и не быть меньше допустимых сроков проведения мониторинга состояния заданий и узлов РВС [62].

## 3. Принцип координации.

Сущность принципа заключается в принятии решений по выбору *стратегии планирования* заданий системными администраторами ВО, которая

определяется на основе статистических характеристик заданий, поступающих на обработку в РВС.

4. Принцип гарантированного соблюдения директивных сроков выполнения заданий.

С целью выбора множества локальных ресурсов РВС для планирования пакетов заданий предлагается метод построения матрицы соответствия, отличающийся использованием директивного срока выполнения задания и реализующий следующие этапы.

*Этап 1.* Формируется текущий пул заданий из входного потока заданий на основе следующих шагов.

*Шаг 1.* Задания входного потока сортируются в порядке неубывания (увеличения) их директивных сроков. В случае одинаковых директивных сроков выбирается задание с меньшим временем выполнения (длительностью).

*Шаг 2.* Отсортированные задания располагаются последовательно в списке заданий пула.

*Этап 2.* Для каждого задания текущего пула определяется ожидаемое время его решения (ожидаемое время его завершения) на локальных ресурсах РВС.

*Этап 3.* Для назначения заданию выбираются те локальные ресурсы РВС, на которых время его выполнения (завершения) меньше директивного срока выполнения.

*Этап 4.* Формируется матрица соответствия, в ячейках которой записывается значение “1”, если время выполнения (завершения) задания меньше директивного срока, и “0” – в противном случае.

Так как на каждой итерации планирования в моменты времени  $T_i$  на основе решения ЗНП эта задача решается несколько раз, пока не будут обслужены все задания, вошедшие в пул, то процесс формирования локальных очередей на ресурсы на различных тактах планирования продолжается до тех пор, пока не будут выполнены следующие условия:

все задания текущего пула обслужены;

при планировании заданий пула ресурсы, на которых эти задания могут быть решены, заняты другими заданиями или недоступны;

все задания глобальной очереди обслужены.

Планирование заданий, вошедших в пул, на локальные ресурсы для их выполнения осуществляется как результат решения ЗНП. При этом в качестве

исходных данных для ее решения используется матрица соответствия, метод формирования которой приведен ранее.

Параметры:

система – количество свободных ресурсов, производительность ресурсов, величина пакета на ресурс, количество неспланированных заданий, отношение количества неспланированных заданий к величине пула (процент возврата), количество решений ЗНП, приходящихся на одно планирование;

задание – временная трудоемкость задания, интенсивность входного потока.

Дальнейшая обработка заданий осуществляется путем их последовательного поступления из локальных очередей на обработку ЛСУР по мере освобождения соответствующего локального ресурса для решения заданий.

Параметры:

система – производительность ресурса, величина пакета (максимальная величина локальной очереди) заданий на ресурс;

задание – временная трудоемкость задания.

С учетом того, что каждое последующее событие процесса планирования задания в данной модели начинается непосредственно после завершения предыдущего и происходит в последовательности, которая определяется событиями: формирование пула, планирование и перепланирование, загрузка и освобождение ресурсов, следует, что эта модель относится к классу дискретно-событийных моделей [54; 61; 60; 62] и включающая следующие события.

Событие 1. Поступление заданий входной очереди в систему и формирование пула.

Событие 2. Формирование матрицы соответствия, распределение заданий пула на ресурсы системы на основе решения ЗНП и возврат неспланированных заданий обратно в пул.

Событие 3. Выполнение заданий на назначенных им ресурсах. После выполнения заданий на ресурсах осуществляется следующий такт планирования заданий, формируемых пул.

Событие 4. Определяется, является ли пустой входная очередь; если да, то работа планировщика завершается; в противном случае осуществляется переход к событию 1.

В рассматриваемой модели планирования заданий используются следующие компоненты [54; 60; 62]:

– глобальная (входная) очередь заданий (GQ), которая формируется по мере поступления заданий пользователей для их решения и характеризуется величиной  $L_{GQ}$  – количеством заданий, поступивших на их решение за время формирования очереди (или период планирования);

– промежуточный пул заданий, который включает выбираемые из глобальной очереди задания, а также непланируемые на предыдущем шаге планирования задания (за исключением стартового планирования), и имеет величину  $L_{POOL}$ ,  $L_{POOL} \leq L_{GQ}$ . Величина пула может изменяться и определяться, например, количеством ресурсов, их динамикой, доступностью, периодом планирования, интенсивностью входного потока заданий и т. д.;

– пакет заданий на ресурс  $R_k$  (локальная очередь,  $L_{GQ}$ ), формируемый как результат планирования заданий, входящих в пул, определяемый решением ЗНП, и имеющий величину  $L_{packet}$ ;

– ресурсы  $\{R_j\}$ ,  $j = \overline{1, n}$ , на которых будут решаться спланированные как результат решения ЗНП задания, вошедшие в пул.

Представим последовательность обработки заданий, находящихся во входной очереди, в виде этапов, на каждом из которых используется функция отображения (преобразования), а обобщенная модель планирования получена с использованием нотации Грэхема [107].

Этап 1. Из глобальной очереди GQ задания выбираются в соответствии с применяемой политикой (например, их приоритетами, порядком поступления в систему и т. д.) до тех пор, пока пул заданий  $L_{POOL}$  не будет заполнен. После этого для каждого задания из пула определяются подходящие для его выполнения ресурсы и далее путем решения ЗНП им назначаются те ресурсы, на которых эти задания могут быть решены. При этом создаются копии задания на каждый из подходящих для ее решения ресурсов и осуществляется резервирование (коаллокация) этих ресурсов до завершения планирования. После выбора ресурсов как результата решения ЗНП, зарезервированные ресурсы, за исключением выбранных (назначенных) заданиям пула, опять становятся доступными для их назначения.

Данный этап назначения заданий пула  $L_{POOL}(T_i)$  на ресурсы системы в виде пакетов заданий (локальных очередей) в момент планирования  $T_i$  описывается преобразованием:

$$Scheduling\_POOL: L_{POOL}(T_i) \rightarrow \left\langle \bigcup_{k=1}^{MC(T_i)} L_{Q_k} \cdot R_k, T_i \right\rangle, \quad (1)$$

где  $MC(T_i)$  – количество ресурсов системы, выбранных как результат решения ЗНП в момент планирования  $T_i$ ;  $LQ_k.R_k$  – пакет заданий (локальная очередь), сформированный на ресурс  $R_k$  в момент планирования  $T_i$ .

Если множество ресурсов  $R = \{R_j\}$ ,  $j = \overline{1, n_T}$ ,  $n_T \leq n$ , на которых в момент планирования  $T_i$  можно решить все задания глобальной очереди  $GQ(T_i)$ , то формирование локальных очередей на основе входной (глобальной) очереди можно представить в виде следующего выражения:

$$GQ(T_i).R = \bigcup_{j=1}^{n_T} LQ_j(T_i).R_j, \quad (2)$$

а величина пакета (локальной очереди)  $LQ_k(T_i).R_k$  на ресурс  $R_k$  определяется как:

$$\text{Task\_Allocation} : LQ_k(T_i).R_k \rightarrow R_k, R_k = \{R_{kl}\}, l = \overline{1, N_{cores}}.$$

После этого получаем распределение заданий локальной очереди на узлы  $R_{kl}$  ресурса  $R_k$ :

$$LQ_k(T_i).R_k = \bigcup_{l=1}^{N_{nodes}} R_{kl}, \quad (3)$$

где  $l = \overline{1, N_{nodes}}$ ,  $N_{nodes}$  – количество узлов (или, например, процессорных ядер  $N_{cores}$ ) ресурса  $R_k$ .

Локальная очередь заданий  $LQ_k(T_i).R_k$  на ресурс  $R_k$  и задания описывается следующими параметрами:

$LQ_k^{\min}.R_k$ ,  $LQ_k^{\max}.R_k$  – количеством заданий, минимальным и максимальным соответственно, которые могут быть размещены в локальной очереди на ресурс  $R_k$ ;

$t_k$  – временем решения одного задания на ресурсе  $R_k$ ;

$t_k^{\min}$  – минимальным временем решения задания из числа заданий, спланированных на ресурс  $R_k$ , для которого требуется  $R_{kl}$  узлов (процессорных ядер) ресурса  $R_k$

$t_k^{\max}$  – максимальным временем решения задания из числа заданий, спланированных на ресурс  $R_k$ , для которого требуется  $R_{kl}$  узлов (процессорных ядер) ресурса  $R_k$ ;

$R_{kl}$  – количеством узлов  $l$  (процессорных ядер,  $N_{cores}$ ) ресурса  $R_k$ , на которых может быть выполнено одно задание.



Этап 2. Локальные очереди на ресурсы  $R_k$  определяют общую загрузку их узлов (процессорных ядер,  $N_{cores}$ )  $WORKLOAD$ , а именно количество заданий и количество узлов ресурса  $R_k$ , необходимых для выполнения каждого задания локальной очереди на этот ресурс:

$$WORKLOAD : LPOOL \times \bigcup_{l=1}^{N_{nodes}} R_{kl} \rightarrow \langle T_{alloc}, R_k \rangle, \quad (4)$$

где  $T_{alloc}$  – задания пула, распределенные на узлы ресурса  $R_k$ .

В начальный (стартовый) момент планирования заданий входной очереди ( $T_0=0$ ) все ресурсы  $R_k$  РВС предполагаются доступными и свободными, то есть выполняется:

$$R_{available}(T_0) = \bigcup_{j=1}^n R_j(T_0).$$

Назовем полностью допустимым расписанием выполнения заданий такое, при котором все задания входной очереди в момент планирования  $T_0=0$  распределяются на ресурсы  $R_k$  системы:

$$GQ(T_0).R = \bigcup_{j=1}^n LQ_j(T_0).R_j. \quad (5)$$

В моменты планирования  $T_i$  количество *свободных и доступных ресурсов*  $R_{available}(T_i)$  определяется временем освобождения ресурсов или временем решения определенных типов заданий на ресурсах (определенных типах ресурсов), которое в общем случае может динамически изменяться в моменты планирования в соответствии с состоянием распределенной среды и определяется следующим образом:

$$R_{available}(T_{i+1}) = \bigcup_{j=1}^n R_j(T_i) \setminus \bigcup_{mc=1}^{MC(T_i)} R_{mc}(T_i), \quad (6)$$

где  $R_{available}(T_{i+1})$  – доступные ресурсы в момент времени  $T_{i+1}$ ;  $R_{mc}(T_i)$  – ресурсы, выбранные (назначенные) как результат решения ЗНП на уровне грид-сегмента из числа *свободных и доступных ресурсов* на момент планирования  $T_i$ ;  $mc = \overline{1, MC(T_i)}$  – мощность выбранного множества ресурсов из числа *свободных* на текущей итерации планирования  $T_i$ .

Этап 3. На следующей итерации планирования  $T_{i+1}$  оставшиеся непланируемыми задания входной очереди  $GQ(T_{i+1})$  формируют очереди на локальные ресурсы:

$$\bigcup_{mc=1}^{n_{mc}(T_{i+1})} LQ_k(T_{i+1}) \cdot R_{available}(T_{i+1}) = GQ(T_i) \setminus \bigcup_{mc=1}^{n_{mc}(T_i)} LQ_k(T_i) \cdot R_{available}(T_i), \quad (7)$$

где  $n_{mc}(T_i)$ ,  $n_{mc}(T_{i+1})$  – мощности множеств назначенных заданиям ресурсов системы в моменты планирования  $T_i$  и  $T_{i+1}$  соответственно.

Назовем частично допустимым расписанием выполнения заданий такое расписание, при котором множество спланированных в моменты  $T_i$  заданий (за исключением момента времени  $T_0=0$ ) включает в себя задания пула, спланированные в очереди на свободные и доступные локальные ресурсы  $R_{available\ k}(T_i)$ , определяемые выражением

$$LPOOL_{shed}(T_i) = \bigcup_{k=1}^{MC(T_i)} LQ_k(T_i) \cdot R_{available\ k}(T_i), \quad (8)$$

и множество неспланированных на текущей итерации планирования  $T_i$  заданий, возвращенных обратно в пул, определяемых следующим образом:

$$LPOOL_{unshed}(T_i) = LPOOL_{shed}(T_i) \setminus \bigcup_{k=1}^{MC(T_i)} LQ_k(T_i) \cdot R_{available\ k}(T_i).$$

Параметры:

система – количество доступных ресурсов; производительность ресурсов; гетерогенность ресурсов;

задание – интенсивность входного потока, формирующего глобальную очередь; гетерогенность; приоритет.

В качестве типа ресурса используется ресурс с параллельно работающими узлами – ресурс  $R_m$ , при этом предполагается, что этот ресурс не является возобновляемым: задание, которое было прервано при его выполнении, должно быть полностью перезапущено после наступления момента времени, когда ресурс становится доступным –  $nr - a$  [107].

Так как в предлагаемой модели планирования используются параметры заданий, влияющие на результаты планирования, в качестве этих параметров выбраны следующие:

$r_{ij}$  – время решения задания  $i$ -го задания на ресурсе  $R_j$ ;

$s_{ij}$  – время старта  $i$ -го задания на ресурсе  $R_j$ ;

$d_i$  – директивный срок выполнения  $i$ -го задания;

$r_{ij}$  – время готовности выполнения  $i$ -го задания на ресурсе  $R_j$ .

Если для эффективной работы локальных ресурсов грид-сегмента требуется обеспечить уровень качества обслуживания – выполнение

директивного срока выполнения заданий, то в качестве характеристики задания следует выбрать параметр *prec* (precedence constraints) [107], определяющий условие предшествования и следования заданий, и директивный срок выполнения задания. При этом множество доступных ресурсов на такте планирования  $T_{i+1}$  определится следующим образом:

$$R_{m_{available}}(T_{i+1}) = \bigcup_{j=1}^n R_{m_j}(T_i) \setminus \bigcup_{mc=1}^{MC} R_{m_{mc}}(T_i), \quad (9)$$

а формальная модель системы планирования на основе нотации Грэхема примет следующий вид:

$$R_{m_{available}}(T_i) | p_{ij}, r_{ij}(T_i), s_{ij}(T_i), d_{ij}, nr - a, prec | (C_{max}, R_{utiliz}), \quad (10)$$

где  $r_{ij}(T_i)$  – время готовности выполнения  $i$ -го задания на ресурсе  $R_j$  в момент планирования  $T_i$ ;

$s_{ij}(T_i)$  – время старта  $i$ -го задания на ресурсе  $R_j$  в момент планирования  $T_i$ ;

$C_{max}$  – максимальное время завершения выполнения заданий входной очереди;

$R_{utiliz}$  – коэффициент использования ресурсов системы.

**Заключение.** Разработанная математическая модель планирования пакетов заданий в двухуровневой РВС позволяет учесть особенности и специфику планирования на основе модели ЗНП и таким образом обобщить полученные результаты в виде модели Грэхема. Полученные результаты на основе предложенных положений и принципов реализации модели позволяют математически формализовать этапы построения модели планирования на основе решения ЗНП и динамический процесс планирования на основе приведенной дискретно-событийной модели планирования.

## РОЗДІЛ 8

# ЗАДАЧИ И АЛГОРИТМЫ УПРАВЛЕНИЯ МАРШРУТИЗАЦИЕЙ ПАКЕТОВ В СЕТЯХ

***Аннотация.** В работе анализируются алгоритмы сбора и обработки информации при решении задачи маршрутизации пакетов в корпоративных коммуникационных сетях обмена данными. Рассматриваются пути совершенствования алгоритмического обеспечения маршрутизации пакетов на основе использования распределенной стратегии управления процессом информационного обмена. Предлагаются алгоритмы, обеспечивающие маршрутизацию пакетов в сетях для различных видов соединений.*

***Ключевые слова:** цифровая сеть, маршрутизация, маршрутная информация, виртуальное соединение, информационный пакет, адаптивный алгоритм, протокол передачи данных, линия связи, топология сети, распределенная стратегия.*

***Abstract.** This paper analyzes the characteristics of the algorithms for collecting and processing information in solving the problem of routing packets in digital networks. The ways to improve the algorithmic routing software packages through the use of distributed control strategy process of information exchange are considered in the article. The algorithms which provide packet routing for networks of different types of compounds are offered in the work.*

***Keywords:** digital network, routing, routing information, virtual connection, information packet, adaptive algorithm, data transfer protocol, communication link, network topology, distributed strategy.*

**Введение и постановка задачи.** Широкое использование методов контроля и управления передачей информации ограничивается отсутствием простых и эффективных алгоритмов маршрутизации пакетов, которые позволяют значительно повысить быстродействие обработки данных не снижая достоверности приема сообщений и эффективности автоматизации управления обменом данными в сетях.

Основные проблемы любого протокола маршрутизации всегда связаны с нестабильностью маршрутов. В любой системе всегда существует хотя бы один нестабильно работающий элемент или элемент, выходящий из строя самый неподходящий момент. Отказы и сбои в системах передачи данных приводят к пропаданию маршрутов. Задача алгоритма маршрутизации в данном случае состоит в том, чтобы как можно скорее найти обходной маршрут взамен вышедшего из строя. Именно в такие моменты некоторые алгоритмы

маршрутизации проявляют себя не лучшим образом [36]. Поэтому, в условиях растущих требований к достоверности передачи сообщений и времени их доставки абонентам актуальной задачей является разработка эффективных алгоритмов маршрутизации пакетов в системах обмена данными, использующих разнообразные протоколы. Эффективное алгоритмическое обеспечение должно быстро адаптироваться к изменяющимся условиям. Для оперативного анализа состояния сети в алгоритмах адаптивной маршрутизации информационных пакетов сообщений в системе обмена данными должно быть предусмотрено сохранение контекста ошибок, и маршрутов пакетов сообщений во временной базе данных.

**Основная часть.** Управление передачей данными в сети с коммутацией пакетов может осуществляться между пользователем и узлом коммутации между входными и выходными узлами, между каждой парой узлов вдоль виртуального канала. При адаптивном управлении потоком пакетов между входными и выходными узлами могут использоваться два варианта контроля правильности прохождения пакетов по маршруту. Один вариант предполагает проверку правильности приема пакета проводить на исходном узле коммутации. При втором варианте проверка правильности прохождения по маршруту осуществляется на каждом узле. При обнаружении ошибки на любом узле пакет стирается. Этот вариант должен быть учтен при разработке алгоритма контроля передачи информации в сетях обмена данными с использованием адаптивной маршрутизации пакетов, а также возможность адаптивного управления потоком пакетов между каждой парой узлов вдоль виртуального канала [56].

На всех узлах коммутации, через которые проходит пакет, хранятся копии в буферных запоминающих устройствах до получения квитанций (при положительных квитанциях – подтверждении правильного приема пакета) или определенный интервал времени (при передаче негативных квитанций - переспрос неправильного принятого пакета). На канальном уровне в каждом узле производится проверка правильности принятого пакета [38]. Пакет будет потерян, если изменен адрес получателя или отправителя, переполнено буферное запоминающее устройство или ложно сформировалась положительная квитанция. Для полной идентификации пакета сообщения он должен иметь несколько информационных полей, которые приведены на рис. 8.1.

1	2	3	4	5
Маршрут пакета от абонента до абонента	Свертка пакета	№ пакета в сообщении	Идентификатор сообщения	Информационная часть пакета сообщения

*Рис. 8.1. Информационные поля пакета сообщения*

Для помощи решения задачи определения маршрута пакета, который передается в канал связи, к служебной информационной части пакета добавляется адрес, на которой должен пройти пакет. Этот адрес представляет собой последовательность узлов маршрутизации и записывается он в заготовочных части пакета. Поэтому поле № 1 пакета (“Маршрут пакета от абонента к абоненту”) в процессе прохождения в сети связи постоянно меняется в узлах маршрутизации (рис. 8.1). Все остальные поля не должны изменяться при прохождении пакета в сети.

Алгоритм приема и обработки сообщения может работать при передаче информации с использованием выделенного, виртуального или дейтаграммного каналов. Поэтому в процессе работы алгоритма могут возникнуть ситуации, когда пакеты поступают абоненту по порядку в соответствии с их отправки из источника сообщения, но возможно возникновение таких случаев, когда необходимо собирать сообщения из пакетов, полученных приемником в порядке не соответствующем их отправке. При этом сбой или ошибка на какой-либо из участков этого сложного и многоэтапного процесса существенно затрудняет восстановление информации в исходное состояние. Кроме того, если часть процесса обработки сообщения прошла успешно, а сбой или ошибка возникли на конечном этапе, то сохранив контекст удачно завершенных операций можно существенно сократить время при повторной обработке информации.

Оценка состояния направления связи производится по минимально необходимому числу каналов для передачи потока информации. Исправным считается такое направление, в котором число каналов заданного вида больше допустимого. Состояние сети связи считается нормальным, когда все направления исправны. Сеть считается неисправной (авария), если ее эффективность снизилась на величину, большую допустимой.

Различают несколько предельных значений состояния каналов связи. Нормальное состояние соответствует интенсивности потока в канале не более 0,7 от максимально возможной. Состояние предупреждения перегрузки соответствует интенсивности потока 0,7 ... 0,8. Сигнал о перегрузке производится при интенсивности потока более 0,8 [39]. Узлы коммутации обмениваются между собой информацией об интенсивности потока. Возможны различные варианты принятия решений по этой информации. Например, если канал находится в состоянии предупреждения перегрузки, то принятый пакет ставится в очередь к этому каналу, а отправителю выдается сигнал о прекращении передачи пакетов. Если же в канале перегрузка, то принятый пакет стирается [56].

Результаты измерений интенсивности потока передаются всем центрам коммутации для внесения изменений в маршрутные таблицы. Для всех пар абонентов определяются основные маршруты. Если нагрузка в одном из пакетов превышает нормальную, то используются обходные маршруты. Если и эти маршруты перегружены, то прием пакетов от абонентов прекращается.

Измерение нагрузки производится с целью прогнозирования, планирования, накопление статистических данных.

Система измерения на узлах коммутации должна определить:

распределение потоков нагрузки и качество обслуживания на всех обслуживаемых устройствах узла;

место возникновения перегрузки, то есть устройство, через которое возник повышенный процент отказов;

необходимое число обслуживаемых устройств для обеспечения требуемого качества обслуживания, кроме того, система должна вести контроль за различными факторами, которые приводят к перегрузке.

Измеряется только интенсивность “обслуженной” нагрузки. Интенсивности входящей нагрузки и потери пакетов определяются через входной поток.

Отличительной чертой узлов коммутации пакетов является их гибкость относительно отказов элементов. Отказ одной или нескольких групп элементов (процессора, модуля памяти и др.), который не ухудшает характеристики узла, не извлекает этот узел из строя. Для выявления неисправностей периодически проводится программное тестирование. При определении отказавших элементов, предусмотрена аппаратная избыточность и осуществляется изменение конфигурации комплекса, в котором этот элемент отказал, при этом он выводится из контура взаимодействия и заменяется резервным [91]. После ремонта такой элемент снова вводится в комплекс.

Измерение времени задержки пакета на узле можно провести путем фиксирования момента его поступления и момента передачи первого бита в канал. Новое значение задержки пакетов может передаваться узлом только тогда, когда оно отличается от предыдущих на величину порога.

Корректирующий пакет посылается всем соседним узлам. При получении этого пакета в узле производится проверка наличия аналогичного пакета. Если такой пакет поступил раньше, то он уничтожается, в противном случае передается дальше всем соседним узлам, включая и тот, от которого был получен. При этом возвращающийся пакет, играет роль квитанции. Если такая квитанция не будет получена в течение тайм-аута, то корректирующий пакет передается повторно.

При решении задачи маршрутизации наибольшее распространение получила распределенная стратегия [56], которая позволяет эффективно контролировать состояние сети, а также оперативно рассчитывать и изменять маршруты передачи пакетов.

Структурная схема алгоритма контроля передачи информации в сетях обмена данными с использованием распределенной стратегии маршрутизации для случая проверки правильности информации в пакетах приведена на рис. 8.2.

В структурной схеме алгоритма (рис. 8.2) не рассматриваются вопросы допуска пакетов в сеть, выбора методов контроля передачи информации, использование протоколов обмена данными. Использование распределенной стратегии маршрутизации предполагает, что основная нагрузка по определению маршрутов несут центры коммутации. В комплекс решаемых ими задач входит опрос подчиненных узлов сети и сбор маршрутной информации о техническом состоянии каналов связи с целью формирования маршрутных таблиц. В блоке 1 «Формирование маршрутной информации в центре коммутации» (рис. 8.2) предполагается решение этого комплекса задач независимо от вида канала связи.

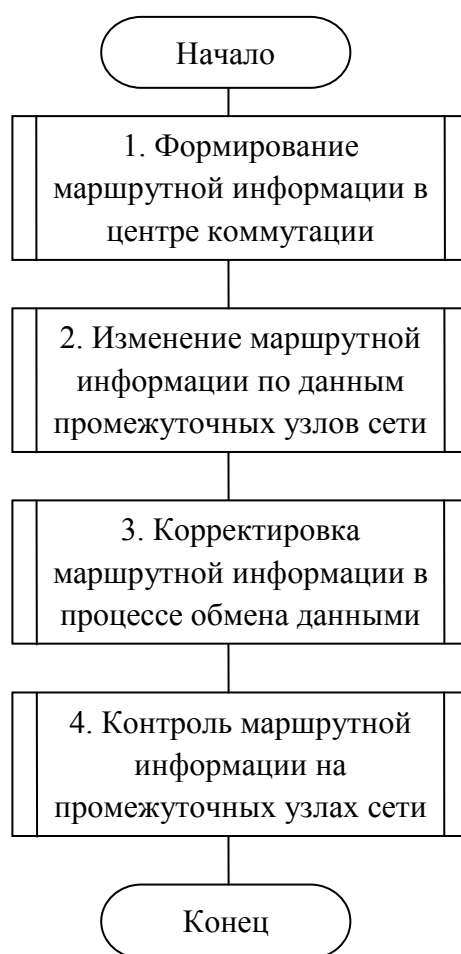
В процессе обмена данными в отдельных каналах связи возможно возникновение перегрузки или аварии. Контроль и выявление таких ситуаций осуществляют промежуточные узлы сети связи. Для того, чтобы центрам коммутации были известны подобные изменения в структурной схеме алгоритма предусмотрен блок № 2 «Изменение маршрутной информации по данным промежуточных узлов сети» (рис. 8.2).

Маршрутная информация в пакете, посланном от абонента источника, может быть изменена в процессе его прохождения к абоненту адресату. Для обеспечения возможности учета таких изменений в структурной схеме



алгоритма используется блок № 3 “Корректировка маршрутной информации в процессе обмена данными” (рис. 8.2).

Проверка правильности следования по маршруту информационных пакетов может осуществляться на каждом узле. При обнаружении ошибки на любом узле пакет стирается. При этом надо вести статистический учет таких маршрутов и каналов связи, на которых часто возникают ошибки. Для определения этой информации и хранение в центрах коммутации и на центральном узле управления в структурной схеме алгоритма формирования маршрутной информации используется блок №4 “Контроль маршрутной информации на промежуточных узлах сети” (рис. 8.2).



**Рис. 8.2. Структурная схема алгоритма формирования маршрутной информации**

Рассмотрим алгоритм формирования маршрутной информации в центре коммутации. При формировании маршрутной информации в центре коммутации необходимо провести анализ ресурсов сети путем сбора информации с узлов.

Сбор информации о состоянии сети осуществляется путем передачи на необходимый узел корректирующего пакета. По этому пакету контролируемый

узел посылает ответный пакет, включающий кроме адреса контролирующего узла всю необходимую для оценки состояния узла информацию. После получения информации контролирующий узел выполняет необходимые расчеты для корректировки маршрутных таблиц посылает результаты этих расчетов на управляемый узел. Поскольку информация для реализации маршрутов является наиболее важной, получение этой информации должно квитироваться. При передаче корректирующего пакета он может быть потерян, принятый с искажением адреса абонента или вызывает, принятые правильно и опознаны [56].

Если корректирующий пакет будет потерян или принят с искажением адреса вызывающего абонента, то возникает необходимость повторения передачи этого пакета через определенный временной интервал, который влияет только на время обмена информацией. На расход ресурсов сети (пропускную способность) это время не оказывает влияния. При правильном приеме корректирующего пакета и приеме его другим адресатом будет выслан в ответ пакет. Этот пакет может быть потерян, пойти не тому адресату, прийти правильно, или с искажением информации о состоянии узла коммутации. В случае правильного приема корректирующего и пакетов ответа, у абонента проводятся расчеты маршрутов и проводится подготовка для передачи управляющих пакетов. Если абонент занят, то вызов будет повторен через интервал времени. При наличии ошибок в пакете ответа также решается задача маршрутизации. Однако она будет решена не оптимально, в результате будет нанесен ущерб сети. Если же корректирующий пакет будет принят другим абонентом (искажена адрес абонента), при условии, что он свободен, будет послан в ответный пакет, при потере которого (либо искажении адреса) через определенный интервал корректирующий пакет будет повторен снова. Если в этих условиях в ответный пакет принят правильно, то за счет реализации процесса определения ошибки сети будет нанесен ущерб. После такого определения корректирующий пакет будет вновь повторен.

Структурная схема алгоритма формирования маршрутной информации в центре коммутации приведена на рис. 8.3.

В блоке № 1 “Запрос центра коммутации на узел сети” предполагается запрос узла о его состоянии. После этого в блоке № 2 определяется какого вида квитирование используется в протоколе обмена данными.

В блоках № 3, 9 выполняется контроль времени положительного или отрицательного квитирования. В блоках № 4, 10 принимается решение о получении положительной квитанции (блок № 4) или необходимости выполнения переспроса пакета из опрашиваемого узла (блок № 10). Результат принятого

решения зависит от используемого протокола обмена данными – с положительными или отрицательными квитанциями. Блоки № 5, 6 выполняют ожидание подтверждений, в соответствии с применяемым протоколом с положительными или отрицательными квитанциями.

В блоках № 7, 8, 14 (рис. 8.3) выполняется проверка правильности передачи ответа узла. Это действие может осуществляться на основе помехоустойчивого кодирования. Результатом такой проверки является установление факта наличия ошибки в адресной и информационной части ответа, а также возможности или невозможности ее устранения. В случае отсутствия ошибок в информационном пакете (блок № 14, рис. 8.3) управление передается блоку № 15, а при наличии ошибки управление передается блоку № 12. Повторение запроса узла, при возникновении ошибок, выполняется с помощью блоков № 12, 1. При этом количество таких повторений ограничивается. Если количество повторений запроса узла центром коммутации превышает заданную, то принимается решение что линия связи находится в аварийном состоянии (блок № 13, рис. 8.3). Информация о состоянии узла и линии связи должна храниться в базе данных центра коммутации (блок № 15, рис. 8.3).

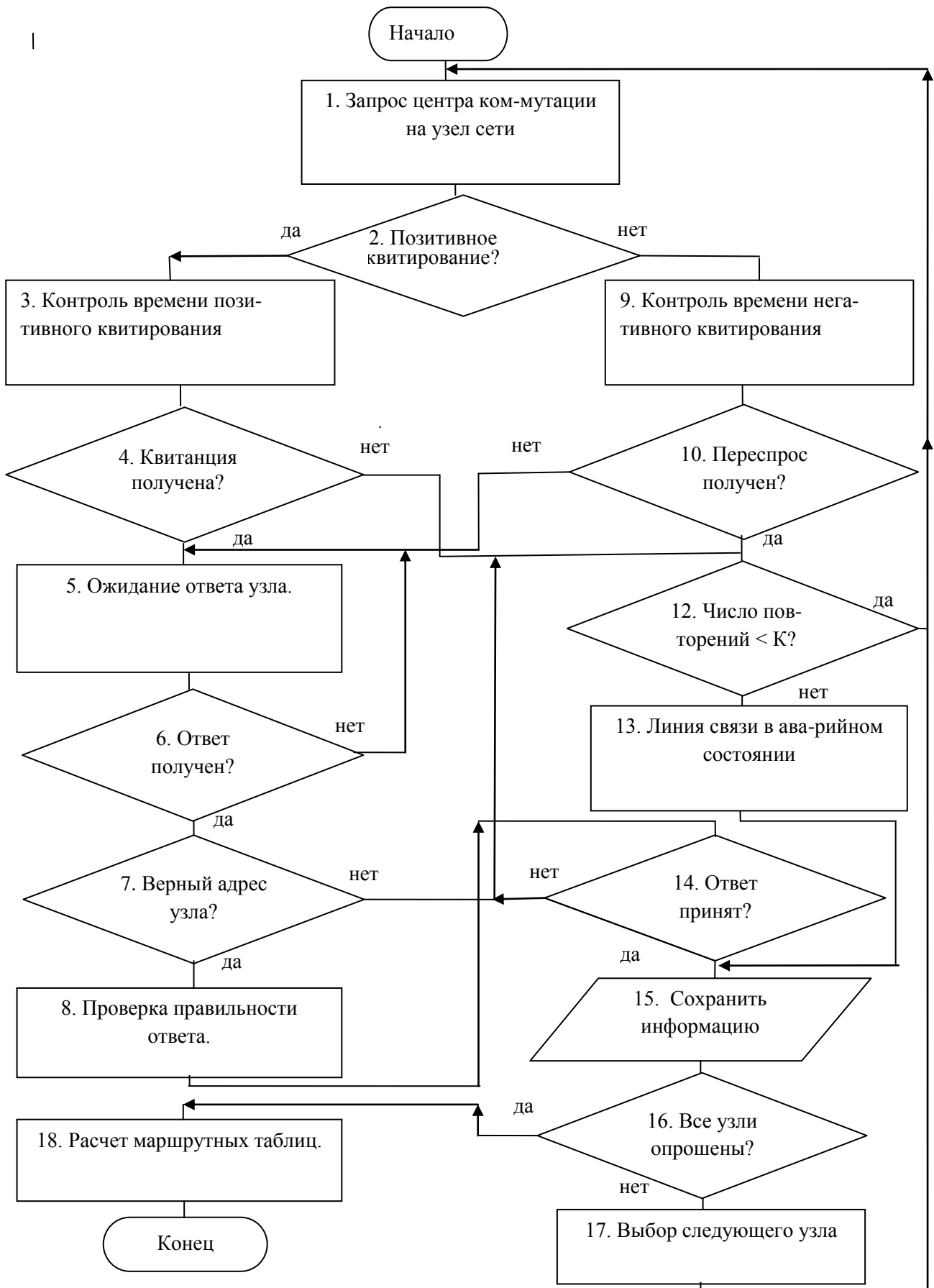
Если все узлы опрошенные (блоки № 16, 17, рис. 8.3), то выполняется расчет маршрутных таблиц (блок № 18, рис. 8.3), если нет, то выполняется опрос следующего узла.

Рассмотрим алгоритм изменения маршрутной информации по данным промежуточных узлов сети.

В процессе обмена данными в отдельных каналах связи возможно возникновение перегрузки или аварии. Контроль и выявление таких ситуаций осуществляют промежуточные узлы сети связи.

Для того, чтобы центрам коммутации были известны подобные изменения предусмотрен алгоритм изменения маршрутной информации по данным промежуточных узлов сети. Структурная схема такого алгоритма приведена на рис. 8.4. Инициатором передачи информации и организации связи в этом алгоритме есть промежуточный узел сети, который контролирует свое состояние и состояние линий связи в пределах своей ответственности. Узел коммутации принимает данные об изменениях состояния соответствующего узла сети или ее линий и обязательно выполняет расчет маршрутных таблиц.

Алгоритм изменения маршрутной информации по данным промежуточных узлов сети предусматривает прием обновляющей информации центром коммутации (блок № 1, рис. 8.4), а также проверку правильности передачи этой информации (блок № 2, рис. 8.4).



**Рис. 8.3. Структурная схема алгоритма формирования маршрутной информации в центре коммутации**

Если информационный пакет о состоянии линии или узла связи принят (блок № 3, рис. 8.4), то обязательно посылается подтверждение этого действия (блок № 4, рис. 8.4). В том случае, если пакет не принимается, например, при наличии неисправленных ошибок, то необходимо посылать дополнительный запрос на узел, который инициировал связь (блок № 10, рис. 8.4).

Структурная схема алгоритма существенно зависит от протокола, который выбран для обмена данными в этом случае. Если связь инициируется промежуточным узлом без запроса на его установку, то центру коммутации обязательно необходимо проверить правильность доставки своего подтверждения к абоненту. Для выполнения этого действия используются блоки № 5, 6 (рис. 8.4). Если подтверждения не принято, опять выполняется блок №10 (рис. 8.4) для посылания дополнительного запроса узлу сети. Количество таких дополнительных запросов не может превышать заранее заданного. Проверка этого процесса выполняется в блоке № 11. Если количество дополнительных запросов превышает максимально возможное значение, то принимается решение об ошибочности данного маршрута и выполняется сохранение его контекста (блок № 10, рис. 8.4). Если обновляющий пакет принят полностью и получено подтверждение его приема и передачи, выполняется сохранение контекста обновляющего пакета в базе данных коммуникационного узла (блок № 7, рис. 8.4). После сохранения данных обязательно выполняется расчет маршрутных таблиц в центре коммутации с учетом обновленной информации (блок № 8, рис. 8.4), которая получена из узла сети или определена центром в процессе обмена данными. При завершении алгоритма информация об изменениях в маршрутах необходимо передать на центральный узел (блок № 9, рис. 8.4).

Рассмотрим алгоритм корректировки маршрутной информации в процессе обмена данными. В наиболее реальном варианте маршрутизации узел, выдает пакет в канал, формирует адрес, по которому должен пройти пакет. Адрес представляет собой последовательность узлов и записывается в заголовке пакета.

Когда узел получает новый пакет, он сверяет его дальнейший адрес со своим маршрутом, вычисленным на основании информации, имеющейся в каждом узле и выбранным из таблицы минимизации маршрутов. При совпадении адреса с маршрутом происходит дальнейшая передача пакета другому узлу или получателю. В случае выхода из строя линии, по которой должен пройти пакет, смежные узлы фиксируют данное событие, внося изменения в таблице смежности.



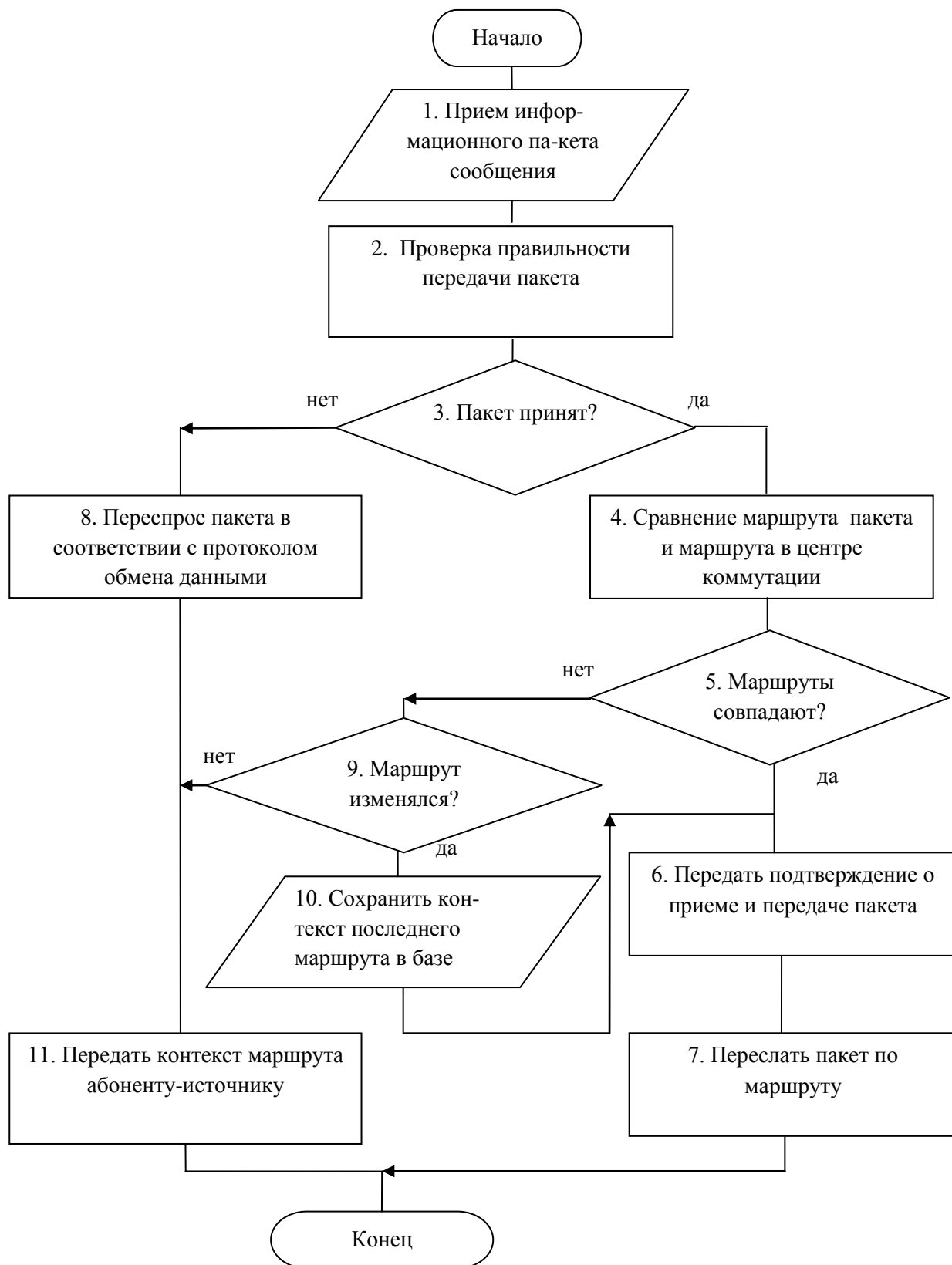
**Рис. 8.4. Структурная схема алгоритма изменения маршрутной информации по данным промежуточных узлов сети**

На основании данных изменений проводится корректировка минимальных маршрутов. Так осуществляется фиксация изменения топологии сети и корректировки маршрутов. Узел, на котором зафиксировано изменение в топологии сети, может принимать функции управления передачей пакетов на себя. При этом данный пакет не сбрасывается, не отсылается оповещение отправителю о формировании нового маршрута, а происходит формирование маршрута в данном узле, что позволит экономить общее время пребывания пакетов в сети. Пакет проходит по новому участку сети, дойдя до узла смежного с линией, вышедшей из строя, он продолжит свой путь по адресу пока не достигнет абонента.

Маршрутная информация в пакете, посланном от абонента источника, может быть изменена в процессе его прохождения к абоненту адресату. Для обеспечения возможности учета таких изменений используется структурная схема корректировки маршрутной информации в процессе обмена данными (рис. 8.5).

При выполнении алгоритма корректировки маршрутной информации в процессе обмена данными информационные пакеты сообщений принимаются и проверяются в блоках № 1, 2 (рис. 8.5). Проверка правильности передачи информационных пакетов может осуществляться на основе помехоустойчивого кодирования. Результатом такой проверки является установление факта наличия ошибки в адресной и информационной части пакетов, возможности или невозможности ее устранения. В случае отсутствия ошибок в информационном пакете принимается решение о приеме пакета (блок № 3, рис. 8.5) и управление передается блоку № 4, а в присутствии неисправимых ошибок – блоку № 8 для выполнения переспроса пакета в соответствии протоколом обмена данными. В блоках № 4, 5 (рис. 8.5) выполняется сравнение маршрута, находящимся в информационном пакете с маршрутом, который имеет центр коммутации. Если маршруты одинаковы, то необходимо передать подтверждение о приеме пакета и передачу его следующему центру коммутации или промежуточному узлу (блоки № 6, 7, рис. 8.5). Если маршруты не совпадают, то в блоке № 9 проверяется факт изменения маршрута за время с момента его определения в передающемся пакете. Если изменения были, то сохраняется контекст последнего определенного маршрута (блок № 10, рис. 8.5) и выполняются блоки № 6, 7.

Рассмотрим алгоритм контроля передачи информации на промежуточных узлах сети.



**Рис. 8.5. Структурная схема алгоритма корректировки маршрутной информации в процессе обмена данными**

Проверка правильности следования по маршруту информационных пакетов может осуществляться на каждом узле. При обнаружении ошибки на

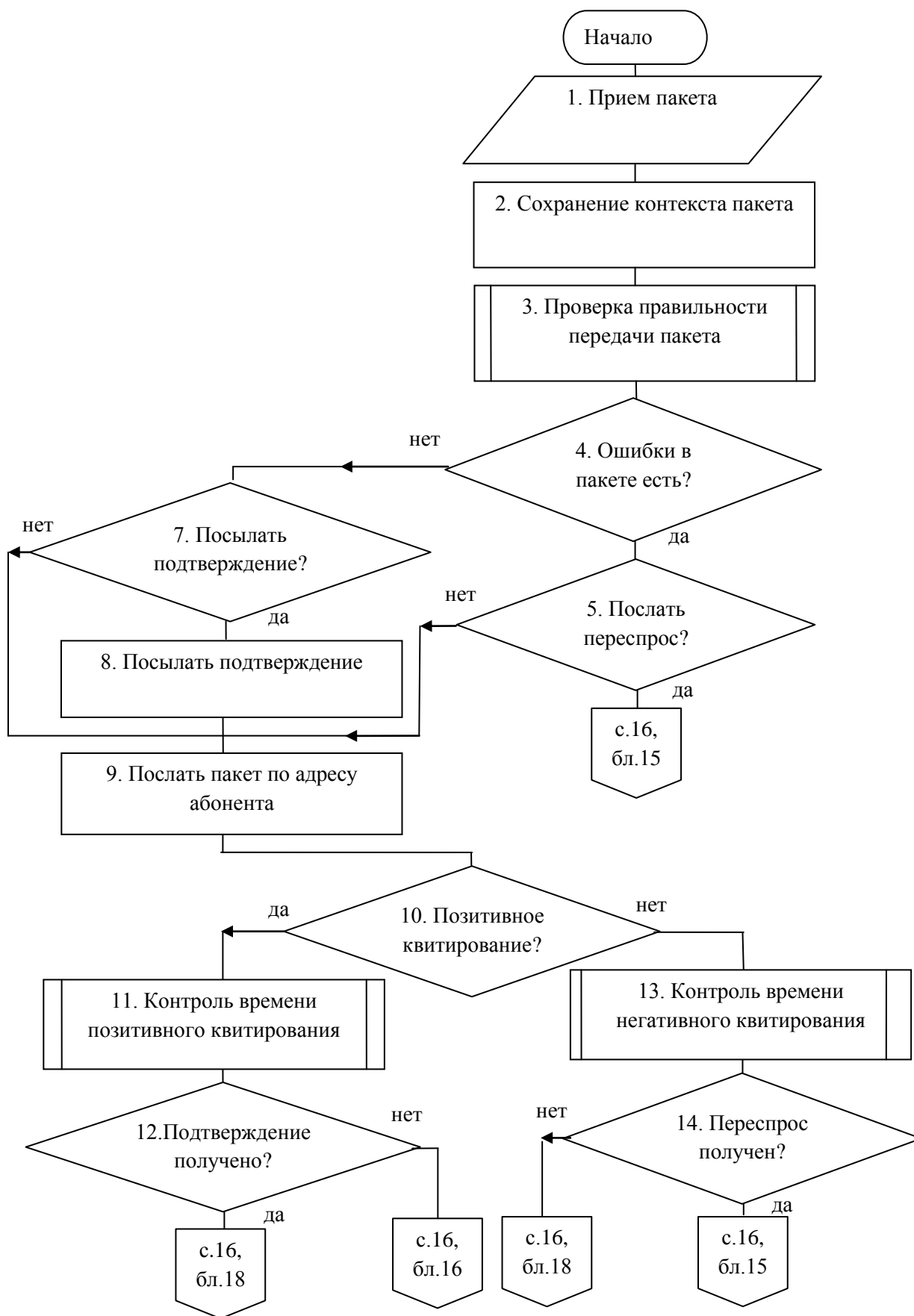


любом узле пакет стирается. Этот вариант контроля и управления передачи информации следует учитывать при разработке алгоритма контроля обмена данными в сетях с использованием протокола окно, а также возможность адаптивного управления потоком пакетов между каждой парой узлов вдоль виртуального или дейтаграммного каналов. Структурная схема алгоритма контроля передачи информации на промежуточных узлах сети при обмене данными с использованием таких протоколов приведена на рис. 8.5–8.6.

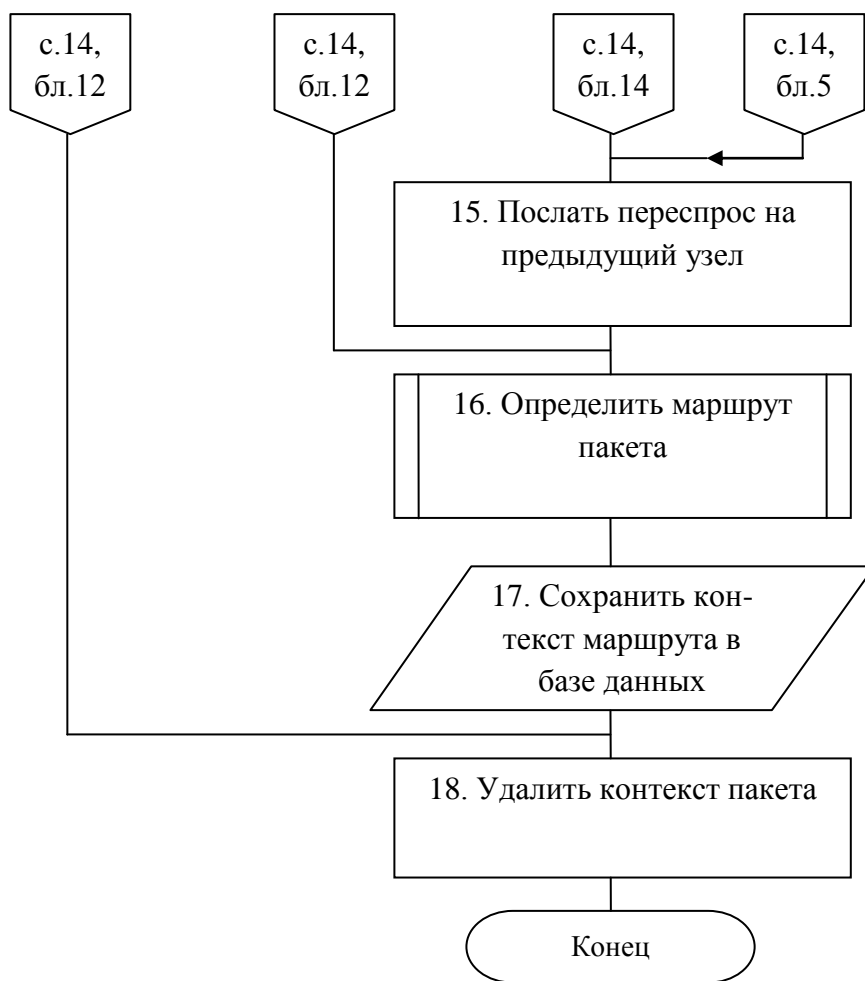
В блоке № 1 “Принятие пакета” предполагается выборка пакета из буферного запоминающего устройства абонента. После этого в блоке № 2 выполняется временное хранение контекста пакета в базе данных (рис. 8.6). В блоке № 3 выполняется проверка правильности передачи пакета. Это действие может осуществляться на основе помехоустойчивого кодирования. Результатом такой проверки является установление факта наличия ошибки, возможности или невозможности ее устранения. В случае отсутствия ошибок в информационном пакете (блок № 4, рис. 8.6) управление передается блоку № 7, а при наличии ошибки – блоку № 5. В блоках № 4, 7 принимается решение отсылать подтверждение о правильности приема пакета (положительную квитанцию блок № 7) или посылать переспрос пакета (блок № 5). Результат принятого решения зависит от используемого протокола обмена данными – с положительными или отрицательными квитанциями.

Блок №8 выполняет отсылку такого подтверждения, если применяется протокол с положительными квитанциями. Если используется протокол с негативными квитанциями, то выполняется блок № 15, в котором осуществляется переспрос пакета на предыдущий узел.

В блоке № 9 информационный пакет передается следующему узлу в соответствии с определенным маршрута. После чего текущий узел коммутации ждет прихода квитанции о результате доставки отправленного пакета, при этом контролируя время квитирования (блоки № 10, 11, 13). Если в течение определенного времени получено подтверждение о правильном приеме пакета, управление передается блоку № 17 и контекст пакета стирается из базы данных. Если подтверждения не получено, то принимается решение о возникновении ошибки или потере пакета, и определяется маршрут, на котором возникли ошибки (блок № 16, рис. 8.7), а информация о нем сохраняется в базе данных (блок № 17, рис. 8.7). При отрицательном квитировании и отсутствия переспроса (блоки № 12, 13, рис. 8.7) принимается решение о правильном приеме пакета, контекст пакета стирается и алгоритм завершается.



**Рис. 8.6. Структурная схема алгоритма контроля передачи информации на промежуточных узлах сети**



*Рис. 8. 7. Структурная схема алгоритма контроля передачи информации на промежуточных узлах сети (продолжение)*

В случае получения переспроса (блок № 14) выполняются блоки № 14–18, действие которых рассмотрена выше. Таким образом, алгоритм позволяет контролировать передачу информации на промежуточных узлах сети обмена данными.

**Заключение.** Постоянное повышение требований к системам и алгоритмам контроля обменом данными, а также методам внедрения программных и аппаратных средств адаптивного управления передачей информации в сетях связи приводит к необходимости совершенствования методического, алгоритмического и программного обеспечения с целью оперативного получения объективной информации о передаче данных и состоянии каналов связи.

В работе предлагается алгоритм адаптивной маршрутизации информационных пакетов сообщений в сетях обмена данными с использованием распределенной стратегии управления. В алгоритме предусмотрено сохранение контекста ошибок, пакетов сообщений и маршрутов

во временной базе данных. Использование разработанного алгоритма предусматривает динамическое управление памятью компьютера и буферных запоминающих устройств, возможности оперативного расширения базы данных и распространение ее в различных СУБД.

Алгоритм адаптивной маршрутизации информационных пакетов сообщений в сетях обмена данными с использованием распределенной стратегии управления осуществляет решение следующих задач:

- формирование маршрутной информации в центре коммутации;

- изменение маршрутной информации по данным промежуточных узлов сети;

- корректировки маршрутной информации в процессе обмена данными;

- изменение маршрутной информации на промежуточных узлах сети;

- обеспечивает статистический учет ошибок, возникающих в каналах по маршрутам прохождения информационных пакетов с целью анализа и возможности диагностирования состояния линий связи.

Алгоритм контроля маршрутной информации, передаваемой между абонентами учитывает одно из важнейших требований, предъявляемых к адаптивным автоматизированным системам управления связью - это возможность их использования в процессе обмена данными по разным каналам (виртуальному, дейтаграммному или выделенному).

Алгоритм контроля маршрутной информации, передаваемой между абонентами осуществляет решение следующих задач:

- проверка правильности передачи маршрутной части пакетов сообщений на узлах коммутации и у абонентов;

- учет особенностей различных каналов связи;

- учет особенностей использования как положительных (подтверждение приема пакета), так и отрицательных (запрос на повторную передачу пакета) квитанций;

- обеспечивает возможность контроля информации, принимаемой как в пакетах, так и в сообщениях.

Алгоритм адаптивной маршрутизации информационных пакетов сообщений в сетях является основой для проведения исследований вероятностно-временных характеристик каналов связи использующих разнообразные протоколы обмена данными.

## РОЗДІЛ 9

# ФІЛЬТРИ ГРАФІЧНИХ ЗОБРАЖЕНЬ НА ОСНОВІ ЧИСЛОВИХ В'ЯЗАНОК

***Анотація.** Розглянуто теоретичні та практичні проблеми побудови фільтрів графічних зображень за допомогою числових в'язанок. Зроблено аналіз та описано необхідні умови для створення якісних графічних фільтрів розмиття (низькочастотні фільтри), різкості (високочастотні фільтри), тиснення та виділення контурів. Дослідження та розробка алгоритмів для вирішення проблеми фільтрації растрових зображень з метою забезпечення достатньої якості зображення стала предметом даної роботи.*

***Ключові слова:** числова лінійка-в'язанка, ідеальна кільцева в'язанка, графічний фільтр.*

***Abstract.** The theoretical and practical problems of construction of filters of graphic images are considered by means of numerical bundles. An analysis is done and necessary terms are described for creation of high-quality graphic filters of blurring (low-frequency filters), sharpness (high-frequency filters), stamping and selection of contours. Research-and-development algorithms for the decision of problem of filtration of bitmapped images with the purpose of providing of sufficient quality of image became the article of this work.*

***Keywords:** numerical line-bundle, ideal ring bundle, graphic filter.*

**Вступ і постановка задачі.** Актуальність. Візуальне представлення даних є набагато інформативнішим за інші методи отримання та сприйняття інформації. поширенням засобів формування візуальної інформації дуже актуальними стали задачі обробки та покращання якості зображень. Серед цих задач одними з базових, класичних є проблеми підвищення контрастності, або навпаки його зменшення, знешумлення зображень та надання додаткових ефектів, наприклад, тиснення. Безліч підходів для фільтрації зображень розділяються на дві досить великі категорії: методи обробки в просторовій області та методи обробки в частотній області. В даній роботі будуть розглянуті просторові (матричні) методи обробки зображень. Просторові методи об'єднують підходи обробки, основані на маніпулюванні пікселями зображення, що опрацьовується.

Загальної теорії поліпшення зображень не існує. Коли зображення обробляється для візуальної інтерпретації, спостерігач є остаточним того, наскільки добре діє конкретний метод.

Візуальне оцінювання якості зображення є досить суб'єктивним процесом, що робить тим самим поняття "ідеального зображення" деяким

невловимим еталоном, за допомогою якого необхідно порівнювати ефективність алгоритму. Але тим не менш, коли проблема дозволяє встановити чіткі критерії якості, зазвичай потребується проведення певної кількості спроб тестування, поки не буде обраний конкретний підхід до поліпшення зображення. Таким чином, актуальність теми дослідження зумовлена необхідністю пошуку нових напрямків удосконалення задачі обробки зображень за допомогою фільтрів на основі матриці перетворення. Все це дозволить підвищити якість та ефективність вже розроблених методів розпізнавання та обробки зображень.

Мета роботи полягає в опрацюванні теоретичних положень, щодо завдання обробки зображень; дослідженні вже існуючих методів фільтрації та застосування числової лінійки-в'язанки для розробки фільтрів. Вище описана мета визначила необхідність вирішення таких задач:

узагальнити теоретичні основи та виявити особливості задачі обробки зображень в сучасних інформаційних умовах;

дослідити використання методів просторової обробки зображень;

узагальнити існуючий досвід використання числових лінійок-в'язанок та ідеальних кільцевих в'язанок у сфері цифрової обробки зображень, розробити й обґрунтувати досліджені методи фільтрації на основі числових лінійок-в'язанок.

Об'єкт дослідження – методи та алгоритми цифрової обробки растрових зображень.

Предмет дослідження - теоретичні і методичні основи обробки зображень за допомогою фільтрації у просторовій області.

Методи дослідження. Теоретичну і методологічну основу дослідження склали роботи провідних вчених та фахівців в області цифрової обробки зображень, а саме такі як Гонсалес Р., Хуанг Т.С., Яне Б., Яншин В.В., Калинин Г.А., Сойфер В.А. ті інші. У роботі були використані загальнонаукові методи: спостереження, порівняння, узагальнення, абстрагування, формалізація, аналіз і синтез, застосовувалися положення теорії множин.

Наукова новизна одержаних результатів. Виходячи з виконаних теоретичних та експериментальних досліджень результат роботи полягає в використанні стандартних алгоритмів фільтрації при застосуванні числових лінійних в'язанок та ідеальних кільцевих в'язанок.

В рамках цього наукового напрямку отримано такі основні результати:

✓ розроблено концептуальну модель фільтрів на основі числових лінійних в'язанок та ідеальних кільцевих в'язанок;

✓ досліджено застосування фільтрації зображень на основі числових лінійних в'язанок та ідеальних кільцевих в'язанок, а саме фільтрів розмиття, різкості, границь, тиснення.

### **Характеристика задачі цифрової фільтрації зображень та концептуальна модель**

Цифровим зображенням прийнято називати двовимірний, дискретний сигнал, що отримується в результаті перетворення безперервного (аналогового) зображення в еквівалентний цифровий масив. Під цифровою обробкою зображення зазвичай розуміється математична обробка сигналу, заданого матрицею дискретних відліків. Обробка виконується шляхом застосування до вхідного зображення різних перетворень, цифрова реалізація яких часто називається цифровою фільтрацією. З математичної точки зору процес фільтрації може бути представлений як перетворення множини вхідних зображень у множину вихідних за допомогою деякого оператора.

Цифрова фільтрація зображенні зазвичай спрямована на досягнення однієї з таких цілей:

перетворення по деякому закону і відповідно до заданих вимог дискретних відліків зображення в результаті чого формується нове зображення; отримання з зображення якоїсь корисної інформації про об'єкт дослідження, наприклад, як деякого вектору атрибутів зображення.

Цілі цифрової фільтрації досягаються використанням різних алгоритмів і методів обробки, кожен з яких спрямований на вирішення конкретного завдання. Завдання цифрової обробки зображень за своїм прикладним призначенням може бути класифіковане, наприклад, наступним чином: підвищення якості зображень, препарування зображень, сегментація зображень, морфологічна обробка зображень, виявлення низькорівневих ознак у зображенні, стиск зображень. Кожна з задач цифрової фільтрації включає множину підзадач, для вирішення яких розроблено велику кількість алгоритмів і методів, оснований на різних теоретичних підходах, і використовуються в залежності від конкретних вимог до результатів їх застосування.

Алгоритми та методи цифрової фільтрації умовно діляться на дві групи. До першої групи відносяться методи, засновані на поданні зображень та імпульсних реакцій фільтрів у вигляді матриць дискретних значень, що називаються методами цифрової фільтрації в просторовій області або методами просторової фільтрації. До другої групи належать методи фільтрації в частотній області або методи частотної фільтрації, засновані на перетвореннях спектрів зображень.

Процеси просторової обробки в загальному випадку описуються рівнянням виду (1):

$$g[x, y] = T(f[x, y]), \quad (1)$$

де  $f(x, y)$  і  $g(x, y)$  - вхідне і вихідне зображення відповідно,

$T$  – оператор над  $f$ , визначений в деякому околі точки  $[x, y]$ .

Оператор  $T$  реалізує будь-яке перетворення, що дозволяє обчислити значення  $g$  в точці  $[x, y]$  на основі значень  $f$  всередині заданої околиці. Один з основних підходів при такій постановці завдання просторової фільтрації базується на використанні масок, значення коефіцієнтів якої визначають сутність процесу. При цьому оператор  $T$  в (1) реалізується за допомогою згортки аналізованого зображення  $f[x, y]$  з маскою (або фільтром)  $h[x, y]$ .

Методи частотної фільтрації ґрунтуються на модифікації сигналу, який формується шляхом застосування до зображення деякого частотного перетворення. У цьому випадку модель обробки зображення описується рівнянням виду:

$$g[x, y] = F^{-1}(F(f[x, y])F(h[x, y])), \quad (2)$$

де  $F$  – пряме частотне перетворення,

$F^{-1}$  – зворотне частотне перетворення.

В даному випадку частотна фільтрація зводиться до обчислення прямого частотного перетворення вхідного зображення, зміни отриманого образу шляхом перемноження його з образом функції фільтра і обчислення зворотного частотного перетворення.

При вирішенні задач цифрової фільтрації вибір методу фільтрації (просторової або частотної) здійснюється на основі вимог до результатів обчислень, швидкості обробки даних, реалізованого алгоритму рішення в просторовому або частотному представленні і т.п. Деякі завдання легко описуються і вирішуються у просторовій області, наприклад, виділення контурів, інші - в частотній, наприклад, підкреслення меж.

Таким чином, вибір способу фільтрації залежить від розв'язуваної задачі, типу сигналу, вимог до точності і швидкості.

### **Розробка алгоритму просторової фільтрації зображень**

Загальний розроблений алгоритм для просторової фільтрації зображень складається з таких пунктів:

- ✓ отримання програмою вхідного зображення;



- ✓ вибір користувачем фільтру для зображення – передання відповідної матриці перетворення до основного модуля обробки зображення;
- ✓ обробка зображення обраним фільтром;
- ✓ виведення результуючого зображення на екран.

Алгоритм самої фільтрації зображення:

- ✓ отримання граничних розмірів зображення та матриці перетворення;
- ✓ множення кожного значення пікселя на відповідне центральний коефіцієнт матриці фільтру, крім цього на відповідні значення множаться навколишні його елементи;
- ✓ сумування всіх результатів множення;
- ✓ ділення результату на суму коефіцієнтів матриці перетворення;
- ✓ перевірка чи значення пікселів не виходять за межі (0 та 255);
- ✓ запис нового значення пікселя в зображення.

### **Аналіз використання числових лінійних в'язанок для розробки цифрових фільтрів**

В результаті проведення дослідження за допомогою розробленої програми для обробки зображень, використовуючи числові лінійки в'язанки на стандартному алгоритмі просторової фільтрації, отримано нові різновиди матриць перетворення [83; 84]. А саме матриці фільтрів розмиття, підвищення різкості, виділення контурів зображення та тиснення. В якості коефіцієнтів матриць перетворення були взяті числові лінійки в'язанки (ЧЛВ), а саме з табл. 9.1.

Таблиця 9.1

**Числові лінійки в'язанки**

<i>N</i> (порядок)	ЧЛВ
3	1, 3, 2
4	3, 1, 5, 2
5	1, 3, 6, 2, 5 1, 7, 3, 2, 4
6	1, 10, 5, 3, 4, 2
6	1, 4, 2, 8, 3, 9
7	1, 3, 5, 6, 7, 10, 2
	4, 1, 12, 2, 6, 3, 7
8	1, 4, 7, 13, 12, 8, 6, 3
	2, 8, 14, 1, 4, 7, 6, 3

Використання даних лінійок дало досить гарні результати. Для отримання ефекту розмиття можна застосовувати такі матриці:

$$H = \begin{pmatrix} 1 & 3 & 2 \\ 2 & 1 & 3 \\ 3 & 2 & 1 \end{pmatrix} \quad H = \begin{pmatrix} 3 & 1 & 2 & 2 \\ 2 & 3 & 5 & 5 \\ 5 & 2 & 1 & 1 \\ 1 & 5 & 3 & 3 \end{pmatrix} \quad H = \begin{pmatrix} 1 & 3 & 6 & 2 & 5 \\ 5 & 1 & 3 & 6 & 2 \\ 2 & 5 & 1 & 3 & 6 \\ 6 & 2 & 5 & 1 & 3 \\ 3 & 6 & 2 & 5 & 1 \end{pmatrix}$$

$$H = \begin{pmatrix} 1 & 7 & 3 & 2 & 4 \\ 4 & 1 & 7 & 3 & 2 \\ 2 & 4 & 1 & 7 & 3 \\ 3 & 2 & 4 & 1 & 7 \\ 7 & 3 & 2 & 4 & 1 \end{pmatrix} \quad H = \begin{pmatrix} 1 & 10 & 5 & 3 & 4 & 2 \\ 2 & 1 & 10 & 5 & 3 & 4 \\ 4 & 2 & 1 & 10 & 5 & 3 \\ 3 & 4 & 2 & 1 & 10 & 5 \\ 5 & 3 & 4 & 2 & 1 & 10 \\ 10 & 5 & 5 & 3 & 4 & 1 \end{pmatrix} \quad H = \begin{pmatrix} 1 & 4 & 2 & 8 & 3 & 9 \\ 9 & 1 & 4 & 2 & 8 & 3 \\ 3 & 9 & 1 & 4 & 2 & 8 \\ 8 & 3 & 9 & 1 & 4 & 2 \\ 2 & 8 & 3 & 9 & 1 & 4 \\ 4 & 2 & 8 & 3 & 9 & 1 \end{pmatrix}$$

$$H = \begin{pmatrix} 1 & 3 & 5 & 6 & 7 & 10 & 2 \\ 2 & 1 & 3 & 5 & 6 & 7 & 10 \\ 10 & 2 & 1 & 3 & 5 & 6 & 7 \\ 7 & 10 & 2 & 1 & 3 & 5 & 6 \\ 6 & 7 & 10 & 2 & 1 & 3 & 5 \\ 5 & 6 & 7 & 10 & 2 & 1 & 3 \\ 3 & 5 & 6 & 7 & 10 & 2 & 1 \end{pmatrix} \quad H = \begin{pmatrix} 4 & 1 & 12 & 2 & 6 & 3 & 7 \\ 7 & 4 & 1 & 12 & 2 & 6 & 3 \\ 3 & 7 & 4 & 1 & 12 & 2 & 6 \\ 6 & 3 & 7 & 4 & 1 & 12 & 2 \\ 2 & 6 & 3 & 7 & 1 & 1 & 2 \\ 12 & 2 & 6 & 3 & 7 & 4 & 1 \\ 1 & 12 & 2 & 6 & 3 & 7 & 4 \end{pmatrix}$$

$$H = \begin{pmatrix} 1 & 4 & 7 & 13 & 12 & 8 & 6 & 3 \\ 3 & 1 & 4 & 7 & 13 & 12 & 8 & 6 \\ 6 & 3 & 1 & 4 & 7 & 13 & 12 & 8 \\ 8 & 6 & 3 & 1 & 4 & 7 & 13 & 12 \\ 12 & 8 & 6 & 3 & 1 & 4 & 7 & 13 \\ 13 & 12 & 8 & 6 & 3 & 1 & 1 & 7 \\ 7 & 13 & 12 & 8 & 6 & 3 & 3 & 4 \\ 4 & 7 & 13 & 12 & 8 & 6 & 6 & 1 \end{pmatrix} \quad H = \begin{pmatrix} 2 & 8 & 14 & 1 & 4 & 7 & 6 & 3 \\ 3 & 2 & 8 & 14 & 1 & 4 & 7 & 6 \\ 6 & 3 & 2 & 8 & 14 & 1 & 4 & 7 \\ 7 & 6 & 3 & 2 & 8 & 14 & 1 & 4 \\ 4 & 7 & 6 & 3 & 2 & 8 & 14 & 1 \\ 1 & 4 & 7 & 3 & 3 & 2 & 8 & 14 \\ 13 & 1 & 4 & 7 & 6 & 3 & 2 & 8 \\ 8 & 14 & 1 & 4 & 7 & 6 & 3 & 2 \end{pmatrix}$$

Після порівняння зображень опрацьованих фільтром розмиття, використовуючи кожну з цих матриць, можна сказати, що зі збільшенням порядку матриці зростає ефект фільтру.

Результатом застосування матриці:

$$H = \begin{pmatrix} 1 & 4 & 7 & 13 & 12 & 8 & 6 & 3 \\ 3 & 1 & 4 & 7 & 13 & 12 & 8 & 6 \\ 6 & 3 & 1 & 4 & 7 & 13 & 12 & 8 \\ -8 & -6 & -3 & -1 & -4 & -7 & -13 & -12 \\ -12 & -8 & -6 & -3 & -1 & -4 & -7 & -13 \\ 13 & 12 & 8 & 6 & 3 & 1 & 4 & 7 \\ 7 & 13 & 12 & 8 & 6 & 3 & 1 & 4 \\ 4 & 7 & 13 & 12 & 8 & 6 & 3 & 1 \end{pmatrix}$$

є фільтр з ефектом горизонтального розмиття.

Для отримання ефекту вертикального розмиття необхідно застосувати матрицю:

$$H = \begin{pmatrix} -1 & -4 & -7 & -13 & -12 & -8 & -6 & -3 \\ 3 & 1 & 4 & 7 & 13 & 12 & 8 & 6 \\ 6 & 3 & 1 & 4 & 7 & 13 & 12 & 8 \\ 8 & 6 & 3 & 1 & 4 & 7 & 13 & 12 \\ 12 & 8 & 6 & 3 & 1 & 4 & 7 & 13 \\ 13 & 12 & 8 & 6 & 3 & 1 & 4 & 7 \\ 7 & 13 & 12 & 8 & 6 & 3 & 1 & 4 \\ -4 & -7 & -13 & -12 & -8 & -6 & -3 & -1 \end{pmatrix}$$

Для збільшення різкості зображення підходить така матриця:

$$H = \begin{pmatrix} -1 & 3 & -2 \\ 2 & 1 & 3 \\ -3 & 2 & -1 \end{pmatrix}$$

Порівнюючи отриману матрицю зі стандартною можна побачити, що не потрібна велика різниця між центральним елементом та його навколишніх. Наступні матриці

$$H = \begin{vmatrix} 3 & 1 & 5 & 2 \\ -2 & -3 & -1 & -5 \\ -5 & -2 & -3 & -1 \\ 1 & 5 & 2 & 3 \end{vmatrix} \quad H = \begin{vmatrix} -1 & -3 & -6 & -2 & -5 \\ 5 & 1 & 3 & 6 & 2 \\ 2 & -5 & -1 & -3 & 6 \\ 6 & 2 & 5 & 1 & 3 \\ -3 & -6 & -2 & -5 & -1 \end{vmatrix}$$

$$H = \begin{vmatrix} -4 & -1 & -12 & -2 & -6 & -3 & -7 \\ -7 & 4 & 1 & 12 & 2 & 6 & -3 \\ -3 & 7 & 4 & 1 & 12 & 2 & -6 \\ -6 & 3 & 7 & 4 & 1 & 12 & -2 \\ -2 & 6 & 3 & 7 & 4 & 1 & -12 \\ -12 & 2 & 6 & 3 & 7 & 4 & -1 \\ -1 & -12 & -2 & -6 & -3 & -7 & -4 \end{vmatrix}$$

можна використовувати для виділення контурів зображення. Ці фільтри можна використовувати для будь-яких растрових зображень, але краще за все вони працюють з зображеннями у відтінках сірого.

Фільтр тиснення (його графічний ефект схожий на гравюру зображення) реалізується за допомогою матриць:

$$H = \begin{vmatrix} 1 & 7 & 3 & 2 & 4 \\ -4 & -1 & -7 & -3 & -2 \\ 2 & 4 & 1 & -7 & 3 \\ -3 & -2 & -4 & -1 & -7 \\ 7 & 3 & 2 & 4 & 1 \end{vmatrix} \quad H = \begin{vmatrix} -1 & -3 & -6 & -2 & -5 \\ 5 & 1 & 3 & 6 & 2 \\ -2 & 5 & 1 & 3 & -6 \\ 6 & 2 & 5 & 1 & 3 \\ -3 & -6 & -2 & -5 & -1 \end{vmatrix}$$

$$H = \begin{vmatrix} 1 & 10 & 5 & 3 & 4 & 2 \\ -2 & -1 & -10 & -5 & -3 & -4 \\ 4 & 2 & 1 & 10 & 5 & 3 \\ -3 & -4 & -2 & -1 & -10 & -5 \\ -5 & -3 & -4 & -2 & -1 & -10 \\ 10 & 5 & 3 & 4 & 2 & 1 \end{vmatrix}$$

## Результати роботи реалізованого алгоритму цифрової обробки зображень

Використовуючи розроблену програму для обробки зображень за допомогою фільтрів, досліджено стандартні загально відомі фільтри та фільтри на основі числових лінійок-в'язанок. Використовуючи фільтри розмиття зображення можна отримати такі результати (рис. 9.1–9.3):



*Рис. 9.1. Вхідне зображення*

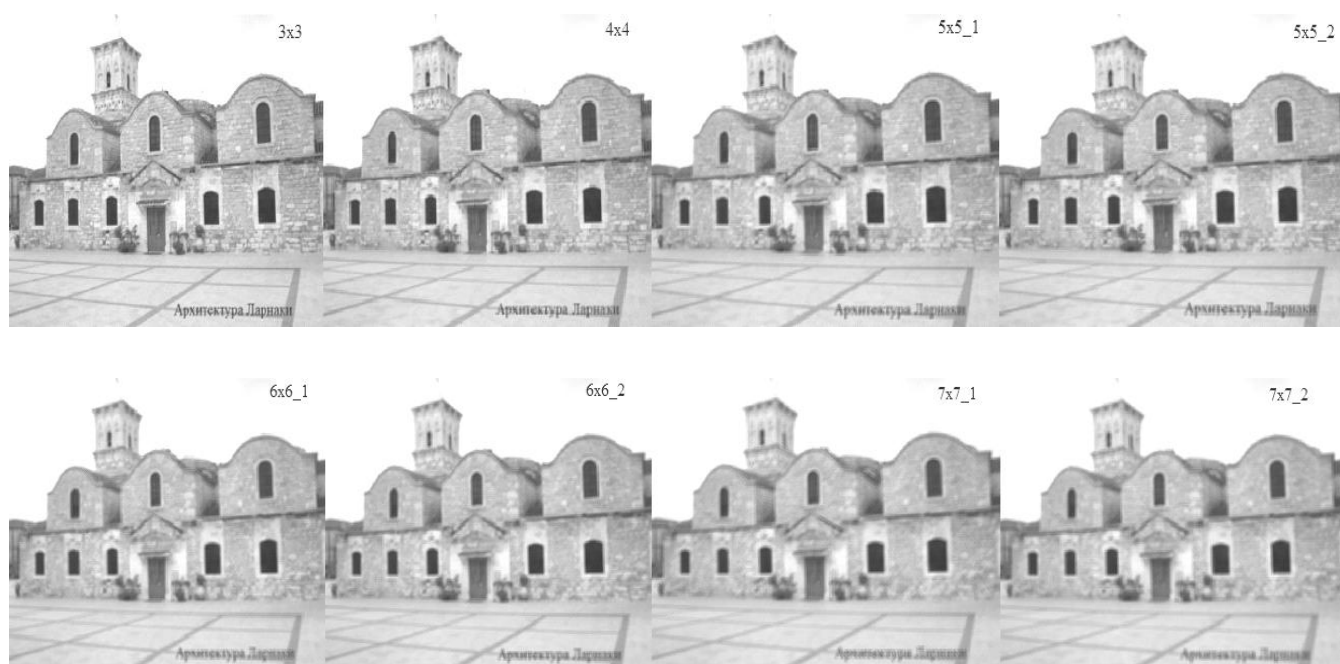


*Рис. 9.2. Звичайний фільтр "Розмиття"*



*Рис. 9.3. Фільтр "Розмиття за Гаусом"*

Ефект розмиття на основі ЧЛВ представлений на рис. 9.4.

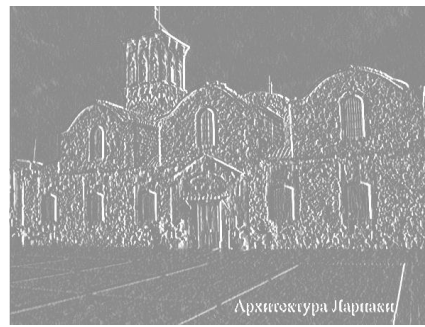


*Рис. 9.4. Фільтр "Розмиття на ЧЛВ" (матриці порядку від 3x3 до 7x7)*

Чим більший порядок матриці, тим сильніший ефект розмиття. Результат роботи фільтрів підвищення різкості (рис. 9.5) та границь (рис. 9.6).



**Рис. 9.5. Фільтр “Різкість на ЧЛВ”**



**Рис. 9.6. Фільтр “Границі на ЧЛВ”**

Фільтр для виділення контурів зображення можна використовувати, як для кольорового зображення, так і для відтінків сірого. Досліджуючи фільтри виділення контурів та тиснення на з вхідним зображенням на рис. 9.7 отримано результати на рис. 9.8–9.14.



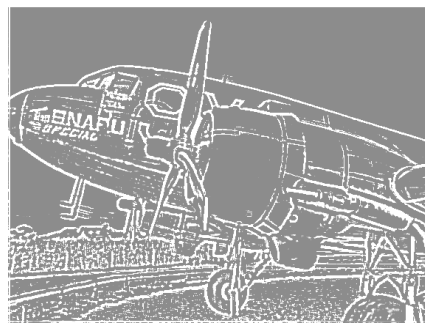
**Рис. 9.7. Вхідне зображення**



**Рис. 9.8. Звичайний фільтр “Границі”**

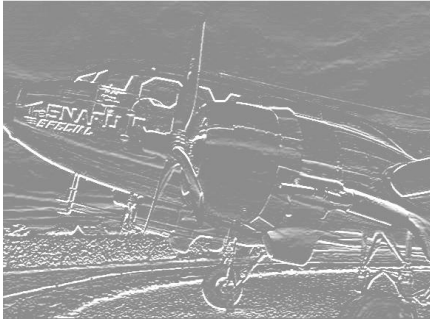


**Рис. 9.9. Зображення опрацьоване фільтром “Границі” (матриця 5x5)**

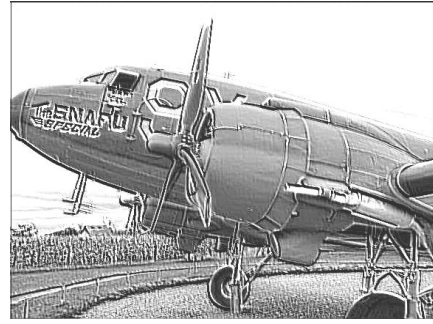


**Рис. 9.10. Зображення опрацьоване фільтром “Границі” (матриця 7x7)**

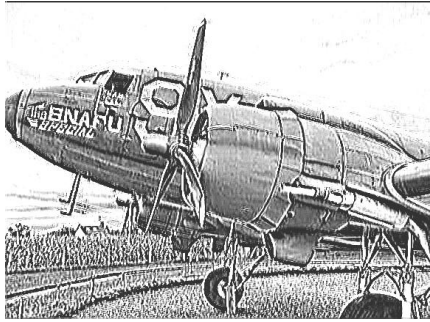
ЧЛВ можна застосовувати для отримання ефекту гравюри фільтром “Тиснення” на основі ЧЛВ з матрицями різної розмірності (рис. 9.12–9.14).



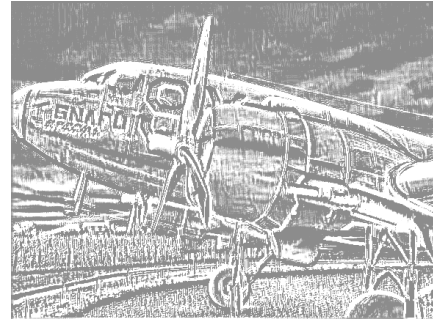
*Рис. 9.11. Звичайний фільтр “Тиснення”*



*Рис. 9.12. Фільтр “Тиснення на основі ЧЛВ” (матриця 4x4)*



*Рис. 9.13. Фільтр “Тиснення на основі ЧЛВ” (матриця 5x5)*



*Рис. 9.14. Фільтр “Тиснення на основі ЧЛВ” (матриця 6x6)*

Провівши порівняння отриманих результатів можна зробити висновок, що деякі фільтри з використанням числових лінійок в'язанок виявилися кращими за вже відомі раніше. Також у розроблених фільтрах простежується закономірність у тому, що чим більша розмірність матриці перетворення, тим сильніший ефект фільтру.

### **Дослідження ідеальних кільцевих в'язанок для розробки цифрових фільтрів**

В результаті проведення дослідження за допомогою розробленого програмного продукту просторової фільтрації на основі ідеальних кільцевих в'язанок (ІКВ) для обробки зображень, отримано нові матриці перетворення: матриці фільтрів розмиття, підвищення різкості, виділення контурів зображення та тиснення [81; 83; 84]. В якості коефіцієнтів матриць перетворення були взяті ідеальні кільцеві в'язанки з табл. 9.2.

## Ідеальні кільцеві в'язанки

N n/n	N (порядок) ІКВ	R (кратність) ІКВ	Ідеальні кільцеві в'язанки
1.	4	1	1 2 6 4
2.			1 3 2 7
3.	4	2	1 1 2 3
4.	5	1	1 3 10 2 5
5.	5	2	1 1 2 3 4
6.	6	1	1 2 5 4 6 13
7.			1 2 7 4 12 5
8.			1 2 3 7 8 10
9.			1 3 6 2 5 14
10.			1 7 3 2 4 14
11.	6	3	1 1 2 1 2 4
12.	7	3	1 1 2 1 3 2 5
13.	8	1	1 2 10 19 4 7 9 5
14.			1 3 5 11 2 12 17 6
15.			1 3 8 2 16 7 15 5
16.			1 4 2 10 18 3 11 8
17.			1 4 22 7 3 6 2 12
18.			1 6 12 4 21 3 2 8
19.	8	4	1 1 1 2 2 1 3 4

Використання даних ІКВ дало непогані результати. Для отримання ефекту розмиття можна застосовувати такі матриці на основі ІКВ (табл. 9.2):

$$\begin{aligned}
 H = \begin{vmatrix} 1 & 4 & 2 \\ 2 & 1 & 4 \\ 4 & 2 & 1 \end{vmatrix} \quad H = \begin{vmatrix} 4 & 1 & 2 & 6 \\ 6 & 4 & 1 & 2 \\ 2 & 6 & 4 & 1 \\ 1 & 2 & 6 & 4 \end{vmatrix} \quad H = \begin{vmatrix} 1 & 5 & 2 & 10 & 3 \\ 3 & 1 & 5 & 2 & 10 \\ 10 & 3 & 1 & 5 & 2 \\ 2 & 10 & 3 & 1 & 5 \\ 5 & 2 & 10 & 3 & 1 \end{vmatrix} \quad H = \begin{vmatrix} 1 & 1 & 2 & 3 & 4 \\ 4 & 1 & 1 & 2 & 3 \\ 3 & 4 & 1 & 1 & 2 \\ 2 & 3 & 4 & 1 & 1 \\ 1 & 2 & 3 & 4 & 1 \end{vmatrix}
 \end{aligned}$$

Після порівняння зображень опрацьованих фільтром розмиття, використовуючи кожен з цих матриць, можна сказати, що зі збільшенням порядку матриці зростає ефект фільтру. Результатом застосування матриці на основі ІКВ (табл. 9.2):



$$H = \begin{pmatrix} 1 & 3 & 8 & 2 & 16 & 7 & 15 & 5 \\ 5 & 1 & 3 & 8 & 2 & 16 & 7 & 15 \\ 15 & 5 & 1 & 3 & 8 & 2 & 16 & 7 \\ -7 & -15 & -5 & -1 & -3 & -8 & -2 & -16 \\ -16 & -7 & -15 & -5 & -1 & -3 & -8 & -2 \\ 2 & 16 & 7 & 15 & 5 & 1 & 3 & 8 \\ 8 & 2 & 16 & 7 & 15 & 5 & 1 & 3 \\ 3 & 8 & 2 & 16 & 7 & 15 & 5 & 1 \end{pmatrix}$$

є фільтр з ефектом горизонтального розмиття.

Для отримання ефекту вертикального розмиття необхідно застосувати наступну матрицю на основі ІКВ:

$$H = \begin{pmatrix} -1 & -3 & -8 & -2 & -16 & -7 & -15 & -5 \\ 5 & 1 & 3 & 8 & 2 & 16 & 7 & 15 \\ 15 & 5 & 1 & 3 & 8 & 2 & 16 & 7 \\ 7 & 15 & 5 & 1 & 3 & 8 & 2 & 16 \\ 16 & 7 & 15 & 5 & 1 & 3 & 8 & 2 \\ 2 & 16 & 7 & 15 & 5 & 1 & 3 & 8 \\ 8 & 2 & 16 & 7 & 15 & 5 & 1 & 3 \\ -3 & -8 & -2 & -16 & -7 & -15 & -5 & -1 \end{pmatrix}$$

Для збільшення різкості зображення підходить така матриця на основі ІКВ:

$$H = \begin{pmatrix} -1 & 4 & -2 \\ 2 & 1 & 4 \\ -4 & 2 & -1 \end{pmatrix}$$

Роблячи порівняння отриманої матриці зі стандартною можна побачити, що не потрібна велика різниця між центральним елементом та його сусідніми. Наступні матриці на основі ІКВ можна використовувати для виділення контурів зображення.

$$H = \begin{pmatrix} 4 & 1 & 2 & 6 \\ -6 & -4 & -1 & -2 \\ -2 & -6 & -4 & -1 \\ 1 & 2 & 6 & 4 \end{pmatrix} \quad H = \begin{pmatrix} -1 & -3 & -10 & -2 & -5 \\ 5 & 1 & 3 & 10 & 2 \\ 2 & -5 & -1 & -3 & 10 \\ 10 & 2 & 5 & 1 & 3 \\ -3 & -10 & -2 & -5 & -1 \end{pmatrix}$$

Фільтр тиснення на основі ІКВ реалізується за допомогою наступних матриць:

$$H = \begin{vmatrix} 1 & 5 & 2 & 10 & 3 \\ -3 & -1 & -5 & -2 & -10 \\ 10 & 3 & 1 & -5 & 2 \\ -2 & -10 & -3 & -1 & -5 \\ 5 & 2 & 10 & 3 & 1 \end{vmatrix} \quad H = \begin{vmatrix} -1 & -3 & -10 & -2 & -5 \\ 5 & 1 & 3 & 10 & 2 \\ -2 & 5 & 1 & 3 & -10 \\ 10 & 2 & 5 & 1 & 3 \\ -3 & -10 & -2 & -5 & -1 \end{vmatrix}$$

### **Розробка алгоритму фільтрації зображень на основі ІКВ**

Досліджений та розроблений алгоритм для фільтрації зображень складається з таких пунктів:

отримання вхідного зображення;

вибір користувачем фільтру для зображення – передання відповідної матриці перетворення до основного модуля обробки зображення;

обробка зображення обраним фільтром за допомогою ІКВ;

виведення результуючого зображення на екран монітору.

Алгоритм самої фільтрації растрового зображення:

отримання граничних розмірів растрового зображення та матриці перетворення;

множення кожного значення пікселя на відповідне центральний коефіцієнт матриці фільтру ІКВ, крім цього на відповідні значення множаться навколишні його елементи;

сумування всіх результатів множення матриці фільтру ІКВ;

ділення результату на суму коефіцієнтів матриці перетворення фільтру ІКВ;

перевірка чи значення пікселів не виходять за діапазон (0 та 255);

запис нового значення пікселя в растрове зображення.

### **Результати роботи алгоритму фільтрації растрових зображень**

Використовуючи розроблений програмний продукт для фільтрації растрових зображень за допомогою ІКВ, досліджено і стандартні загально відомі фільтри та фільтри на основі ІКВ [81; 85; 82]. Для стандартних фільтрів та фільтрів на основі ІКВ отримано такі результати (рис. 9.15–9.17).



*Рис. 9.15. Вхідне зображення*



*Рис. 9.16. Фільтр “Розмиття за Гаусом”*



*Рис. 9.17. Фільтр “Розмиття на основі ІКВ”*

Використовуючи фільтри розмиття зображення можна отримати такі результати (рис. 9.18–9.19):



*Рис. 9.18. Фільтр “Розмиття за Гаусом”*



*Рис. 9.19. Фільтр “Розмиття на основі ІКВ”*

Результат роботи фільтру підвищення різкості (рис. 9.20–9.22):



*Рис. 9.20. Вхідне зображення*

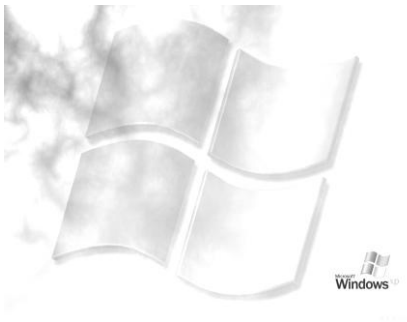


*Рис. 9.21. Звичайний фільтр “Різкість”*



*Рис. 9.22. Фільтр “Різкість на основі ІКВ”*

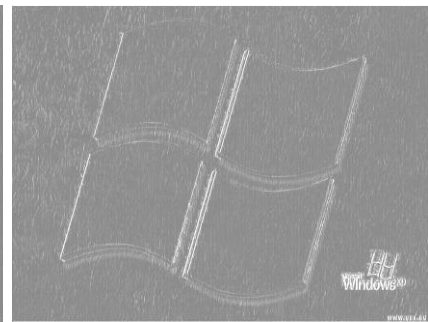
Фільтр для виділення контурів зображення можна використовувати, як для кольорового зображення, так і для чорно-білого зображення (рис. 9.23–9.25). Це видно з результату роботи фільтру:



*Рис. 9.23. Вхідне зображення*

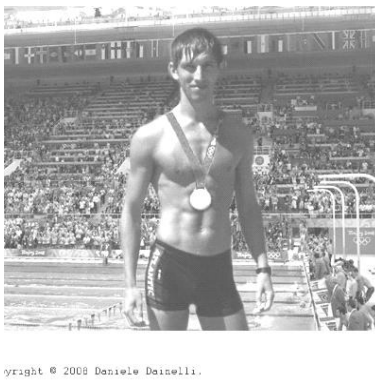


*Рис. 9.24. Звичайний фільтр “Границі”*

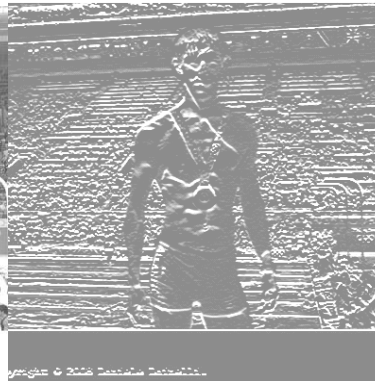


*Рис. 9.25. Фільтр “Границі на основі ІКВ”*

Застосувавши фільтр для тиснення зображення отримано такий результат (рис. 9.26–9.28):



*Рис. 9.26. Вхідне зображення*



*Рис. 9.27. Звичайний фільтр “Тиснення”*



*Рис. 9.28. Фільтр “Тиснення на основі ІКВ”*

Фільтри реалізовані з використанням ідеальних кільцевих в'язанок (ІКВ). Дані результати зображень ми отримали використовуючи матриці перетворення з використанням ІКВ. Отримані фільтри дали ефект розмиття (рис.9.16), підвищення різкості (рис. 9.21), виділення границь (рис. 9.24) та тиснення зображення (рис. 9.27).

Досліджено, що чим більший порядок матриці на основі ІКВ, тим сильніший ефект розмиття. Провівши порівняння отриманих результатів можна зробити висновок, що деякі фільтри з використанням ІКВ виявилися кращими за існуючі стандартні фільтри. Також у розроблених фільтрів можна простежити закономірність у тому, що чим більша розмірність матриці перетворення на основі ІКВ, тим сильніший ефект фільтру.

**Висновки.** Дослідження та розробка алгоритмів для вирішення проблеми фільтрації растрових зображень з метою забезпечення достатньої якості зображення стала предметом даної роботи.

Відповідно до поставленої мети було вирішено такі задачі:

досліджено теоретичні основи та виявлено особливості задачі обробки зображень в сучасних інформаційних умовах;

досліджено використання методів просторової обробки зображень;

досліджено існуючий досвід використання числових лінійок-в'язанок та ідеальних кільцевих в'язанок у сфері цифрової обробки зображень, розроблено й обґрунтовано досліджені методи фільтрації на основі числових лінійок-в'язанок та ідеальних кільцевих в'язанок;

розроблений метод фільтрації зображення з використанням числових лінійок в'язанок та ідеальних кільцевих в'язанок дозволяє використовувати його як один з складових загальної оцінки візуальної якості зображення, що не залежить від суб'єктивного сприйняття.

## РОЗДІЛ 10

# НЕЙРОСЕТЕВОЙ МЕТОД КЛАСТЕРИЗАЦИИ ЭЛЕКТРОННЫХ ТЕКСТОВ В РАСПРЕДЕЛЕННОЙ ФАЙЛОВОЙ СИСТЕМЕ

***Аннотация.** Предложен метод кластеризации и индексации электронных текстов, основанный на применении алгоритма SOINN, лингвистических дескрипторов и модульной системы обработки файловых массивов. Метод позволяет формировать динамичную структуру кластеров входной информации и создавать единые индексы электронной библиотеки. Результаты тестирования подтверждают эффективность применения метода для обработки и классификации документов в текстовых массивах.*

***Ключевые слова:** кластеризация текстов, нейронная сеть, лингвистический дескриптор, индексация файлов, программный модуль*

***Abstract.** The method of clusterization and indexation of e-texts, based on application of algorithm SOINN, linguistic descriptors and module system of treatment of filearrays, is offered. A method allows to form the dynamic structure of clusters of entrance information and create the single indexes of e-library. Testing results confirm efficiency of application of method for treatment and classification of documents in text arrays.*

***Keywords:** clusterization of texts, neural network, linguistic descriptor, indexation of files, programmatic module.*

**Введение и постановка задачи.** Отличительной особенностью методов кластеризации, используемых в современных системах анализа корпусов электронных документов, является способность автоматически выделять группы в потоке входных данных. В контексте обработки текстов на естественном языке это свойство является особенно привлекательным, когда возникает необходимость оперативно выделить тематические кластеры в большом массиве текстовых документов. Такая кластеризация, например, актуальна для информационно-поисковых систем, когда пользователю необходимо получить общее представление обо всем списке найденных документов, не просматривая при этом их большую часть. При этом возникает задача синтеза аннотаций кластеров, кратко отражающих тематику их документов, важной особенностью которой является необходимость модификации структуры формируемых кластеров в реальном времени, позволяющей учитывать изменение характера анализируемых данных.

К наиболее эффективным подходам к аннотированию кластеров, осуществляемому для их дальнейшего использования в задачах анализа текстов, является присвоение уникального индекса каждому документу из формируемого кластера. Данный способ позволяет однозначно и оперативно определять степень тематической близости документов с возможностью быстрого оценивания последствий вхождения новых документов в структуру кластера и модифицировать эту структуру при существенном изменении характера данных. При кластеризации больших объемов текстовой информации эффективным является выделение и анализ лингвистических дескрипторов. Лингвистический дескриптор – лексическая единица (слово, словосочетание, аббревиатура), служащая для описания основного смыслового содержания документа или формулировки запроса при поиске документа в информационно-поисковой или классифицирующей системе [94]. Актуальной является задача разработки метода кластеризации данных с анализом лингвистических дескрипторов в массивах с распределенным хранением информации, позволяющего учитывать возможность изменения местонахождения файла в файловой структуре с сохраняемой иерархией. Для решения такой задачи можно использовать размещение дополнительной информации в системном разделе метаданных файла, что позволяет реализовать удобное и компактное хранение предобработанных данных этого файла с маркировкой по дате вытяжки. В комбинации с такими стандартными свойствами, как дата создания и дата изменения, этот подход позволяет получить минимально необходимую распределенную систему контроля версии предобработанного вектора дескрипторов для каждого текстового документа в формируемой библиотеке.

Целью данной статьи является разработка и тестирование нейросетевого метода, позволяющего с применением лингвистических дескрипторов решить проблему кластеризации и индексации больших объемов электронной текстовой информации по пользовательским категориям, учитывая возможность распределенного хранения массивов информации.

Задачами программной системы, реализующей предлагаемый метод, являются автоматический анализ данных от различных источников с применением алгоритмов машинного обучения для формирования динамичной структуры кластеров входной текстовой информации и создание единого индекса электронной библиотеки, позволяющего повысить эффективность поиска информации. При этом должна быть реализована возможность поддержания кластерной системы в актуальном и исправном состоянии в условиях изменчивости среды.

**Основная часть.** Рассмотрим задачу кластеризации и кластерной индексации большого массива документов, имеющего политематический характер. Таким массивом может быть, например, электронный сборник материалов конференции, содержащих аннотации и основной текст; массив документов большой электронной библиотеки, посвященных некоторому общему научному направлению, и т.п. Проблемами, возникающими при реализации такой задачи, являются: наличие служебных элементов и посторонних блоков текста, не относящихся к основной тематике документа; наличие в обучающем массиве аномальных документов (пустых, в неизвестных кодировках и т.п.); сложность автоматического формирования решающих правил для рубрик из-за негативного влияния посторонней информации; снижение качества классификации из-за наложения нескольких рубрик друг на друга; сложность интерпретации результатов классификации из-за неопределенности расположения в тексте информации [95].

Перспективным направлением повышения эффективности реализации подхода, связанного с автоматической кластеризацией политематических текстовых ресурсов, является применение нейросетевых методов, основанных на возможности учета в схеме кластеризации операций выделения из текстов дескрипторов и ключевых слов. В соответствии с поставленной задачей, для большого объема разнородных текстов необходимо получить множество массивов, содержащих документы определенной тематики. Рассмотрим предлагаемую процедуру построения соответствующего модуля кластеризации с предварительной фильтрацией исходного массива текстовых документов и выделением лингвистических дескрипторов. Как правило, дескриптор может быть однозначно поставлен в соответствие группе ключевых слов естественного языка, отобранных из текста, относящегося к определенной области знаний. Это позволяет создать уникальные словари дескрипторов для учета их в схеме обработки входных текстов с целью повышения точности кластеризации. При этом для характеристики анализируемых текстов используется векторная модель представления документов  $d$  исходного массива:

$$d = (t_1, t_2, \dots, t_n), \quad (1)$$

где  $t_1, t_2, \dots, t_n$  – дескрипторные термины (признаки) текстов;  $n$  – общее количество учитываемых терминов.

Компонентам вектора (1) могут быть поставлены в соответствие бинарная или частотная функции взвешивания. Бинарные векторы



представляются как последовательность нулей и единиц: если конкретный термин из словаря выборки встречается в тексте – вес термина будет равен 1, в противном случае – 0. Частотные векторы формируются на основе количества вхождений определенного термина в классе документов.

При реализации процедуры кластеризации для каждого встреченного в текстах слова рассчитывается и сохраняется его вес – оценка вероятности того, что текст с этим словом принадлежит к одному из возможных классов (одна из возможных аппроксимаций такой оценки состоит в замене бинарной функции взвешивания терминов  $t_i$  в векторной модели (1) частотной функцией, ставящей в соответствие этим терминам частоту их появления в анализируемом документе). На основе данных о классификации лингвистических дескрипторов при составлении локальных словарей производится расчет целесообразности выбора тех или иных его вариантов с учетом требований к системе, по каждому выбранному классу дескрипторов в различных классификациях. При этом учитывается общее соотношение слов, распределенных для каждой из категорий выбранных подклассов дескрипторов. Для формирования локальных словарей дескрипторов для каждого из формируемых в дальнейшем классов рассматриваются такие виды классификации, как “Части речи” и “Частотность”. На основе анализа соотношения числа дескрипторов и возможностей их извлечения из неструктурированной текстовой информации можно утверждать, что для повышения точности классификации: рациональней всего использовать существительные; по частотной встречаемости видов дескрипторов: целесообразно использовать высокочастотные и низкочастотные дескрипторы. Наиболее соответствующей этим требованиям структурой являются аббревиатуры. Аббревиатуры – это существительные, состоящие из усеченных слов, входящих в исходное словосочетание, или из усеченных частей исходного сложного слова, а также из названий начальных букв этих слов (или их частей). Аббревиатуры являются низкочастотными дескрипторами, которые легко получить из текстов (особенно из их аннотаций). В случае, если в текстах отсутствуют аббревиатуры, целесообразно использовать высокочастотные дескрипторы для максимального охвата всего массива текстовой информации.

Для формирования входных данных системы кластеризации был использован алгоритм выделения дескрипторов из текста, приведенный в [95]. Векторные дескрипторы разных типов, поступающие на вход блока кластеризации, накапливаются в верхнем суперклассе программной системы

(для обеспечения доступа к списку всех уникальных дескрипторов при кластеризации документа) и сохраняются в метаданных файлов для исключения возможности повторного анализа одного и того же файла.

Сформированные векторы служат основой для дальнейшего анализа и составления кластерной структуры данных, последующей индексации и организации работы поисковой машины. Рассмотренная в [95] схема предварительной обработки данных характеризуется универсальной применимостью вне зависимости от используемого формата файлов и позволяет эффективно использовать содержащуюся в каждом файле информацию. В результате такой обработки у текстового файла появляются следующие дополнительные свойства: дата индексации; полное индексное имя; входной вектор файлов дескрипторов в сокращенном виде.

В случае, если в файле отсутствует возможность записи в блок метаданных, то все метаданные записываются в первую строку файла. Отметим, что существуют проблемные форматы, модифицирование которых может привести к их повреждению. Среди них:

- форматы сканированных или автогенерированных книг-изображений (например, .pdf и .djvu);

- форматы изображений без возможности создания особых метатегов (например, .jpg, .png, .gif и т.п.);

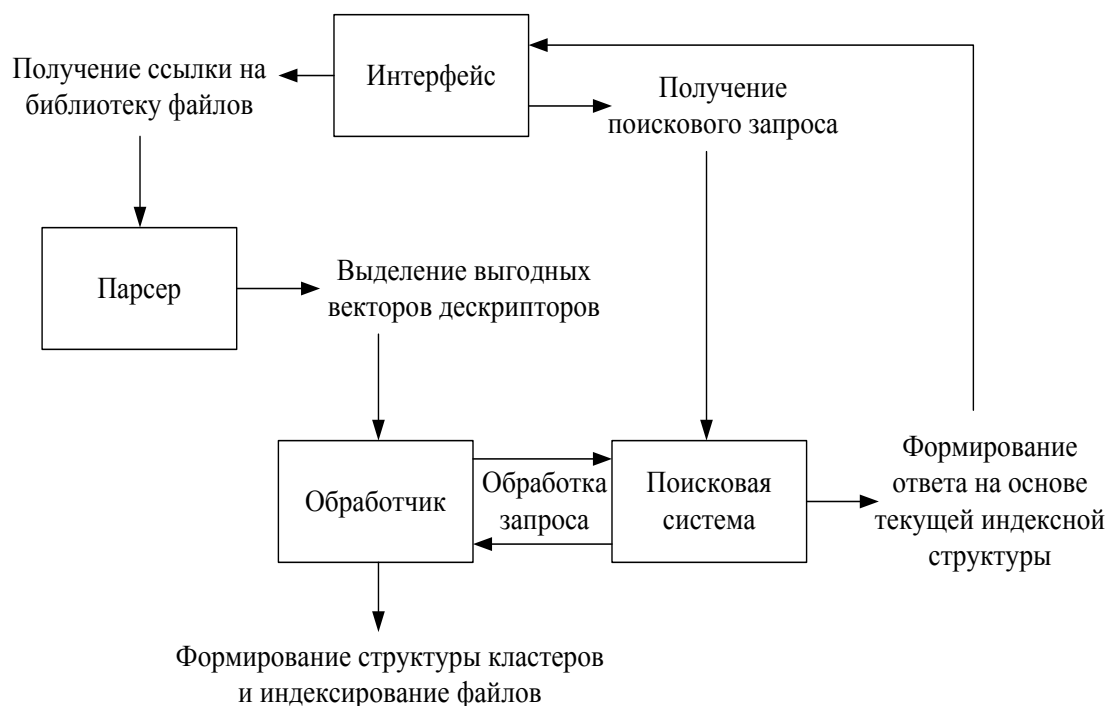
- медиа форматы, отвечающие за видео и звук (например, .mp3, .mpg, .mkv, .avi и т. д.);

- рабочие файлы визуальных редакторов, и исходные коды приложений (например, .crd, .cpp, .h, .rb и т. д.);

- системные и исполняемые файлы приложений: .exe, .dll, .config, .db.

Учитывая данные ограничения, основное внимание уделим рассмотрению поддерживаемых текстовых форматов в процессе анализа структуры файлов с целью корректной организации поиска. Для построения эффективной масштабируемой системы кластеризации и индексации файлов необходимо предусмотреть перспективы расширения приложения, увеличения объема получаемых данных, возможность модификации без значительных изменений в коде системы и удобный порядок взаимодействия пользователя с системой.

Структура предлагаемой программной системы “CLASTFIND” содержит четыре основных модуля: “Интерфейс”, “Парсер”, “Обработчик” и “Поисковая машина”. Такое разделение позволяет вносить изменения в систему в любом из модулей, улучшая его работу, при сохраняющемся интерфейсе взаимодействия с пользователем. Схема передачи данных в системе приведена на рис. 10.1.



**Рис. 10.1. Схема передачи данных в системе «CLASTFIND»**

Модуль “Интерфейс” обеспечивает получение необходимой информации от пользователя и логику распределения задач в зависимости от текущего состояния системы. В этом модуле осуществляется взаимодействие с пользователем от получения ссылки на библиотеку классифицируемых данных до выдачи пользователю данных о принадлежности к определенному файлу его поискового запроса. Кроме того, он формирует сообщения об ошибках последовательности взаимодействия с системой и координирует выдачу релевантных результатов просмотра текстов.

Модуль “Парсер” осуществляет предварительную обработку документов библиотеки и последующий анализ текстов с целью составления полной карты распределения входных дескрипторов по кластерам. Данный модуль, получая адрес библиотеки текстов, анализирует содержимое текстовых массивов и формирует универсальный базовый блок-вектор, который является основой для дальнейших аналитических и алгоритмических преобразований и выводов. Универсальный блок-вектор включает в себя наименование слова и его частотная характеристика относительно анализируемого документа. Данный блок-вектор абсорбируется общим частотным словарем, который составляется из всех документов входящих в базовый кластер.

Модуль “Обработчик” осуществляет создание структуры кластеров, а также их индексацию, хранение и дальнейшее использование для

классификации входных запросов. Схема работы данного модуля представлена на рис. 10.2.



**Рис. 10.2. Схема работы модуля «Обработчик»**

Полученная из парсера информация поступает в модуль “Обработчик” для формирования кластеров и их последовательного индексирования. Под индексацией кластеров понимается присвоение кластеру уникального

номера и составление для него соответствующего блока векторных дескрипторов путем слияния векторов входящих в него документов.

Задача формирования кластеров связана, прежде всего, с нахождением топологической структуры распределения входных данных. Существует различные алгоритмы решения этой задачи. Например, алгоритм самоорганизующихся карт Кохонена является методом проецирования многомерного пространства в пространство с более низкой размерностью с предопределенной структурой. При этом возникают дефекты проецирования, анализ которых является сложной задачей. Одной из альтернатив данному подходу является гибридное применение методов конкурентного обучения Хебба и нейронного газа [90]. Однако для практической реализации такого подхода необходимы априорные знания о размере сети и адаптация скорости ее обучения. Алгоритм кластеризации документов по векторным дескрипторам должен удовлетворять следующим требованиям: обучение в режиме онлайн; разделение на классы без предварительных знаний о входных данных; разделение данных с нечеткой границей и выявление структуры кластеров.

В основу процедуры онлайн кластеризации, используемой в модуле “Обработчик”, положен алгоритм SOINN (Self-Organizing Incremental Neural Network), который частично решает отмеченные выше проблемы [109; 108]. SOINN представляет собой нейронную сеть с двумя слоями. Первый слой используется для определения топологической структуры кластеров, а второй для определения числа кластеров и выявления узлов-прототипов. Сначала обучается первый слой сети, а затем, по результатам этого обучения, обучается второй слой сети. Для задачи классификации без учителя необходимо определить, принадлежит ли входной образ одному из ранее полученных кластеров. Пусть два образа принадлежат одному кластеру, если расстояние (по заранее заданной метрике) между ними меньше порога расстояния  $T$ . Если  $T$  будет слишком большим, то все входные образы будут отнесены к одному кластеру. Если  $T$  слишком маленький, то каждый образ будет изолирован как отдельный кластер. Для получения разумного разбиения на кластеры,  $T$  должно быть больше внутриклассового расстояния и меньше межклассового. Для каждого из слоев сети SOINN необходимо вычислить порог  $T$ . Первый слой не имеет априорных знаний о структуре исходных данных, поэтому  $T$  подбирается адаптивно, с учетом знаний структуры уже построенной сети и текущего входного образа. При обучении второго слоя можно рассчитать внутриклассовое и межклассовое расстояния и подобрать постоянное значение порога  $T$ . Для представления топологической структуры в

онлайнном режиме обучения рост сети является важным фактором снижения ошибки и адаптации к меняющимся условиям при сохранении старых данных. В то же время, неконтролируемое увеличение числа узлов приводит к перегрузке сети и ее “переобучению” в целом. Для принятия решений о добавлении узлов, SOINN использует схему, при которой вставка происходит в регионе с максимальной ошибкой. При этом для оценки полезности такой вставки используется так называемый “радиус ошибки”. Применение этой оценки контролирует прирост узлов, и в конце концов стабилизирует их количество. Для составления связей между нейронами используется конкурентное правило Хебба, предложенное для определения топологий в сетях. В соответствии с этим правилом для каждого входного сигнала объединяются два ближайших узла. Доказано, что формируемый при этом граф оптимально представляет топологию входных данных. Узлы, которые являются соседними на ранней стадии, возможно, не будут близкими в более поздней стадии. При этом появляется необходимость удаления соединений, которые в последнее время не обновлялись. В общем случае происходит также перекрытие существующих кластеров. Чтобы определить количество кластеров точно, предполагается, что входные данные являются разделяемыми: плотности вероятности в центральной части каждого кластера выше, чем плотность в части между кластерами, а перекрытие кластеров имеет низкую плотность вероятности. Разделение кластеров происходит путем удаления узлов из регионов с низкой плотностью вероятности. Алгоритм SOINN предлагает следующую стратегию удаления узлов: если число сигналов, генерируемых до текущего шага, является целым кратным заданному параметру, то удалить те узлы, которые имеют только одного топологического соседа, или не имеют соседей вовсе и если локальный накопленный уровень сигнала является маленьким, то считаем такие узлы лежащими в области низкой плотности вероятности и удаляем их. Для решения задачи кластеризации текстовых документов была использована модификация базовой версии алгоритма (SOINN-index). В базовой версии в качестве меры расстояния выбирается евклидова метрика, а в модифицированной версии – косинусная метрика и мера Жаккара для оценивания расстояний между документами, представленными векторной дескрипторной моделью (1).

Приведем формальное описание алгоритма обучения как первого, так и второго слоев сети SOINN-index. Отметим, что входные данные для обучения второго слоя порождаются первым слоем и при обучении второго слоя

используются знания о топологической структуре первого слоя для вычисления постоянного порога подобия.

В алгоритме используются следующие обозначения:  $A$  – множество, используемое для хранения узлов;  $N_A$  – число узлов в множестве  $A$ ;  $C$  – множество связей между узлами;  $N_C$  – число ребер в  $C$ ;  $W_i$  –  $n$ -мерный вектор весов для узла  $i$ ;  $E_i$  – локальный аккумулятор ошибки для узла  $i$ ;  $M_i$  – локальный аккумулятор сигнала для узла  $i$ ;  $R_i$  – радиус ошибки для узла  $i$  ( $R_i = E_i / M_i$ );  $C_i$  – метка кластера для узла  $i$ ;  $Q$  – число кластеров;  $T_i$  – порог подобия для узла  $i$ ;  $N_i$  – набор топологических соседей для узла  $i$ ;  $\text{age}(i, j)$  – возраст связи между узлами  $i$  и  $j$ .

Алгоритм SOINN-index:

1. Инициализировать множество узлов  $A$  узлами  $c_1$  и  $c_2$ :  $A = \{c_1, c_2\}$ .

Инициализировать множество ребер  $C$  пустым множеством:  $C = \{\}$ .

2. Подать на вход новый сигнал  $x$  из  $R^n$ .

3. Найти в множестве  $A$  узлы победителя  $s_1$  и второго победителя  $s_2$ , как узлы с ближайшим и следующим за ним векторами весов (по некоторой заданной метрике). Если расстояние между  $x$ ,  $s_1$  и  $s_2$  больше порогов подобия  $T_{s_1}$  или  $T_{s_2}$ , то создать новый узел и перейти к шагу 2.

4. Если ребро между  $s_1$  и  $s_2$  отсутствует, то создать его и принять возраст ребра между ними равным 0.

5. Увеличить на 1 возраст всех дуг, выходящих из  $s_1$ .

6. Прибавить расстояние между входным сигналом и победителем к локальной суммарной ошибке  $E_{s_1}$ .

7. Увеличить локальное количество сигналов узла  $s_1$ :  $M_{s_1} = M_{s_1} + 1$ .

8. Адаптировать векторы весов победителя и его прямых топологических соседей:  $W_{s_1} = W_{s_1} + e_1(t)(x - W_{s_1})$ ,  $W_{s_i} = W_{s_i} + e_2(t)(x - W_{s_i})$ , где  $e_1(t)$  и  $e_2(t)$  – коэффициенты обучения победителя и его соседей.

9. Удалить ребра с возрастом, превышающим заданное пороговое значение.

10. Если число генерируемых до сих пор входных сигналов кратно параметру  $\lambda$ , вставить новый узел и удалить узлы в областях низкой плотности вероятности по следующим правилам:

10.1. Найти узел  $q$  с максимальной ошибкой.

10.2. Найти среди соседей  $q$  узел  $f$  с максимальной ошибкой.

10.3. Добавить узел  $r$  таким образом, чтобы его весовой вектор был средним арифметическим весовых коэффициентов  $q$  и  $f$ .

10.4. Рассчитать накопленную ошибку  $E_r$ , суммарное количество сигналов  $M_r$ , и наследованный радиус ошибки:  
$$= \alpha_1 * (E_q + E_f) M_r = \alpha_2 * (M_q + M_f) R_r = \alpha_3 * (R_q + R_f).$$

10.5. Уменьшить суммарную ошибку узлов  $q$  и  $r$ :  
$$q = \beta * E_q E_f = \beta * E_f.$$

10.6. Уменьшить накопленное количество сигналов:  
$$M_q = \gamma * M_q M_f = \gamma * M_f.$$

10.7. Определить, успешно ли прошла вставка нового узла. Если вставка не может уменьшить среднюю погрешность данной области, то добавленная вершина удаляется, а все параметры возвращаются в исходные состояния. Иначе обновляется радиус ошибки всех задействованных во вставке узлов:  
$$R_q = E_q / M_q R_f = E_f / M_f R_r = E_r / M_r.$$

10.8. Если вставка прошла успешно, создать связи  $q \leftrightarrow r$  и  $r \leftrightarrow f$  и удалить связь  $q \leftrightarrow f$ .

10.9. Среди всех узлов в  $A$  найти такие, которые имеют только одного соседа, затем сравнить накопленное количество входных сигналов со средним значением всех узлов. Если узел содержит всего одного соседа и счетчик сигналов не превышает адаптивного порога, удалить его из набора узлов. Например, если  $L_i = 1$  и  $M_i < c * \sum_{j=1 \dots N_A} M_j / N_A$  (где  $0 < c < 1$ ), то удалить вершину  $i$ .

10.10. Удалить все изолированные узлы.

10.11. Проверить условие останова алгоритма: если каждому классу из входных данных соответствует компонента связности в построенном графе, то обновить метки классов и завершить обучение; иначе перейти к пункту 2 и продолжить обучение.

Обобщенная схема алгоритма SOINN-index приведена на рис. 10.3.



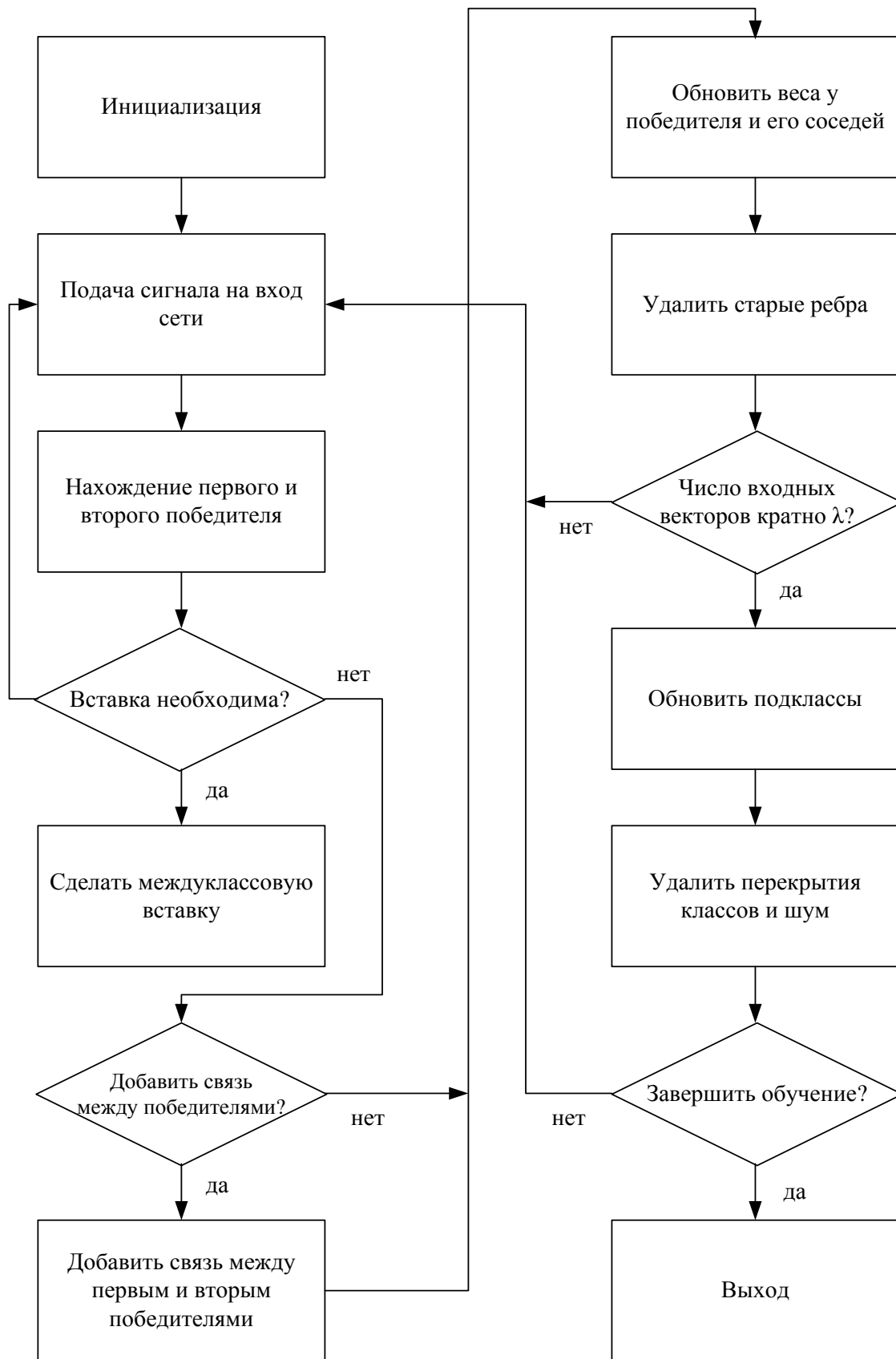


Рис. 10.3. Схема алгоритма SOINN-index

Рассмотрим процедуру вычисления порогового значения  $T$  для алгоритма SOINN-index.

Для первого слоя порог подобия  $T$  вычисляется адаптивно по следующему алгоритму:

1. Инициализировать порог подобия для новых узлов равным  $+\infty$ .

2. Когда узел является первым или вторым победителем, обновить значение порога подобия по следующему правилу: если узел имеет прямых топологических соседей ( $L_i > 0$ ), обновить значение  $T_i$  как максимальное расстояние между узлом и его соседями:  $T_i = \max \|W_i - W_j\|$ ; если узел не имеет соседей,  $T_i$  устанавливается как минимальное расстояние между узлом и другими узлами в множестве  $A$ :  $T_i = \min \|W_i - W_c\|$ .

Для второго слоя рассчитывается постоянный порог подобия:

1. Вычислить внутриклассовое расстояние следующим образом:  
 $dw = 1 / N_c * \sum \{(i, j) \text{ in } C\} \|W_i - W_j\|$ .

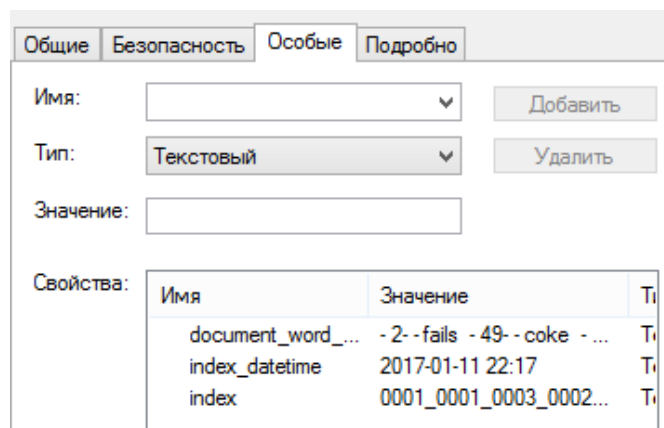
2. Вычислить межклассовые расстояния. Расстояние между классами  $C_i$  и  $C_j$  вычисляется следующим образом:  $db(C_i, C_j) = \max \{i \text{ in } C_i, j \text{ in } C_j\} \|W_i - W_j\|$

3. В качестве постоянного порога подобия  $T_c$  принять минимальное межклассовое расстояние, которое превышает внутриклассовое расстояние.

Модуль “Поисковая машина” выполняет предварительную обработку запросов пользователя, направление этих запросов на получение индекса в модуль “Обработчик” и вывод релевантных результатов в соответствии с индексом. Данный модуль частично использует результаты работы модулей “Парсер” и “Обработчик” с привнесением своей специфики в процессы обработки и анализа. Поисковый запрос сначала проходит предварительную обработку в модуле “Парсер”, затем кластеризуется модулем “Обработчик”, после чего из текущей иерархии кластеров формируется путем сложения индексов кластеров искомым индекс файла. По индексам файла производится поиск в индексной базе и формируется конечный путь к требуемым данным.

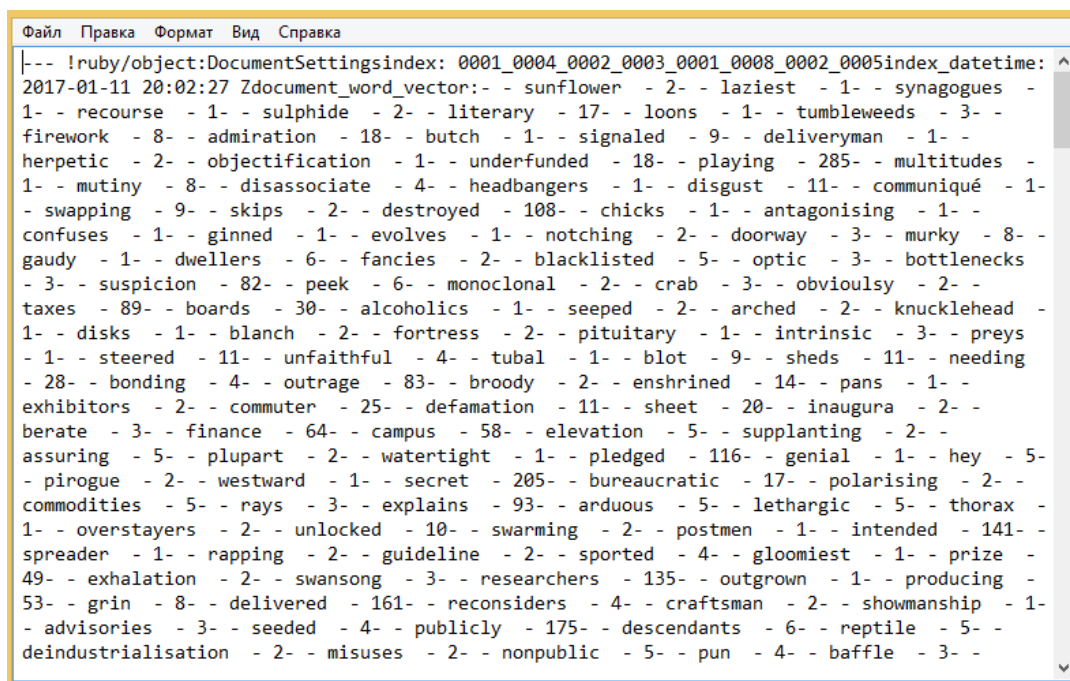
Поисковые индексы кластеров хранятся в базе в виде обратного поискового индекса с учетом контекста, что позволяет экономить пространство на диске за счет использования числового обозначения групп. Поскольку одному кластеру сразу ассоциирован полный набор файлов (с учетом вложенности), то это исключает необходимость проведения дополнительных операций над множествами в пределах базы. При необходимости актуализации

кластерного индекса для минимизации вычислительных нагрузок и поддержания высокого качества поиска в системе предусмотрен алгоритм актуализации. В случае изменения промежуточных кластеров модификация индексов происходит только в рамках измененной группы, что повышает общую стабильность системы. Данные о структуре и содержимом файла хранятся в свойствах метаданных файла в формате “.doc” (рис. 10.4).



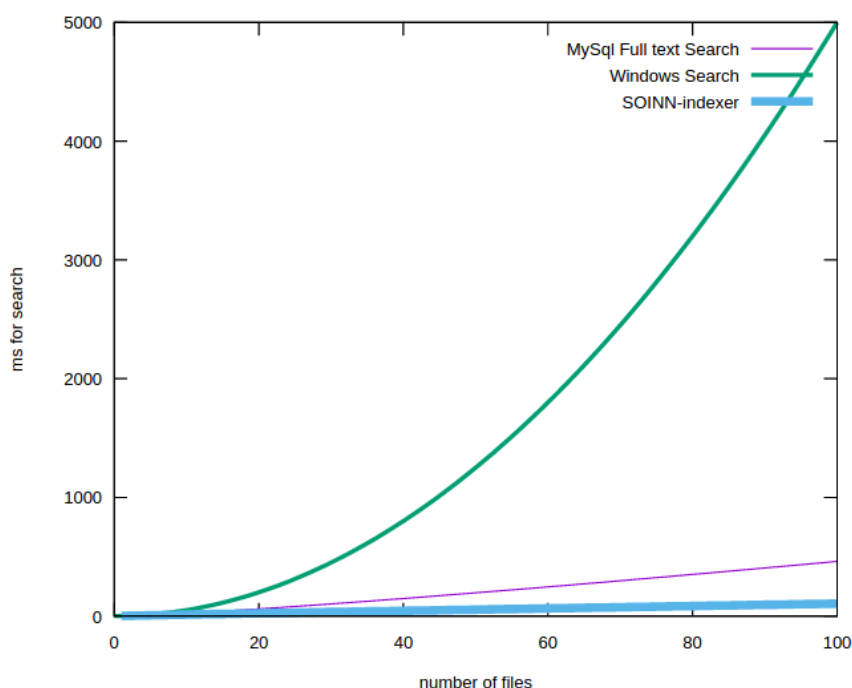
**Рис. 10.4. Структура метаданных файла в формате «.doc»**

Если в файле отсутствует возможность записи в блок метаданных, то метаданные записываются в первую строку файла в формате “.txt” (рис. 10.5).



**Рис. 10.5. Структура метаданных файла в формате «.txt»**

Результаты тестирования программной системы кластеризации и индексирования “CLASTFIND” с применением алгоритма SOINN-index, приведенные на рис. 10.6, позволяют сделать вывод, что предлагаемый подход является более эффективным, чем применение стандартного полнотекстового поиска, встроенного в операционную систему, и полнотекстового поиска MySQL. При проведении тестирования были следующие параметры алгоритма SOINN-index: Для проведения экспериментов, были выбраны следующие параметры алгоритма:  $\lambda = 100$ ,  $c = 1$ ,  $\alpha_1 = 0.16$ ,  $\alpha_2 = 0.25$ ,  $\alpha_3 = 0.25$ ,  $\beta = 0.67$ ,  $\gamma = 0.75$ .



**Рис. 10.6. Результаты тестирования программной системы «CLASTFIND»**

**Заключение.** Рассмотренный нейросетевой метод позволяет с применением лингвистических дескрипторов решить проблему кластеризации и индексации больших объемов электронной текстовой информации по пользовательским категориям, учитывая возможность распределенного хранения массивов информации. Метод является перспективным для классификации по тематическим рубрикам больших массивов электронных текстовых документов, имеющих политематический характер. Такими массивами могут быть, например, электронные сборники материалов конференции, содержащих аннотации и основной текст; массив документов большой электронной библиотеки, посвященных некоторому общему научному направлению, и т.п.

Структура предлагаемой программной системы “CLASTFIND” содержит четыре основных модуля: “Интерфейс”, “Парсер”, “Обработчик” и “Поисковая машина”. Такое разделение позволяет вносить изменения в систему в любом из модулей, улучшая его работу, при сохраняющемся интерфейсе взаимодействия с пользователем. Для решения задачи кластеризации текстовых документов была использована модификация базовой версии алгоритма SOINN (Self-Organizing Incremental Neural Network), который основан на построении нейронной сети с двумя слоями. Первый слой используется для определения топологической структуры кластеров, а второй для определения числа кластеров и выявления узлов-прототипов.

Результаты тестирования программной системы кластеризации и индексирования с применением алгоритма SOINN-index подтверждают целесообразность его использования при решении широкого класса задач политематической классификации электронных текстов. Перспективным развитием метода является проведение экспериментов по усовершенствованию процедуры поиска электронных документов путем модификации алгоритма индексации текстовых данных и хранения индексов.

# *Захист інформації в інформаційних та комунікаційних системах*

## РОЗДІЛ 11

### КОМПЛЕКСНІ СИСТЕМИ БЕЗПЕКИ КІБЕРНЕТИЧНОГО ПРОСТОРУ КІБЕРФІЗИЧНОЇ СИСТЕМИ НА ОСНОВІ КОНЦЕПЦІЇ “ОБ’ЄКТ – ЗАГРОЗА – ЗАХИСТ”

*Анотація.* Розглянуті комплексні системи безпеки кіберфізичної системи на рівні кібернетичного простору “ресурси – системи – процеси” на основі концепції “об’єкт – загроза – захист”.

*Ключові слова:* кіберфізична система, кібернетичний простір, інформаційні ресурси, інформаційні системи, інформаційні процеси, комплексна система безпеки, концепція “об’єкт – загроза – захист”.

*Abstract.* Complex security systems of a cyber-physical system were considered on the cyberspace level “sources – systems – processes” based on the concept “object – threat – protection”.

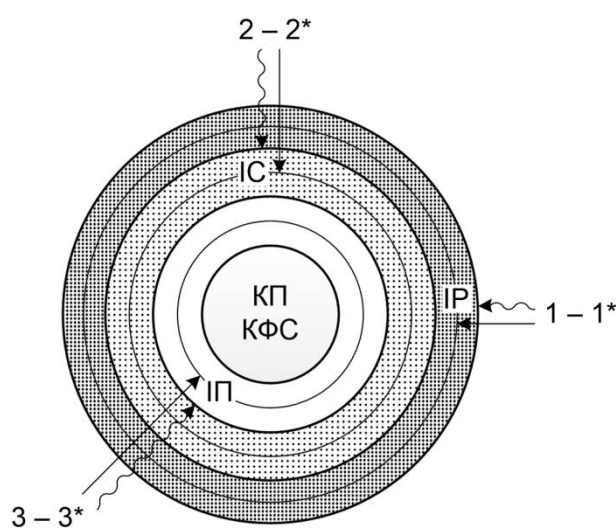
*Keywords:* cyber-physical system, cyberspace, information resources, information systems, information processes, complex security system, concept "object – threat – protection".

**Вступ і постановка задачі.** Засади та завдання забезпечення кібернетичної безпеки України [77; 76] сьогодні є стратегічним фундаментом для розроблення та впровадження комплексних систем захисту інформаційно-комунікаційних технологій (ІКТ). Кіберфізичні системи (КФС) є одним із сегментів ІКТ, особливо у розрізі забезпечення захисту об’єктів критичної інформаційної інфраструктури. Стратегічна безпека КФС має своєю особливістю нову модель захищеного обміну даних, зокрема інформації з обмеженим доступом (ІЗОД) – “багаторівнева система – багаторівневий захист”, що спрямована на забезпечення інтеграції рівнів КФС згідно концепції “об’єкт – загроза – захист”: кібернетичний простір (КП), комунікаційне середовище (КС), фізичний простір (ФП) – модель загроз – комплексні системи безпеки (КСБ). З метою забезпечення інтеграції рівнів КФС необхідним етапом в реалізації інформаційної безпеки є – створення КСБ кожного з рівнів зокрема. Кібернетичний простір КФС формується сегментами – інформаційні ресурси (ІР), інформаційні системи (ІС), інформаційні процеси (ІП) і є інструментарієм

функціонування інформації в просторі: контроль – обробка – управління. Кібернетичний простір КФС системно реалізує: відбір інформації з фізичного рівня КФС – давачів, вбудованих в різномірні об’єкти/ пристрої розумної інфраструктури; опрацювання та збереження інформації; прийняття рішення на управління об’єктами/ пристроями фізичного простору КФС.

### **1. Багаторівнева модель безпеки кібернетичного простору КФС на основі концепції “об’єкт – загроза – захист”**

Багаторівнева модель захисту інформації в кібернетичному просторі КФС представлена трьома рівнями комплексних систем безпеки: КСБ інформаційних ресурсів, КСБ інформаційних систем, інформаційних процесів (рис. 11.1).



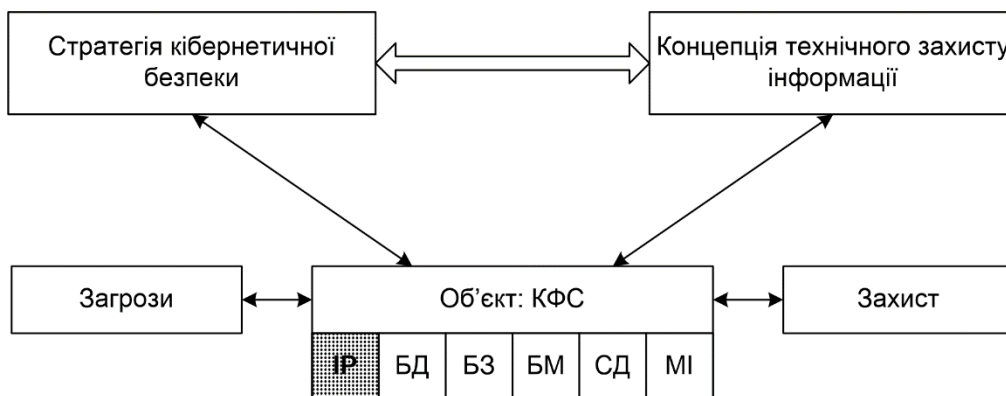
**Рис. 11.1. Багаторівнева модель КСБ кібернетичного простору КФС**

На рис. 11.1: 1 – рівень ІП (1 – загроза /1\* – захист); 2 – рівень ІС (2 – загроза/2\* – захист); 3 – рівень ІП (3 – загроза/ 3\* – захист). Кожний рівень представленої інформаційної моделі безпеки кібернетичного простору КФС має певні особливості, що відображаються комплексними системами безпеки відповідною структурою концепції “об’єкт: ІР/ ІС/ІП – загроза: апаратна, програмна/ випадкова, систематична/ випадкова, цілеспрямована – захист: апаратний, програмний”.

### **2. Комплексна система безпеки інформаційних ресурсів на основі концепції “об’єкт – загроза – захист”**

*Інформаційні ресурси* – один з рівнів інформаційних технологій, який пов’язаний із завданнями Стратегії кібернетичної безпеки та концепції технічного захисту інформації в контексті забезпечення функціональної та інформаційної безпеки згідно концепції “об’єкт – загроза – захист” (рис. 11.2).

За функціональним змістом ІР розрізняють як: бази даних (БД), бази знань (БЗ), бази моделей (БМ), сховища даних (СД), масиви інформації (МІ), а за рівнем доступу поділяються на: ресурси відкритого доступу; ресурси обмеженого доступу – державна таємниця; конфіденційна інформація (службова таємниця, комерційна таємниця, персональні дані).



*Рис. 11.2. Інформаційні ресурси: концепція “об’єкт – загроза – захист”*

**Комплексні системи безпеки баз даних та масивів інформації.**  
Розглянемо КСБ ІР згідно концепції “об’єкт – загроза – захист” на прикладі БД та МІ за впливу апаратному/ програмному рівні (табл. 11.1–11.2).

Таблиця 11.1

**КСБ ІР на рівні БД згідно концепції “об’єкт – загроза – захист”**

Бази даних			
Загрози		Захист	
Апаратні	Програмні	Апаратний	Програмний
1	2	3	4
– введення невірних даних; – знищення інформації і засобів її обробки; – розкрадання/ копіювання ІЗОД і засобів обробки; – пошкодження обладнання;	– відмова ПЗ; – зміна даних; – модифікація даних; – витік, порушення цілісності, справжності й збереженості інформації при її обробці;	– підвищення достовірності введення даних; – контроль та обмеження доступу; – реагування на атаки; – контроль звернень до захисних компонентів ІС;	– захист паролем; – шифрування даних/ програм; – захист полів та записів таблиць БД; – розділення прав доступу до об’єктів БД; – забезпечення цілісності зв’язків таблиць;



1	2	3	4
<ul style="list-style-type: none"> <li>– помилки операторів;</li> <li>– помилки при обслуговуванні;</li> <li>– НСД до адміністративної частини;</li> <li>– виведення з ладу підсистем забезпечення функціонування системи</li> </ul>	<ul style="list-style-type: none"> <li>– “чистка сміття” на диску або в оперативній пам’яті;</li> <li>– установка неперевічених виконуваних модулів і командних процедур;</li> <li>– створення або зміна записів БД захисту</li> </ul>	<ul style="list-style-type: none"> <li>– реагування при несанкціонованому доступі (НСД)</li> <li>– моніторинг активності в системі управління захистом;</li> <li>– реєстрація у журналі дій та операцій</li> </ul>	<ul style="list-style-type: none"> <li>– антивірусне ПЗ;</li> <li>– цифровий псевдонім;</li> <li>– запобігання створенню несанкціонованої інформації;</li> <li>– управління потоком захищених процедур/програм</li> </ul>

Таблиця 11.2

### КСБ ІР на рівні МІ згідно концепції “об’єкт – загроза – захист”

Масиви інформації			
Загрози		Захист	
Апаратні	Програмні	Апаратний	Програмний
<ul style="list-style-type: none"> <li>– втрата / знищення інформації;</li> <li>– НСД до ІзОД;</li> <li>– втрата носіїв інформації;</li> <li>– витік інформації під час зберігання/обробки даних;</li> <li>– помилки передавання;</li> <li>– перевантажений трафік;</li> <li>– несанкціоноване використання мережевих засобів;</li> <li>– збої послуг зв’язку /мережевих послуг</li> </ul>	<ul style="list-style-type: none"> <li>– перехоплення інформації під час пересилання/отримання;</li> <li>– модифікація інформації;</li> <li>– блокування інформації;</li> <li>– несанкціоноване видалення ІзОД;</li> <li>– хибна адреса відправлення повідомлень;</li> <li>– підміна довіреного об’єкта в мережі</li> <li>– віддалений запуск додатків</li> </ul>	<ul style="list-style-type: none"> <li>– перевірки стану роботи системи, її компонентів;</li> <li>– контроль доступу;</li> <li>– виявлення сторонніх засобів;</li> <li>– швидка реакція на процеси НСД;</li> <li>– відеоспостереження;</li> <li>– контроль процесу передавання;</li> <li>– захист обробки даних;</li> <li>– створення резервних копій;</li> <li>– міжмережевий екран</li> </ul>	<ul style="list-style-type: none"> <li>– шифрування передавання інформації;</li> <li>– захист доступу до даних;</li> <li>– розмежування доступу до об’єктів;</li> <li>– реєстрація всіх звернень до інформації, що підлягає захисту;</li> <li>– аутентифікація;</li> <li>– надійний спосіб зберігання даних;</li> <li>– антивірусне ПЗ;</li> <li>– алгоритми хешування</li> </ul>

### 3. Комплексна система безпеки інформаційних систем на основі концепції “об’єкт – загроза – захист”

**Функціональна та інформаційна безпека ІС.** Сегмент КП на рівні ІС представляють: інформаційно-аналітичні системи (ІАС), вимірювальні інформаційні системи (ВІС), автоматизовані системи управління (АСУ), системи підтримки прийняття рішень (СППР), експертні системи (ЕС). Враховуючи спільні функціональні особливості ІАС, ВІС, АСУ, СППР та ЕС, розглянемо узагальнену комплексну систему функціональної/інформаційної безпеки (ФБ/ІБ) ІС відповідно до випадкових /систематичних загроз. *Функціональна безпека* – спрямована на створення моделей захисту компонент кібернетичного простору кіберфізичних систем на основі методів і засобів апаратно-програмного забезпечення відповідно до ймовірних загроз: невизначеність; відмова; аварія. *Інформаційна безпека* – спрямована на створення моделей захисту компонент кібернетичного простору КФС на основі методів і засобів апаратно-прогамного забезпечення відповідно до ймовірних загроз: витік; модифікація; знищення. Комплексна система безпеки ІС, як компоненти КП КФС у контексті структури гарантоздатності охоплює ФБ та ІБ на апаратно-програмному рівні. Засадами створення КСБ ІС є: випадкові і систематичні загрози; методи та засоби протидії загрозам апаратного і програмного забезпечення ІС, що є підґрунтям забезпечення гарантоздатності КФС для різних функціональних задач промислової інфраструктури.

**Комплексна система безпеки ІС на апаратно-програмному рівні.** Розглянемо узагальнені загрози ФБ/ІБ інформаційних систем на апаратно-програмному рівні (табл. 11.3).

Таблиця 11.3

#### Загрози ФБ/ІБ інформаційних систем на апаратно-програмному рівні

Інформаційні системи: загрози на апаратно-програмному рівні	
<b>Загроза</b>	<p>1. <i>Об’єктивні</i> – залежать від особливостей побудови і технічних характеристик обладнання, застосовуваного в ІС.</p> <p>1.1. Перехоплення випромінювання технічних засобів: перехоплення побічного випромінювання елементів технічних засобів; перехоплення наведень електромагнітних випромінювань; перехоплення звукових коливань;</p> <p>1.2. Закладки: апаратні, програмні;</p> <p>1.3. Перехоплення сигналів електроакустичного перетворення;</p>

<b>Інформаційні системи: загрози на апаратно-програмному рівні</b>	
<b>Загроза</b>	<p>1.4. Перехоплення даних з каналів обміну інформацією;</p> <p>1.5. Незаконне підключення до ліній зв'язку та модифікація потоку даних від давача до комп'ютерної системи.</p> <p>2. <i>Суб'єктивні</i> – залежать від дій операторів/адміністраторів.</p> <p>2.1. Помилки при: підготовці та використанні ПЗ; управлінні складними системами; експлуатації технічних засобів; виконанні задач, пов'язаних з надсиланням хибних даних.</p> <p>2.2. Порушення режиму: експлуатації технічних засобів; використання інформації за життєвим циклом, конфіденційності, опитування давачів, відбору та модифікації алгоритму обробки даних з давачів; розкриття шифрів криптозахисту інформації.</p> <p>3. <i>Випадкові</i> – залежать від особливостей навколишнього середовища ІС та непередбачених обставин.</p> <p>3.1. Збої і відмови: відмови і несправності технічних засобів; старіння і розмагнічування носіїв інформації; збої ПЗ; збої електропостачання; помилкові спрацювання давачів; надходження хибних даних з давачів; стихійні лиха та аварії;</p> <p>3.2. Пошкодження: життєзабезпечувальних комунікацій; огорожувальних конструкцій.</p>

Розглянемо технології забезпечення ФБ інформаційних систем від впливу випадкових загроз на апаратний рівень (табл. 11.4).

Таблиця 11.4

### **Забезпечення ФБ ІС на апаратному рівні від впливу випадкових загроз**

<b>Методи і засоби забезпечення ФБ від випадкових загроз на апаратне забезпечення ІС</b>	
<b>Гаранто-здатність</b>	<p><i>Методи і засоби забезпечення ФБ від випадкових загроз на апаратне забезпечення</i></p> <p>1. <i>Електричні</i>: виявлення відмов в режимі online; моніторинг контактів реле; компаратор; схема голосування за мажоритарним принципом; відсутність електроживлення.</p> <p>2. <i>Електронні</i>: принципи динамічного управління; стандартний тестовий порт доступу і архітектура граничного сканування;</p>

<b>Методи і засоби забезпечення ФБ від випадкових загроз на апаратне забезпечення ІС</b>	
<b>Гаранто-здатність</b>	<p>надлишковий контроль; електричні / електронні компоненти з автоматичною перевіркою; контроль аналогових сигналів; зниження максимальних значень.</p> <p>3. <i>Модулі обробки</i>: програмне самотестування: гранична кількість комбінацій; самотестування забезпечуване обладнанням; запрограмована обробка; програмне виявлення неспівпадінь.</p> <p>4. <i>Постійна пам'ять</i>: збереження слів з багатобітною надлишковістю; модифіковувана контрольна сума; сигнатура з одного слова (8 біт); сигнатура з двох слів (16 біт); дублювання блоку.</p> <p>5. <i>Оперативний запам'ятовуючий пристрій (ОЗП)</i>: тести “шахова дошка” або “марш” для ОЗП; тест “блудлива траєкторія” для ОЗП; тест ОЗП “GALPAT”/ “Прозорий GALPAT”; біт парності для ОЗП; контроль ОЗП за допомогою модифікованого коду Хемінга/ виявлення збоїв даних за допомогою кодів виявлення і корекції помилок (EDC); дублювання в порівнянні ОЗП з апаратними/ програмними засобами і тестування читанням / записом.</p> <p>6. <i>Пристрої вводу / виводу та інтерфейси (зовнішній обмін)</i>: тестуюча комбінація; кодовий захист; багатоканальний паралельний вихідний пристрій; засіб контролю виходів; порівняння на вході.</p> <p>7. <i>Інформаційні канали (внутрішній обмін)</i>: однобітна апаратна надлишковість; багатобітна апаратна надлишковість; повна апаратна надлишковість; аналіз з використанням тестових комбінацій; надлишковість при передачі; інформаційна надлишковість.</p> <p>8. <i>Джерело живлення</i>: захист від стрибків напруги із захистом від короткого замикання або відключенням / підключенням до другого джерела живлення; контроль напруги (вторинний) з безпечним відключенням / підключенням до другого джерела живлення; відключення системи безпеки при зниженні живлення/ підключення до другого джерела живлення.</p>

<b>Методи і засоби забезпечення ФБ від випадкових загроз на апаратне забезпечення ІС</b>	
<b>Гаранто-здатність</b>	<p>9. <i>Послідовність виконання програм</i>: контрольний датчик часу з окремою тимчасовою базою без тимчасового вікна; контрольний датчик часу з окремою тимчасовою базою і тимчасовим вікном; логічний контроль послідовності виконання програм; комбінація тимчасового і логічного контролю послідовності виконання програм; початковий тест при включенні.</p> <p>10. <i>Вентиляція і температура</i>: давач температури; управління вентиляцією; безпечне вимкнення з використанням плавкого запобіжника; порогове повідомлення від термодавачів і умовна тривога; під'єднання пристрою охолодження повітря і індикатора стану.</p> <p>11. <i>Пристрій зв'язку та запам'ятовуючий пристрій великої ємності</i>: розділення ліній живлення і ліній передачі інформації; підвищення стійкості до електромагнітних впливів.</p> <p>12. <i>Давачі</i>: еталонний давач.; позитивно-керуючий перемикач.</p> <p>13. <i>Виконавчі елементи (приводи)</i>: моніторинг; перехресний контроль групових приводів.</p> <p>14. <i>Засоби проти фізичної взаємодії із зовнішнім середовищем</i>.</p>

Розглянемо технології забезпечення ФБ ІС від впливу випадкових та систематичних загроз на програмний рівень (табл. 11.5).

Таблиця 11.5

**Забезпечення ФБ ІС на програмному рівні від впливу випадкових та систематичних загроз**

<b>Методи і засоби забезпечення ФБ від випадкових та систематичних загроз на програмне забезпечення ІС</b>	
<b>Гаранто-здатність</b>	<p><i>Методи і засоби забезпечення ФБ на програмному рівні від випадкових і систематичних загроз</i></p> <p>1. <i>Вимоги і детальне проектування</i>: структурні методи; діаграми потоків даних; структурні діаграми; формальні методи; програмування з захистом; стандарти по проектуванню і кодуванню; структурне програмування; обмеження доступу/</p>

<b>Методи і засоби забезпечення ФБ від випадкових та систематичних загроз на програмне забезпечення ІС</b>	
<b>Гаранто- здатність</b>	<p>інкапсуляція інформації; модульний підхід; використання довірювальних/ провірених елементів ПЗ; прослідковуваність; проектування програмного забезпечення, не зберігаючого стан; чисельний аналіз в автономному режимі; діаграми послідовності повідомлень.</p> <p>2. <i>Архітектурне проектування</i>: аналіз і діагностика збоїв; коди виявлення і виправлення помилок; програмування з перевіркою помилок; методи контролю; багатоваріантне програмування; відновлення попереднього стану; механізми повторних спроб відбиття збоїв; поступове відключення функцій; виправлення помилок методами штучного інтелекту; динамічна реконфігурація; безпека і робота в жорсткому реальному часі; UML.</p> <p>3. <i>Інструменти розробки і мови програмування</i>: суворо типізовані мови програмування; підмножина мови; сертифіковані засоби і сертифіковані транслятори; інструментальні засоби і транслятори; вибір відповідної мови програмування; автоматична генерація ПЗ; управління тестуванням і засоби автоматизації.</p> <p>4. <i>Верифікація і модифікація</i>: імовірнісне тестування; реєстрація і аналіз даних; тестування інтерфейсу; аналіз граничних значень; введення помилок; розділення вхідних даних і класи еквівалентності; структурне тестування; аналіз потоків управління; аналіз потоків даних; тестування на символічному рівні; метрика важкості ПЗ; формальні перевірки; наскрізний контроль ПЗ; аналіз проекту; макетування / анімація; моделювання процесу; вимоги до реалізації та моделювання; перевірка на критичні загрузки; обмеження на час відповіді/ час пам'яті; управління конфігурацією ПЗ; регресивне підтвердження відповідності; анімація специфікації і проектування; тестування, що ґрунтується на моделі (генерація тестів).</p> <p>5. <i>Оцінка ФБ</i>: таблиця рішень; дослідження небезпеки і роботоздатності ПЗ; аналіз відмов; структурні схеми надійності.</p>

Розглянемо технології забезпечення ІБ ІС від впливу випадкових та систематичних загроз на апаратний і програмний рівень (табл. 11.6).

**Забезпечення ІБ ІС на апаратному і програмному рівні від впливу  
випадкових та систематичних загроз**

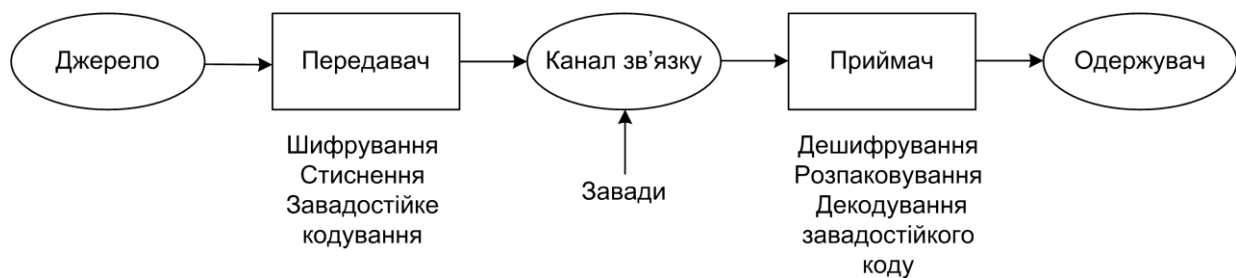
<b>Методи і засоби забезпечення ІБ від випадкових і систематичних загроз на апаратне та програмне забезпечення ІС</b>	
<b>Гаранто-здатність</b>	<p><b>Методи і засоби забезпечення ІБ – апаратне забезпечення</b></p> <ul style="list-style-type: none"> <li>• контроль доступу до підсистеми керування станцією з боку віддалених терміналів обслуговування;</li> <li>• очищення робочих областей пам'яті ЕОМ після завершення роботи з даними, що захищаються;</li> <li>• контроль цілісності програмної та інформаційної частин системи розмежування доступу;</li> <li>• виявлення позаштатних впливів на елементи устаткування, програми, дані і процеси;</li> <li>• реєстрація позаштатних впливів;</li> <li>• ідентифікація і розпізнавання процесів;</li> <li>• введення в дію засобів протидії непередбаченим змінам параметрів середовища експлуатації ІС;</li> <li>• керування засобами нейтралізації позаштатних впливів на параметри середовища експлуатації ІС;</li> <li>• виявлення закладних пристроїв/ реєстрація їх впливів/нейтралізація</li> <li>• виявлення побічного електромагнітного випромінювання і наведень (ПЕМВН) від елементів ІС;</li> <li>• виявлення електромагнітних наведень в елементах ІС;</li> <li>• послаблення/ екранування ПЕМВН від елементів ІС;</li> <li>• нейтралізація інформативних складових ПЕМВН;</li> <li>• нейтралізація електромагнітних наведень в елементах ІС від джерел інформативних випромінювань у зонах розміщення елементів ІС;</li> <li>• моніторинг з метою виявлення перевищень припустимих значень ПЕМВН;</li> <li>• тестування засобів підсистеми захисту від витоку через канали ПЕМВН.</li> <li>• виявлення спроб несанкціонованого впливу на інформацію.</li> </ul> <p><b>Методи і засоби забезпечення ІБ – програмне забезпечення</b></p> <ul style="list-style-type: none"> <li>• ізоляція програм, що виконуються в інтересах суб'єкта, від</li> </ul>

<b>Методи і засоби забезпечення ІБ від випадкових і систематичних загроз на апаратне та програмне забезпечення ІС</b>	
<b>Гаранто-здатність</b>	<p>інших суб'єктів;</p> <ul style="list-style-type: none"> <li>• керування потоками даних і команд з метою запобігання помилкових з'єднань/ помилкового надання послуг/ відмов в обслуговуванні ;</li> <li>• ідентифікація та аутентифікація суб'єктів;</li> <li>• реєстрація дій суб'єкта та його процесу;</li> <li>• надання можливостей вилучення /включення нових суб'єктів і об'єктів доступу, а також зміни повноважень суб'єктів;</li> <li>• шифрування інформаційних ресурсів;</li> <li>• контроль цілісності програмної та інформаційної частин засобів захисту від позаштатних впливів;</li> <li>• тестування засобів захисту від позаштатних впливів;</li> <li>• виявлення програмних закладок/ реєстрація їх впливів/ нейтралізація;</li> <li>• ідентифікація і виявлення моментів активізації інформаційно уразливих режимів/ функцій/ послуг;</li> <li>• сигналізація про активний стан інформаційно уразливих режимів/ функцій/ послуг;</li> <li>• придушення каналів витоку інформації в інформаційно уразливих режимах/ функціях/ послугах;</li> <li>• аварійне завершення активних процесів;</li> <li>• керування надлишковими ресурсами з метою протидії збоєм і відмовам у роботі ІС;</li> <li>• тестування засобів підсистеми захисту від збоїв і відмов;</li> <li>• виявлення спроб несанкціонованого впливу на інформацію;</li> <li>• реєстрація спроб впливу на інформацію;</li> <li>• реакція на спроби несанкціонованих впливів;</li> <li>• ідентифікація та аутентифікація суб'єктів доступу;</li> <li>• виявлення / реєстрація фактів реалізації загроз;</li> <li>• тестування засобів системи ліквідації наслідків загроз.</li> </ul>



#### 4. Комплексна система безпеки інформаційних процесів на основі концепції “об’єкт – загроза – захист”

**Класифікація інформаційних процесів та загроз.** Інформаційні процеси функціонують внаслідок передавання/приймання інформації від джерела до одержувача в структурі “система – канал – система” із застосуванням процедур безпечного обміну інформації в безпроводному/ провідному каналі зв’язку, що передбачає комплексний захист від перехоплення даних злоумисником та впливу комплексу завад (рис. 11.3).



**Рис. 11.3. Інформаційні процеси – схема передавання/ приймання інформації: джерело інформації – передавач – канал зв’язку – приймач**

Інформаційні процеси класифікуються як: 1) фази; 2) операції; 3) обробка. Розглянемо класифікацію ІП на рівні: 1) фаз – 1.1. сприйняття / збір / відбір; 1.2. передавання; 1.3. обробка; 1.4. зберігання; 1.5. представлення / вплив; 2) операцій – 2.1. збір / відбір даних; 2.2. формалізація даних; 2.3. фільтрація даних; 2.4. сортування даних; 2.5. групування даних; 2.6. архівування даних; 2.7. захист даних; 2.8. транспортування даних; 2.9. перетворення даних; 3) обробки – 3.1. автоматизована обробка; 3.2. в режимі реального часу; 3.3. неупорядкована; 3.4. послідовна; 3.5. стиснення даних; 3.6. хмарні обчислення.

**Загрози інформаційним процесам** – сукупність навмисних і не навмисних дій, що можуть призвести до витоку, модифікації чи знищення інформації. Загрози розділяють на випадкові і цілеспрямовані. *Випадковими* загрозами є дії, які здійснюються персоналом/ користувачами із-за неухважності/ недбалості/ незнання, але без навмисного наміру. До комплексу випадкових загроз відносяться: відмови і збої апаратури; завади в лініях зв’язку від впливів зовнішніх факторів; помилки людини, як складової системи; схемні і системотехнічні помилки проєктантів; структурні, алгоритмічні і програмні

помилки; аварійні ситуації та інші впливи. *Цілеспрямовані* загрози пов'язані з діями людини (причинами яких можуть бути певні невдоволення життєвою ситуацією/ матеріальна зацікавленість/ проста розвага т. і.). До комплексу цілеспрямованих загроз відносяться: порушення режимів функціонування систем життєзабезпечення; впровадження і використання комп'ютерних вірусів, закладних (апаратних і програмних) і прослуховуючих пристроїв, інших засобів розвідки; використання засобів перехоплення ПЕМВН; використання персоналу з корисливою метою; крадіжки носіїв інформації; несанкціоноване копіювання носіїв інформації; зчитування залишкової інформації з оперативної пам'яті ЕОМ, зовнішніх накопичувачів.

***Комплексні системи безпеки ІІ на рівні операцій та обробки.***

Розглянемо КСБ інформаційних процесів згідно концепції “об’єкт – загроза – захист” на прикладі операцій та обробки. В табл. 11.7 представлена комплексна система безпеки ІІ на рівні операцій. У контексті безпечного передавання/приймання інформації з фізичного простору КФС в кібернетичний простір для повноінформативного забезпечення прийняття управлінського рішення важливим є *збір/ відбір даних* – як операція ІІ, яка полягає в отриманні інформаційними службами повідомлень усіх видів по різних каналах зв’язку.

Таблиця 11.7

**Комплексна система безпеки ІІ на рівні операцій**

Об’єкт: ІІ	Загрози		Захист інформації	
	Цілеспрямовані	Випадкові	Апаратний	Програмний
2. Операції				
1	2	3	4	5
2.1. Збір / відбір даних	<ul style="list-style-type: none"> <li>• модифікація даних;</li> <li>• підміна даних;</li> <li>• перехоплення даних</li> </ul>	<ul style="list-style-type: none"> <li>• збої / нестабільність роботи технічних пристроїв;</li> <li>• стихійні лиха;</li> <li>• відмова систем живлення;</li> <li>• помилки людини;</li> <li>• завади в лініях зв’язку від зовнішніх впливів</li> </ul>	<ol style="list-style-type: none"> <li>1. Secure IDE</li> <li>2. Guardant</li> <li>3. "EXPERT"</li> <li>4. SenseLock</li> </ol>	<ol style="list-style-type: none"> <li>1. PPTP</li> <li>2. StarForce</li> <li>3. Adobe Digital Editions</li> <li>4. Fairplay</li> <li>5. StarForce</li> </ol>

1	2	3	4	5
2.2. Формалізація даних	<ul style="list-style-type: none"> <li>• порушення конфіденційності інформації;</li> <li>• порушення цілісності інформації;</li> <li>• втрата інформації</li> </ul>	<ul style="list-style-type: none"> <li>• некоректні дії експертів;</li> <li>• НСД до інформації;</li> <li>• вплив електромагнітного випромінювання;</li> <li>• стихійні лиха</li> </ul>	<ol style="list-style-type: none"> <li>1. Luna G5</li> <li>2. Luna PCI</li> <li>3. Protect-Server Gold</li> <li>4. nShield Solo</li> <li>5. KOKON-R</li> <li>6. M-590</li> </ol>	<ol style="list-style-type: none"> <li>1. Crypto Composer</li> <li>2. R-Crypto Disk Security</li> <li>3. Secret Disk Server NG</li> <li>4. InfoWatch</li> <li>5. Encrypt Easy</li> </ol>
2.3. Фільтрація даних	<ul style="list-style-type: none"> <li>• передавання по мережі шкідливих програм;</li> <li>• віддалений запуск додатків</li> </ul>	<ul style="list-style-type: none"> <li>• витік, порушення цілісності/ достовірності/ збереженості інформації при її фільтрації</li> <li>• технічні несправності фільтрів</li> </ul>	<ol style="list-style-type: none"> <li>1. KOKON-R</li> <li>2. M-575</li> <li>3. Protect-Server Gold</li> <li>4. “Бар’єр-301”</li> <li>5. Luna CA4</li> <li>6. nShield</li> </ol>	<ol style="list-style-type: none"> <li>1. TrueCrypt</li> <li>2. R-Crypto Disk Security</li> <li>3. Secret Disk Server NG</li> <li>4. BestCrypt Volume Encryption</li> </ol>
2.5. Групування даних	<ul style="list-style-type: none"> <li>• виявлення паролів;</li> <li>• перехоплення сигналу;</li> <li>• аналіз назв груп</li> </ul>	<ul style="list-style-type: none"> <li>• відмова систем живлення;</li> <li>• збої чи нестабільність роботи технічних пристроїв;</li> </ul>	<ol style="list-style-type: none"> <li>1. Luna CA4</li> <li>2. Luna G5</li> <li>3. Protect-Server Gold</li> <li>4. M-590</li> </ol>	<ol style="list-style-type: none"> <li>1. Info-Watch Crypto-Storage</li> <li>2. КриптоПро</li> <li>3. Secret Disk Server NG</li> <li>4. R-Crypto</li> </ol>
2.6. Архівування даних	<ul style="list-style-type: none"> <li>• перехоплення інформації</li> <li>• ініціювання хибних знань в архівах;</li> <li>• втрата інформації</li> </ul>	<ul style="list-style-type: none"> <li>• НСД до системи захисту;</li> <li>• порушення цілісності сигналу;</li> <li>• зміна змісту архівної інформації</li> </ul>	<ol style="list-style-type: none"> <li>1. Гряди-61</li> <li>2. KOKON-M</li> <li>3. nShield Connect</li> <li>4. nShield Solo</li> <li>5. Luna PCI</li> </ol>	<ol style="list-style-type: none"> <li>1. TrueCrypt</li> <li>2. R-Crypto Disk Security</li> <li>3. Secret Disk Server NG</li> <li>4. Encrypt Easy</li> </ol>

1	2	3	4	5
2.7. Захист даних	<ul style="list-style-type: none"> <li>• навмисне виведення з ладу системи захисту інформації</li> </ul>	<ul style="list-style-type: none"> <li>• створення або зміна записів бази даних захисту;</li> <li>• помилки операторів;</li> <li>• некоректні дії експертів</li> </ul>	<ol style="list-style-type: none"> <li>1. EPOS eFlash</li> <li>2. "EXPERT"</li> <li>3. Luna XML</li> <li>4. Protect-Server Gold</li> </ol>	<ol style="list-style-type: none"> <li>1. Safe 2.10</li> <li>2. Snort</li> <li>3. Thales</li> <li>4. Cisco UVPN-ZAS</li> </ol>
2.8. Транспортування даних	<ul style="list-style-type: none"> <li>• отримання віддаленого НСД;</li> <li>• затримання повідомлення;</li> </ul>	<ul style="list-style-type: none"> <li>• відмова систем живлення;</li> <li>• помилки людини;</li> <li>• завади в лініях зв'язку від впливів зовнішніх факторів</li> </ul>	<ol style="list-style-type: none"> <li>1. КриптоПро HSM</li> <li>2. Атлікс hsm</li> <li>3. Luna SA</li> <li>4. Канал-301</li> </ol>	<ol style="list-style-type: none"> <li>1. Silent Phone</li> <li>2. PPTP</li> <li>3. Whisper-Core</li> <li>4. Encrypt Easy</li> </ol>
2.8. Транспортування даних	<ul style="list-style-type: none"> <li>• порушення цілісності/ конфіденційності інформації;</li> </ul>		<ol style="list-style-type: none"> <li>5. nShield Connect</li> <li>6. Гряда-301</li> <li>7. Luna XML</li> </ol>	<ol style="list-style-type: none"> <li>5. S2end</li> <li>6. Secret Disk Server NG</li> </ol>
2.9. Перетворення даних	<ul style="list-style-type: none"> <li>• модифікація даних;</li> <li>• блокування інформації;</li> <li>• втрата інформації</li> </ul>	<ul style="list-style-type: none"> <li>• перешкоджання коректному функціонуванню елементів управління;</li> <li>• установка неперевірених виконуваних модулів і командних процедур</li> </ul>	<ol style="list-style-type: none"> <li>1. M-590</li> <li>2. M-575</li> <li>3. Secure IDE</li> <li>4. Luna CA4</li> <li>5. Luna PCM</li> <li>6. Protect-Server External</li> </ol>	<ol style="list-style-type: none"> <li>1. S2end</li> <li>2. M-506A</li> <li>3. Encrypt Easy</li> <li>4. Thales</li> <li>5. Crypto Composer</li> </ol>

В табл. 11.8 представлена комплексна система безпеки ІІ на рівні обробки згідно концепції “об’єкт – загроза – захист”. В контексті забезпечення інтеграції рівнів КФС у сегменті ІІ актуальними є хмарні обчислення. *Хмарні обчислення (модель)* – це модель забезпечення повсюдного та зручного доступу на вимогу через мережу до спільного пулу обчислювальних ресурсів, що підлягають налаштуванню (наприклад, до комунікаційних мереж, серверів, засобів збереження даних, прикладних програм та сервісів), і які можуть бути оперативно надані та звільнені з мінімальними управлінськими затратами та

зверненнями до провайдера. *Хмарні обчислення* (технологія) – технологія обробки даних, в якій ПЗ надається користувачеві як інтернет-сервіс.

Таблиця 11.8

**Комплексна система безпеки ІІІ на рівні обробки**

Об'єкт: ІІІ	Загрози		Захист інформації	
3. Обробка	Цілеспрямовані	Випадкові	Апаратний	Програмний
1	2	3	4	5
3.1. Авто-магізована	<ul style="list-style-type: none"> <li>вірусні атаки;</li> <li>неправомірна відмова в наданні послуг;</li> </ul>	<ul style="list-style-type: none"> <li>збої чи нестабільність роботи технічних пристроїв;</li> </ul>	<ol style="list-style-type: none"> <li>Крипто-Про HSM</li> <li>Атлікс hsm</li> <li>Luna SA</li> </ol>	<ol style="list-style-type: none"> <li>TrueCrypt</li> <li>М-506А-ХР</li> <li>Крипто-</li> </ol>
3.1. Авто-магізована	<ul style="list-style-type: none"> <li>несанкціоноване встановлення зв'язку</li> </ul>	<ul style="list-style-type: none"> <li>стихійні лиха;</li> <li>відмова систем. живлення;</li> <li>помилки людини</li> </ul>	<ol style="list-style-type: none"> <li>Luna SP</li> <li>Luna XML</li> </ol>	<ol style="list-style-type: none"> <li>Про CSP 3.6</li> <li>Encrypt Easy</li> <li>Thales</li> </ol>
3.2. В режимі реального часу	<ul style="list-style-type: none"> <li>порушення цілісності інформації;</li> <li>порушення конфіденційності</li> </ul>	<ul style="list-style-type: none"> <li>технічні несправності мережі і компонентів;</li> <li>пошкодження ліній</li> </ul>	<ol style="list-style-type: none"> <li>nShield Connect</li> <li>Грядя-301</li> <li>Бар'єр-301</li> <li>Канал-301</li> </ol>	<ol style="list-style-type: none"> <li>Secret Disk Server NG</li> <li>Secure Pack Rus</li> <li>Silent Ph</li> <li>S2end</li> </ol>
3.3. Невпорядкована	<ul style="list-style-type: none"> <li>зміна змісту переданої інформації;</li> <li>зміна маршруту напряму повідомлень</li> </ul>	<ul style="list-style-type: none"> <li>помилки передавання;</li> <li>витік інформації при випуску документів з результатуючими даними</li> </ul>	<ol style="list-style-type: none"> <li>Protect-Server External</li> <li>Luna G5</li> <li>KOKON-R</li> <li>М-590</li> </ol>	<ol style="list-style-type: none"> <li>ЛОЗА-2</li> <li>"Криптосервер"</li> <li>Deskman</li> <li>Kerio Ctrl</li> <li>Spyware Terminator</li> </ol>
3.4. Послідовна	<ul style="list-style-type: none"> <li>витік, порушення цілісності, дійсності та збереженості інформації при її обробці;</li> <li>злам криптошифрів</li> </ul>	<ul style="list-style-type: none"> <li>використання мережевих засобів несанкціонованим способом;</li> <li>перешкоджання коректному функціонуванню елементів управління</li> </ul>	<ol style="list-style-type: none"> <li>М-575</li> <li>Secure IDE</li> <li>Luna CA4</li> <li>Luna PCM</li> </ol>	<ol style="list-style-type: none"> <li>KeePass Password</li> <li>Safe 2.10</li> <li>Snort</li> <li>WinSCP</li> <li>Cisco UVPN-ZAS</li> </ol>

1	2	3	4	5
3.5. Стиснення даних	<ul style="list-style-type: none"> <li>• пошкодження вторинних знань;</li> <li>• помилки стиснення;</li> <li>• зміна змісту стисненої інформації</li> </ul>	<ul style="list-style-type: none"> <li>• алгоритмічні помилки при проектуванні;</li> <li>• установка неперевіраних виконуваних модулів і командних процедур</li> </ul>	<ol style="list-style-type: none"> <li>1. “Кристал-1”</li> <li>2. “Алмаз-1К”</li> <li>3. CRYPTOIP-448</li> <li>4. EPOS eFlash</li> <li>5. “EXPERT”</li> </ol>	<ol style="list-style-type: none"> <li>1. My Lockbox 8.7</li> <li>2. Passwords Keeper 1.8</li> <li>3. Encoding Decoding Free</li> <li>4. AxCrypt 1.7</li> <li>5. S-Tools 4.0</li> </ol>
3.6. Хмарні обчислення	<ul style="list-style-type: none"> <li>• порушення цілісності інформації;</li> <li>• порушення конфіденційності інформації</li> <li>• втрата інформації на сервері</li> </ul>	<ul style="list-style-type: none"> <li>• некоректні дії адміністраторів;</li> <li>• збої/ нестабільність роботи технічних пристроїв серверів</li> </ul>	<ol style="list-style-type: none"> <li>1. M-484</li> <li>2. ASA 5512-X</li> <li>3. “Плазма-3В”</li> <li>4. “ЗАРМ-24-2”</li> </ol>	<ol style="list-style-type: none"> <li>1. TLS</li> <li>2. PPTP</li> <li>3. SSL</li> <li>4. S/MIME</li> <li>5. Ipsec</li> </ol>

**Висновок.** Представлено багаторівневу модель захисту інформації в КП КФС. Розглянуто комплексні системи безпеки IP від цілеспрямованих/ випадкових загроз у сегменті апаратного/ програмного захисту даних на основі концепції “об’єкт – загроза – захист”, які спрямовані на забезпечення цілісності, достовірності і конфіденційності інформації в кібернетичному просторі.

На основі концепції “об’єкт – загроза – захист” розглянуто комплексні системи функціональної та інформаційної безпеки для ІС, як методи і засоби протидії випадковим/ систематичним загрозам на апаратному/ програмному рівнях, що є підставою для забезпечення гарантоздатності КФС на рівні структури “невизначеність – відмова – аварія: витік – модифікація – втрата”.

Розглянуто КСБ інформаційних процесів на рівні операцій та обробки згідно концепції “об’єкт – загроза – захист” відповідно до імовірних випадкових/ цілеспрямованих загроз, що є підґрунтям забезпечення інтеграції рівнів кіберфізичної системи.

## РОЗДІЛ 12

# ПІДХІД ДО ВИБОРУ ЗАСОБІВ ЗАХИСТУ З УРАХУВАННЯМ ЇХ ОЦІНОК

***Анотація.** В даній роботі запропонований підхід до вибору засобів захисту інформації. Дається короткий огляд вибору засобів захисту, та як і коли концепція базової безпеки може бути при цьому використана. Є два головних підходи до вибору засобів захисту, а саме: використання базового підходу та проведення детального дослідження ризиків.*

*Наведені основні компоненти для визначення типу інформаційної системи. Проаналізовано поняття фізичної безпеки інформаційної системи. Показано в чому полягає розробка базової безпеки організації, як проводиться вибір засобів захисту відповідно до детальних оцінок, а також що містить у собі процес планування безперебійної роботи.*

***Ключові слова:** захист інформації, засоби захисту інформації, вибір засобів захисту, оцінка засобів захисту.*

***Abstract.** In this paper, the proposed approach to the selection of information security. We give a brief overview of the choice of remedies and how and when basic safety concept can be used at the same time. There are two main approaches to the choice of remedies, namely the use of base approach and a detailed study of the risks.*

*The basic components to determine the type of information system. Analyzed the concept of physical security information system. What is shown in the development of basic security organizations such as the selection of remedies under detailed ratings and that includes planning process continuity.*

***Keywords:** information security, data protection, choice of remedies, rating remedies.*

**Введення та постановка задачі.** Настанову з вибору засобів захисту застосовують тоді, коли приймають рішення про вибір засобів захисту інформаційної системи:

- відповідно до типу і характеристик інформаційної системи;
- відповідно до загального оцінювання загроз та наявних потреб безпеки;
- відповідно до результатів детального аналізування ризиків.

Перехресні посилання надані для того, щоб показати, де вибір засобів захисту може бути підтриманий використанням загально доступних довідників, що містять засоби захисту (ДСТУ ISO/IEC TR 13335-4:2005).

Дається короткий огляд вибору засобів захисту, та як і коли концепція базової безпеки може бути при цьому використана. Є два головних підходи до вибору засобів захисту, а саме: використання базового підходу та проведення детального дослідження ризиків.

Проведення детального аналізу ризику має ту перевагу, що досягається повна картина ризиків. Це необхідно для вибору тих засобів захисту, що зумовлені ризиками, і, відповідно, мають бути реалізовані. Це запобігає використанню занадто великого чи занадто малого захисту. Оскільки цей підхід може вимагати значної кількості часу, зусиль та кваліфікації, він найбільш підходить для інформаційних систем з високим ризиком, тоді як простіший підхід може виявитись достатнім для систем з низьким рівнем ризику. Використання високорівневого аналізування ризиків може визначити системи з низьким рівнем ризику. Цей високорівневий аналіз ризиків не потребує формалізованого чи складного процесу. Засоби захисту для систем з низьким рівнем ризику можуть бути обрані шляхом застосування базової безпеки. Базова безпека забезпечує мінімальний рівень безпеки, визначений організацією для кожного типу інформаційної технології системи. Цей рівень базової безпеки досягається реалізацією мінімального набору засобів захисту, що відомі як базові засоби.

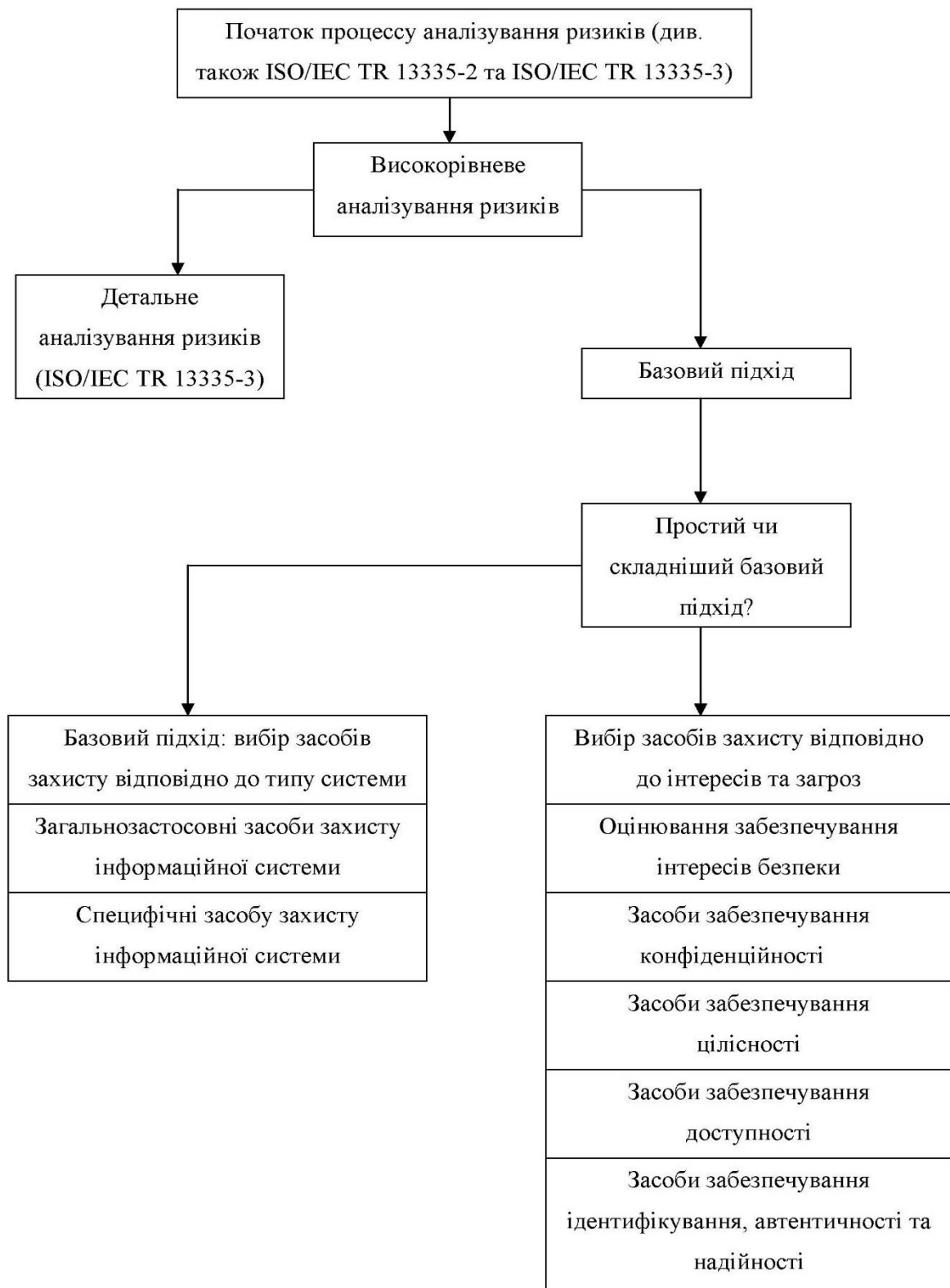
Внаслідок відмінностей в процесі вибору засобів захисту, в цьому розділі розглядаються два різні шляхи застосування базового підходу:

- використання базового підходу, в якому засоби безпеки рекомендовано вибирати відповідно до типу та характеристик інформаційної технології розглядуваної системи;
- використання базового підходу, в якому засоби безпеки рекомендовано вибирати відповідно до проблем та загроз безпеки з урахуванням системи, що розглядається.

На рис. 12.1 розглянуто способи вибору засобів захисту.

Базовий підхід треба вибирати відповідно до ресурсів, які можуть бути витрачені у процесі вибору усвідомлених проблем безпеки, а також типу і характеристик інформаційної системи, що розглядається. Якщо організація не бажає витратити багато часу та зусиль на вибір засобів захисту (з будь-якої причини), то можна скористатися базовим підходом, що пропонує засоби захисту без подальшого оцінювання. Однак, якщо ділові процеси організації до певної міри залежать від інформаційної системи чи послуг та (або) оброблювана інформація контрольована, то дуже ймовірно, що будуть необхідні додаткові засоби захисту.





**Рис. 12.1. Способи вибору засобів захисту**

В цьому випадку настійно рекомендується, проводити високорівневий огляд важливості інформації та можливих загроз для того, щоб мати краще уявлення про засоби захисту, потрібні для найефективнішого захисту інформаційної системи. Якщо ділові процеси організації сильно залежать від

інформаційної системи чи послуг, та (або) оброблена інформація є дуже чутливою, ризики можуть бути високі, і детальне аналізування ризиків є найкращим шляхом визначення прийнятних засобів захисту.

Специфічні засоби захисту повинні призначатися на основі детального аналізування ризиків, якщо:

- тип інформаційної системи, що розглядається, не відповідає типам, описаним у цьому стандарті;
- діяльність чи потреби безпеки не відповідають рішенням, запропонованим у цих розділах;
- детальніше оцінювання є виправданим у разі потенційно високих ризиків, чи важливості інформаційної системи для діяльності організації.

Треба зазначити, що навіть, коли виконано детальне аналізування ризиків, все ще доцільно застосувати до системи базові засоби захисту.

Перше рішення, яке повинна ухвалити організація – чи використовувати базовий підхід сам по собі, чи як частину більш повної стратегії аналізування ризиків. У разі прийняття цього рішення треба зазначити, що під час використання базового підходу самого по собі, результатний процес вибору засобів захисту може дати менш оптимізовану безпеку, ніж прийнята ширша стратегія аналізування ризиків. Однак, менші кошти та ресурси, необхідні під час вибору засобів забезпечення безпеки, та досягнення принаймні мінімального рівня безпеки для всіх інформаційних систем можуть бути причинами для прийняття рішення про використання тільки базового підходу.

Базовий захист для інформаційної системи може бути досягнутий через визначення та застосування набору відповідних засобів захисту, що є прийнятним за обставин наявності низького ризику, тобто вони задовольняють, принаймні, мінімальні потреби безпеки. Наприклад, прийнятні засоби захисту можуть бути визначені через каталоги, що містять набори засобів захисту безпеки від більшості загальних загроз для різних типів інформаційних технологій. Ці каталоги засобів захисту містять інформацію про категорії засобів захисту чи про окремі засоби, але загалом не зазначають, які засоби захисту треба застосовувати в конкретних обставинах. Можливо, якщо інформаційні системи організації (чи частини організації) є дуже схожі за природою та послугами, які вони надають, засоби захисту, вибрані за базовим підходом, можуть бути застосовані до всіх систем інформаційних технологій. На рис. 12.2 показано різні способи використання базового підходу.

Якщо організація вирішує впровадити базову безпеку до організації в цілому або її підрозділів, необхідно вирішити, для яких підрозділів організації прийнятні засоби захисту, і який рівень безпеки повинен забезпечувати цей захист. У більшості випадків, коли використовують базову безпеку, не застосовують менший рівень безпеки, доки не будуть реалізовані додаткові засоби захисту, обґрунтовані та необхідні для керування середніми та великими ризиками. Як альтернатива, базова безпека може визначити середній рівень для організації, тобто дозволяються винятки вище і нижче базового рівня, якщо вони були обґрунтовані, наприклад, результатами аналізування ризиків. Однією з переваг базової безпеки є те, що її застосовують до груп інформаційних систем, і всюди в цій групі можна покладатися на визначений рівень безпеки. В цих умовах зазвичай найкориснішим є розробити і вести базовий каталог засобів захисту в межах організації чи відділу.

**Основна частина.** Процес вибору засобів захисту завжди потребує певного знання типу та характеристик інформаційної системи, що розглядається, (наприклад, окрема робоча станція або робоча станція, включена до мережі), оскільки це має суттєвий вплив на вибрані засоби захисту. Корисно мати інфраструктури, які містять будівлі, кімнати тощо. Інший важливий чинник, пов'язаний із вибором засобів захисту – це оцінювання наявних та (або) запланованих засобів захисту. Це звільнює від зайвої роботи та марнування часу, зусиль та коштів. Тому настійно рекомендується оцінювання використовувати як основу для вибору засобів захисту. Коли вибирають засоби захисту, треба брати до уваги вимоги бізнесу та підхід організації до безпеки. Нарешті, необхідно визначити, чи надають ці результати оцінювання достатньо інформації для вибору базових засобів захисту, чи необхідне детальніше оцінювання або детальне аналізування ризиків (рис. 12.2).

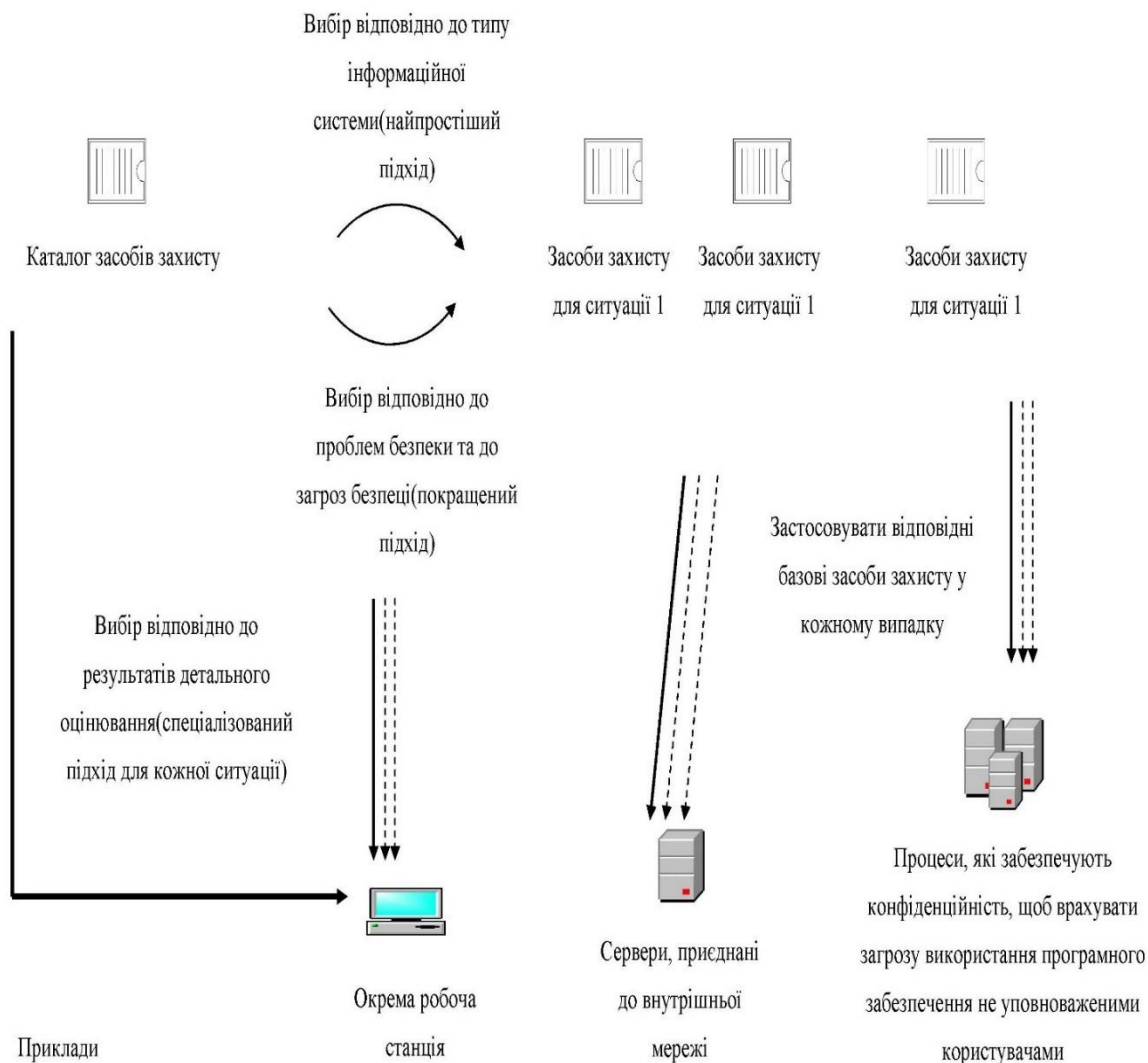
#### Визначення типу інформаційної системи

Для оцінювання наявної чи запланованої інформаційної системи її треба порівняти з нижченаведеними компонентами та визначити складники, визначальні для системи. Засоби захисту, що пропонуються для кожного з перерахованих нижче компонентів. Компоненти для вибору такі:

- окрема робоча станція;
- робоча станція (клієнт без спільних ресурсів), підключена до мережі;
- сервер чи робоча станція зі спільними ресурсами, підключена до мережі.

#### Визначення фізичних умов та умов навколишнього середовища

Оцінювання оточення охоплює визначання фізичної інфраструктури, що підтримує наявну чи заплановану інформаційну систему, так само як і пов'язані з нею наявні та (або) заплановані засоби захисту. Оскільки всі засоби захисту мають бути сумісними з навколишнім середовищем, оцінки є суттєвими для успішного вибору. Нижче наведені питання, які можуть надати допомогу під час досліджування інфраструктури. Необхідно також враховувати вплив навколишнього середовища та будь-які інші обставини.



**Рис. 12.2. Базове оцінювання під час вибору засобів захисту**

*Розташування та будівля:*

- де розташована будівля – на своїй власній ділянці з огорожею по периметру, чи на вулиці, в місці з інтенсивним рухом транспорту тощо?
- будівля зайнята однією організацією чи зайнята багатьма організаціями?
- якщо будівля зайнята багатьма організаціями, то хто ці мешканці?

- де знаходяться зони безпеки?

*Керування доступом:*

- хто має доступ до будівлі?
- чи є система керування фізичним доступом?
- наскільки міцна конструкція будівлі?
- наскільки міцні двері, вікна тощо та як вони захищені?
- чи охороняється будівля, якщо так, то цілодобово чи тільки протягом робочого часу?
  - чи є в будівлі та (або) кімнаті, в якій розташоване критичне інформаційне обладнання, сигналізація для захисту від вторгнень?

*Захист на місці:*

- як захищається кімната (кімнати), що містить інформаційну систему?
- яка система виявлення пожежі, сигналізація та система гасіння встановлені, та де вони знаходяться?
- яка система виявлення витoku води/рідини, можливості її виведення та сигналізації?
- чи є допоміжні засоби, такі як система безперебійного живлення, водопровід та кондиціонування повітря (для контролю температури та вологості)?

Відповідаючи на ці питання, можна легше виявити наявні фізичні та пов'язані з ними засоби захисту. Варто відмітити, що під час дослідження місцезнаходження будівлі, необхідно одночасно з'ясувати все, що стосується дверей, замків та контролю і порядку фізичного доступу, ця процедура не займає багато часу.

Оцінювання наявних/планованих засобів захисту

Після оцінювання умов навколишнього середовища та компонентів інформаційної системи треба визначити всі інші засоби захисту: вже наявні та заплановані. Це необхідно, щоб уникнути повторного вибору наявних чи запланованих засобів, а знання цих засобів захисту допомагає вибрати інші засоби, що будуть діяти разом з ними. Коли вибирають засоби захисту, треба розглядати сумісність наявних засобів захисту з вибраними. Засіб захисту може бути несумісний з іншими засобами захисту чи унеможливити успішну діяльність та захист, що вже забезпечується.

Для визначення наявних чи планованих засобів захисту, можуть бути корисними такі дії:

- перегляньте документи, що містять інформацію про засоби захисту (наприклад, плани чи концепції інформаційної безпеки) – якщо процес

убезпечення добре документований, всі наявні чи заплановані засоби та статус їх реалізації повинні бути там перераховані;

- перевірте з відповідальними особами (наприклад, керівник інформаційної безпеки, управитель будинком чи директор-розпорядник) та користувачами, які засоби захисту дійсно реалізовані для інформаційної системи, що розглядається;

- продивіться схему розташування засобів захисту в будівлі, порівняйте запроваджені засоби захисту зі списком тих, що мають бути, та перевірте як запроваджено засоби захисту, чи вони працюють коректно та ефективно.

Може бути виявлено, що наявні засоби захисту перевищують поточні потреби. В цьому випадку, треба розглянути можливість видалення цих засобів. Якщо розглядати вилучення надлишкових засобів чи засобів, що не є необхідними, треба взяти до уваги чинники безпеки та вартості. Оскільки засоби захисту впливають один на одного, видалення надлишкових засобів може зменшити загальну безпеку. Треба зазначити, що іноді дешевше залишити ці засоби на місці, ніж вилучати їх, чи, особливо, якщо засоби захисту мають високу вартість обслуговування, дешевше вилучити їх.

Є два різні набори засобів захисту, механізмів та (або) процедур, які можна застосовувати для захисту інформаційних систем. З одного боку, є доволі багато організаційних категорій засобів захисту, які є загальноприйнятими для кожної інформаційної системи за конкретних обставин, незалежно від окремих компонентів. Внаслідок їх загальної застосовності, засоби, які належать до цих категорій треба завжди розглядати. До того ж, багато з них є недорогими для впровадження, оскільки вони стосуються організаційних структур та процедур.

З іншого боку, є специфічні засоби захисту інформаційної системи – вибір цих засобів захисту залежить від типу та характеристик інформаційної системи, що розглядається.

Звичайно, можливо, що одна чи більше з цих категорій або специфічних засобів захисту не є застосовними до інформаційної системи. Наприклад, шифрування може не бути необхідним, якщо відправлена чи отримана інформація не потребує конфіденційності, а цілісність може бути перевірена іншим чином.

Після того, як визначені всі типи засобів захисту, застосовні до обговорюваної інформаційної системи, перед реалізацією вибраних засобів захисту треба ретельно перевірити, чи їх немає серед наявних та (або) запланованих.

Якщо засоби захисту вибрані відповідно до інших критеріїв (наприклад базові та додаткові засоби), остаточний набір засобів для впровадження треба

робити обережно. Після перегляду декількох інформаційних систем, потрібно розглянути, чи можна запровадити базову безпеку для всієї організації.

Інша можливість вибору засобів захисту без детального розгляду – це застосування баз, пов'язаних з конкретним використанням. Але перед вибором того, які засоби захисту треба впроваджувати, корисно розглянути потреби та проблеми безпеки.

#### Засоби захисту загального застосування

Категоріями загального застосування засобів захисту є:

- керування інформаційною безпекою та політики безпеки;
- перевіряння узгодженості безпеки;
- реагування на порушення;
- персонал;
- питання експлуатації;
- планування неперервності бізнесу;
- фізична безпека.

Засоби захисту, які належать до цих категорій формують основу успішного керування інформаційною безпекою, їх не треба недооцінювати. Також важливо забезпечити взаємодію цих засобів з більш технічними засобами. Організація визначає обсяги робіт у цих сферах, залежно від її потреб, проблем та доступних ресурсів.

Звичайно, багато інших категорій засобів захисту застосовні в більшості випадків, але спосіб реалізації зазвичай є відповідним конкретним обставинам (наприклад, засоби, які забезпечують контроль доступу для мережі відрізняються від тих засобів, що забезпечують контроль доступу для автономної інформаційної системи).

Коли засоби захисту вибирають з категорій загально застосовних засобів, корисно розглядати розмір організації так само, як потреби безпеки, оскільки він впливає на межі, в яких реалізуються ці засоби захисту. Наприклад, маленька організація не буде мати ні потреби, ні персоналу для створення комітету інформаційної безпеки, проте має бути хтось, хто виконує ці функції. Тому всі засоби захисту мають бути відповідно зважені, коли б це не знадобилося.

#### Специфічні засоби захисту інформаційної системи

На додаток до засобів захисту загального застосування, для кожного відповідного типу системного компонента треба вибирати специфічні засоби захисту системи. Нижченаведена таблиця дає приклад того, як починати процес вибору специфічних засобів системи. В цьому прикладі 'X' означає засоби, що

мають реалізуватись за нормальних обставинах, та '(X)' означає засоби, що можуть бути необхідними за деяких обставин (табл. 12.1).

Таблиця 12.1

### Процес вибору специфічних засобів системи

	Автономна робоча станція	Робоча станція (клієнт без спіль- них ресурсів), під'єднана до мережі	Сервер чи робо- ча станція з спі- льними ресур- сами, під'єдна- на до мережі
<b>Автентифікація</b>			
Автентифікація на основі інформації, якою володіє користувач	X	X	X
Автентифікація на основі дечого, чим володіє користувач	X	X	X
Автентифікація на основі того, ким є користувач	(X)	(X)	(X)
<b>Контроль логічного доступу та аудит</b>			
Політика контролю доступу			X
Доступ користувачів до комп'ютерів	X	X	X
Доступ користувачів до даних, служб та програм	X	X	X
Перегляд і оновлення прав доступу			X
Журнали аудиту	X	X	X
<b>Зловмисний код</b>			
Сканери	X	X	X
Програми перевіряння цілісності	X	X	X
Контроль за обігом переносних носіїв інформації	X	X	X
Процедурні засоби захисту	X	X	X
<b>Керування мережею</b>			
Методика експлуатації			X
Планування системи			X
Конфігурація мережі			X
Відокремлення мережі			X
Моніторинг мережі			X
Виявлення вторгнень			X
<b>Криптографія</b>			
Захист конфіденційності даних	(X)	(X)	(X)
Захист цілісності даних	(X)	(X)	(X)
Неспровтовність		(X)	(X)
Автентичність даних	(X)	(X)	(X)
Керування ключами	(X)	(X)	(X)

Вибір засобів захисту відповідно до проблем та загроз безпеці можна використовувати таким чином.



Перший крок – визначити та оцінити проблеми безпеки. Треба розглянути вимоги до конфіденційності, цілісності, доступності, спостережності, автентичності та надійності. Міцність та кількість вибраних засобів захисту має відповідати оціненим проблемам безпеки.

Другий – для кожної проблеми безпеки визначають типові загрози і для кожної загрози пропонують засоби захисту інформаційної системи, що розглядається. У такий спосіб можливо задовольнити специфічні потреби безпеки та досягнути захисту там, де він дійсно необхідний.

#### Оцінювання проблем безпеки

Для ефективного вибору прийнятих засобів захисту необхідно розуміти проблеми безпеки підтримуваних ділових операцій інформаційною системою, що розглядається. Засоби захисту треба вибирати за допомогою визначення проблем безпеки, беручи до уваги відповідні загрози, що можуть призвести до цих проблем.

Вибір засобів захисту відповідно до детальних оцінок здійснюють згідно з принципами, наведеними в попередніх розділах. Детальне аналізування ризиків дозволяє враховувати спеціальні вимоги та обставини інформаційної системи та її цінностей. Відмінність від використання, наведеного у попередніх розділах – це обсяг робіт та подробиці, зібрані протягом процесу оцінювання. Тому можливе кваліфіковане обґрунтування вибраних засобів захисту.

Якщо оцінювання, проведене згідно з положеннями цього підрозділу, виявляє дуже великі проблеми безпеки, рекомендується деталізованіший підхід для визначення прийняттого захисту.

Проблеми безпеки мають враховувати:

- втрату конфіденційності;
- втрату цілісності;
- втрату доступності;
- втрату спостережності;
- втрату автентичності;
- втрату надійності.

Оцінювання має охоплювати саму інформаційну систему, інформацію, що зберігається чи обробляється на ній, та ділові операції, які вона виконує. Це оцінювання визначає цілі вибраних засобів захисту. Різні частини інформаційної системи або інформація, що зберігається чи обробляється, можуть мати різні проблеми безпеки. Важливо пов'язувати проблеми безпеки безпосередньо з цінностями, оскільки це впливає на загрози, які можуть з'являтися, і, таким чином, на вибір засобів захисту.

### Взаємозв'язок ISO/IEC TR 13335-3 та ISO/IEC TR 13335-4

В ISO/IEC TR 13335-3 описано керування інформаційною безпекою. Крім того, наведено інші питання, можливі варіанти стратегії аналізування ризиків та рекомендований підхід до аналізування ризиків. Головними варіантами стратегій для використання в організації є:

- використання базового підходу для всіх інформаційних технологій систем;
- використання детального аналізування ризиків для всіх інформаційних технологій систем;
- використання “рекомендованого підходу”, тобто дотримання високорівневого аналізування ризиків для всіх інформаційних технологій систем, базового підходу для інформаційних технологій систем малого ризику і детального аналізування ризиків для інформаційних технологій систем високого ризику.

Якщо для визначення засобів захисту було вирішено використовувати детальне аналізування ризиків для всіх інформаційних технологій систем, потрібна інформація про те, як вибирати засоби захисту, і як ефективно використовувати результати детального аналізування ризиків. Проте може використовуватися інформація про засоби захисту для специфічних інформаційних технологій систем та зв'язок між проблемами безпеки, загрозами і засобами захисту.

Значимість проблем безпеки може бути оцінена залежно від того, чи порушення безпеки спричинює серйозні ушкодження ділової діяльності, або ж тільки завдає легкої шкоди, чи не впливає зовсім. Розгляд можливих загроз може допомогти з'ясувати проблеми безпеки. Оцінювання, описане нижче, треба проводити окремо для кожної цінності, оскільки проблеми безпеки для різних цінностей можуть бути різними. Однак, якщо є достатніми знання з проблем безпеки, цінності з однаковими чи схожими діловими потребами та проблемами безпеки можуть бути об'єднані в групи.

#### Принципи вибору

Є чотири основних аспекти, на які спрямований засіб захисту: впливи, загрози, вразливості та ризики самі по собі. Засоби спрямовують на самі ризики, коли приймається рішення зменшити чи уникнути ризиків, а не приймати їх, наприклад, для зменшення ризику – проведення страхування, а прикладом, щоб уникнути ризику, є переміщення контрольованої інформації на іншу інформаційну систему. Компоненти, що всі разом створюють ризики,

тобто впливи, загрози та вразливості, є головними цілями засобів захисту. Способи, якими засоби можна направляти на ці аспекти, такі:

1) *загрози* – засоби захисту можуть зменшити ймовірність виникнення загрози (наприклад, розглянемо загрозу втрати даних через помилки користувача, тоді навчальний курс для користувачів зменшить кількість цих помилок), чи, у випадку зловмисного нападу, вони можуть спинити його через збільшення технічної складності успішної атаки;

2) *вразливість* – засоби захисту можуть усунути вразливість чи зробити її менш серйозною приклад, якщо внутрішня мережа, що з'єднана із зовнішньою мережею, вразлива до несанкціонованого доступу, то реалізація відповідного брандмауера зробить з'єднання менш вразливим, а роз'єднання усуне цю вразливість;

3) *вплив* – засоби захисту можуть зменшити чи усунути вплив (якщо зловмисний вплив являє собою недоступність інформації, він зменшується через створення копій інформації, які надійно зберігаються в іншому місці, та готовність до активування плану неперервності бізнесу). Добре організоване реєстрування і аналіз журналів аудиту та засобів сигналізації може допомогти ранньому виявленню інциденту та знизити зловмисний вплив на бізнес.

Як і де використовують засіб захисту, може бути суттєва різниця від тієї користі, яку отримано завдяки його запровадженню. Дуже часто, загрози можуть використовувати більше ніж одну вразливість. Тому, якщо засіб захисту використовують, щоб запобігти виникненню такої загрози, він може бути спрямованим на декілька вразливостей одночасно. Зворотне також вірно – засіб захисту, що захищає вразливість, може бути спрямованим на декілька загроз. Ці переваги бажано розглядати під час вибору засобів захисту. Ці додаткові переваги потрібно завжди документувати, щоб мати повноту вимог безпеки, яким задовольняє будь-який засіб захисту.

Взагалі, засоби захисту можуть забезпечувати один чи більше з таких типів захисту: запобігання, стримування, виявлення, зменшення, відновлення, виправлення, моніторинг та обізнаність. Яка з цих властивостей найкраща, залежить від конкретних обставин та від призначення кожного засобу, в багатьох випадках засоби захисту забезпечують більше одного типу захисту, що знову ж таки, забезпечує додаткові переваги. За можливості, треба обирати ті засоби, які мають багато переваг.

Безпека повинна завжди показувати розумний баланс щодо ефектів, згаданих вище. Якщо занадто великий акцент робиться на типі засобу захисту, малоімовірно, що загальна безпека буде ефективною. Наприклад, якщо

більшість засобів стримування використовують без адекватних засобів виявлення, щоб визначити, коли стримування не спрацювало, загальна безпека не буде ефективною. Перед реалізацією, запропоновані засоби захисту треба порівняти з наявними засобами, щоб оцінити, що треба розширяти чи оновлювати. Якщо є, то це може бути дешевше, ніж запровадження нових засобів захисту.

Під час вибору засобів захисту важливо зважувати на вартість реалізації засобів захисту відносно вартості цінностей, що захищаються, та строки повернення інвестицій, пов'язаних зі зниженням ризику. Вартість реалізації та обслуговування засобу може бути набагато вищою, ніж вартість самого засобу, тому це треба враховувати під час вибору.

Технічні обмеження, а саме: вимоги продуктивності, керованості (вимоги обслуговування діяльності) та питань сумісності, – можуть заважати використанню деяких засобів захисту. В цих випадках керівники системи та безпеки мають працювати разом для прийняття оптимальних рішень. Може трапитися випадок, коли засіб захисту буде знижувати продуктивність. Знову таки, керівник системи та безпеки разом повинні прийняти рішення, що дозволить забезпечити необхідну продуктивність за умови гарантованої достатньої безпеки. Такі аспекти, як законодавство про захист приватного життя та права можуть вимагати, щоб були наявні певні засоби захисту, тому використовується визначення незмінних базових елементів. Коли організація вирішує запровадити базову безпеку до всієї організації чи до її частин, треба розглянути такі питання.

- які частини організації чи систем можуть бути захищені певним базовим рівнем, а які потребують іншого, і чи може той самий базовий рівень запроваджуватись для цілої організації?

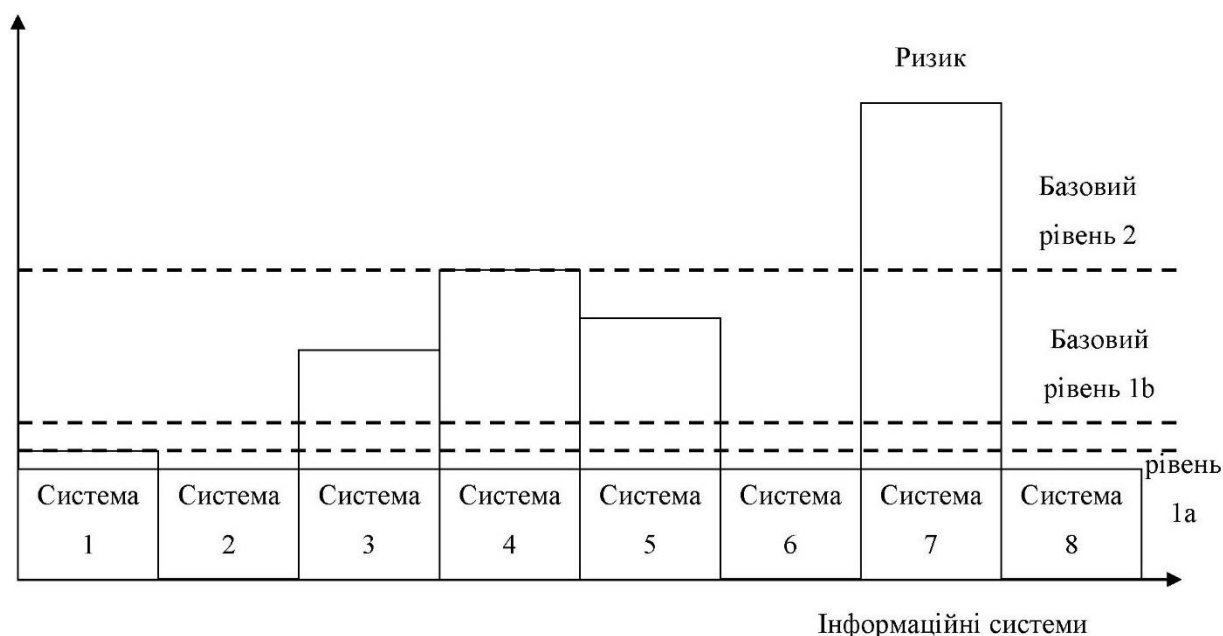
- на який рівень безпеки має бути орієнтована базова безпека (чи різні базові безпеки)?

- як можуть бути визначені засоби захисту, що формують іншу базову безпеку (за потреби)? На рис. 12.3 зображено різні способи застосування базової безпеки.

Перевага застосування різних базових рівнів в одній організації – це те, що більшість систем будуть захищені належним чином, тобто застосовується не надмірно малий та не надмірно великий захист (наприклад, для інформаційних технологій з базовим рівнем 1 – системи 1, 2, 6, 8 та інформаційних технологій з базовим рівнем 2 – це системи 3, 4 та 5, як зазначено на рис. 12.3). Якщо інформаційні системи з різними вимогами безпеки є «насправді різними»

(в тому сенсі, що більшість засобів захисту, які потрібні для захисту інформаційних технологій системи, різні), тоді для організації рекомендовано використовувати різні бази. Якщо вимоги до безпеки фундаментально відрізняються, рішення про використання базового підходу має бути переглянуто.

З іншого боку, якщо єдина відмінність між різними базовими рівнями – це та, що виникає потреба у деяких додаткових засобах захисту для формування вищих базових рівнів, тоді можливо не варто запроваджувати декілька різних базових рівнів. Якщо запроваджений тільки один базовий рівень, накладні видатки організації можуть бути значно зменшені, і кожен в організації зможе покладатися на наявність однакового рівня безпеки.



**Рис. 12.3. Різні базові рівні**

**Заключення.** Рівень базової безпеки, на який треба орієнтуватися, звичайно, залежить від рішення, скільки рівнів базової безпеки можна логічно запровадити: один чи більше. Якщо вибрані різні базові рівні, ці рівні можуть бути встановлені достатньо точно до вимог безпеки інформаційних технологій систем, які потрібно захищати. Загалом, жоден базовий рівень не повинен бути спрямований на безпеку нижче найнижчих вимог до безпеки інформаційних технологій систем, які треба захищати (наприклад, нижче вимог інформаційних технологій системи 2 на рис. 12.3). Доцільно орієнтуватись на рівень, що є достатнім для більшості (базовий рівень 1a на рис. 12.3) чи всіх (базовий рівень 1b) інформаційних технологій систем, призначених для захисту. Часто доцільно

орієнтуватись на найвищий рівень безпеки інформаційних систем для захисту їх базовими засобами, оскільки це зазвичай не дуже дорого, але забезпечує достатню безпеку для всіх задіяних інформаційних технологій систем. Необхідно уважно розглянути запроваджені інформаційні технології систем для прийняття остаточного рішення, які інформаційні технології системи будуть захищатись тим же базовим рівнем. Деякі інформаційні технології системи багато в чому схожі за сутністю та (або) за вимогами до захисту – в цьому випадку є корисно захищати їх тим же самим базовим рівнем. З іншого боку, якщо декілька інформаційних технологій повністю відрізняються в своїх вимогах до захисту, дуже часто найпростіше розглядати їх окремо.

Теж саме і у випадку, якщо організація вирішує реалізовувати однаковий базовий рівень по всій організації. Ця базова безпека може бути орієнтована на три різні рівні:

- низький рівень, долучаючи специфічні засоби захисту, щоб захистити всі інформаційні технології системи з вищими вимогами;
- середній рівень, долучаючи специфічні засоби захисту, щоб захистити всі інформаційні технології системи з вищими вимогами, чи
- високий рівень, що є достатнім для захисту всіх інформаційних технологій систем, передбачених для захисту базовою безпекою.

Як описано вище, середній та високий рівні базової безпеки можуть бути зручними для багатьох організацій.

## РОЗДІЛ 13

# СИНЕРГЕТИЧЕСКИЕ МОДЕЛИ ОЦЕНКИ БЕЗОПАСНОСТИ В АВТОМАТИЗИРОВАННЫХ БАНКОВСКИХ СИСТЕМАХ

***Аннотация.** раскрыта сущность построения синергетической модели оценки безопасности банковской информации (БИН), предложены формальное математическое описание синергетической модели безопасности БИН на основе методологии и синергетическом подходе к обеспечению безопасности БИН и оцениванию безопасности информационных технологий автоматизированных банковских систем (АБС) организаций банковского сектора (ОБС) Украины, а также частных моделей: инфраструктурной модели АБС, синергетической модели угроз, модели нарушителя и модели проведения оценки защищенности АБС. Рассмотрена модель оценки экономической целесообразности внедрения того или иного механизма ТСЗИ в АБС ОБС в зависимости от ценности БИН.*

***Ключевые слова:** синергетическая модель безопасности банковской информации, модель нарушителя, модель защищенности АБС.*

***Abstract.** The article reveals the essence of building the synergetic model for assessing the security of banking information (BIn), offered a formal mathematical description of synergetic security model Bin based on the methodology and synergetic approach to security BIn and security evaluation of information technologies automated banking systems (ABS), banking organizations (BOs) in Ukraine, as well as special models: infrastructure ABS model, synergetic model threats, intruder's models and security assessment ABS models. The model of evaluation of the economic feasibility of the introduction of a mechanism TSZI in ABS OBS is considered depending on the value BIn.*

***Keywords:** synergetic model of banking information security, intruder's model, ABS security model.*

**Введение и постановка задачи.** Анализ последних исследований и публикаций показал, что при построении защиты информации сложился подход, основанный на представлении процесса ее обработки в виде абстрактной вычислительной среды, в которой работают множество субъектов (пользователей и процессов) с множеством объектов (ресурсы и наборы данных). При этом построение системы защиты заключается в создании защитной среды в виде некоторого множества ограничений и процедур, способных под управлением ядра безопасности запретить несанкционированный и реализовать санкционированный доступ субъектов к объектам и защиту последних от преднамеренных и случайных внешних и внутренних угроз. Данный подход опирается на теоретические модели

безопасности Хартсона, Белла –Лападулы, MMS Лендвера и Мак Лина, Биба, Кларка – Вилсона и др. Считается, что перечисленные модели являются инструментарием при разработке политик безопасности, определяющих множество требований, которые должны быть выполнены в конкретной реализации системы. При построении защиты информации используют два подхода, использующих представление процесса ее обработки в виде абстрактной вычислительной среды, в которой работают множество субъектов (пользователей и процессов) с множеством объектов (ресурсы и наборы данных). При первом подходе используется модель на основе системы управления информационной безопасностью (СУИБ), второй основывается на использовании системы менеджмента информационной безопасности (СМИБ). В обоих подходах для оценки рисков используются теоретические модели безопасности, основанные на различных моделях разграничения доступа. Однако основными недостатками обоих подходов являются формирование моделей информационной безопасности на основе модели триады CIA (обеспечения конфиденциальности, целостности и доступности), отсутствие разграничений в понятиях “информационная безопасность” (ИБ) и “безопасность информации” (БИ), формальное комплексирование угроз, без учета их особенностей, что не позволяет получить синергетический выигрыш и эмерджентные свойства СБ АБС.

**Целью исследования** является раскрытие сущности построения синергетической модели безопасности БИИ на основе методологии и синергетическом подходе к обеспечению Б БИИ и оцениванию безопасности информационных технологий АБС ОБС Украины.

Основным результатом формирования методологических основ обеспечения безопасности ИП, в соответствии с системным подходом является идеализированная или эталонная модель (ЭМ) защищенной АБС ИКП, реализующая принципиально безопасные технологии циркуляции БИИ. Кроме этого, ЭМ обеспечивает потенциальную возможность реализации решений стандартизации и унификации архитектурных подходов, путем разработки регламентов и стандартов в области безопасности БИИ. Для построения модели безопасности на основе синергетического подхода к оценке угроз БИИ, независимо от составляющей безопасности: информационной безопасности (ИБ), кибербезопасности (КБз), безопасности информации (БИ), целесообразно применять принципы Риск-менеджмента, который позволит при грамотном использовании основных его процедур своевременно определить и классифицировать угрозы, и, в соответствии с вероятностью наступления



негативных последствий от их возможного проявления адекватно организовать систему обеспечения безопасности БИИ. Очевидно, что одной из важнейших задач оптимального построения комплексной системы защиты информации является выбор из множества средств такого их набора, который позволит обеспечить нейтрализацию всех потенциально возможных угроз с наилучшим качеством и минимально возможными затратами ресурсов. С этой целью используются модели безопасности, позволяющие синтезировать настройки параметров безопасности АБС, уменьшив трудозатраты и повысив степень соответствия нормативных документов на протяжении всего цикла использования ТСЗИ в АБС. На рис. 13.1 приведен обобщенный подход к построению синергетической модели безопасности БИИ в АБС.

Анализ рис. 13.1 показывает, что основным отличием предлагаемого подхода моделирования модели безопасности от известных является, во-первых, использование синергетического подхода при построении модели угроз, что дает эмерджентный эффект получения комплексированной оценки угроз БИИ, во-вторых, обеспечению успешности выполнения бизнес-процессов посредством функций безопасности БИИ (ФББИИ), выделенных элементов АБС, основанных на требованиях:

- обеспечение конфиденциальности информации;
- обеспечение доступности информации, сервисов и сетевых, и аппаратных подсистем;
- обеспечение целостности информации;
- обеспечение непрерывности бизнес-процессов.

Введем определения безопасности БИИ, основных механизмов и процедур, в рамках построения модели безопасности БИИ на основе синергетического подхода:

*Банковская информация (БИИ)* – информация, возникшая в результате банковской деятельности, а также сведения, характеризующие сам банк, его финансовое положение, надёжность и выполнение требований законодательства.

*Безопасность банковской информации (Б БИИ)* – состояние защищенности банковской информации, характеризуемое способностью пользователей, технических средств и информационных технологий обеспечить конфиденциальность, целостность аутентичность и доступность БИИ при ее обработке в автоматизированной банковской системе (АБС).

*Информационная безопасность банковской информации (ИБ БИИ)* – состояние защищенности информационной среды банковского сектора,

обеспечивающее ее формирование, использование и развитие в интересах граждан и организаций банковского сектора.

*Кибербезопасность банковской информации (КБрБ БИИ)* – набор средств, стратегий, принципов обеспечения безопасности, гарантий безопасности, подходов к управлению рисками, действий, профессиональной подготовки, страхования и технологий, которые используются для защиты киберсреды АБС, ресурсов и пользователей ОБС;

*Объектами угроз БИИ* выступают сведения о составе, состоянии и деятельности банка (персонала, материальных и финансовых ценностей, информационных ресурсов банка).

*Угрозы безопасности БИИ* – совокупность условий и факторов, создающих опасность несанкционированного, в том числе случайного, доступа к банковским данным, результатом которого может стать уничтожение, изменение, блокирование, копирование, распространение БИИ, а также иных НСД при их обработке в АБС.

Угрозы информации выражаются в нарушении ее доступности, целостности, аутентичности и конфиденциальности.

*Синергетический показатель Б БИИ* в АБС – синергетическая оценка эффективности комплексного применения сил и средств обеспечения безопасности банковской информации в условиях антагонистического противодействия системы банковской защиты случайным и целенаправленным угрозам безопасности.

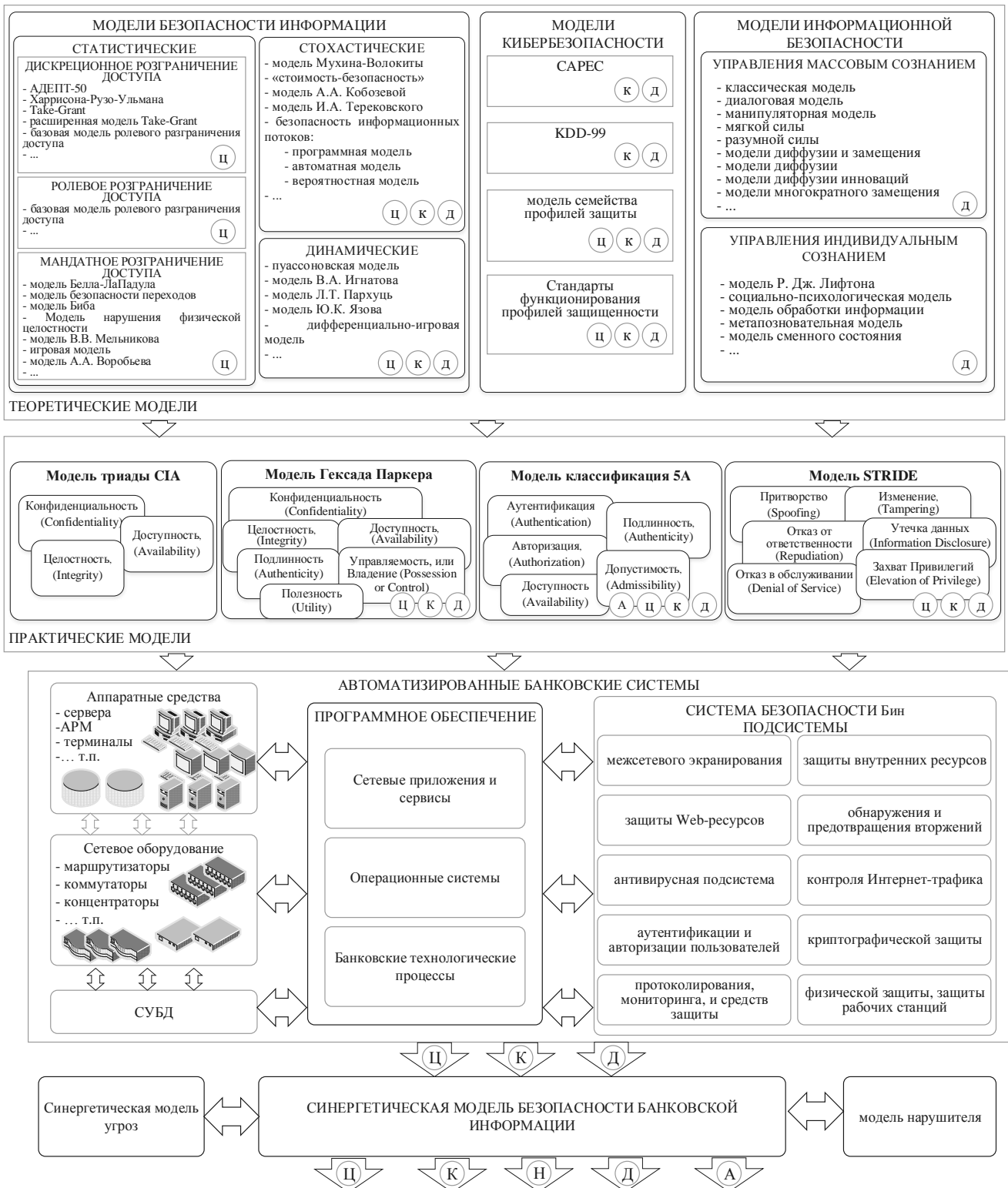
*Источниками угроз* выступают конкуренты, злоумышленники-хакеры, баккеры и инсайдеры. Источники угроз преследуют при этом следующие цели: ознакомление с банковской информацией, их модификация в корыстных целях и уничтожение для нанесения прямого материального ущерба.

Неправомерное овладение конфиденциальной информацией возможно за счет ее разглашения, утечки БИИ через технические средства и несанкционированного доступа к БИИ.

*Источниками конфиденциальной информации* являются персонал, банковские процессы, документы, технические носители БИИ, технические средства обеспечения банковских транзакций.

*Основными направлениями защиты информации* являются правовая, организационная и инженерно-техническая защиты информации как выразители комплексного подхода к обеспечению информационной безопасности БИИ.

Средствами защиты информации являются физические, аппаратные, программно-аппаратные средства и криптографические методы.



(Ц) - целостность (К) - конфиденциальность (Д) - доступность (А) - аутентичность (Н) - непрерывность

**Рис. 13.1. Обобщенный подход формирования синергетической модели безопасности Бин**

В качестве *способов защиты* выступают организационно-технические меры, способы и действия, обеспечивающие упреждение противоправных действий, их предотвращение, пресечение и противодействие несанкционированному доступу к БИИ. В обобщенном виде рассмотренные компоненты в виде *концептуальной синергетической модели безопасности БИИ* приведены на рис. 13.2. Концептуальная синергетическая модель безопасности БИИ предлагаемая автором формируется на основе предложенной автором методологии и синергетическом подходе к обеспечению безопасности БИИ и оцениванию безопасности информационных технологий (ИТ) АБС Украины [113; 23], а также частных моделей: инфраструктурной модели АБС, синергетической модели угроз и модели проведения оценки защищенности АБС. Инфраструктурная модель АБС представляет собой следующую формальную модель:

$$G^{ABS} = \{\{O^{ABS}\}, \{L^{ABS}\}, \{I_A\}\}, \quad (1)$$

где  $O^{ABS}$  – множество объектов среды, описывающих элементы АБС и их принадлежность к уровням иерархии ИКП,  $L^{ABS}$  – множество связей между элементами, определяемое матрицей смежности  $A^{ABS} = \|a_{ij}^{ABS}\|$ .



**Рис. 13.2. Структурная схема концептуальной синергетической модели безопасности БИИ**

$\{I_A\}$  – множество элементов информационных активов. Каждый элемент  $I_{A_i} \in \{I_A\}$  описывается вектором  $I_{A_i} = (Type, A^C, A^D, A^A, A^K, C_Y)$ .  $Type$  – тип информационного актива, описывается множеством базовых значений  $Type = \{BT, PID, RrD, KT, StO, Ol, YI, PD\}$ , где  $BT$  – банковская тайна,  $PID$  – платежные документы,  $KrD$  – кредитные документы,  $KT$  – коммерческая тайна,  $StO$  – статистические отчеты,  $Ol$  – общедоступная информация,  $YI$  – управляющая информация,  $PD$  – персональные данные.  $A^K$  – конфиденциальность,  $A^C$  – целостность,  $A^D$  – доступность,  $A^A$  – аутентичность,  $C_Y$  – непрерывность – свойства информации, которые необходимо обеспечивать. Принимают значение 1 – если свойство необходимо, 0 – в противном случае.

Каждый элемент  $O_l \in \{O^{ABS}\}$ , описывается вектором  $O_l = \{Y^{ABS}, TO\}$ , где  $Y^{ABS}$  – уровень иерархии информационной структуры, определяемое множеством  $Y^{ABS} = \{FL, NL, OSL, DBL, BL\}$ , где  $FL$  – физический уровень,  $NL$  – сетевой уровень,  $OSL$  – уровень операционных систем (ОС),  $DBL$  – уровень систем управления базами данных,  $BL$  – уровень банковских технологических приложений и сервисов. Для указания типа связи и существующего отношения  $IO^R$  между информационными активами и объектами среды использования используется правило:

$$IO^R = \parallel IO_{il}^R \parallel \quad (2)$$

где  $IO_{il}^R$  – отображает наличие и тип связи между  $i$ -м информационным активом и  $l$ -м объектом среды. При этом  $\forall i \in \{I_A\}$ , а  $\forall l \in \{O^{ABS}\}$ :

$$IO_{il}^R = \begin{cases} 0, \text{ связь отсутствует} \\ cs, \text{ включает и хранит} \\ pt, \text{ обрабатывает или передает} \\ so, \text{ поддерживает функционирование} \end{cases} .$$

Синергетическая модель угроз формально может быть представлена в виде:

$$GR^{ABS} = \{ \{DF^{ABS}\}, \{T_{risk}\}, \{T_P\}, \{T_U\}, \{VH\} \}. \quad (3)$$

Множество источников угроз безопасности АБС представлено кортежем  $DF^{ABS} = \{V^{NS}, V^{AS}\}$ , в котором  $V^{NS}$  – класс естественных источников угроз,  $V^{AS} = \{V^{ASIB}, V^{ASBI}, V^{ASKBr}\}$  – класс антропогенных угроз, где  $V^{ASIB}$  – множество

угроз информационной безопасности,  $V^{ASBI}$  – множество угроз безопасности информации,  $V^{ASKBr}$  – множество угроз кибербезопасности.  $T_{risk}$  – качественный показатель риска,  $T_p$  – множество базовых термов вероятности реализации хотя бы одной угрозы  $j$ -му активу,  $T_U$  – множество базовых термов величины ущерба от реализации угрозы  $u_i$ ,  $VH$  – множество деструктивные состояния элементов АБС, под которыми понимается нежелательное и незапланированное состояние компонента АБС, в котором он оказался в результате реализации одной или нескольких угроз [70].

Для получения синергетического эффекта повышения уровня защищенности БИИ необходимо учитывать комплексирование угроз:

$$DF^{ABS} = \{V^{NS}\} \cup \{V^{AS}\}, \text{ где } \{V^{AS}\} = \{V^{ASBI}\} \cap \{V^{ASIB}\} \cap \{V^{ASKBr}\} \quad (4)$$

Каждый элемент из множества угроз  $DF_i \in \{DF^{ABS}\}$ , может быть представлен следующим вектором значений  $DF_i(p, u, risk)$ , где  $p$  – вероятность реализации угрозы,  $u$  – потенциальный ущерб,  $risk$  – риск, выраженный в качественной форме и принимающий одно из двух состояний  $T_{risk} = \{\text{допустимый}, \text{недопустимый}\} = \{\alpha_{r1}, \alpha_{r2}\}$ .

Оценка вероятности реализации  $i$ -й угрозы к  $j$ -му активу определим на основе предложений авторов в работе [4]: для учета связей между источниками угроз и элементами АБС задается матрица  $A^{DF} = \|a_{ij}^{DF}\|$ , размерностью  $n$  на  $m$ , где  $n$  – количество угроз,  $m$  – количество активов. Для каждой  $i$ -й угрозы к  $j$ -му активу определяется вероятность реализации  $pr_{ij}$  на основе либо накопленных статистических данных, характерных для данного региона и условий эксплуатации (в количественной и/или качественной форме), либо экспертным путем.

Расчет вероятности реализации хотя бы одной угрозы для каждого актива выполняется по формуле:

$$p_{rj} = 1 - \prod_{i=1}^m (1 - pr_{ij}), \quad (5)$$

где  $p_{rj}$  – вероятность реализации хотя бы одной угрозы  $j$ -му активу.

Предполагается, что в случае реализации для  $j$ -го актива хотя бы одной из угроз из множества  $V^{AS} = \{V^{ASIB}, V^{ASBI}, V^{ASKBr}\}$ , ущерб равняется стоимости актива на основе детализации активов и тщательного выбора актуальных угроз:

$$q_j = u_j \quad (6)$$

Считается, что угрозы могут быть реализованы независимо друг от друга, тогда цена риска  $R_j$  для каждого  $j$ -го актива определяется по формуле:

$$R_j = pr_j \times q_j. \quad (7)$$

Цена полного риска равна сумме цен риска всех активов:

$$R_{полн} = \sum_{j=1}^n R_j. \quad (8)$$

Таким образом, вероятность реализации среды  $p_{rj}$ , с областью определения  $P = [0, 1]$  зададим множеством базовых термов  $T_p = \{\text{нереализуемая, минимальная, средняя, высокая, критичная}\} = \{\alpha_{x1}, \alpha_{x2}, \alpha_{x3}, \alpha_{x4}, \alpha_{x5}\}$ .

Оценка потенциально возможного ущерба от реализации угрозы тесно связано с капиталом (см. выражение (6)) и формируется на основе экспертных оценок. Величина ущерба от реализации угрозы  $u_i$  задается множеством базовых термов  $T_U = \{\text{минимальная, средняя, высокая, критичная}\} = \{\alpha_{y1}, \alpha_{y2}, \alpha_{y3}, \alpha_{y4}, \alpha_{y5}\}$ . Для перехода между качественными и количественными значениями используем правило, предложенное в [86].

Для определения значения рисков воспользуемся правилом, предложенным в работе [18] на основе системы нечетких высказываний:

$$\tilde{L}^1 = \left\{ \begin{array}{l} \tilde{L}_1 : \langle E_{11} \cup E_{12} \cup E_{13} \cup E_{14} \cup E_{21} \cup E_{22} \cup E_{23} \cup E_{31} \cup E_{32} : risk_i \text{ есть } \alpha_{r1} \rangle; \\ \tilde{L}_2 : \langle E_{24} \cup E_{33} \cup E_{34} \cup E_{42} \cup E_{43} \cup E_{44} \cup E_{51} \cup E_{52} \cup E_{53} \cup E_{54} : risk_i \text{ есть } \alpha_{r2} \rangle \end{array} \right\}, \quad (9)$$

где  $E_{kj}$ : “ $p_{ri}$  есть  $\alpha_{xk}$  и  $u_i$  есть  $\alpha_{yj}$ ”

В ходе анализа документов по моделированию угроз, оценке рисков и теории надежности определены следующие *деструктивные состояния элементов АБС* (множество  $\{vH\}$ ):

а) *информационный актив*: недоступен (нарушена доступность),  $I_A^{[D]}$ ; скомпрометирован (нарушена конфиденциальность),  $I_A^{[K]}$ ; изменен (нарушена целостность),  $I_A^{[C]}$ ; нарушена метка безопасности (цифровая подпись) (нарушена аутентичность),  $I_A^{[A]}$ ;

б) *программное обеспечение*: недоступно (произошел сбой),  $SW^{[B]}$ ; взломано (получен несанкционированный доступ (НСД) злоумышленником или повышены привилегии пользователя),  $SW^{[I]}$ ; нарушение доступности,  $SW^{[U]}$ ; изменено (не санкционированно изменен код и/или конфигурация),  $SW^{[M]}$ ;

в) *техническое средство*: недоступно (произошел временный сбой),  $HW^{[B]}$ ; нарушение доступности,  $HW^{[U]}$ ; неработоспособно (произошел отказ,

требующий ремонт или замена),  $HW^{[D]}$ ; утеряно (произошла потеря или кража у законного владельца),  $HW^{[L]}$ ; взломано (получен несанкционированный доступ (НСД) злоумышленником или повышены привилегии пользователя),  $HW^{[I]}$ ;

г) *линия связи*: недоступна (произошел сбой или отказ),  $CL^{[D]}$ ; нарушение доступности,  $CL^{[U]}$ ; взломана (получен НСД злоумышленником),  $CL^{[I]}$ .

Формальную модель злоумышленника определим с учетом предложений авторов [5] в которых определены категории и действия злоумышленников:

$$G_{IA}^{ABS} = \{aid_i, pur_i, T_{IA}, S_{\max_i}, pr_j, MS_i^{ABS}\} \forall i \in n, \forall j \in m, \quad (10)$$

где  $aid_i \in \{aid\}$  – идентификатор нарушителя,  $pur_i \in \{pur_i\}$  – цель нарушителя,  $T_{IA}$  – время успешной реализации угрозы,  $S_{\max_i}$  – вероятностный ущерб

системы,  $MS_i^{ABS} = \{ms_i\}_{i=1}^{N_{MS^{ABS}}}$  – рекомендации по выявлению, реагированию

ТСЗИ,  $N_{MS^{ABS}}$  – количество рекомендаций известных АБС,  $n$  – количество угроз,  $m$  – количество активов.

Под *источником угроз* понимается субъект доступа, материальный объект или физическое явление, являющиеся причиной возникновения угрозы безопасности информации.

Множество источников угроз включает источники четырех видов:

$$DF^{ABS} = \{V^{NS}, V^{AS}, TS, PI, NI\}, \quad (11)$$

где  $DF^{ABS}$  – множество источников угроз безопасности АБС, в котором  $V^{NS}$  – класс естественных источников угроз,  $V^{AS} = \{V^{ASIB}, V^{ASBI}, V^{ASKBr}\}$  – класс антропогенных угроз, где  $V^{ASIB}$  – множество угроз информационной безопасности,  $V^{ASBI}$  – множество угроз безопасности информации,  $V^{ASKBr}$  – множество угроз кибербезопасности;  $TS$  – технические средства и системы;  $PI$  – преднамеренные нарушители;  $NI$  – непреднамеренные нарушители (злоумышленники).

Сценарием реализации угроз называется один или несколько связанных переходов компонентов АБС в деструктивные состояния в результате воздействий источников угроз. Один или несколько сценариев реализации угроз могут быть представлены ориентированным графом  $G(V, H)$ , в котором: начальной вершиной ( $v_0$ ) является множество, один из видов или конкретный источник угроз; промежуточными и конечными вершинами ( $v_n$ ) являются деструктивные состояния компонентов АБС; дугами ( $h_{ij}$ ) соединятся две вершины, одна из которых является причиной ( $v_i$ ), а вторая – следствием и



результатом перехода ( $v_j$ ), Сценарий реализации угроз конфиденциальности рассмотрен в работе [70].

Для оценки показателей степени опасности нарушителей и степени реализации защитных мер определим наборы взвешенных метрик, принимающих значения в интервале  $[0; 1]$ . Каждая метрика характеризует степень соответствия некоторого признака нарушителя или защитной меры заданному целевому значению. Для оценки степени опасности нарушителя предлагается использовать следующие метрики, сформированные с учетом положений [70]: мотивация, оснащенность (имеющееся оборудование), техническая компетентность, знание информации о АБС и ТСЗИ, права доступа (до реализации угроз), время доступа (до момента обнаружения и реагирования).

Степень опасности  $i$ -го нарушителя определяется по формуле:

$$d_i = \prod \left( M_{ih}^{DFABS} \right)^{w_{ih}^{DFABS}} \quad (12)$$

где  $M_{ih}^{DFABS}$  – значение  $h$ -ой метрики  $i$ -го нарушителя;

$w_{ih}^{DFABS}$  – весовой коэффициент  $h$ -ой метрики  $i$ -го нарушителя,  $\sum_h w_{ih}^{DFABS} = 1$ .

Метрики степени реализации защитных мер, подразделяемых на превентивные (предотвращение перехода элемента АБС в деструктивное состояние)  $\psi_j$  и корректирующие (снижающие величину ущерба от перехода)  $\psi'_j$  определим по формуле:

$$\psi_j = \prod_g \left( \sum_l w_{gl}^{SZABS} \times M_{gl}^{SZABS} \right)^{w_{jg}^K}, \quad (13)$$

где  $M_{gl}^{SZABS}$  – значение метрики  $l$ -ой защитной меры  $g$ -ой категории;

$w_{gl}^{SZABS}$  – весовой коэффициент  $l$ -ой защитной меры  $g$ -ой категории,  $\sum_l w_{gl}^{SZABS} = 1$ .

$w_{jg}^K$  – весовой коэффициент  $g$ -ой категории,  $\sum_g w_{jg}^K = 1$ .

Степень реализации корректирующих защитных мер  $\psi'_j$  по аналогии с  $\psi_j$  определяется по формуле (6). Вектор весовых коэффициентов  $W$

определяется путем нормирования результирующего вектора приоритетов, определяемого экспертным путем:

$$w_i = \bar{b}_i / \sum_{i=1}^m \bar{b}_i, \forall i \in [1; m], \bar{b}_i = \sqrt[K_E]{\prod_k b_{ik}}, \quad (14)$$

где  $\bar{b}_i$  – результирующий приоритет  $i$ -го элемента;

$b_{ik}$  – приоритет  $i$ -го элемента, оцененный  $k$ -м экспертом;

$m$  – размерность матрицы парных сравнений;

$K_E$  – число экспертов.

Формирование экспертной группы (число экспертов) вычислим за формулой:

$$K_E \geq 0,5(0,33/\beta + 5), \quad (15)$$

где  $\beta$  – ошибка результата экспертного анализа или допустимая вероятность ошибки.

Согласованность полученных оценок определяется дважды. Сначала оценивается индекс согласованности оценок эксперта:

$$C_E = \frac{\lambda_{k_{\max}} - m}{m - 1}, \quad (9)$$

где  $\lambda_{k_{\max}}$  – максимальное собственное число матрицы парных сравнений  $k$ -го эксперта;  $m$  – размерность матрицы парных сравнений.

Оценки эксперта считаются согласованными, если отношение согласованности  $CR = C_E/CIS$ , где  $CIS$  – среднее значение индекса согласованности, определяемый в диапазонах (табл. 13.1).

Согласованность мнений группы экспертов определяется по правилу трех сигм. несогласованные оценки не учитываются при расчете результирующего вектора приоритетов  $\bar{B} = (\bar{b}_1, \bar{b}_2, \dots, \bar{b}_m)^T$ .

Таблица 13.1

**Значения  $CIS$  и  $CR$  от  $m$**

$m$	3	4	5	6	7	8	9	10	11	12
$CIS$	0,58	0,90	1,12	1,24	1,32	1,41	1,45	1,49	1,51	1,48
$CR$	[0;0,05]	[0;0,08]	[0;0,1]							

Доверительный интервал  $\delta_i$  определяется по формуле:

$$\delta_i = t_{cm} \times \sigma_{gi} / \sqrt{K_E}, \quad (16)$$

где  $t_{cm}=0,95$  – критерий Стьюдента;

$\sigma_{gi}$  – геометрическое стандартное отклонение.

Для построения метрик угроз на основе синергетического подхода, предложенного в работе [113] воспользуемся подходом построения классификатора угроз на основе информационно-аналитической модели метода двойных троек, предложенного авторами в работе [70]. В отличие от известного при построении классификатора содержательная часть каждой из четырех платформ включает в себя соответственно: *первая платформа* – классификация угроз по отношению к составным обеспечения безопасности БИИ в АБС ОБС: информационная безопасность (ИБ) (01), безопасность информации (БИ) (02), кибербезопасность (КБр) (03).

*вторая платформа* – классификация угроз по характеру направлений: нормативно-правовое (01), организационное (02), инженерно-техническое (03);

*третья платформа* – классификация угроз в соответствии с основными особенностями информации: конфиденциальность (01), целостность (02), доступность (03), аутентичность (04);

*четвертая платформа* – классификация угроз по уровням иерархии инфраструктуры АБС: *FL* – физический уровень (01), *NL* – сетевой уровень (02), *OSL* – уровень операционных систем (ОС) (03), *DBL* – уровень систем управления базами данных (04), *BL* – уровень банковских технологических приложений и сервисов (05). На рис. 13.3 приведена взаимосвязь структурной схемы классификатора угроз с АБС ОБС.

Впервые методологию построения классификатора угроз, принципы и методику представления, семантику и систему кодирования различных классов угроз государственных информационных ресурсов (ГИР), а также классификатор для первого широкого класса угроз ГИР, сформированных на основе нормативно-правового направления представлены в работах Юдина О.К., Бучика С.С. В работе [25] предлагается модифицированный классификатор, основанный на синергетическом подходе к формированию моделей нарушителя и оценки угроз, с учетом специфики угроз и инфраструктуры в АБС ОБС.

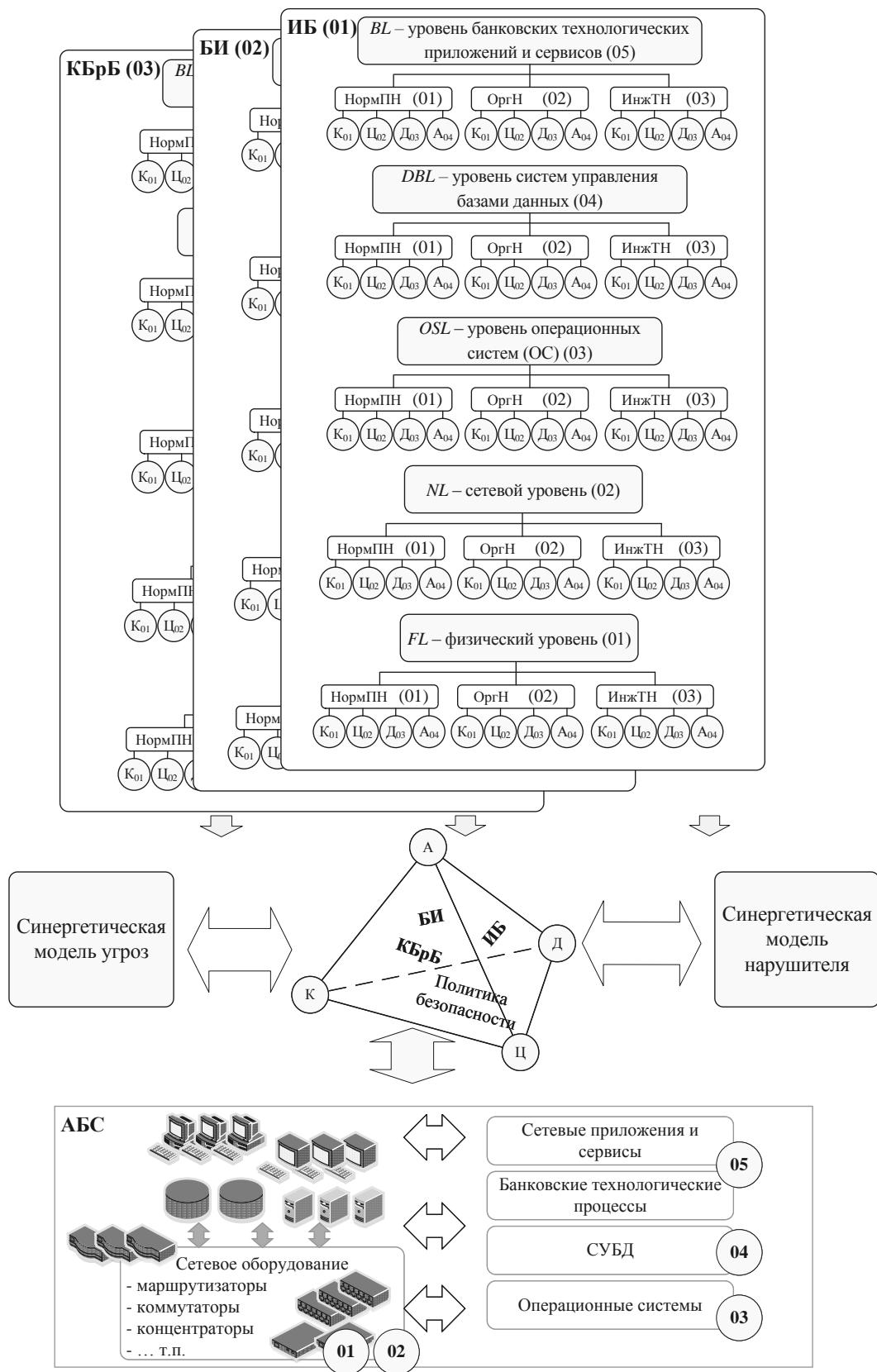


Рис. 13.3. Взаимосвязь структурной схемы классификатора угроз с АБС ОБС

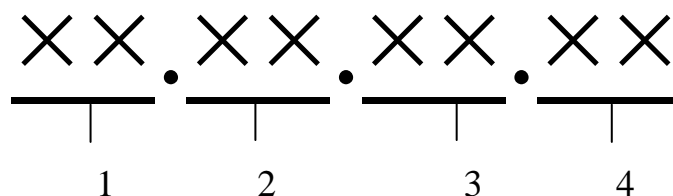
Описание модифицированного классификатора угроз состоит из четырех числовых величин:

- составная обеспечения безопасности БИИ в АБС ОБС: информационная безопасность (ИБ) (01), безопасность информации (БИ) (02), кибербезопасность (КБр) (03);

- характер направлений: нормативно-правовое (01), организационное (02), инженерно-техническое (03);

- основные особенности информации: конфиденциальность (01), целостность (02), доступность (03), аутентичность (04);

- уровни иерархии инфраструктуры АБС: *FL* – физический уровень (01), *NL* – сетевой уровень (02), *OSL* – уровень операционных систем (ОС) (03), *DBL* – уровень систем управления базами данных (04), *BL* – уровень банковских технологических приложений и сервисов (05). Части классификатора разделяются точкой и имеют вид, представленный на рис. 13.4.



**Рис. 13.4. Части классификатора**

(1 – синергетическая составная обеспечения безопасности БИИ, 2 – характер направлений; 3 – особенности информации; 4 – уровни иерархии инфраструктуры АБС).

Представленная классификация позволяет сформировать соответствующие метрики угроз и превентивных защитных мер.

На основании полученных от модели угроз данных осуществляется соотношение угроз с возможностями нарушителя той или иной категории. Перечень угроз классифицируется на основе уровней иерархии информационной структуры ( $DF_i$ ). Для определения связей между категориями нарушителей уровнями иерархии АБС задается матрица  $A^{DF} = \|a_{ij}^{DF}\|$ , в которой  $a_{ij}^{DF} = 1$ , если источник угроз  $DF_i$  может реализовать угрозу в отношении  $j$ -го актива АБС  $O_l \in \{O^{ABS}\}$ , а иначе  $a_{ij}^{DF} = 0$ . Для описания модели оценки

защищенности АБС воспользуемся методологией оценивания безопасности ИТ АБС [23], формально предлагается следующая модель:

$$G_{OZ}^{ABS} = \left\{ \{I_A\}, \{O^{ABS}\}, \{DF^{ABS}\}, \{RR^{ABS}\}, \{SZ^{ABS}\}, \{ROZ^{ABS}\}, \{UZ_r^{ABS}\} \right\}, \quad (17)$$

где  $\{I_A\}$  – множество элементов информационных активов;

- $\{O^{ABS}\}$  – множество объектов среды, описывающих элементы АБС и их принадлежность к уровням иерархии ИКП;
- $\{DF^{ABS}\}$  – множество источников угроз безопасности АБС;
- $\{RR^{ABS}\}$  – множество требований регуляторов к обеспечению безопасности БИИ;
- $\{SZ^{ABS}\}$  – множество возможных ТСЗИ;
- $\{ROZ^{ABS}\}$  – данные учета о результатах оценки защищенности АБС;
- $\{UZ_r^{ABS}\}$  – уровень защищенности АБС.

Для определения связей между угрозами и информационными активами используется матрица бинарных отношений  $A^{DF} = \|a_{ij}^{DF}\|$ , при этом  $\forall j \in \{I_A\}$ , а  $\forall i \in \{DF_i\}$ .

$$\|A^{DF}\| = \begin{cases} 1, & \text{если для } j\text{-го информационного актива существует } i \text{ угроза} \\ 0, & \text{если для } j\text{-го информационного актива не существует } i \text{ угроза} \end{cases}. \quad (18)$$

Каждый механизм защиты БИИ в АБС  $SZ_i \in \{SZ^{ABS}\}$  характеризуется вектором  $SZ_i = (T_{SZ}, T_V, C_{SZ})$ , где  $T_{SZ}$  – тип средства защиты,  $T_V$  – время внедрения,  $C_{SZ}$  – стоимость. Для описания связи между угрозами и ТСЗИ используется матрица  $A^{DFSZ} = \|a_{ij}^{DFSZ}\|$ , где  $a_{ij}^{DFSZ}$  – отображает наличие связи между  $i$ -й угрозой нарушения безопасности  $DF_i \in \{DF^{ABS}\}$  и  $j$ -м ТСЗИ  $SZ_j \in \{SZ^{ABS}\}$ . В модели предлагается использовать следующие типы связей:

$MZ$  – имеется механизм защиты, обеспечивающий противодействие ее деструктивному воздействию  $VH_i \in \{VH\}$ ;

$NMZ$  – нет механизма защиты, для обеспечения противодействия  $i$ -й угрозы.

При этом  $a_{ij}^{DFSZ} \in \{MZ, NMZ\}$ ,  $MZ$ ,  $NMZ$  – наличие связи типа, определенного между  $i$ -й угрозой и  $j$ -м ТСЗИ. Для элементов матрицы значения определяются по правилу:

$$\|a_{ij}^{DFSZ}\| = \begin{cases} MZ, & \text{если } i \text{ угроза рззакрывается } j\text{-м ТСЗИ} \\ NMZ, & \text{если } i \text{ угроза рззакрывается } j\text{-м ТСЗИ} \end{cases}. \quad (19)$$

Если для всех  $i = m$   $a_{mj}^{DFSZ} = NMZ$ , то делается вывод что ТСЗИ АБС не способны защитить БИИ от данного деструктивного воздействия, и для повышения уровня защищенности АБС необходимо внедрить дополнительные средства и механизмы защиты.

Множество требований регуляторов  $\{RR^{ABS}\}$  включает в себя требования к обеспечению безопасности БИИ –  $\{R_{BBI}\}$ , определенных в международных и национальных стандартах, систематизация источников представлена в работе [98], множества оценок степени выполнения требований безопасности –  $\{OV_{BBI}\}$ , итоговый уровень соответствия безопасности БИИ требованиям из множества  $\{R_{BBI}\}$  –  $\{IU_{BBI}\}$  и определяется:

$$\{RR^{ABS}\} = \{R_{BBI}\} \cup \{OV_{BBI}\} \cup \{IU_{BBI}\}. \quad (20)$$

Общий показатель уровня защищенности АБС, позволяющий оценить, уровень соответствия ТСЗИ требованиям регуляторов, на основе комплексного подхода оценки рисков и синергетической модели угроз определяется по формуле:

$$OPZ^{ABS} = \sum_{i=1}^k OPZ_i, \quad (21)$$

где  $k$  – количество частных показателей безопасности,  $OPZ_i$  – частный показатель, принимающий значения из множества  $\{0, 1\}$ , принимающий значения в соответствии с правилами [98]:

$OPZ_1$  – отсутствие недопустимых рисков, в случае если в организации банковского сектора (ОБС) при составлении модели угроз/модели нарушителя и оценки рисков выявлены недопустимые по своему уровню риски, то  $OPZ_1 = 0$ , в противном случае –  $OPZ_1 = 1$ ;

$OPZ_2$  – отсутствие опасных угроз, незакрытых механизмами и ТСЗИ,  $OPZ_2 = 0$ , в случае, если в ОБС при составлении модели выявлены “незакрытые” угрозы –  $OPZ_2 = 1$ ;

$OPZ_3$  – уровень соответствия безопасности БИИ требованиям регуляторов признан рекомендуемым –  $OPZ_3 = 1$ , в случае, если признан нерекондуемым –  $OPZ_3 = 0$ .

На основании полученных данных системе присваивается один из трех уровней защищенности  $UZ^{ABS} = \{\text{низкий, средний, высокий}\}$  в соответствии с правилом:

$$UZ^{ABS} = \begin{cases} \text{высокий, если } OPZ^{ABS} = 3; \\ \text{средний, если } 1 \leq OPZ^{ABS} \leq 3; \\ \text{низкий, если } OPZ^{ABS} = 0. \end{cases} \quad (22)$$

Полученная в результате аудита оценка защищенности БИИ позволяет определить наиболее ценные информационные активы, эффективность используемых средств для их защиты, а также степень соответствия системы ТСЗИ ОБС требованиям к защите и уровню защищенности регуляторов, выявить наиболее уязвимые места и выработать рекомендации по повышению, в случае необходимости, защищенности АБС ОБС.

Для оценки экономической целесообразности внедрения того или иного механизма ТСЗИ в АБС ОБС в зависимости от ценности БИИ, циркулируемой в АБС введем следующие обозначения:

$V_{BIn}^{ABS}$  – ценность БИИ для ОБС (стороны, обладающей информацией, и пытающейся ее защитить),  $V_{BIn}^{IA}$  – ценность БИИ для атакующей стороны (пытающейся добыть информацию);

$SZ^{ABS}$  – средства возможных ТСЗИ;  $SV^{AS} = \{SV^{ASIB}, SV^{ASBI}, SV^{ASKBr}\}$  – средства, выделяемые на добывание БИИ,  $SV^{ASIB}$  – средства взлома механизмов и ТСЗИ информационной безопасности,  $SV^{ASBI}$  – средства взлома механизмов и ТСЗИ безопасности информации;  $SV^{ASKBr}$  – средства взлома механизмов и ТСЗИ кибербезопасности:

$$SV^{AS} = \{SV^{ASIB}\} \cup \{SV^{ASBI}\} \cup \{SV^{ASKBr}\}; \quad (23)$$

$P_{Vj}$  – вероятность реализации хотя бы одной  $i$ -й угрозы  $j$ -му активу (вероятность успеха нападающей стороной);  $p_{Zj}$  – вероятность защиты от  $i$ -й угрозы  $j$ -му активу (вероятность успеха защищаемой стороной). Очевидным признается факт, что бессмысленно вкладывать средства в защиту или добывание информации больше, чем ценность БИИ:

$$SZ^{ABS} \leq V_{BIn}^{ABS}, \quad SV^{AS} \leq V_{BIn}^{ABS}. \quad (24)$$

Предположим, вероятности определяются по формулам:

$$p_{Zj} = \frac{q_Z \times SZ^{ABS}}{q_Z \times SZ^{ABS} + q_V \times SV^{AS}}, \quad (25)$$

$$p_{Vj} = \frac{q_V \times SV^{AS}}{q_V \times SV^{AS} + q_Z \times SZ^{ABS}}, \quad (26)$$



где  $q_Z, q_V$  – весовые коэффициенты, определяющие насколько каждая из сторон близка к цели.

Предположим, что сумма средств, выделенных атакующей стороной равна ценности БИИ, и ценность БИИ одинакова для обеих сторон, и противоборствующие стороны находятся в равных условиях, тогда экономическая стоимость затрат на защиту БИИ не должна превышать:

$$SZ^{ABS} = V_{BIИ}^{ABS} \times \frac{\sqrt{5}-1}{2}. \quad (27)$$

Эффективность предлагаемой модели оценки экономических затрат зависит от точности формулировки вероятности успеха защиты и определения ценности БИИ.

**Выводы.** Предложенная в работе синергетическая модель оценки безопасности банковской информации (БИИ) разработана на основе методологии и синергетическом подходе к обеспечению безопасности БИИ и оцениванию безопасности ИТ АБС ОБС позволяет переосмыслить подход построения политик безопасности БИИ на основе выявления эмерджентных свойств с использованием синергетической модели угроз, что позволяет комплексированно подходить к оценке рисков, с учетом главенствования киберугроз. Модель нарушителя позволяет строить типовые модели нарушителя в соответствии с требованиями регуляторов, при этом используется однозначная классификация нарушителей прав доступа, что позволяет избежать привлечения экспертов на этапе предпроектного обследования. Предложенный модифицированный классификатор угроз в АБС ОБС обеспечивает связь модели нарушителя с моделью угроз, позволяет сформировать соответствующие метрики угроз и превентивных защитных мер, семантику и систему кодирования различных классов угроз в АБС ОБС.

## РОЗДІЛ 14

# МОДЕЛІ ПОБУДОВИ ПОЛІТИКИ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ ПІДПРИЄМСТВА

***Анотація.** Розглянуто основні моделі політик та типів політик інформаційної безпеки. В результаті цього було виявлено, що в залежності від особливостей організації можна реалізувати ту чи іншу модель політики безпеки, яка дозволить захистити дані компанії від несанкціонованого доступу, а також зменшить втрати компанії у випадку дій з боку співробітників компанії, які можуть нанести шкоду організації.*

***Ключові слова:** Політика безпеки, кібербезпека, моделі загроз, моделі безпеки, інформаційна безпека.*

***Abstract.** The problems of The main types of model policies and information security policy. As a result, it was found that depending on the characteristics of the organization can implement a particular model of security policy that will protect company data from unauthorized access and reduce losses in the case of actions by employees, which may harm the organization.*

***Keywords:** The Security policy, the cyber security, the threat model, the security model, the information security.*

**Вступ та постановка задачі.** Важливу роль в забезпеченні кібербезпеки будь-якої організації відіграє політика безпеки, яка організована в ній. Визначити інформацію, що потребує захисту та рівень втрат від несанкціонованого спотворення, руйнування чи поширення інформації з боку третіх осіб, під силу саме тій особі, що має чітке уявлення мети організації, а також розуміє ті умови, в яких вона функціонує.

Визначивши політику безпеки, потрібно вирішити питання стосовно технології, що буде використана для її реалізації в автоматизованому контурі. Розробка деякої формальної моделі, що дає можливість ефективно програмувати налюбій формальній мові, дозволяє реалізувати норми та правила політики безпеки, що сформульовані в термінах природної мови.

Політику безпеки інформаційної системи необхідно формалізувати з метою опису поглядів керуючої гілки компанії на суть загроз інформаційній безпеці організації, а також на технології, за допомогою яких можна забезпечити безпеку її інформаційних ресурсів. Для політики безпеки можна визначити дві основні частини. Перша з них – загальні принципи роботи з інформаційними ресурсами (базами даних) для кожної із категорій

користувачів. Друга – чітко визначені правила такої роботи. При побудові політики безпеки потрібно розуміти, що вона завжди буде компромісом між рівнем захищеності інформаційних ресурсів системи, який ми бажаємо отримати, тим, на скільки зручно користувачам буде працювати із системою і, звичайно тією витратою коштів, що необхідні для її експлуатації.

Оформлення політики безпеки повинно проводитись документально з розділенням на певну кількість рівнів управління. Документ, в якому визначені відповідальні за реалізацію політики безпеки, її цілі та структура повинен бути визначений для найвищої ланки керівництва. Деталізація основного документу відбувається адміністраторами безпеки інформаційних систем, які повинні враховувати принципи діяльності організації, наявність ресурсів, а також співвідношення важливості цілей, які прагне досягнути компанія, реалізувавши політику інформаційної безпеки. В результаті необхідно отримати детальні рішення, що будуть складатись із ясно визначених методів захисту різних типів ресурсів (технічних, інформаційних) та інструкцій, які визначають поведінку співробітників у певних ситуаціях.

На даний момент часу інформація відіграє важливу роль у функціонуванні будь-якої структури, тому стала актуальною задача, яка включає в себе запобігання несанкціонованого доступу до неї з боку сторонніх осіб. Для реалізації такої задачі необхідно контролювати канали, за якими інформація може бути вкрадена. Таких каналів декілька і головних з них є так званий інсайдер. Інсайдер – це людина, яка працює на ту компанію, інформація якої цікавить зловмисника. Для того щоб мінімізувати втрати від такої атаки, потрібно правильно проводити політику безпеки і надавати співробітникам доступ тільки до тієї інформації, що їм необхідна в роботі. Під політикою безпеки в даній статті мається на увазі саме політика інформаційної безпеки. На цей момент є кілька моделей побудови такої політики. Саме вони і будуть розглянуті в даній роботі

**Основна частина.** Почнемо розгляд з моделі *Bell-LaPadula (BLP)* [73]. Дана модель заснована на політиці конфіденційності і визначає поняття захищеного стану. Реалізується за рахунок надання визначених привілеїв користувачам. Це може бути тільки читання, тільки запис або ж і читання і запис.

Модель визначає доступ заснований на рівні доступу, який пов'язаний як з предметом так і з об'єктом.

У моделі доступ базується на двох головних властивостях: простий захищений стан і властивість із \*. Простий захищений стан відповідає за можливість читання, властивість із \* – за можливість запису.

Модель повністю описується наступними складовими:

Елементи – складові частини системи:

суб'єкти – активні об'єкти (користувачі, програми);

об'єкти – пасивні об'єкти (паролі, інші дані);

атрибути доступу – всілякі дії суб'єктів над об'єктами: читання, зміна, доповнення (без читання!), Пошук, виконання. Можуть бути різні варіації і доповнення до наявного списку;

рівні безпеки – визначають способи взаємодії між об'єктами та суб'єктами і є доповненням до них. Об'єкти мають рівень безпеки, що складається з однієї частини в той час, як рівень безпеки суб'єктів розділяється на два окремих рівні:

рівень доступу, що визначає можливість доступу суб'єкта до певного класу інформації: повністю секретно > секретно > конфіденційність > для загального користування. Більш високий рівень доступу для суб'єктів відкриває також їм доступ до інформації, що знаходиться на нижчому рівні доступу;

рівень категорій доступу – на цьому рівні суб'єкти отримують певну область доступу. Одному суб'єкту може належати кілька категорій, однак жоден суб'єкт не має доступу до категорій, що йому не належать. Компоненти - структури, повністю описують стан системи:

поточний стан доступу складають трійки суб'єкт – об'єкт + атрибути доступу;

ієрархія об'єктів визначає відносини головний-наступний в структурі об'єктів, складається з кореневих дерев і ізольованих точок. Головний об'єкт має рівень допуску > = рівню допуску подальшого об'єкта;

матриця допустимого доступу:

суб'єкти по рядках, об'єкти по одному стовпчику, поточні стану доступу на відповідних перетинах;

- функція рівня визначає рівень доступу для суб'єктів і об'єктів.

Властивості:

система працює за принципом поточний стан -> запит -> рішення -> подальший стан. Робота системи регулюється набором властивостей і правил;

властивість простої безпеки (*Simple Security Property*): якщо в поточному стані доступу присутній трійка суб'єкт-об'єкт-атрибути доступу і атрибути доступу дозволяють перегляд, то рівень доступу суб'єкта домінує над рівнем

доступу об'єкта, що означає те, що не можна переглядати об'єкти більш високого рівня;

властивість '\*': не можна записувати в об'єкти нижчого рівня;

властивість самостійного захисту (*Discretionary Security Property*): всі дії суб'єкта над об'єктом обмежені матрицею допустимого доступу. У той час, як перші дві властивості визначають по суті всі можливі напрямки передачі інформації, *DS* властивість визначає то, як контролюється доступ до системи в цілому;

мандатний контроль за доступом.

Правила визначають переходи системи з поточного стану в наступне (тобто задають всілякі пари запит-рішення з позитивним рішенням), в базовому наборі було 8 правил, далі список правил розширювався. Правила, додані в *MULTICS* моделі дозволили змінювати рівень безпеки об'єкта і додати можливість вільного контролю за доступом в систему.

Теореми і докази підтверджують захищеність системи. Основна теорема безпеки: якщо початковий стан системи є безпечним і всі наступні переходи системи з одного стану в інший є безпека, то система повністю безпеки.

В цілому модель *BLP* стала першою значною моделлю політики безпеки, яка застосовується для комп'ютерів, і до сих пір в зміненому вигляді застосовується у військовій галузі. Модель повністю формалізована математично. Основний упор в моделі робиться на конфіденційність, але крім неї фактично більше нічого не представлено. Ще з недоліків моделі варто відзначити неможливість передачі інформації від вищого рівня до нижніх, оскільки це значно знижує можливості управління суб'єктами. В рамках моделі можливе створення незахищених систем.

Наступна модель називається моделлю *Biba* [59]. Вона є першою спробою створення інтегрованої моделі. Основні відмінності від попередньої моделі в наявності рівнів інтеграції, і наявності додаткової властивості (властивості виклику). Дана властивість відповідає за можливість суб'єкта посилати сервісні запити. Інші властивості схожі з попередньою моделлю, тільки в них йде прив'язка до рівня інтеграції, на якому знаходиться об'єкт і суб'єкт (у попередній моделі рівні класифікації).

Для додаткового управління цілісністю введені понижуючі водяні знаки (порушують заборону на спілкування):

якщо суб'єкт читає об'єкт більш низького рівня, то його рівень цілісності знижується до рівня цілісності об'єкта;

якщо суб'єкт доповнює об'єкт більш високого рівня, то рівень цілісності об'єкта знижується до рівня цілісності суб'єкта.

В даній моделі наявні переваги і недоліки моделі *BLP*. Крім того модель має додаткові власні недоліки. Основний з них – це рівень цілісності, який практично неможливо контролювати. Такий рівень обмежує доступ суб'єктів до об'єктів в такому вигляді, що або обмеження ізолює рівні цілісності, або, якщо їх визначити, то відбувається постійне, неконтрольоване зниження цього рівня, що позбавляє його функціональності. На момент написання статті, модель *Biba* використовувалася тільки у військових організаціях.

Модель *Clark-Wilson (CW)* [30] переслідує три основні мети:

- запобігання можливості роботи зміни НЕ авторизованими користувачами;
- підтримка внутрішньої та зовнішньої узгодженості;
- запобігання авторизованих користувачів від прийняття неправомірних змін.

Реалізація моделі забезпечена за рахунок набору правил. Математичної формалізації для цієї моделі проведено не було. Особливість моделі у відсутності у суб'єктів прямого доступу до об'єктів. Між ними знаходиться рівень із програм. Саме вони отримують доступ до об'єкта у випадку необхідності. Чітко визначеного контролю за доступом до системи не має.

Контроль за доступом можна розділити на дві групи. В першій групі для кожного типу даних визначаються операції доступу, які можна над ним виробляти. Причому для кожної групи об'єктів визначається чіткий набір програм. В другій групі описуються ті операції доступу, які можуть бути зроблені конкретним суб'єктом. Причому для кожного суб'єкту визначається набір програм, до яких він має доступ.

Всі дані в моделі *CW* розділені на 2 класи:

необхідний елемент даних (*CDI*);

спонтанний елемент даних (*UDI*).

Взаємодію з обома типами даних регулює певний набір правил. Перше правило – всі *CDI* повинні знаходитись в достовірному стані в процесі роботи початкових процедур перевірки (*IVP*), причому саме вони повинні в цьому переконатися. Друге – необхідно забезпечити сертифікацію для всіх процедур зміни (*TP*). Перехід повинен відбуватися із достовірних *CDI* в достовірні. Для кожної процедури зміни визначається певний набір *CDI*. Третє – правила доступу повинні відповідати всім вимогам поділу обов'язків. Четверте –

процедури зміни можна записувати виключно до журналу, в якому відкрито доступ тільки на додавання. П'яте – якщо процедура зміни отримала на вхід спонтанний елемент даних, то вона повинна його відхилити чи перетворити на необхідний елемент даних.

Набір правил моделі *Clark-Wilson (CW)* розроблений таким чином, щоб в повній мірі була забезпечена безпека і підзвітність переходів у системі за рахунок вибору необхідного для такої ситуації режиму роботи з даними. Головне досягнення цих правил в порівнянні з моделлю Біба - поділ процедур з перевірки цілісності і процедур зміни. Дозволяє запобігти або виправити більшість нелегальних дій, що здійснюються зсередини комерційної організації.

Для посилення захисту в моделі *CW* був введений ще один набір правил (*Enforcements Rules*):

- система повинна підтримувати і захищати список (*Tpi: CD1a, CD1b, ...*), що зіставляє *TP* і *CDI* і сертифікуючий доступ до них;

- система повинна підтримувати і захищати список (*UserID, Tpi: CD1a, CD1b, ...*), що визначає, які *TP* користувач може виконувати;

- при запиті на виконання процедури зміни кожен користувач повинен пройти процедуру аутентифікації;

- тільки допущений до сертифікації правил доступу для *TP* суб'єкт може змінювати відповідні записи в списку. Цей суб'єкт не повинен мати прав на виконання даного *TP*. Права на зміну записів у списку має тільки суб'єкт, що допущений до сертифікації правил доступу для *TP*. Одночасно такому суб'єкту заборонено мати права на виконання даної *TP*.

За рахунок правил, що приведені вище ми отримуємо додатковий захист для процедур зміни. Модель не має чітко визначених недоліків крім одного. Таку модель досить складно реалізувати на практиці, тому що в ній не закладено методів реалізації наборів правил, які б дозволили накласти на неї роботи реальної системи.

Модель “Китайська стіна” [14].

У 1989 році Бювером і Нешем була розроблена модель “Китайська стіна”, спочатку замислювався як протилежність БЛП подібним моделям, але надалі це твердження було спростовано. Основна область застосування моделі – фінансові аналітичні організації, для яких важливо уникнути конфлікту інтересів.

Модель складається з наступних компонентів:

- суб'єкти – аналітики;

- об'єкти – дані на одного клієнта;

набір даних компанії – ставить у відповідність кожному об'єкту його набір даних компанії;

класи конфлікту інтересів компанії – суперники (приєднуються до кожного об'єкту конфліктуючих компаній);

теги – набір даних компанії і класи конфлікту інтересів;

інформація, що оздоровлює – не має обмежень до доступу;

Властивості:

До об'єкту можна отримати доступ тільки у випадку, якщо він вільний. Об'єкт вважається вільним у випадку, якщо ті об'єкти, які намагаються отримати до нього доступ, належать до одного набору даних чи знаходяться в різних класах конфлікту інтересів;

'\*' - властивість: суб'єкту буде заборонений доступ на запис до об'єкту тільки якщо у нього немає доступу на читання будь-якого іншого об'єкта, який знаходиться в іншому наборі даних компанії і нездоровий.

Ці властивості дозволяють уникнути прямої участі аналітичної організації в конфліктах компаній їх клієнтів, але допускають непряму участь: відомості про певний об'єкт можуть послідовно оновлювати (за рахунок своєї інформації) компанії – суперники, але в цьому вже не буде провини аналітичної компанії (*Indirect Information Flow*). Контроль за доступом до системи є проміжним між класичними *DAC* і *MAC*.

В цілому модель орієнтована на реалізацію в дуже окремому випадку (забезпечення захисту від одночасного доступу до даних) і не претендує на значну спільність, але в досить повній мірі реалізує закладені в неї вимоги в життя.

Модель Гогена-Мезігера [93].

Модель Гогена-Мезігера (*Goguen-Meseguer*), представлена ними в 1982 році, заснована на теорії автоматів. Суть цієї моделі в тому, що система може із одного дозволеного стану тільки в кілька інших. Суб'єкти і об'єкти в даній моделі захисту розбиваються на групи – домени. При необхідності змінити стан системи потрібно скористатись таблицею дозволів. В такій таблиці зазначені ті операції, які можуть виконувати суб'єкти, що знаходяться в одному домені над об'єкті, що розташовані іншому. Для зміни стану в даній моделі використовуються транзакції, що забезпечують цілісність системи.

Сазерлендський модель [15].

Сазерлендський (від англ. *Sutherland*) модель захисту, опублікована в 1986 році, заснована на взаємодії суб'єктів і потоків інформації. Так само як і в попередній моделі, тут використовується машина станів з безліччю дозволених



комбінацій станів і деяким набором початкових позицій. У даній моделі досліджується поведінка множинних композицій функцій переходу з одного стану в інший.

Дискреційна (матрична) модель [100].

Розглянемо так звану матричну модель захисту (її ще називають дискреційною моделлю). Вона, на момент написання статті була самою розповсюдженою на практиці. Стан системи захисту можна описати наступною трійкою (на основі термінів матричної моделі):

$$(S, O, M), \quad (1)$$

де  $S$  – множина суб'єктів, які є активними структурними елементами моделі;

$O$  – множина об'єктів доступу, є пасивними захищеними елементами моделі. Для ідентифікації об'єкта використовується його ім'я;

$M$  – матриця доступу. Для визначення права права доступу суб'єкта до об'єкта використовується значення елемента матриці  $M$ . Звернення до різних типів об'єктів доступу з боку суб'єкта необхідно здійснювати, керуючись правами доступу, в яких описані способи такого звернення. Зазвичай права доступу суб'єктів до файлових об'єктів визначають як читання ( $R$ ), запис ( $W$ ) і виконання ( $E$ ).

Аналіз рядка матриці доступу за зверненням суб'єкта до об'єкта береться за основу реалізації управління доступом. Він проводиться наступним чином, а саме: для перевірки обирається відповідний до об'єкта рядок матриці. Під час перевірки визначається наявність необхідних прав доступу для суб'єкта. По результатах перевірки робиться відбувається надання чи заборона доступу.

Наочність і гнучкість налаштувань політики доступу до ресурсів у матричних моделях є їх великим плюсом, але на противагу йому встає ряд недоліків такої моделі. До них можна віднести зайвий деталізований рівень опису відносин суб'єктів і об'єктів. Він призводить до підвищення складності адміністрування системи захисту під час завдання параметрів і їх підтримці в актуальному стані при включені до схеми розмежування доступу нових елементів (об'єктів чи суб'єктів, чи і тих і інших одночасно). В результаті ми отримуємо ризик допустити достатньо багато помилок при адмініструванні. Виходячи з цього цей недолік можна назвати основним для дискреційної моделі.

Модель *Take-Grant* [99].

Модель *Take-Grant* – є формальною моделлю. Її використовують в області комп'ютерної безпеки, при необхідності аналізу систем дискреційного

розмежування доступу. Вона дозволяє підтвердити або спростувати ступень захищеності системи обраної автоматизованої системи, що повинна задовольняти регламентованим вимогам. Модель створена у вигляді направленої графу, вузли якого можуть бути або об'єктами, або суб'єктами. З'єднують їх маркіровані дуги, значення яких вказують права, що надані об'єкту чи суб'єкту. Домінуючими у моделі є два правила (давати та брати).

В цілому існує 4 правила перетворення:

- право "брати";
- право "давати";
- право "створити";
- право "видалити".

Використовуючи ці правила, можна відтворити стани, в яких буде перебувати система в залежності від розподілу і зміни прав доступу. Отже, можна проаналізувати можливі загрози для даної системи.

Звичайна модель.

$O$  – множина об'єктів (файли, сегменти пам'яті і т.д.).

$S$  – множина суб'єктів (користувачі, процеси системи).

$R = \{r1, r2, r3, r4, \dots, rn\} \cup \{t, g\}$  – множина прав доступу.

$t$  [take] - право брати "права доступу".

$g$  [grant] – право давати "права доступу".

$G = (S, O, E)$  – кінцевий, позначений, орієнтований граф без петель.

$\times$  – об'єкти, елементи множини  $O$ .

$\bullet$  – суб'єкти, елементи множини  $S$ .

$E \in O \times O \times R$  - дуги графа. Стан системи описується її графом.

Правило "Брати".

Брати = take ( $r, x, y, s$ ),  $r \in R$ .

Нехай  $s \in S, x, y \in O$  – вершини графа  $G$ .

Тоді граф  $G$  (рис. 14.1).

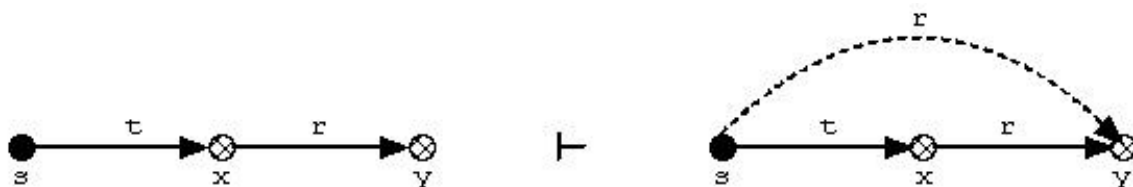


Рис. 14.1 Суб'єкт  $S$  бере у об'єкта  $X$  права  $r$  на об'єкт  $Y$

Правило "Давати".

Давати = grant ( $r, x, y, s$ ),  $r \in R$ .

Нехай  $s \in S$ ,  $x, y \in O$  – вершини графа  $G$ .  
Тоді граф  $G$  (рис. 14.2).

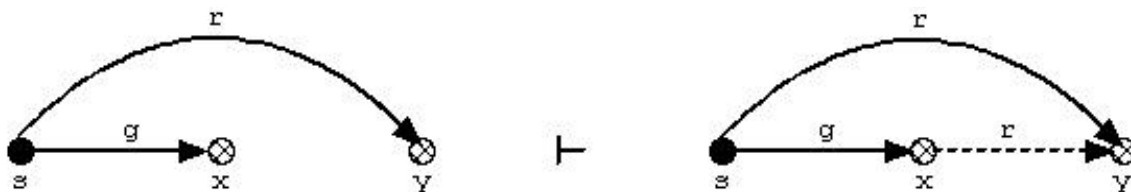


Рис. 14.2. Суб'єкт  $S$  надає об'єкту  $X$  права  $r$  на об'єкт  $Y$

Правило “Створити”.

Створити = *create* ( $r, x, s$ ),  $r \in R$ .

Нехай  $s \in S$ ,  $x, y \in O$  – вершини графа  $G$ .

Тоді граф  $G$  (рис. 14.3).



Рис. 14.3. Суб'єкт  $S$  бере  $r$ -доступний об'єкт  $Y$

Правило “Видалити”

Видалити = *remove* ( $r, x, s$ ),  $r \in R$ ,

Нехай  $s \in S$ ,  $x, y \in O$  – вершини графа  $G$

Тоді граф  $G$  (рис 14.4).

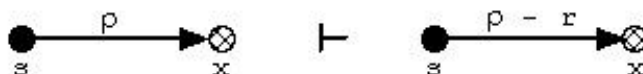


Рис. 14.4. Суб'єкт  $S$  видаляє  $r$ -доступний об'єкт  $Y$

На даний момент ця модель практично не використовується в програмному забезпечення. Однак, в розподіленій операційній системі *EI* дана модель успішно реалізована. На даний момент, в основному, дана модель використовується для аналізу уразливості різних систем. Вже написано досить велика кількість статей і літератури, де описані методи аналізу уразливості систем програмного забезпечення і мереж за допомогою моделі *Take-Grant*.

Модель типізованої матриці доступу (Модель Харрісона-Руззо-Ульмана)

Дана модель реалізує дискреційне (довільне) управління доступом суб'єктів до об'єктів і контроль за поширенням прав доступу.

Критерій безпеки моделі Харрісона-Руззо-Ульмана формулюється в такий спосіб:

Для заданої системи початковий стан є безпечним щодо права  $r$ , якщо не існує застосовної до послідовності команд, в результаті якої право  $r$  буде занесено в осередок пам'яті матриці  $M$ , в якій воно було відсутнє в стані.

Зміст даного критерію полягає в тому, що для безпечної конфігурації системи суб'єкт ніколи не отримує право  $r$  доступу до об'єкта, якщо він не мав його спочатку.

В результаті видалення суб'єкта або об'єкта видаляються всі права у відповідному рядку (стовпці) матриці, але розмір матриці при цьому залишається сталим

Видалення суб'єкта або об'єкта призводить до знищення всіх прав у відповідному рядку або стовпці матриці, але не тягне за собою знищення самого стовпчика або рядка і скорочення розміру матриці. Отже, якщо в якийсь осередку в початковому стані існувало право  $r$ , і після видалення суб'єкта або об'єкта, до яких відносилось це право, осередок буде очищена, але згодом з'явиться знову (в результаті створення суб'єкта або об'єкта), і в цей осередок з допомогою відповідної команди *enter* знову буде занесено право  $r$ , то це не буде означати порушення безпеки.

З критерію безпеки слід, що для даної моделі ключову роль грає вибір значень прав доступу та їх використання в умовах команд. Хоча модель не накладає ніяких обмежень на зміст прав і вважає їх рівнозначними, ті з них, які беруть участь в умовах виконання команд, фактично являють собою не права доступу до об'єктів, а права керування доступом, або права на здійснення модифікацій осередків матриці доступу.

Таким чином, дана модель описує не тільки доступ суб'єктів до об'єктів, а й розподіл прав доступу від суб'єкта до суб'єкта, тому що саме зміна змісту осередків матриці доступу визначає можливість виконання команд (в тому числі команд, що модифікують саму матрицю доступу), які потенційно можуть призвести до порушення критерію безпеки.

Позитивні сторони моделі Харрісона-Руззо-Ульмана [18].

простота реалізації, що досягається шляхом використання простих алгоритмів;

висока ефективність в управлінні, за рахунок точного управління повноваженнями користувача;;

критерій безпеки даної моделі є досить сильним в практичному плані (тому що дозволяє гарантувати відсутність певної інформації для користувачів, яким спочатку не видані відповідні повноваження).

Недоліки моделі Харрісона-Руззо-Ульмана:

доведено, що в загальному випадку не існує алгоритму, який може для довільної системи, її початкового стану і загального правила  $r$  вирішити, чи є дана інформація безпечною;

існує вразливість до атаки за допомогою “троянського коня” (тому що в дискреційних моделях контролюються тільки операції доступу суб’єктів до об’єктів, а не потоки інформації між ними).

Основні моделі політик інформаційної безпеки розглянуто, тепер перейдемо до моделей типів політик безпеки.

Існують наступні типи політик безпеки: дискреційна, мандатна та рольова [18].

Основою дискреційної (дискретної) політики безпеки є дискреційне управління доступом (*Discretionary Access Control, DAC*), яке визначається двома властивостями:

всі суб’єкти і об’єкти повинні бути ідентифіковані;

права доступу суб’єкта до об’єкта системи визначаються на підставі деякого зовнішнього по відношенню до системи правила.

Дискреційна політика безпека є досить простою якщо потрібно реалізувати певні механізми захисту, тому це можна віднести до її переваги. Тому, на цей момент часу, автоматизовані системи (АСУ БС, АСУ СН) забезпечують виконання положень саме цієї політики безпеки.

До переваг дискреційної політики безпеки можна віднести відносно просту реалізацію відповідних механізмів захисту. Цим обумовлений той факт, що більшість поширених в даний час автоматизовані системи, в тому числі і АСУВ (АСУ БС, АСУ СН), забезпечують виконання положень саме цієї політики безпеки.

В якості прикладу реалізацій дискреційної політики безпеки в АС можна привести матрицю доступів, рядки якої відповідають суб’єктам системи, а стовпці – об’єктам; елементи матриці характеризують права доступу. Модель є статичною, що є її значним недоліком, тому що автоматична система може змінюватись з певною динамікою, а політикою безпеки цей факт не враховується.

Також необхідно звернути увагу на те, що використання такої моделі потребує відповіді на запитання: як поширювати права доступу і аналізувати їх вплив на безпеку АС. У загальному випадку при використанні даної політики безпеки перед монітором безпеки об’єктів (МБО), який при санкціонуванні доступу суб’єкта до об’єкта керується деяким набором правил, варто вирішити алгоритмічно нерозв’язну задачу: перевірити чи приведуть його дії до порушення безпеки чи ні.

У той же час є моделі АС, що реалізують дискреційну політику безпеки (наприклад, модель *Take-Grant*), які надають алгоритми перевірки безпеки.

Однак вони не є тим механізмом, який би дозволив реалізувати ясну і чітку систему захисту інформації в АС, що обумовлює пошук більш досконалих політик безпеки.

Основу мандатної (повноважної) політики безпеки складає мандатний управління доступом (Mandatory Access Control - MAC), що має на увазі, що:

всі суб'єкти і об'єкти системи повинні бути однозначно ідентифіковані;

задає лінійно упорядкований набір міток секретності;

кожному об'єкту системи привласнена мітка секретності, яка визначає цінність міститься в ньому інформації – його рівень секретності в АС;

кожному суб'єкту системи присвоєна мітка секретності, яка визначає рівень довіри до нього в АС – максимальне значення мітки секретності об'єктів, до яких суб'єкт має доступ; мітка секретності суб'єкта називається його рівнем доступу.

Запобігання витоку інформації від об'єктів, що мають високий рівень доступу до тих, чий рівень нижче є основною метою мандатної політики безпеки. Отже така політика безпеки запобігає виникненню в АС інформаційних каналів зверху вниз.

Найчастіше мандатну політику безпеки описують в термінах, поняттях і визначеннях властивостей моделі Белла-Лапалуда. В рамках даної моделі доводиться важливе твердження, яке вказує на принципову відмінність систем, що реалізують мандатний захист, від систем з дискреційним захистом: якщо початковий стан системи безпечно, і всі переходи системи зі стану в стан не порушують обмежень, сформульованих політикою безпеки, то будь-який стан системи безпечно.

На додаток до того, що сказано вище потрібно відзначити, що АС з мандатною політикою безпеки є більш надійні, ніж АС з дискреційною політикою. Це досягається за рахунок відслідковування з боку МБО стану самої АС, а ні як в дискреційній моделі тільки правила доступу суб'єктів системи до об'єктів.

Коли мова йде про велику кількість користувачів, то традиційні підсистеми управління доступом потрібно адмініструвати, використовуючи об'єктно-орієнтовані рішення, що дозволяють знизити складність адміністрування. Це необхідна міра, тому що число зв'язків пропорційно добутку кількості користувачів на кількість об'єктів, що робить процес адміністрування надскладною задачею.

Є кілька різновидів об'єктно-орієнтованих рішень. Наприклад: рольове управління доступом (РУД). Реалізується воно шляхом додавання проміжних сутностей (ролей) між користувачами та їх привілеями. Це дозволяє користувачу в різні проміжки часу мати різні права, за рахунок зміни ролі. Один користувач може мати кілька ролей.

При використанні рольового доступу можна спростити процес адміністрування системи, оскільки збільшення його складності при зростанні кількості користувачів відбувається значно повільніше. Досягається це за рахунок абстрагування від конкретних видів і способів перевірки прав користувачів та встановлення зв'язків між ролями. Ролей в такому випадку потрібно значно менше, ніж користувачів. Відповідно число зв'язків, які потрібно адмініструвати стає пропорційним сумі, а не добутку кількості користувачів і об'єктів.

У 2001 році Національний інститут стандартів і технологій США запропонував проект стандарту рольового управління доступом [14].

Створивши роль, ми можемо визначити для неї права доступу та зв'язати з нею користувача. За рахунок цього між користувачем та його правами отримаємо відносини “багато до багатьох”. Великій кількості користувачів можна надати одну і ту саму роль або кілька ролей одному користувачу. Коли користувач починає сеанс роботи, то активізуються одразу всі його ролі. В результаті він отримує усі права, від кожної із ролей, разом. Кожен користувач в один момент часу може мати не обмежену кількість сеансів.

У випадку необхідності можна встановити спадкування між ролями. Воно дозволяє ролі, що є спадкоємицею отримати всі права від ролі спадкодавця і доповнити її тими правами, які необхідні для комплексної реалізації створеної ролі. Можна провести відповідність ролей у РУД та класів у об'єктно орієнтованому програмуванні (ООП). Права доступу у РУД, в такому випадку, відповідають методам класів в ООП, а користувачам – об'єкти.

Робити спадкування ми можемо з будь-якою глибиною ієрархії, причому права доступу будуть більшими для тих користувачів, у яких ця глибина більша. Причому кожна роль може успадковувати права у будь-якої кількості ролей і надавати у спадок свої права також не обмеженій кількості інших ролей.

Спадкування починається з найвищого рівня абстрагування. При побудові політики інформаційної безпеки вищою ланкою абстракції буде співробітник компанії, а далі, в залежності від особливостей роботи будуть створюватись додаткові ролі з більш розширеними правами. Надаючи права тій чи іншій ролі, необхідно чітко розуміти ті обов'язки, які вона виконує. Роль керівника не означає необмежені права. Вони в такій ролі просто не потрібні.

Важливим принципом інформаційної безпеки є поділ обов'язків. Він може бути як статичним, так і динамічним.

При статичному розподілу ролей користувач, фіксовано приписується до однієї ролі і не може бути після цього приписаний до множини інших ролей. В такому розподілі реалізація відбувається в наступному вигляді, а саме: створюється визначена кількість ролей із однаковими правами (не менше двох) і до них приписується визначена кількість співробітників (також не менше

двох). Таким чином створюється пара: багато ролей – число. Наприклад в компанії може існувати три ролі адміністратора і десять адміністраторів. Тоді число буде дорівнювати чотирьом.

Додаючи успадкування, потрібно слідкувати за тим, щоб кількість співробітників, що отримують ролі не перевищила ту, що зафіксована у політиці безпеки організації, причому відслідковувати її необхідно по всій ієрархічній гілці.

У динамічному розподіленні обов'язків є одна особливість, що відрізняє його від статичного. При ньому розглядаються ролі, що одночасно активні для даного користувача. Наприклад один користувач може мати ролі вантажника і водія, але не одночасно. Водій повинен привезти машину на місце, підготувати її для розвантаження. Потім перевдягнутися у форму вантажника і розвантажити її. Причому у формі вантажника він не може вести автомобіль. Так ми отримаємо тимчасове обмеження довіри за рахунок надання мінімальних привілеїв. Стандарт регламентує три функції, що необхідні для адміністрування РУД. До них відносяться функції адміністрування, що включають в себе створення нових ролей та супровід існуючих. Наприклад створення чи видалення ролі чи користувача, надання користувачеві право участі у асоціації; створення нової асоціації чи видалення існуючої; створення нового відношення спадкоємства між ролями чи видалення існуючого; створення нової ролі, як спадкоємиці існуючої чи навпаки попередниці; видалення обмежень для статичного чи динамічного поділу обов'язків. Додатково до них можна віднести допоміжні функції, що використовуються для обслуговування сеансів роботи користувача. Наприклад активація нової ролі користувача, перегляд роботи існуючого користувача, перевірка правомірності доступу.

**Висновки.** В роботі було розглянути основні моделі політик та типів політик інформаційної безпеки. В результаті цього, можна зробити висновок, що в залежності від особливостей організації можна реалізувати ту чи іншу модель політики, яка дозволить захистити дані компанії від несанкціонованого доступу, а також зменшить втрати компанії у випадку дій з боку співробітників компанії, які можуть нанести шкоду організації.

Таким чином, наведена інформація може використовуватись як основа для розробки політики безпеки не тільки приватних підприємств але і вищих навчальних закладів

Для реалізації цієї мети авторами було обрано три моделі. Наприклад, для розробки політики безпеки ВНЗ Ми будемо використовувати третю з них, а саме модель CW, тому що ми робимо припущення, що ця модель буде найбільш ефективною.



## РОЗДІЛ 15

# РАЗРАБОТКА И ИССЛЕДОВАНИЕ МЕТОДОВ РЕВЕРСИВНЫХ ЦИФРОВЫХ ЗНАКОВ (ЦВЗ) ДЛЯ ВЕРИФИКАЦИИ МЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ

*Аннотация.* Рассмотрены метод идентификации и верификации медицинских диагностических изображений, основанный на реверсивных ЦВЗ. Метод был реализован и протестирован с использованием реальных данных. На основании метода разработана система позволяющая проводить дальнейшие исследования.

**Ключевые слова:** Медицинская информация, цифровой водяной знак, стегосигнал, стегообраз, стегосообщение.

**Abstract.** This paper focuses the method of authentication and verification of medical diagnostic images, based on reversible digital watermarking. A method was realized and tested with the use of the real data. Basis on the method is the system which worked out allowing to do further researches.

**Keywords:** Medical information, The digital water mark, steganographic signal, steganographic image, steganographic message.

**Введение и постановка задачи.** Конфиденциальность и целостность медицинской информации, как правило, регламентируется законодательными, как впрочем, и этическими нормами. В настоящее время темпы развития информационных технологий, медицинского диагностического оборудования явно опережают разработку соответствующих законодательных норм. Вышесказанное обусловило необходимость развития технологий защиты медицинской информации, представленной в цифровом виде.

После определения подоптимальной структуры системы, выработки требований к основным параметрам, обеспечивающих достижение заданных значений вероятностей ошибок детектирования ЦВЗ и надежности восприятия стегосигнала в условиях простейших преобразований стегосигнала, представляется возможным адаптировать систему относительно более сложных и характерных для конкретного практического применения преобразований стегосигнала в канале обработки и/или несанкционированного пользователя. Но даже для простейших преобразований стегосообщения необходимо с учетом отклонения от принципа Кергоффа с точки зрения теоретико-игрового подхода рассматривать различные режимы работы системы с ЦВЗ, динамику изменения показателей системы при достижении баланса между двумя сторонами (разработчик системы и несанкционированный пользователь), либо временного

преимущества какой-либо из сторон [103]. Адаптации подоптимальных систем с ЦВЗ для практического использования в условиях более сложных преобразований канала обработки подразумевает коррекцию структуры системы или ее отдельных составляющих, изменение параметров, вероятностных мер ЦВЗ, позволяющих не снизить показатели системы, полученные при оценке в условиях простейших преобразований.

В общем случае цели разработчика системы с ЦВЗ: минимизация вероятностей ошибок детектирования ЦВЗ; увеличение скорости передачи, пропускной способности, сохранение на заданном уровне надежности восприятия стегообраза, минимизация отношения энергии одного бита ЦВЗ к спектральной мощности шума, минимизация конструктивной сложности, вычислительной нагрузки и стоимости системы и т.д.

Для решаемой задачи верификации медицинских изображений особую важность приобретают вопросы восстановления изображения после процедуры верификации и аутентификации.

**Основная часть.** При решении задач верификации технологии ЦВЗ в отличие от решающих аналогичные задачи криптографических методов, а именно, формирования цифровой подписи (ЦП), не увеличивают размер сообщения, не могут быть просто отброшены без ухудшения надежности восприятия стегосообщения. С другой стороны, в качестве ЦВЗ можно использовать ЦП некоторого сообщения при выполнении требований по надежности восприятия стегосообщения и устойчивости к преобразованиям стегосигнала, определяемыми особенностями практического использования.

Основная проблема верификации на основе ЦП, заключающаяся в несанкционированном удалении ЦП, для ЦВЗ не актуальна. Однако возникает другая проблема: ЦВЗ должны быть инвертируемы, т.е. необходимо точное восстановление исходного основного покрывающего сообщения после детектирования.

При использовании технологий ЦВЗ для контроля над копированием, предотвращения несанкционированного копирования, мониторинга рекламного вещания и т.д. полагается, что некоторые изменения основного покрывающего сообщения (ОПС) в результате погружения ЦВЗ являются допустимыми и на приемной части восстановление ОПС не требуется. Однако для подтверждения целостности, в частности в медицинском менеджменте, криминалистике, требуется точное восстановление исходного ОПС. Например, если ОПС содержит  $N$  бит и после процедуры сжатия останется только  $L$  бит, то в  $N-L$  бит ОПС допустимо погружать ЦВЗ, которые будут утрачены при сжатии

(полухрупкие системы с ЦВЗ). Практическое применение такого подхода ограничено [103].

Модульный детектор. Для решения данной проблемы рассмотрим возможность формирования и детектирования ЦВЗ на основе модульной арифметики. С точки зрения повышения эффективности детектирования ЦВЗ представляется интересным рассмотреть в данном практическом приложении использование кодов, исправляющих ошибки.

Стеганографическое сообщение  $S$  в случае одномерного представления ОПС при модульном погружении ЦВЗ

$$s(n) = (c(n) + w(n)) \bmod 256, \quad n \in A_n, \quad (1)$$

где  $s(n) \in S$ ,  $c(n) \in C$ ,  $w(n) \in W$ ,  $A_N = \{1, \dots, N\}$  и без потери общности исследований в качестве ОПС рассматриваются монохромные изображения с градацией яркости одного пикселя  $0 \dots 255$ .

Для восстановления ОПС на приемной части выполняется операция вычитания по модулю из принятого стегосообщения секретного ключа в виде ЦВЗ:

$$c(n) = (s(n) - w(n)) \bmod 256, \quad n \in A_n. \quad (2)$$

В отличие от обычной арифметики при модульном сложении исключается эффект усечения при превышении порога яркости. Однако погружение на основе (1) характеризуется следующими недостатками:

- появление мелкоструктурного шума на фрагментах стегообраза;
- непригодность преобразований (1), (2) для изображений, чьи гистограммы яркости близки к равномерным

Искажения в виде мелкоструктурного шума зависят от типа ОПС и могут быть устранены соответствующими предварительными преобразованиями гистограммы яркости ОПС [114]. Второй недостаток также может быть учтен и скорректирован посредством предварительной обработки ОПС и изменения гистограммы ОПС [114].

Для ОПС в виде монохромного изображения, каждый пиксель которого представлен 8 битами с градацией яркости в диапазоне от 0 до 255, можно сформировать ЦП, используя произвольные стандарты хеширования и криптографической обработки. Формирование ЦВЗ в виде ЦП ОПС

производится на основе какого-либо асимметричного криптографического стандарта, например RSA [58].

Таким образом, для процедуры идентификации и верификации на основе ЦП, погруженной в ОПС как ЦВЗ необходимо выполнение следующих требований:

- отличия стегосообщения и ОПС не должны превышать заданный уровень в соответствии с требуемой надежностью восприятия;
- на основе стегосообщения при наличии СК в виде ЦВЗ восстановление ОПС должно характеризоваться очень высокой вероятностью.

Использование кодов, исправляющих ошибки, позволит повысить эффективность детектирования ЦВЗ. Рассмотрим некоторый бинарный систематический исправляющий ошибки код  $Z \in (m, L, d)$ , где  $m$  – размер блока,  $L$  – число информационных бит,  $d$  – минимальное кодовое расстояние.  $b_{ij}$  является  $i$ -м битом  $j$ -го блока кода, причем  $i = 1, \dots, m, j = 1, \dots, 2^L$ . Если в ЦП всего  $R$  бит, то для погружения ЦП необходимо  $R_o = R/L$  кодовых блоков. Причем для погружения одного кодового символа ЦВЗ, являющихся ЦП ОПС, потребуется  $N_o = N/R_o m = NL/Rm$  пикселей ОПС. Правило кодового блока можно представить в виде:

$$s(n) = (c(n) + \alpha(-1)^{\left\lfloor \frac{n}{n_o} \right\rfloor + 1, j} w_r(n)) \bmod 256, \quad n = 1, \dots, N, \quad n_o = 1, \dots, N_o, \quad (3)$$

где  $\alpha$  – целое число,  $0 \leq \alpha \leq 255$ ,  $w_r(n) = \{\pm 1\}$  – ПСП, полученная, например, с помощью линейного рекуррентного регистра сдвига [114].

Предположим, что сложение в (3) выполняется не в модульной, а в обычной арифметике и без потери общности передачу и прием кодового блока из  $R/L$  символов ЦП. В структуре секретной системы с неинформированным детектором в качестве СК используется ЦВЗ и предполагается синхронизированность приемной и передающей частей. Корреляционный детектор принимает решение о том, какое кодовое  $j$ -е кодовое слово принято на основании определения максимального значения функционала детектора

$$j = \text{Arg max}_{0 \leq j_1 \leq 2^L} \sum_{n=1}^{\frac{NL}{R}} (s(n) - E\{C\}) \alpha(-1)^{\left\lfloor \frac{n}{n_o} \right\rfloor + 1, j_1} w_r(n), \quad n \in A_n \quad (4)$$

где ОПС или рассматриваемый фрагмент ОПС предполагается стационарным процессом в широком смысле.

При погружении ЦВЗ по правилу (3) детектор (4) не обеспечивает оптимальное детектирование. Однако для частного случая ОПС в виде стационарного гауссова процесса и обычном аддитивном погружении ЛКД является оптимальным. Используя аналогию с теорией широкополосных систем связи и на основе центральной предельной теоремы, верхняя граница вероятности ложного обнаружения кодового блока

$$P_{fd} \leq (2^L - 1)Q\left(\frac{\alpha}{\sigma_c} \sqrt{\frac{NLd}{Rm}}\right) = (2^L - 1)Q\left(\frac{\alpha}{\sigma_c} \sqrt{\frac{N}{R} Vd}\right), \quad (5)$$

где  $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} \exp\left(-\frac{t^2}{2}\right) dt$ ;

$\sigma_c^2$  – дисперсия ОПС;

$V = L/m$  – скорость кодирования для кода  $Z$ .

Для ЛКД значение (5) не зависит от гистограммы яркости ОПС, но зависит от значения дисперсии ОПС  $\sigma_c^2$ . С другой стороны, анализ (5) позволяет сделать вывод о том, что эффективность детектирования ЦВЗ для выбранного кода  $Z$  в первом приближении определяется значением произведения  $Vd$  (табл. 15.1). Например, если число пикселей ОПС  $N = 64 \times 10^3$ , число бит для ЦП  $R=64$ , требование к вероятности ошибки  $P_{fd} \leq 10^{-3}$ , то для тривиального кода  $Z: \{m, m, 1\}$ , т.е.  $Vd=1$ , необходимо 64 блока с  $10^3$  пикселями ОПС для каждого. Тогда вероятность неправильного приема каждого блока будет не выше  $1,56 \times 10^{-5}$  и отношение сигнал/шум после погружения ЦВЗ в ОПС  $\eta_w = \sigma_c^2/\alpha^2$  составит 55,5. При использовании кода Голя  $Z: \{m, m, 1\} = \{24, 12, 8\}$ ,  $Vd=4$  для обеспечения той же вероятности ошибки обнаружения кодового блока отношение сигнал/шум после погружения составит  $\eta_w = 160$ , что значительно лучше с точки зрения надежности восприятия стегосигнала, чем для системы с ЦВЗ без использования кодов, исправляющих ошибки.

Если при формировании стегосообщения используется оператор модульного сложения (3), то для оценки вероятности  $P_{fd}$  выражение (5)

не корректно, поскольку процедура обнаружения по правилу (4) при использовании модульной арифметики в общем случае ухудшается. С другой стороны, только использование модульных операций позволяет точно восстанавливать ОПС при правильном декодировании кодового слова  $b_{1j}b_{2j}\dots b_{Lj}$ . Для разрешения данного противоречия необходимо адаптировать алгоритм детектирования системы с ЦВЗ при аддитивном модульном погружении ЦВЗ (3).

Таблица 15.1

**Сравнительная оценка кодов**

Код	Тривиальный	Хэмминга	Хэмминга	Хэмминга	Голея	БЧХ
$(m, L, d)$	$\{m, m, 1\}$	$\{8, 4, 4\}$	$\{255, 8, 128\}$	$\{1024, 10, 512\}$	$\{24, 12, 8\}$	$\{63, 45, 6\}$
$Vd$	1	2	4	5	4	4

Для простоты изложения рассмотрим систему с нулевым битом, т.е. (3) преобразуется к виду:

$$s(n) = (c(n) + \alpha(-1)^b w_r(n)) \bmod 256, n \in A_n, \tag{6}$$

где  $b \in \{0, 1\}$ .

Прореженный детектор. Адаптация алгоритма при модульном алгоритме формирования стегосигнала требует изменения структуры детектора. Алгоритм прореженного детектирования (ПД) основан на том, что для  $n$  пикселей результаты модульных и обычных операций будут совпадать.

Обозначим

$$F_\alpha = \{n \leq N \mid (s(n) - \alpha) \bmod 256 \leq 255 - \alpha; (s(n) + \alpha) \bmod 256 \geq \alpha\}.$$

Тогда правило обнаружения преобразуется к виду:

$$\Lambda^{ПД} = \sum_{n \in F_\alpha} (s(n) - C_0)w(n), n \in A_N \Rightarrow b = \begin{cases} 1, & \text{если } \Lambda \geq 0, \\ 0, & \text{если } \Lambda \leq 0, \end{cases} \tag{7}$$

где  $C_0 = E\{C\}$  является средним значением пикселей ОПС  $c(n)$ ,  $n \in F_\alpha$ .

С учетом центральной предельной теоремы оценка вероятности ошибки такого обнаружителя

$$P_e = Q\left(\frac{\alpha}{\sigma_c} \sqrt{N\mu}\right), \quad (8)$$

где  $\mu = \frac{N_o}{N}$ ,  $N_o = |F_\alpha|$ .

В частном случае изображения с равномерной гистограммой яркости в диапазоне  $0 \dots 255$ , т.е. когда

$$\forall \Delta = 0, 1, \dots, 255, n = 1, \dots, N: \text{Pr ob}(c(n) = \Delta) = \frac{1}{256} \quad (9)$$

получим

$$\mu = \frac{256 - 2\alpha}{256} = 1 - \frac{\alpha}{128}, \quad \sigma_c^2 = \frac{255^2}{12} = 5418. \quad (10)$$

В результате подстановки (10) в (8) можно сделать вывод, что для некоторого приемлемого значения  $\alpha$  вероятность ошибки  $P_e$  устремится к нулю при увеличении  $N$ , т.е.  $\lim_{N \rightarrow \infty} P_e = 0$ . Однако данный факт противоречит известному свойству равномерного распределения на некотором интервале  $\{0, \dots, J-1\}$ : если к равномерно распределенной величине добавить или отнять по модулю  $J$  некоторую константу, то результирующая величина так же характеризуется равномерным распределением. Из данного свойства очевидно, что для изображений с равномерной гистограммой яркости детектор не сможет различать гипотезы  $b=1$  и  $b=0$ . Решением данной парадоксальной ситуации и является алгоритм прореженного детектора (ПД). Если вероятность ошибки детектирования ЦВЗ оценить выражением

$$P_e = Q\left(\frac{E\{\Lambda^{ПД}\}}{\sqrt{\text{Var}\{\Lambda^{ПД}\}}}\right), \quad (11)$$

где  $E\{\Lambda^{ПД}\}$ ,  $\text{Var}\{\Lambda^{ПД}\}$  – математическое ожидание и дисперсия функционала ПД, соответственно, причем для  $b=0$ :

$$E\{\Lambda_0^{ПД}\} = N\mu \left[ \alpha + \frac{\sum_{c(n) \in A_C} c(n)P(C) - \sum_{c(n) \in D_C} c(n)P(C)}{1 + \sum_{c(n) \in B} P(C)} \right], \quad (12)$$

для  $b = 1$ :

$$E\{\Lambda_1^{ПД}\} = -E\{\Lambda_0^{ПД}\}, \quad (13)$$

для  $b = 1$  и  $b = 0$ :

$$\begin{aligned} \text{Var}\{\Lambda_0^{ПД}\} &= \text{Var}\{\Lambda_1^{ПД}\} = \text{Var}\{\Lambda^{ПД}\}, \\ \text{Var}\{\Lambda^{ПД}\} &= \frac{N\mu}{1 + \sum_{c(n) \in B_C} P(C)} \left[ \left( \sum_{c(n)=0}^{255} c(n)^2 P(C) + \sum_{c(n) \in B_C} c^2(n) P(C) \right) - 2C_o \left( C_o + \sum_{c(n) \in B_C} c^2(n) P(C) \right) - \right. \\ &\left. 2C_o \alpha \left( \sum_{c(n) \in A_C} P(C) - \sum_{c(n) \in D_C} P(C) \right) - \frac{\left[ \sum_{c(n) \in A_C} c(n) P(C) - \sum_{c(n) \in D_C} c(n) P(C) \right]^2}{1 + \sum_{c(n) \in B_C} P(C)} \right], \quad (14) \end{aligned}$$

где  $P(C)$  – вероятность распределения (гистограмма) ОПС;

$$A_C = \{0, 1, \dots, 255 - 2\alpha\};$$

$$B_C = \{2\alpha, 2\alpha + 1, \dots, 255 - 2\alpha\};$$

$$D_C = \{2\alpha, 2\alpha + 1, \dots, 255\}.$$

Для ОПС с равномерной гистограммой яркости математические ожидания (12) и (13) будут нулевыми и события  $b = 1$  и  $b = 0$ , не различимы. Но для ОПС с другими типами гистограмм использование структуры с ПД будет более эффективным, чем ЛКД, определенного для пространства  $A_N = \{1, \dots, N\}$ . Для детектирования с обычным корреляционным детектором, т.е. при использовании структуры с ЛКД при  $n \in A_N$  оценка вероятности ошибочного детектирования ЦВЗ (11) справедлива, но изменяются аналитические оценки математического ожидания и дисперсии функционала детектора  $\Lambda^{ЛКД}$ , а именно, для  $b = 0$ :

$$E\{\Lambda_0^{ЛКД}\} = N \left[ \alpha - 128 \left( \sum_{c(n) \in D_C} P(C) + \sum_{c(n) \in A_C} P(C) \right) \right] \quad (15)$$

для  $b = 1$ :



$$E\{\Lambda_1^{JKD}\} = -E\{\Lambda_0^{JKD}\} \quad (16)$$

для  $b = 0$ ,  $b = 1$ :

$$\begin{aligned} \text{Var}\{\Lambda_0^{JKD}\} &= \text{Var}\{\Lambda_1^{JKD}\} = \text{Var}\{\Lambda^{JKD}\} \\ \text{Var}\{\Lambda^{JKD}\} &= N \left[ \sigma_c^2 + \frac{256^2}{2} (\sum_{c(n) \in A_N} P(C) + \sum_{c(n) \in D_N} P(C)) - \right. \end{aligned} \quad (17)$$

$$\left. - 256 (\sum_{c(n) \in A_N} (c(n) - C_0) P(C) - \sum_{c(n) \in D_N} (c(n) - C_0) P(C)) - 128^2 [\sum_{c(n) \in A_N} P(C) + \sum_{c(n) \in D_N} P(C)]^2 \right]$$

где  $A_c = \{0, 1, \dots, \alpha - 1\}$ ;

$$D_c = \{255 - \alpha + 1, \dots, 255\}.$$

Для ОПС с гистограммой яркости (9)  $E\{\Lambda_0^{JKD}\} = E\{\Lambda_1^{JKD}\} = 0$  и гипотезы  $b = 1$  и  $b = 0$  не различимы.

Описанные алгоритмы прореженного и обычного детектирования не являются оптимальными, даже при ОПС с гауссовым распределением, если погружение ЦВЗ осуществлено на основе (6). В соответствии с критерием максимального правдоподобия алгоритм принятия решения оптимального модульного детектора (МД)

$$\begin{aligned} \Lambda^{MD} &= \sum_{n \leq N} (((s(n) - \alpha w(n)) \bmod 256) - C_0)^2 - \sum_{n \leq N} (((s(n) + \alpha w(n)) \bmod 256) - C_0)^2 \Rightarrow \\ \Rightarrow b &= \begin{cases} 0, & \text{если } \Lambda^{MD} < 0, \\ 1, & \text{если } \Lambda^{MD} \geq 0. \end{cases} \end{aligned} \quad (18)$$

Для МД (18) справедлива оценка (11), но после коррекции выражений для математического ожидания и дисперсии функционала детектора, а именно, для  $b = 0$ :

$$\begin{aligned} E\{\Lambda_0^{MD}\} &= N \left[ -4\alpha^2 - \left( \frac{256^2}{2} - 512\alpha - 256C_0 \right) \right. \\ &\left. \left[ \sum_{c(n) \in A_c} P(C) - \left( \frac{256^2}{2} - 512\alpha + 256C_0 \right) \sum_{c(n) \in B_c} P(C) - \right. \right. \end{aligned}$$

$$\left. - 256(\sum_{c(n) \in A_C} c(n)P(C) - \sum_{c(n) \in B_C} c(n)P(C)) \right] \quad (19)$$

для  $b = 1$ :

$$E\{\Lambda_1^{MD}\} = -E\{\Lambda_0^{MD}\} \text{ для } b = 1, \text{ для } b = 0:$$

$$Var\{\Lambda_0^{MD}\} = Var\{\Lambda_1^{MD}\} = Var\{\Lambda^{MD}\}$$

$$Var(\Lambda^{MD}) = \frac{N}{2} \left[ \sum_{c(n)=0}^{255} [(c(n) - C_0)^2 - (c(n) + 2\alpha) \bmod 256 - C_0]^2 P(C) + \right. \\ \left. + \sum_{c(n)=0}^{255} [(c(n) - C_0)^2 - (c(n) - 2\alpha) \bmod 256 - C_0]^2 P(C) \right]^2 - \frac{1}{N^2} [E\{\Lambda_0^{MD}\}]^2, \quad (21)$$

где  $A_C = \{0, 1, \dots, 2\alpha - 1\}$ ;

$$B_C = \{255 - 2\alpha + 1, \dots, 255\}.$$

В отличие от (14) и (17) расчет (20) возможен численными методами. Подстановка (9) в (18) и (19) приводит к нулевому результату, как и ранее, т.е. для ОПС с плоскими гистограммами яркости различение гипотез  $b = 1$  и  $b = 0$  не возможно. Но для других типов гистограмм ОПС при использовании модульного детектора достигается более высокая эффективность, чем при использовании прореженного детектора.

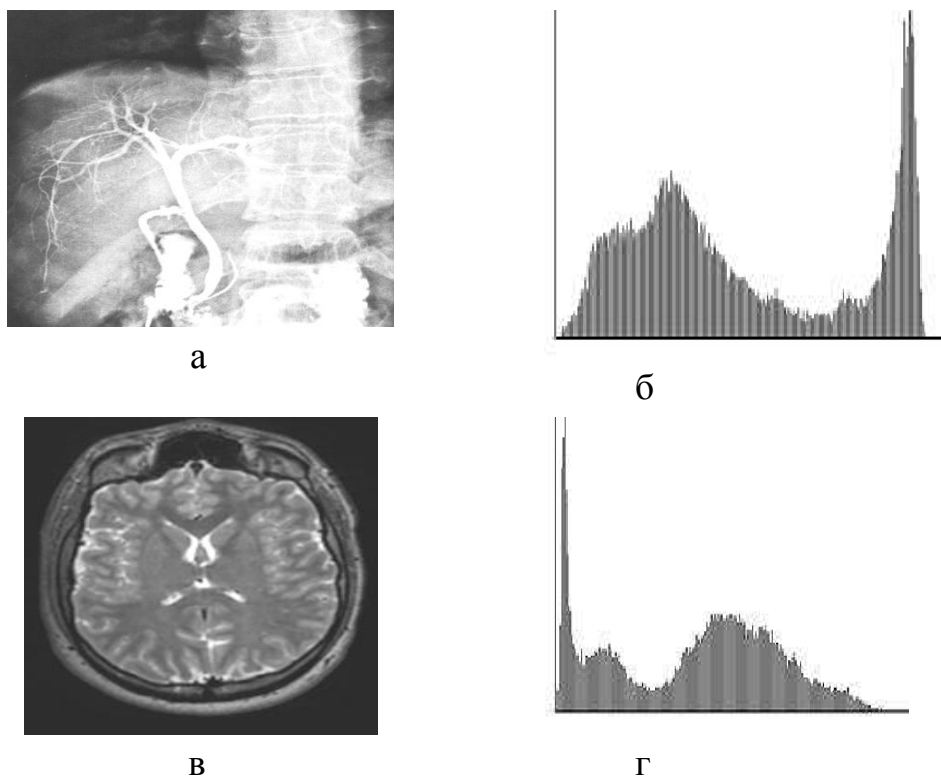
Таблица 15.2

**Оценка вероятности ошибки детектирования ЦВЗ на основе имитационного моделирования**

$P_e/\alpha$	Детектор	Параметр ЦВЗ $\alpha$					
		3	4	5	6	10	30
$P_e$ , ОПС-1	ЛКД	0,027	0,01	0,004	0,001	0.00003	0,002
	ПД	0,03	0,013	0,004	0,001	0.00001	0,00001
	МД	0,032	0,014	0,005	0,002	0,00002	0,001
$P_e$ , ОПС-2	ЛКД	0,074	0,128	0,348	0,541	0,592	0,297
	ПД	0,067	0,123	0,353	0,535	0,597	0,301
	МД	0,02	0,04	0,1	0,17	0,13	0,09

Анализ полученных результатов (табл. 15.2) показывает, что для ОПС-2 эффективность детектирования при использовании МД значительно хуже, чем для структур с ЛКД и ПД, в то время как для ОПС-1 все три типа детекторов

демонстрируют практически одинаковый уровень эффективности детектирования ЦВЗ, что подтверждает теоретические исследования (рис. 15.1).



**Рис. 15.1. ОПС–1 (а), ОПС–2 (в) и гистограммы распределения яркости (б, г), соответственно**

Оценка вероятности ошибки при использовании ПД на основе (15), (16), (17) продемонстрировала существенную зависимость эффективности детектирования ЦВЗ от параметра  $\alpha$  (табл. 15.2).

Данный эксперимент выявил существенное отличие модульного алгоритма погружения от обычного. Вероятность ошибки при модульном погружении не является монотонной функцией интенсивности ЦВЗ и существует некоторое пороговое значение  $\alpha$ , для которого  $P_e$  максимально, что также подтверждает аналитические исследования. Результаты аналитического моделирования (табл. 15.3) подтверждаются результатами имитационного моделирования (табл. 15.2).

Таблица 15.3

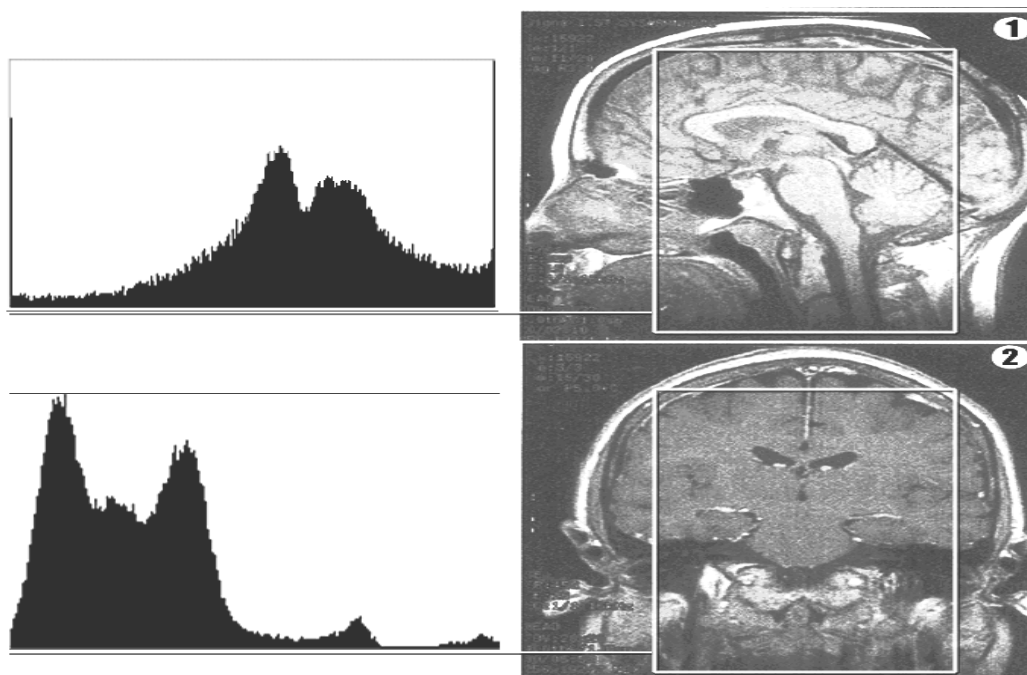
**Оценка вероятности ошибки детектирования ЦВЗ на основе аналитического моделирования**

$\alpha$	3	4	5	6	10	20	30
$P_e$	0,05	0,12	0,33	0,49	0,44	0,35	0,30

В качестве примера рассмотрена система с ЦВЗ при использовании МД,  $\alpha=3$ , генерация для каждого бита ЦВЗ 10 ПСП (чипов) длиной 1000. Для простоты эксперимента биты ЦВЗ полагались нулевыми, ошибка соответствовала появлению единицы после декодирования. Повторение ПСП при передаче бита ЦВЗ несколько раз позволяет существенно снизить число ошибочно декодированных бит. В декодере первоначально может реализовываться инвертирование одиночных битов стегосообщения после детектирования, на что потребуется при 64 бит ЦВЗ всего 64 итерации. Если результат верификации будет отрицательным, то потребуется инвертировать 2016 удвоенных бит, далее 41664 утроенных бит и т.д., что, однако, уже существенно удлинит и усложнит процедуру верификации.

Визуальная оценка стегосообщения при модульном погружении и  $\alpha=3$  (рис. 15.2) продемонстрировала наличие точечных искажений, которые, однако, будут отсутствовать после инвертирования ЦВЗ.

Конечно, существуют более оптимальные коды с исправлением ошибок. Однако их использование потребует еще большей избыточности относительно числа бит, необходимых для формирования ЦВЗ, что сделает стегосообщение более уязвимыми для атак оцениванием ЦВЗ. При моделировании были исследованы тесты для различных реализаций ЦВЗ, полученных в виде линейной или нелинейной функции некоторого секретного ключа.



**Рис. 15.2. Оценка надежности визуального восприятия при модульном формировании стегообраза,  $\alpha=3$**

При внедрении цифровых водяных знаков в изображение некоторые его элементы неизбежно претерпевают изменение. Таким образом, в существующих системах ЦВЗ идет речь о достоверном установлении источника распространения изображения, а не о его информационной целостности, которая нарушается уже в момент добавления в изображения какого-либо идентификатора.

Однако, для некоторых видов изображений, в частности, для медицинских снимков магниторезонансной томографии, гарантия целостности изображения является одним из приоритетных критериев системы ЦВЗ. Целостность изображения в таком случае может гарантироваться так же, как и для других типов файлов путем хеширования файла изображения. Хеш-суммы изображений могут храниться в отдельной защищенной базе данных, но для удобства системы также желательно, чтобы хеш, или любой другой идентификатор, гарантирующий целостность изображения, был внедрен в само изображение.

Проблема реализации указанной системы состоит в том, что такая система должна удовлетворять таким, на первый взгляд противоречивым, требованиям. Реализация такой системы возможна исключительно с применением реверсивных ЦВЗ – водяных знаков, алгоритм добавления которых предусматривает возможность полного восстановления изображения после извлечения скрытого идентификатора. Как правило, данный метод реализуется с помощью предварительного преобразования ОПС [58; 117].

Для решения этой проблемы предлагается метод внедрения реверсивных ЦВЗ в изображения формата BMP, а также система идентификации и верификации изображений медицинской диагностики, основанная на предложенном методе.

Изначально для каждого изображения необходимо сформировать его уникальный идентификатор, который может иметь произвольную форму. Эксперименты проводились с использованием 160-битного идентификатора, являющегося цифровой подписью SHA-1 исходного изображения. Затем изображение в виде матрицы значений яркости каждого пикселя подвергается предварительному преобразованию, в ходе которого оценивается дискриминационная функция преобразования уровня. Полученный идентификатор изображения погружается как невидимая цифровая метка в изображение, не увеличивая его объем. Процедуры идентификации и верификации включает следующие этапы: извлечение уникального идентификатора, погруженного в изображение, восстановление изображения, вычисление уникального

идентификатора восстановленного изображения и сравнение двух этих идентификаторов. При их совпадении целостность изображения подтверждается.

Также к системе предъявляется специфическое требование, заключающееся в том, что последовательные операции погружения и извлечения идентификатора не должны повлечь за собой изменения даже одного бита восстановленного изображения по отношению к исходному [117]. Уникальный идентификатор изображения **ID** как правило, содержит не менее, чем 64 бита. Особенность медицинского приложения заключается в том, что требуется обеспечить надежный уровень восприятия изображения до момента его восстановления: в состоянии хранения изображения с ЦВЗ врач может потребовать хранения некоторого фрагмента изображения в неизменном состоянии, но с обеспечением его верификации. При этом возможно формирование идентификатора для некоторой части изображения, и погружение этого идентификатора в оставшуюся часть, что, однако, потребует дополнительных усилий по сохранению основных параметров эффективности системы и надежности восприятия всего изображения.

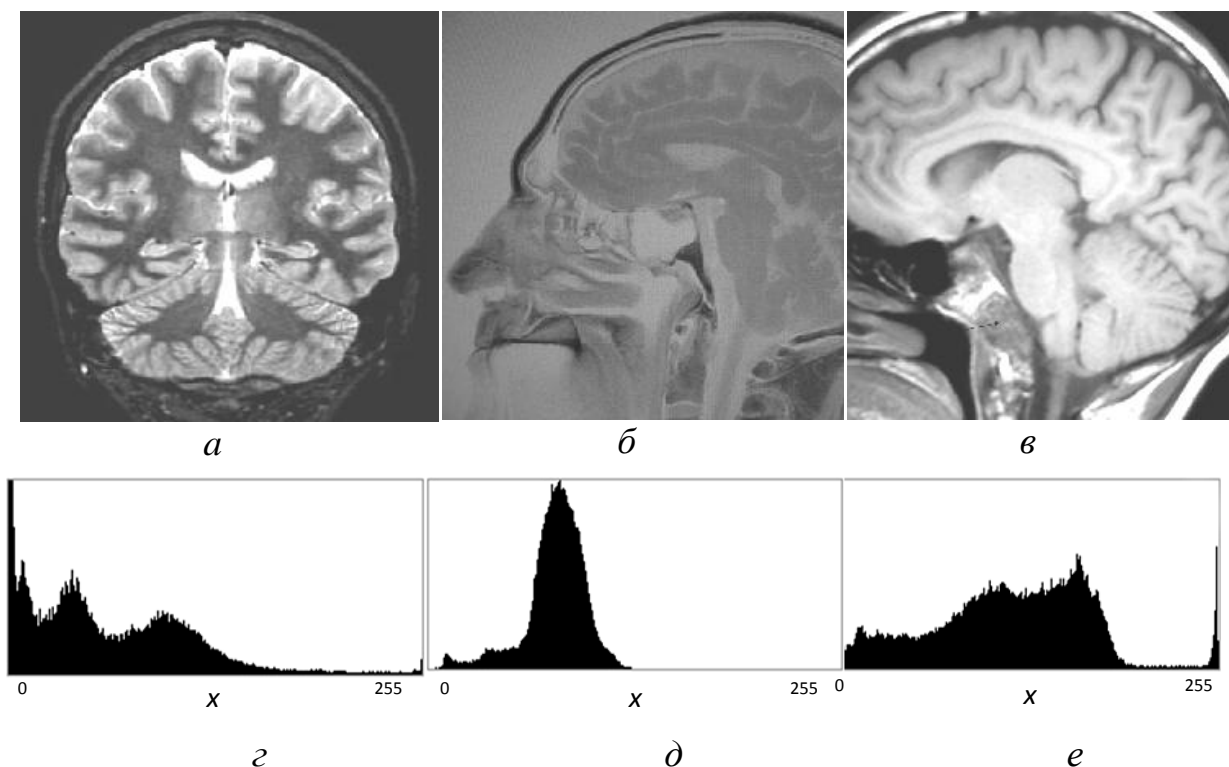
Предлагается метод погружения реверсивных ЦВЗ для идентификации и верификации изображений в формате BMP, позволяющий восстановить исходное изображение с точностью до одного бита. При этом возможно погружать каждый бит идентификатора как в отдельно взятый пиксель, так и в группу пикселей. Предложенная система ЦВЗ несекретна, т.е. не требуется знание секретного ключа. Другими словами, в системе не реализуется принцип Керкгоффа [57]. При извлечении ЦВЗ не требуется знание исходного изображения, т.е. детектор ЦВЗ является неинформированным.

На основе данного метода была реализована программная версия системы верификации и идентификации медицинских изображений. Программа разработана на платформе Microsoft .NET. В программе реализовано внедрение в изображение его hash-значения SHA1. Система апробирована для базы данных изображений диагностики мозга. Три изображения диагностики мозга и соответствующие им гистограммы распределения яркости представлены на рис. 15.3. Для арифметического кодирования вектора  $R$  использовалась программа адаптивного арифметического кодирования.

С точки зрения надежности восприятия изображения с погруженным идентификатором рассматривались амплитуды преобразования уровней  $A = 2, 4, 8$  и размеры групп  $N = 4, 16, 64$ .

Пропускная способность системы верификации и аутентификации зависит от значений параметров предварительного преобразования

$L, L_1, L_2, A, N_1, N_2$ , которые, в свою очередь, определяются гистограммой яркости изображения. Для светлых изображений верификация при обеспечении незаметности ЦВЗ возможна при  $N = 64, A = 4$ . Однако, для темных изображений верификация выполнима при  $N \leq 16, A \geq 6$  и для обеспечения незаметности ЦВЗ могут потребоваться дополнительные усилия.



**Рис. 15.3.** Исходные изображения размером  $256 \times 256$  пикселей (а, б, в), соответствующие гистограммы распределения яркости (г, д, е).

Очевидна зависимость обеспечения незаметности ЦВЗ от параметра преобразования ЦВЗ  $A$  и размера матрицы  $N_1 N_2$ . Для каждого изображения существует оптимальное соотношение данных параметров.

Разработанные в разделе метод верификации изображения может быть применен как в секретном режиме, так и при использовании секретного ключа и соответствующих криптографических алгоритмов, так и в открытом.

Однако, в ряде случаев вполне достаточным может быть проведение процедуры верификации без применения криптографических стандартов. Действительно, ранее указывалось, что после арифметического сжатия разница  $\Delta L = L - L_1$  соответствует числу бит, которые могут быть использованы для записи идентификатора изображения, длина которого  $L_2$ . Если  $L_2 < \Delta L$ , то возникает возможность погружения дополнительной

информации. Однако, для открытой системы верификации возможно использовать все  $\Delta L = L = L_1$  для сокрытого хранения конфиденциальной медицинской информации.

**Заключение.** Для верификации изображений на основе погружения ЦВЗ, представляющего собой ЦП данного изображения, использование алгоритма погружения на основе модульного сложения позволяет выполнять восстановление ОПС после детектирования ЦВЗ, но приводит к ухудшению эффективности детектирования ЦВЗ. Полученные аналитические выражения оценки вероятности ошибочного декодирования одного бита ЦВЗ при использовании трех типов детекторов: ЛКД и предлагаемых адаптивных (ПД, МД) позволили формализовать процедуру оптимизации. Выявлено важное свойство модульного аддитивного погружения, заключающееся в том, что вероятность ошибки не является монотонной функцией интенсивности ЦВЗ и существует некоторое оптимальное значение параметра  $\alpha$ , которому соответствует минимальное значение вероятности ошибки. Для всех типов детекторов ОПС в виде изображений с плоской гистограммой яркости не подлежат верификации без специальных мер предварительной обработки.

Важным выводом является то, что уже на этапе погружения ЦВЗ в виде ЦП представляется возможным оценить возможность верификации ОПС. Разработчик системы может изменять значение параметра  $\alpha$ , реализацию ЦВЗ, изменять гистограмму ОПС, если это необходимо и возможно [57]. Однако данные преобразования ОПС должны быть обратимыми и известными на приемной части системы. Чтобы не передавать дополнительную информацию возможно на приемной части использовать некоторую «кодую книгу», задающую типы преобразований, значения параметров. При этом процедура верификации потребует проведения тестов до получения наилучшего результата.

Предложен метод идентификации и верификации медицинских диагностических изображений, основанный на реверсивных ЦВЗ. Параметры преобразования, такие как размер группы преобразования  $N$  и параметр преобразования  $A$ , могут быть адаптивно настроены, что позволяет обеспечить необходимый уровень гибкости для достижения баланса между емкостью внедряемой в изображение информации и незаметностью внедрения. Метод был реализован и протестирован с использованием реальных данных. Результаты теста показывают, что он может быть использован в распределенной базе данных медицинских изображений, где критерии



целостности изображений и незаметности водяного знака являются определяющими.

На основании метода разработана система, реализующая преимущества описанного метода, и позволяющая проводить дальнейшие исследования с целью его дальнейшего развития и оптимизации. Представляется интересным рассмотреть возможности реализации различных режимов секретности процедуры верификации изображения.

Полученные в данной работе результаты имеют практическое значение [58; 117]. Использование алгоритма погружения на основе модульного сложения позволяет выполнять восстановление ОПС после детектирования ЦВЗ, но приводит к ухудшению эффективности детектирования ЦВЗ. Посредством аналитических исследований теоретически обоснована и проверена экспериментально оптимизация системы верификации ОПС на основе модульного способа погружения ЦВЗ, представляющего собой ЦП ОПС. Важным выводом является то, что уже на этапе погружения ЦВЗ в виде ЦП представляется возможным оптимизировать верификацию конкретного ОПС, уменьшить вероятности ошибок детектирования ЦВЗ более чем на два порядка (оптимизация  $\alpha$ , обратимые модификации гистограммы ОПС, выбор типа детектора). Выявлено важное свойство модульного аддитивного погружения, заключающееся в том, что вероятность ошибки не является монотонной функцией интенсивности ЦВЗ.

Предложен метод идентификации и верификации медицинских диагностических изображений, основанный на реверсивных ЦВЗ. Параметры преобразования, такие как размер группы преобразования  $N$  и параметр преобразования  $A$ , могут быть адаптивно настроены, что позволяет обеспечить необходимый уровень гибкости для достижения баланса между емкостью внедряемой в изображение информации и незаметностью внедрения. Метод был реализован и протестирован с использованием реальных данных. Результаты теста показывают, что он может быть использован в распределенной базе данных медицинских изображений, где критерии целостности изображений и незаметности водяного знака являются определяющими.

Рассмотрена возможность реализации секретного и открытого режимов верификации медицинских изображений.

## РОЗДІЛ 16

### РАЗРАБОТКА МЕТОДА УПРАВЛЕНИЯ РИСКАМИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

*Аннотация.* В данном разделе задача управления рисками разработки программного обеспечения при условии ограниченности средств (финансовых, технических и др.) выделенных на устранение ошибок безопасности, рассматривается в виде полумарковской модели принятия решений для управляемого процесса в непрерывном времени с критерием минимума расходов на устранение аномалий. Разработан метод управления рисками разработки программного обеспечения, отличающийся от известных использованием псевдобулевых методов бивалентного программирования с нелинейной целевой функцией и линейными ограничениями для определения оптимальной стратегии устранения эксплуатационных ошибок. В качестве примера рассмотрены ситуации возникновения ошибок безопасности программного обеспечения, и определена оптимальная стратегия управления для устранения указанной аномальной ситуации.

**Ключевые слова:** управление рисками, разработка программного обеспечения, псевдобулевые методы бивалентного программирования

*Abstract.* In this paper, the problem of risk management software development, provided the limited resources (financial, technical, etc.) allocated to the elimination of security bugs, regarded as a semi-Markov decision model for the controlled process in continuous time with a minimum criterion to eliminate anomalies costs. A risk management software development method, which differs from the known methods using pseudo bivalent programming with nonlinear objective function and linear constraints to determine the optimal strategy for elimination of operational errors. For example, consider the situation of occurrence of software security bugs and determined the optimal control strategy to eliminate this abnormal situation.

**Keywords:** risk management, software development, pseudo-dual-mode programming methods

**Введение и постановка задачи.** Проведенные исследования, а также анализ литературы [8; 16; 119; 55; 45] показали, что управление риском разработки ПО состоит в заблаговременном выявлении связанных с риском финансовых, технических, психологических, и др. опасностей, и принятии мер по снижению риска путем целенаправленного изменения этих факторов с учетом эффективности принимаемых мер. Управление риском разработки ПО включает систему мероприятий, осуществляемых как до проявления негативного события, так и после его реализации. Однако, как показали

исследования, превентивный анализ и учет большинства возможных эксплуатационных ошибок позволит снизить финансовые и др. затраты в жизненном цикле разработки ПО.

Ряд авторов [8; 16; 119; 55; 45] под термином “управление риском” понимают разработку и обоснование оптимальных программ деятельности, призванных эффективно реализовать решения в области обеспечения безопасности. При этом главным элементом такой деятельности является процесс оптимального распределения ограниченных ресурсов с учетом характерных эксплуатационных, экономических и социальных факторов.

Рассматриваемую задачу управления рисками разработки ПО при определенных ограничениях на мероприятия по тестированию качества и безопасности, сформулируем в виде полумарковской модели принятия решений для управляемого марковского процесса в непрерывном времени и дисконтированными доходами (с коэффициентом  $0 < \alpha < 1$  в нормальных условиях процесса создания ПО) или расходами (в условиях с отклонениями от плана, связанными с пренебрежением невыявления уязвимостей (ошибок) безопасности). При этом данный вид эксплуатационных рисков отождествляются с последовательно соединенными независимыми элементами, восстанавливаемыми за конечное время.

Оптимальную нерандомизированную стационарную стратегию управления определим с помощью псевдодобулевых методов бивалентного программирования, находя все решения системы ограничений. Эти решения определяются на основе алгоритма пересечения решений отдельных неравенств-ограничений, предложенного в работе [8; 16; 119; 55; 45] для нахождения базисных решений системы линейных неравенств с булевыми переменными.

В таких условиях сформулируем основную задачу. Пусть каждому состоянию  $i \in S$ , где  $S = \{0, 1, 2, \dots, N\}$  рассматриваемой системы управления рисками разработки ПО поставлено в соответствие конечное множество  $R_i$  решений, элементы которого обозначим как  $r = 1, 2, \dots, r_i$ . Если система находится в состоянии  $i \in S$  и принимается решение  $r \in R_i$ , то ее дальнейшее поведение определяется вероятностным законом

$$Y_{ij}^r(t) = P_{ij}^{(r)} F_{ij}^{(r)}(t), \quad j \in S, \quad (1)$$

где  $P_{ij}^{(r)}$  – вероятность перехода системы из состояния  $i$  в состояние  $j$ ;  $F_{ij}^{(r)}(t)$  – функция распределения времени пребывания системы в состоянии  $i$  при принятии решения  $r$  и при условии, что следующий переход произойдет в состояние  $j$ .

При этом сделаем допущение, что выполнены следующие условия:

Состояние  $i=0$  соответствует нормальному процессу разработки ПО, а  $i \neq 0$  – ситуация ошибки безопасности.

Функции  $F_{0j}^{(r)}(t)$  и  $F_{j0}^{(r)}(t)$ ,  $j \in \tilde{S} = S/\{0\}$ ,  $r \in R_j$ , вместе со своими первыми производными непрерывны при  $t > 0$ , за исключением конечного числа точек, и возрастают в соответствии с экспоненциальным законом распределения.

За единицу времени пребывания в состоянии  $i$  в случае принятия решения  $r$  тратится в среднем  $k_i^{(r)}$  средств (при  $i \neq 0$  число  $k_i^{(r)}$  отрицательно и равно издержкам системы за единицу времени пребывания в состоянии  $i$  при условии выхода из этого состояния с учетом решения  $r$ ).

Величины  $|k_i^{(r)}|$  ограничены при всех  $i \in S$ ,  $r \in R_i$  и вероятности  $P_i^{(r)}$  удовлетворяют соотношениям:

$$\sum_{j \in S} P_i^{(r)} = 1, \quad i \in S, \quad r \in R_i, \quad P_{ij}^{(r)} \geq 0, \quad i, j \in S, \quad r \in R_i.$$

Таким образом, в каждом состоянии  $i \in S$  существует  $r_i$  решений из конечного множества  $R_i$ . Выбор некоторого решения  $r$  из этого множества  $R_i$  в состоянии  $i \in S$  означает задание величин  $Y_{ij}^r(t)$ ,  $P_{ij}^{(r)}$ ,  $F_{ij}^{(r)}(t)$ ,  $k_i^{(r)}$ ,  $j \in S$ .

При  $i=0$ ,  $R_0 = \{0\}$ , вероятность  $P_{0j}^{(r)} \neq 0$ ,  $j \in S$  является вероятностью перехода в состояние  $j$ . Вероятность  $P_{0j}^{(r)} \neq 0$ ,  $j \in S$  вычисляется на практике как доля состояний с ошибками безопасности типа  $j$  в общей совокупности уязвимостей безопасности различных типов на основе данных предыстории процесса разработки ПО. В этом случае  $F_{0j}^{(r)}(t)$  – функция распределения времени тестовой эксплуатации ПО между выявленными ошибками безопасности типа  $j$ .

При  $i=1, \dots, N$  для любого  $r \in R_i$ ,  $P_{i0}^{(r)} = 1$ ,  $P_{ij}^{(r)} = 0$ ,  $j \neq 0$ , функция  $F_{i0}^{(r)}(t)$  это функция распределения времени устранения уязвимостей безопасности с использованием решения  $r$  при ошибке типа  $j$ .

При условии непрерывности во времени исследуемого процесса будем пользоваться переоценкой экспоненциального вида с нормой  $\alpha$ , то есть если в некоторый момент времени затраты составляют какую-то единичную величину, то через время  $t$  эти затраты уже будут  $e^{-\alpha t}$  единичных величин. Тогда если  $k_i$  – расход за единицу времени, то суммарный расход за время  $t$  имеет вид

$$\int_0^t k_i e^{-\alpha\tau} d\tau = \frac{k_i}{\alpha} (1 - e^{-\alpha t}). \quad (2)$$

Обозначим  $i_n$  состояние системы после  $n$ -го перехода,  $u_n$  – принятое решение, а  $\tau_n$  – время пребывания в этом состоянии ( $n=0,1,2,\dots$ ),  $i_0$  – начальное состояние. Допустимую стратегию  $\beta$  для системы управления разработкой ПО определим как последовательность  $\{\beta_0, \beta_1, \beta_2, \dots\}$ , где  $\beta_n(\bullet / z_n)$  – вероятностная мера, сосредоточенная на функции ограничения  $U(S)$  на принятые решения (управления), определяемые системой неравенств

$$\sum_{j \in S} c_{rj} x_{rj} \leq b_r, \quad r \in R = \bigcup_{j \in S} R_j, \quad (3)$$

и зависящая от истории управляемой системы к моменту  $n$ -го перехода  $z_n = (i_0, u_0, \tau_0, \dots, i_{n-1}, u_{n-1}, \tau_{n-1}, i_n)$ . Мера  $\beta_n(\bullet / z_n)$  задает рандомизированное правило выбора решения  $u_n$  на основе информации  $z_n$ . Такую стратегию  $\beta$  можно назвать рандомизированной.

Стратегия  $\beta$  является марковской, если  $\beta_n(\bullet / z_n) = \beta_n(\bullet / i_n)$ , где  $n=0,1,2,\dots$ . Марковская стратегия называется стационарной, если  $\beta_n(\bullet / i_n) = \beta_n(\bullet / i_n)$ . Плотность меры такой стратегии при  $i_n = i$ ,  $u_n = r$ , ( $r \in R_i$ ) обозначим  $d_i^{(r)}$ . Если стратегия  $\beta$  – марковская стационарная, то управляемый процесс является полумарковским.

Анализ литературы показал, что наиболее популярная информация о полумарковских процессах и управляемых полумарковских моделях с дополнительными расходами и дивидендами изложена в работах [8; 16; 119; 55; 45].

Обозначим через  $g_i(t, \alpha, \beta)$  суммарный расход системы, управляемой в соответствии со стратегией  $\beta$ , с нормой переоценки  $\alpha$ , за время  $t$  жизненного цикла разработки ПО. Обязательным условием является то, что процесс начинается в момент  $t=0$  из состояния  $i$ . Через  $v_i(t, \alpha, \beta) = g_i(t, \alpha, \beta) / t$  обозначим суммарный средний расход системы за время  $t$  при тех же условиях.

Пусть  $c_{rj}$  – затраты, связанные с реализацией мероприятия  $r$  в случае события нарушения безопасности ПО  $j$  и  $x_{rj}$  – булева переменная:  $x_{rj} = 1$ , если  $r$  применяется при событии  $j$ ,  $x_{rj} = 0$  в противном случае.

Предположим, что общий объем средств, отпущенных для устранения недостатков безопасности ПО (мероприятия типа  $r$ ) ограничен константой  $b_r$ , т.е. выполняется неравенство (3).

Если затраты  $c_{ij}$  позволяют выполнить каждое из ограничений (3), то реализованная на основании (3) система определяет в пространстве  $\mathcal{R}^d$ ,  $d = \dim R$ , некоторое конечное множество дискретных точек. Тогда в соответствии с работами [8; 16; 45; 55; 119] существует нерандомизированная стационарная стратегия  $\beta^*$ , называемая  $\beta$  – оптимальной, которая минимизирует суммарный средний расход  $v(\alpha, \beta)$  при произвольной стратегии  $\beta$  и норме переоценки  $\alpha$  ( $\alpha > 0$ ). При этом  $v(\alpha, \beta)$  есть  $(N+1) \times 1$ - мерный вектор  $(v_0(\alpha, \beta), v_1(\alpha, \beta), \dots, v_N(\alpha, \beta))$ , где

$$v_i(\alpha, \beta) = \lim_{t \rightarrow \infty} v_i(t, \alpha, \beta), i \in S. \quad (4)$$

Необходимо найти  $\alpha$  – оптимальную нерандомизированную марковскую стационарную стратегию  $\beta^*$ , которая минимизирует суммарный средний расход  $v(\alpha, \beta)$  при произвольном начальном распределении процесса

$$y = (y_0, y_1, \dots, y_N), \quad (5)$$

$$\sum_{i \in S} y_i = 1, y_i \geq 0, i \in S. \quad (6)$$

Не уменьшая общности, в качестве начального распределения возьмем вектор  $y = (1, 0, \dots, 0)$ , т.е. начальное состояние системы. На основе полумарковской модели принятия решений данную задачу приведем к эквивалентной задаче бивалентного программирования с использованием псевдобулевых методов.

### **Оптимизационная стратегия полумарковской модели принятия решений**

Вероятности переходов рассматриваемого, для системы разработки ПО, полумарковского процесса принятия решений в моменты скачков из состояния  $i$  в состояние  $j$  при принятии решения  $r \in R_i$  определяется стохастической  $(N+1) \times (N+1)$  матрицей  $P^{(r)} = \{p_{ij}^{(r)}\}$ , которая задает так называемую вложенную цепь Маркова. Элементы  $p_{ij}^{(r)}$  при любых  $i, j \in S$  и  $r \in R_i$  позволяют определять по формуле (1) совместную вероятность  $Q_{ij}^{(r)}(t)$  того, что длительность пребывания в состоянии  $i$  не превосходит время  $t$  из состояния  $i$  при  $r \in R_i$  процесс переходит в состояние  $j$  с вероятностью  $p_{ij}^{(r)}$ . Функции  $Q_{ij}^{(r)}(t)$  в (1) удовлетворяют условиям

$$Q_{ij}^{(r)}(0) = 0, i, j \in S, r \in R_i, \quad (7)$$

$$\sum_{j \in S} Q_{ij}^{(r)}(\infty) = \sum_{j \in S} p_{ij}^{(r)} = 1, \quad i, \in S, r \in R_i. \quad (8)$$

С помощью матрицы  $Q_{ij}^{(r)}(t) = \{Q_{ij}^{(r)}(t)\}$  переходных распределений, определим функцию

$$H_i^{(r)}(t) = \sum_{j \in S} Q_{ij}^{(r)}(t), \quad i, \in S, r \in R_i, \quad (9)$$

являющуюся функцией распределения времени пребывания процесса в состоянии  $i$  при принятии решения  $r \in R_i$ .

Случайный процесс  $(Z_t), t \geq 0$  со значениями  $Z_t = i$ , если в момент  $t$  система находится в состоянии  $i$ , является полумарковским, и задается величинами  $N, \gamma, Q_{ij}^{(r)}(t), i, j \in S, r \in R_i$ .

Полумарковский процесс называется регулярным, если за конечный промежуток времени он с вероятностью  $p_p = 1$  перейдет в любое состояние не более конечного числа раз. Таким образом, регулярный полумарковский процесс за конечный промежуток времени всегда совершает лишь конечное число переходов. Далее в разделе будем рассматривать только регулярные полумарковские процессы.

В случае одноэлементных множеств решений  $R_i$  в результате стандартных для теории восстановления [8; 16; 119; 55; 45] рассуждений получаем следующее уравнение восстановления

$$v_i(t) = (1 - H_i(t)) \frac{k_i}{\alpha} (1 - e^{-\alpha t}) + \sum_{j \in S} \int_0^t \left( \frac{k_i}{\alpha} (1 - e^{-\alpha t}) + e^{-\alpha t} v_j(t - \tau) \right) dQ_{ij}(\tau), \quad i \in S,$$

где  $v_i(t)$  – краткая запись суммарного среднего расхода  $v_i(t, \alpha, \beta)$  за время  $t$ .

В случае конечных множеств  $R_i$  уравнение восстановления с учетом вероятностей  $d_i^{(r)}$  принятия решений  $r$  в состоянии  $i$  запишем в виде

$$v_i(t) = \sum_{r \in R_i} d_i^r (1 - H_i^{(r)}(t)) \frac{k_i^{(r)}}{\alpha} (1 - e^{-\alpha t}) + \sum_{j \in S} \sum_{r \in R_i} \int_0^t d_i^r \left( \frac{k_i^{(r)}}{\alpha} (1 - e^{-\alpha t}) + e^{-\alpha t} v_j(t - \tau) \right) dQ_{ij}^{(r)}(\tau), \quad i \in S, \quad (10)$$

где  $k_i^{(r)}$  – расход системы за единицу времени пребывания в состоянии  $i$  при решении  $r \in R_i$ ;  $v_j(t)$  – суммарный средний расход с учетом переоценки (2), при условии, что процесс начинается в момент  $t = 0$  из состояния  $j$ .

Величины  $v_i(\alpha, \beta)$  из выражения 4 можно записать в виде  $v_i(\alpha)$ , и для этого уравнения воспользоваться основными положениями уравнения

(интеграла) Лапласа-Стилтьеса. В соответствии с работами [8; 16; 119; 55; 45] для любой функции  $F(t)$ , производная  $F'(t)$  которой является функцией-оригиналом, удовлетворяющей неравенству  $F'(t) < Ce^{\alpha t}$  для всех  $t < 0$ , при всех комплексных  $s$ , когда  $\operatorname{Re} s > \alpha$  существует функция

$$F^*(s) = L_s^* \langle F(t) \rangle = \int_0^{\infty} e^{-st} dF(t), \quad (11)$$

то есть функция  $e^{-st}$  при  $\operatorname{Re} s > \alpha$  интегрируема по функции  $F(t)$ . Функцию  $F^*(s)$  называют преобразование Лапласа-Стилтьеса функции  $F(t)$ .

Из выражений 8 и 9 следует, что  $H_i^{(r)}(\infty) = 1$ ,  $i \in S$ ,  $r \in R_i$ , поэтому первая сумма в выражении 10 при  $t \rightarrow \infty$  обращается в нуль. Интегрируя по частям выражение 11 для  $L_s^* \langle F(t) \rangle$ , получаем

$$sL_s^* \langle F(t) \rangle = L_s^* \langle F(t) \rangle - F(0), \quad (12)$$

где

$$F(s) = L_s \langle F(t) \rangle = \int_0^{\infty} e^{-st} F(t) dt$$

есть преобразование Лапласа функции  $F(t)$ . Из 12 при  $s \neq 0$  находим

$$L_s \langle F(t) \rangle = \frac{1}{s} (L_s^* \langle F(t) \rangle - F(0)). \quad (13)$$

Интегрируем по частям с учетом выражения 9 находим

$$\sum_j \int_0^t (1 - e^{-\alpha \tau}) dQ_{ij}^{(r)}(\tau) = (1 - e^{-\alpha t}) \sum_j dQ_{ij}^{(r)}(\tau) \Big|_0^t - \sum_j \alpha \int_0^t e^{-\alpha \tau} H_i(\tau) d\tau \quad (14)$$

Проводя преобразования, переходя в выражении 14 к пределу  $t \rightarrow \infty$  и применяя формулу 13 для  $s = \alpha$ , ( $\alpha > 0$ ), с учетом соотношений 7 и 8 получим

$$\sum_j \int_0^t (1 - e^{-\alpha \tau}) dQ_{ij}^{(r)}(\tau) = (1 - \alpha) L_{s=\alpha} \langle H_i^{(r)}(\tau) \rangle = 1 - \alpha \frac{1}{\alpha} L_{s=\alpha}^* \langle H_i^{(r)}(\tau) \rangle = 1 - h_i^{(r)}(\alpha),$$

где  $h_i^{(r)}(\alpha) = L_{s=\alpha}^* \langle H_i^{(r)}(t) \rangle$ .

Применяя к функции  $\Phi_i^{(r)}(t) = \int_0^t e^{-\alpha t} v_j(t - \tau) dQ_{ij}^{(r)}(\tau)$  теорему о предельном переходе в интеграле по параметру, от которого зависят пределы интегрирования и подынтегральная функция [8; 16; 119; 55; 45], при  $t \rightarrow \infty$  получаем



$$\Phi_i^{(r)}(\infty) = \int_0^{\infty} e^{-\alpha t} v_j(\alpha) dQ_{ij}^{(r)}(\tau) = v_j(\alpha) q_{ij}^{(r)}(\alpha), \quad (16)$$

где  $q_{ij}^{(r)}(\alpha) = L_{s=\alpha}^* \langle Q_i^{(r)}(\alpha) \rangle$ .

Переходя в выражении 10 к пределу при  $t \rightarrow \infty$ , с учетом 15 и 16 получаем следующее аналитическое выражение:

$$v_i(t) = \sum_{r \in R_i} d_i^{(r)}(\zeta_i^{(r)}(\alpha)) + \sum_{j \in S} q_{ij}^{(r)}(\alpha) v_j(\alpha) \quad (17)$$

где 
$$\zeta_i^{(r)}(\alpha) = \frac{k_i^{(r)}}{\alpha} (1 - h_i^{(r)}(\alpha)). \quad (18)$$

Пусть 
$$\zeta_i(\alpha) = \sum_{r \in R_i} d_i^r(\rho_i^{(r)}(\alpha)) \quad \text{и} \quad \mathfrak{Z}(\alpha) = (\zeta_0(\alpha), \dots, \zeta_N(\alpha))^T,$$

$\wp(\alpha) = (v_0(\alpha), \dots, v_N(\alpha))^T$ , (T – символ транспонирования матрицы). Тогда

$$\wp(\alpha) = \mathfrak{Z}_0(\alpha) + q(\alpha)\wp(\alpha) \quad (19)$$

где  $q(\alpha) = \{q_{ij}(\alpha)\}$ ,  $q_{ij}(\alpha) = \sum_{r \in R_i} d_i^{(r)}(q_{ij}^{(r)}(\alpha))$ .

Из выражения 19 найдем

$$\wp(\alpha) = \{I - q(\alpha)\}^{-1} \mathfrak{Z}_0(\alpha) \quad (20)$$

Данное выражение справедливо, так как при  $\alpha > 0$  матрица  $\{I - q(\alpha)\}$  – невырожденная, I – единичная матрица размера  $(N \times 1) \times (N \times 1)$ .

Умножив обе части равенства 19 слева на вектор  $y$ , получим следующее

$$y\wp(\alpha) = \sum_{i \in S} \sum_{j \in \tilde{S}} \sum_{r \in R_i} y_i \mu_{ij}(\alpha) \zeta_j^{(r)}(\alpha) d_i^{(r)},$$

$$\{I - q(\alpha)\}^{-1} = \{\mu_{ij}(\alpha)\}. \quad (21)$$

Величины  $\mu_{ij}(\alpha)$  зависят от  $d_i^{(r)}$ ,  $r \in R_i$ ,  $i \in S$ , так как элементы матрицы  $\{I - q(\alpha)\}$  можно выразить через  $d_i^{(r)}$ ,  $r \in R_i$ ,  $i \in S$ .

Пусть  $\{d_i^{(r)}\}$  ( $r \in R_i$ ) – нерандомизированная марковская стационарная стратегия системы разработки ПО в состоянии  $j$ .

$$d_j^{(r)} \in \{0, 1\}, \quad \sum_{j \in S} d_j^{(r)} = 1,$$

и  $x_{00} = 1$ ,  $x_{rj} = d_j^{(r)}$ ,  $r \in R_i$ ,  $j \in \tilde{S}$ . Минимизация расходов (выражение 21) приводит к следующей задаче оптимизации для булевых переменных  $X = \{x_{rj}\}$ ,

$r \in R_i$ ,  $j \in \tilde{S}$ :

$$f(\alpha, X) = \sum_{i \in S} \sum_{j \in \tilde{S}} \sum_{r \in R_i} y_i \mu_{ij}(\alpha, X) \zeta_j^{(r)} x_{rj} \rightarrow \min, \quad (22)$$

$$\sum_{r \in R_i} x_{rj} = 1, j \in \tilde{S}, \quad (23)$$

$$\sum_{j \in \tilde{S}} c_{rj} x_{rj} \leq b_r, r \in R_i, j \in \tilde{S}, \quad (24)$$

$$x_{rj} \in \{0,1\}, j \in \tilde{S}, r \in R_i. \quad (25)$$

### Построение оптимальной нерандомизированной марковской стационарной стратегии

Обозначим систему 24, 25 как система С. Она является системой псевдобулевых неравенств.

Подключив далее дополнительное условие 23 можно обозначить систему как  $\tilde{C}$ . А через  $X_r^{(k)} = \{x_{r1}^{(k)}, \dots, x_{rN}^{(k)}\}$ ,  $k=1, \dots, k_r$  – допустимые решения  $r$ -го неравенства системы С.

Для построения решений системы  $\tilde{C}$  при известных допустимых решениях каждого неравенства 24 применим следующий подход. Решения системы  $\tilde{C}$  находятся в виде  $Z = \{s_j\}$ ,  $j=1, \dots, N$ , где  $s_j$  – множество номеров  $r$ , для которых допустимо равенство  $x_{rj} = 1$ . Решения находятся за  $m$  шагов, где  $m$  – число ограничений 24. В исходном состоянии каждое из множеств  $s_j^{(0)}$  вектора  $Z^{(0)}$  включает все возможные значения  $r \in R_i$ . На  $r$ -м шаге происходит пересечение вектора  $Z^{(r-1)}$  с одним из решений  $r$ -го неравенства. Допуская, что  $r$ -му неравенству соответствует  $r = r_1$ , а также, что  $\alpha_j$  является  $j$ -м элементом допустимого решения данного неравенства,  $\alpha_j \in \{0,1,\varphi\}$ , где  $\varphi$  – неопределенный параметр из множества  $\{0,1\}$ , называемый в дальнейшем почерком, можно сформулировать следующие правила для  $r$ -го шага алгоритма построения решений системы  $\tilde{C}$ .

1. Если значение  $\alpha_j$  не фиксировано, то  $s_j^{(r)} = s_j^{(r-1)}$ .

2. Если  $\alpha_j = 1$ , то при  $r_1 \in s^{(r-1)}$  допускаем  $s^{(r)} = \{r_1\}$ , а при  $r_1 \notin s^{(r-1)}$  допускаем  $s^{(r)}$  равно пустому множеству.

3. Если  $\alpha_j = 0$  то допускаем  $s^{(r)} = s^{(r-1)} / \{r_1\}$ .

При этом пересечение семейств решений осуществляется с учетом дополнительных ограничений 23.

На  $m$  шаге алгоритма получается вектор  $Z^{(m)} = \{\alpha_1^{(m)}, K, \alpha_N^{(m)}\}$ , каждая компонента  $\alpha_j^{(m)}$ , которого является одноэлементным множеством  $\{r\}$ ,  $r \in R$ ,  $R = \{1, \dots, m\}$  и следовательно,  $Z^{(m)}$  есть решение системы  $\tilde{C}$ . Исключения составляют случаи, когда  $\tilde{C}$  представляет собой некоторый набор чисел  $r$  из множества  $R$ . В этом случае с помощью сочетания элементов многозначных компонент из вектора  $Z^{(m)}$  можно получить несколько решений системы  $\tilde{C}$ . Затем в результате находим совокупность всех решений системы  $\tilde{C}$ , из которых выбираем оптимальное решение, доставляющее минимум целевой функции  $f(\alpha, X)$ . Это решение может находиться разными известными методами линейного программирования, или просто путем непосредственного сравнения значений  $f(\alpha, X)$  при определении  $X$  системы  $\tilde{C}$ .

Численная реализация изложенного метода управления рисками безопасности индустрии программного обеспечения для полумарковской модели принятия решений при аномальных ситуациях безопасности представлена в следующем примере.

### **Рекомендации по использованию метода управления рисками разработки программного обеспечения**

Рассмотрим две возможные ситуации ошибок безопасности ПО: ситуация 1 – наличие функциональных ошибок ПО (функциональные возможности ПО, не описанные или не соответствующие описанным в документации, при использовании которых возможно нарушение конфиденциальности, доступности или целостности обрабатываемой информации); ситуация 2 – наличие нефункциональных ошибок ПО (избыточные ФО и ИО, переполнение буфера, утечки памяти, ошибки типов данных, ошибки указателей и др.).

Альтернативные мероприятия (упрощенно) в случае первой аномальной ситуации следующие:

- отзыв ПО (для устранения ошибок);
- декомпозиция программы и оценка метрик сложности;
- сигнатурный анализ.

В случае наличия нефункциональных ошибок ПО альтернативные мероприятия следующие:

- отзыв ПО (для устранения ошибок);
- анализ указателей, зависимостей по данным и интервальный анализ.

Предположим, что в обеих ситуациях первое мероприятие (отзыв ПО) осуществляется за одно и то же время  $T=1$  у.е. Поэтому, учитывая малую

стоимость соответствующих функций тестирования по сравнению с убытками при простое ПО, можно считать первые мероприятия в обоих случаях одинаковыми по затратам.

Представим мероприятия в случае первой аномальной ситуации в виде  $R = \{r_1; r_2; r_3\}$ , а мероприятия для устранения второй аномальной ситуации в виде  $R = \{r_1; r_4\}$ . Тогда число элементов системы составит  $N = 2$ , а общее число различных мероприятий для их устранения  $m = 4$ .

Так же представим описанные аномальные ситуации  $j$  по отношению с отдельными модулями и блоками ПО как последовательные соединения (по надежности). Ошибки (отказы) и их устранение не влияют на надежность других модулей и элементов ПО. Время устранения ошибки не зависит от того, выявляются ли другие ошибки в оставшихся модулях. Другими словами, каждая ошибка выявляется и устраняется независимо друг от друга. В этом случае поток ошибок и, устраняющих их, мероприятий есть сумма  $N$  независимых процессов устранения с конечным временем восстановления.

Обозначим через  $F_j(t)$  функцию распределения времени эксплуатации системы между двумя последовательными аномальными ситуациями типа  $j$ , а через  $G_j^{(k)}(t)$  – функцию распределения времени устранения ошибки после аномальной ситуации типа  $j$  при принятии решения  $r$ . Будем полагать, что  $F_j(t)$  и  $G_j^{(k)}(t)$  подчиняются экспоненциальным законам с функциями интенсивности соответственно  $\lambda_j$  и  $\mu_j^{(k)}$ :

$$F_j(t) = 1 - e^{-\lambda_j t}, \quad G_j^{(k)}(t) = 1 - e^{-\mu_j^{(k)} t}, \quad (26)$$

где  $\lambda_j = 1/T_{j1}$ ;  $\mu_j^{(r)} = 1/T_{j2}$ ;  $T_{j1}$  – среднее время эксплуатационного этапа жизненного цикла разработки ПО между двумя аномальными ситуациями типа  $j$ ;  $T_{j2}$  – среднее время восстановительного этапа жизненного цикла разработки ПО после выявления ошибки типа  $j$  при решении  $r$ .

Пусть  $d_i^{(k)}$  – нерандомизированная стационарная стратегия системы в состоянии  $i$  ( $i \in S$ ) при решении  $k$  (то есть, вероятность принятия решения  $r$  в состоянии  $i$ ),

$$d_i^{(r)} \in \{0,1\}, \quad \sum_{k \in K} d_i^{(k)} \in 1, \quad i \in S. \quad (27)$$

Тогда закон распределения эксплуатации и закон распределения устранения ошибок в целом запишем в виде

$$F_j(t) = 1 - e^{-\lambda t}, \quad G_j^{(r)}(t) = 1 - e^{-\mu t},$$

где  $\lambda = \sum_{j=1}^N \lambda_j$ ;  $\mu = \sum_{j=1}^N \sum d_j^{(r)} \mu_j^{(r)}$ .

Пусть

$$\begin{aligned} T_{11} = 8 \text{ у.е.}, \quad T_{21} = 8 \text{ у.е.}, \quad T_{12}^{(1)} = 1 \text{ у.е.}, \quad T_{12}^{(2)} = 2 \text{ у.е.}, \quad T_{12}^{(3)} = 1 \text{ у.е.}, \\ T_{22}^{(1)} = 1 \text{ у.е.}, \quad T_{22}^{(4)} = 0,5 \text{ у.е.} \end{aligned} \quad (28)$$

Предположим, что  $i=0$  фиксирует нормальное состояние,  $i=1$  аномальное состояние (ситуация 1),  $i=2$  аномальное состояние (ситуация 2);  $S = \{0,1,2\}$ ,  $\tilde{S} = \{1,2\}$ . В соответствии с условием 28 и учитывая выражения 25-27 получим:

$$\begin{aligned} F_0(t) = F(t) = 1 - e^{-0,25t}, \quad F_j(t) = 1 - e^{-0,125t}, \quad j = 1,2; \\ G_1^{(1)}(t) = 1 - e^{-t}, \quad G_1^{(2)}(t) = 1 - e^{-0,5t}, \quad G_1^{(3)}(t) = 1 - e^{-t}, \\ G_2^{(1)}(t) = 1 - e^{-t}, \quad G_2^{(4)}(t) = 1 - e^{-2t}. \end{aligned}$$

Пусть издержки от невыполнения своих функций программной системы в течении времени  $T=1\text{ч}$  составят 70000 у.е. Обозначим через  $c_{k,j}$  затраты на мероприятие  $k$  в случае аномальной ситуации  $j$ . Тогда, пренебрегая затратами на тестирование в мероприятии  $r=1$ , получаем  $c_{1,1} = c_{1,2} = c_1 = 70000$  у.е. Пусть далее  $c_{2,1} = 300$ ,  $c_{3,1} = 400$ ,  $c_{4,2} = 600$  у.е. Будем считать, что в состоянии  $i=0$  принято единственное решение ( $r=0$ ) – продолжить нормальное функционирование, и в этом состоянии задано следующее распределение вероятностей:

$$p_{00}^{(0)} = 0,7, \quad p_{01}^{(0)} = 0,1, \quad p_{02}^{(0)} = 0,2. \quad (29)$$

Функции (1) и (9) запишем как

$$\begin{aligned} Q_{00}^{(0)}(t) = 0,7(1 - e^{-0,25t}), \quad Q_{01}^{(0)}(t) = 0,1(1 - e^{-0,125t}) \\ Q_{02}^{(0)}(t) = 0,2(1 - e^{-0,125t}), \quad Q_{10}^{(1)}(t) = (1 - e^{-t}), \quad Q_{10}^{(2)}(t) = (1 - e^{-0,5t}), \\ Q_{10}^{(3)}(t) = (1 - e^{-t}), \quad Q_{11}^{(k)}(t) = Q_{12}^{(k)}(t) = 0, \quad (k = 1,2,3), \\ Q_{20}^{(1)}(t) = (1 - e^{-t}), \quad Q_{20}^{(4)}(t) = (1 - e^{-2t}), \\ H_0^{(0)}(t) = 0,7(1 - e^{-0,25t}) + 0,3(1 - e^{-0,125t}), \\ H_1^{(0)}(t) = (1 - e^{-0,25t}), \quad H_1^{(2)}(t) = (1 - e^{-0,5t}), \quad H_1^{(3)}(t) = (1 - e^{-t}), \end{aligned}$$

$$H_2^{(1)}(t) = (1 - e^{-t}), H_2^{(4)}(t) = (1 - e^{-2t}),$$

С учетом обозначений  $x_{00} = 1, x_{k,j} = d_j^{(k)}, r \in R_j, j \in \tilde{S}$ , матрица  $q(\alpha, x) = [q_{i,j}(\alpha, x)] (i, j \in S)$  с элементами

$$q(\alpha, x) = \sum_{r \in R_j} x_{k,i} q_{i,j}^{(r)}(\alpha) (i, j \in S)$$

примет вид

$$q(\alpha, x) = \begin{bmatrix} \frac{0,175}{\alpha + 0,25} & \frac{0,0125}{\alpha + 0,125} & \frac{0,025}{\alpha + 0,125} \\ \frac{x_{11} + x_{31}}{\alpha + 1} + \frac{x_{21}}{\alpha + 0,5} & 0 & 0 \\ \frac{x_{12}}{\alpha + 1} + \frac{x_{42}}{\alpha + 2} & 0 & 0 \end{bmatrix}$$

Найдем определитель матрицы  $[I - q(\alpha, x)]$ :

$$D(\alpha, x) = -\frac{0,025}{\alpha + 0,125} \left( \frac{x_{12}}{\alpha + 1} + \frac{2x_{42}}{\alpha + 2} \right) + 1 - \frac{0,175}{\alpha + 0,25} - \frac{0,0125}{\alpha + 0,125} \left( \frac{x_{11}}{\alpha + 1} + \frac{x_{21}}{\alpha + 0,5} \right).$$

Матрица  $[I - q(\alpha, x)]^{-1} = [\mu_{i,j}(\alpha, x)] (i, j \in S)$  имеет элементы

$$\mu_{00}(\alpha, x) = 0, \mu_{01}(\alpha, x) = \frac{1}{D(\alpha, x)} \frac{0,0125}{\alpha + 0,125}, \mu_{02}(\alpha, x) = \frac{1}{D(\alpha, x)} \frac{0,025}{\alpha + 0,125},$$

$$\mu_{10}(\alpha, x) = \frac{1}{D(\alpha, x)} \left( \frac{x_{11} + x_{31}}{\alpha + 1} + \frac{x_{21}}{\alpha + 0,5} \right),$$

$$\mu_{11}(\alpha, x) = \frac{1}{D(\alpha, x)} \left( 1 - \frac{0,175}{\alpha + 0,25} - \frac{0,0125}{\alpha + 0,125} \left( \frac{x_{12}}{\alpha + 1} + \frac{2x_{42}}{\alpha + 2} \right) \right),$$

$$\mu_{12}(\alpha, x) = \frac{1}{D(\alpha, x)} \frac{0,025}{\alpha + 0,125} \left( \frac{x_{11} + x_{31}}{\alpha + 1} + \frac{x_{21}}{\alpha + 0,5} \right),$$

$$\mu_{20}(\alpha, x) = \frac{1}{D(\alpha, x)} \left( \frac{x_{12}}{\alpha + 1} + \frac{2x_{42}}{\alpha + 2} \right),$$

$$\mu_{21}(\alpha, x) = \frac{1}{D(\alpha, x)} \frac{0,0125}{\alpha + 0,125} \left( \frac{x_{12}}{\alpha + 1} + \frac{2x_{42}}{\alpha + 2} \right),$$

$$\mu_{22}(\alpha, x) = \frac{1}{D(\alpha, x)} \left( 1 - \frac{0,0175}{\alpha + 0,25} - \frac{0,0125}{\alpha + 0,125} \left( \frac{x_{11} + x_{31}}{\alpha + 1} + \frac{x_{21}}{\alpha + 0,5} \right) \right).$$

В выражении (18)  $k_i^{(r)}$  принимают следующие значения:

$$k_0^{(0)} = 70000 \text{ y.e.}, k_1^{(1)} = -70000 \text{ y.e.}, k_1^{(2)} = -150 \text{ y.e.}, k_1^{(3)} = -400 \text{ y.e.},$$

$$k_2^{(1)} = -70000 \text{ y.e.}, k_2^{(4)} = -12000 \text{ y.e.},$$

а величины  $\zeta_i^{(r)}$  записываются в виде:

$$\zeta_0^{(0)}(\alpha) = \frac{70000}{\alpha} \left( 1 - \frac{0,175}{\alpha + 0,25} - \frac{0,0375}{\alpha + 0,125} \right), \quad \zeta_1^{(1)}(\alpha) = -\frac{70000}{\alpha} \left( 1 - \frac{1}{\alpha + 1} \right),$$

$$\zeta_1^{(2)}(\alpha) = -\frac{150}{\alpha} \left( 1 - \frac{0,5}{\alpha + 0,5} \right), \quad \zeta_1^{(3)}(\alpha) = -\frac{400}{\alpha} \left( 1 - \frac{1}{\alpha + 1} \right),$$

$$\zeta_2^{(1)}(\alpha) = -\frac{70000}{\alpha} \left( 1 - \frac{1}{\alpha + 1} \right), \quad \zeta_2^{(4)}(\alpha) = -\frac{1200}{\alpha} \left( 1 - \frac{2}{\alpha + 2} \right).$$

Найдем решения системы  $\tilde{C}$  с помощью алгоритма пересечения решений отдельных неравенств (24), полагая, что правые части  $b_k$  этих неравенств удовлетворяют условиям:

$$c_{11} < b_1 < 2c_1, \quad b_2 > c_{21}, \quad b_3 > c_{31}, \quad b_4 > c_{42}. \quad (30)$$

С учетом условий (30) находим следующие решения отдельных неравенств системы  $\tilde{C}$ :

$$r_1 = 1 : 1 - (1,0); \quad 2 - (0,1), \quad 3 - (0,0);$$

$$r_1 = 2 : 1 - (1,0); \quad 2 - (0,0);$$

$$r_1 = 3 : 1 - (1,0); \quad 2 - (0,0);$$

$$r_1 = 4 : 1 - (0,1); \quad 2 - (0,0).$$

При наличии  $Z^{(0)} = \{\{1,2,3\}, \{1,4\}\}$  на последнем шаге алгоритма получаем:

$$Z_{1221}^{(4)} = \{\{1\}, \{4\}\},$$

$$Z_{1222}^{(4)} = \{\{1\}, \text{пустое множество}\},$$

$$Z_{2121}^{(4)} = \{\{2\}, \text{пустое множество}\},$$

$$Z_{2122}^{(4)} = \{\{2\}, \{1\}\},$$

$$Z_{2211}^{(4)} = \{\{3\}, \text{пустое множество}\},$$

$$Z_{2212}^{(4)} = \{\{3\}, \{1\}\},$$

$$Z_{3121}^{(4)} = \{\{2\}, \{4\}\},$$

$$Z_{3122}^{(4)} = \{\{2\}, \text{пустое множество}\},$$

$$Z_{3211}^{(4)} = \{\{3\}, \{4\}\},$$

$$Z_{3212}^{(4)} = \{\{3\}, \text{пустое множество}\}.$$

Таким образом, решениями системы  $\tilde{C}$  являются векторы:

$$1 - \{\{1\}, \{4\}\}, \quad 2 - \{\{2\}, \{1\}\}, \quad 3 - \{\{3\}, \{1\}\}, \quad 4 - \{\{2\}, \{4\}\}, \quad 5 - \{\{3\}, \{4\}\}.$$

Им соответствуют следующие значения булевых переменных  $\{x_{k,j}\}$ :

1)  $x_{1,1} = 1, x_{4,2} = 1$ ;

2)  $x_{2,1} = 1, x_{1,2} = 1$ ;

3)  $x_{2,1} = 1, x_{4,2} = 1$ ;

4)  $x_{3,1} = 1, x_{1,2} = 1$ ;

5)  $x_{3,1} = 1, x_{4,2} = 1$ .

Не указанные переменные в каждом  $i$ -ом решении равны нулю.

С учетом выражений 28 и 29  $f(\alpha, x)$  при  $\alpha = 0,1$ , что соответствует инфляции, равной 10%, и начальном распределении  $y = (1, 0, \dots, 0)$  принимает значения:

$$f_1 = -51903; f_2 = -5329040; f_3 = -17161; f_4 = -26939; f_5 = -3325/$$

Таким образом, при  $\alpha = 0,1$  и  $y = (1, 0, \dots, 0)$  оптимальной нерандомизированной марковской стационарной стратегией будет стратегия  $x_{1,1} = 0, x_{2,1} = 0, x_{3,1} = 1, x_{1,2} = 0, x_{4,2} = 1$ , соответствующая решению  $\{\{3\}, \{4\}\}$  системы  $\tilde{C}$ .

**Заключение.** В разделе усовершенствован метод управления рисками разработки ПО. В основу данного метода была положена полумарковская модель принятия решений для управляемого марковского процесса в непрерывном времени. Отличительной особенностью предложенного метода является использование псевдобулевых методов бивалентного программирования с нелинейной целевой функцией и линейными ограничениями для определения оптимальной стратегии устранения эксплуатационных ошибок.

Проведенные исследования показали, что используемые в данном разделе теоретические положения в достаточном объеме отражают стандарты и возможности современных методологий тестирования ПО.

В качестве примера рассмотрены ситуации возникновения ошибок безопасности ПО, и определена оптимальная стратегия управления для устранения указанной аномальной ситуации.

Следует заметить, что представленный в разделе метод целесообразно использовать не только при управлении рисками безопасности ПО, но и при функциональном, нагрузочном, стрессовом и других видах тестирования для предотвращения возможных потерь.



## РОЗДІЛ 17

# АНАЛІЗ СИСТЕМНИХ ВЛАСТИВОСТЕЙ ТА ОБГРУНТУВАННЯ ЕФЕКТИВНОСТІ МЕТОДІВ ПЕРЕТВОРЕННЯ ФОРМИ ІНФОРМАЦІЇ

***Анотація.** Здійснено систематизацію та аналіз базових методів перетворення форми інформації, визначено властивості та обгрунтовано ефективність їх застосування для забезпечення системних функцій інфотехнології. Проаналізовано класи дискретно гармонічних та дискретно нерегулярних систем функцій та базисів.*

***Ключові слова:** перетворення форми інформації, система функцій, базис, код, система кодування.*

***Abstract.** Systematize and analyze of the basic information form transformation methods are done, defined the specified properties and proved the effectiveness of their application for the system functions of information technology. Analyzed discrete harmonic and irregular discrete functions systems and bases.*

***Keywords:** information form transformation, function system, basis, cod, code system.*

**Вступ та постановка задачі.** Реалізація системної функція перетворення форми інформації є вирішальною при визначенні докладності первинного перетворення, достовірності та інтенсивності інфообміну [129; 130]. В роботі досліджено методи аналізу і синтезу дискретних пристроїв перетворення форми інформації та цифрової обробки даних за допомогою апарату теорії ортогональних рядів. Розвивається єдиний підхід до задач аналізу та синтезу схем, виконаних на елементах з бістабільними та багатастабільними дискретними станами, що ґрунтується на представленні систем функцій алгебри логіки ортогональними та лінійно-незалежними рядами. При використанні вказаного підходу ряд важливих задач теорії проектування дискретних систем отримують простіше вирішення. З єдиних позицій викладені результати методів побудови дискретних перетворювачів форми інформації і засобів цифрової обробки сигналів на підставі аналізу теоретико-числових базисів та систем функцій [131].

Останнім часом широкого розповсюдження в практиці набули складні територіально розосереджені інформаційні системи, які включають засоби реалізації системних функцій формування, перетворення, передачі, обробки та

документування повідомлень. При побудові інформаційних систем проблема вибору методу кодування та відповідної системи чи базису функцій розглядається на рівнях [132; 133]:

- теорії джерел інформації та формування повідомлень;
- перетворення форми інформації;
- передачі та прийому дискретних повідомлень;
- цифрової обробки, відображення та зберігання інформації.

Здійснено класифікацію та аналіз кодів, кодових систем і системних функцій з метою теоретичного обґрунтування, узагальнення результатів експериментальних досліджень та широкого застосування рекурсивних кодових систем щодо реалізації системних функцій генерування, формування, перетворення, обробки, стискування та запам'ятовування повідомлень [129].

**Основна частина.** Клас унітарних, розрядно-позиційних, число-імпульсних, Лібова-Крейга та Ліпелла цифрових кодів визначаються як первинний, що породжують похідні дискретні базиси та системи функцій (Радемахера, Уолша тощо). Внаслідок базисної надлишковості названі коди та кодові системи набули обмеженого застосування переважно для реалізації системних функцій генерування, проміжного перетворення при аналого-цифровому перетворенні та частково інфообміні дискретних повідомлень [131].

Проаналізована теорія формування рекурсивних кодів показує, що такі коди мають широкий набір типів, просту структуру та забезпечують повний набір арифметико-логічних операцій, які реалізуються в Булевому базисі на однорозрядних процесорах.

Проте, розрядна надлишковість унітарної, розрядно-позиційної та число-імпульсної кодових систем потребує відповідного розрядного розширення обчислювальних потужностей, що зумовлює зменшення експлуатаційної надійності та подорожання апаратної реалізації. Зменшити в два рази розрядність обчислювальних засобів дозволяє перехід до кодової системи Лібова-Крейга.

Властивості одиничних кодів Лібова-Крейга-Ліпелла визначають їхнє ефективне застосування в якості проміжного представлення при перетворенні форми інформації, розробці скануючих аналого-цифрових перетворювачів Галуа та в пристрої обчислення ковзного математичного очікування. Зменшити кількість розрядів та підняти ефективність кодової системи дозволяє перехід до позиційних кодів із редукованою надлишковістю.

Клас позиційних кодів (двійкові, вісімкові, шістнадцяткові, десяткові, двійково-десяткові) визначається більш ефективними породжуючими базисами

із щільною упаковкою кодових елементів, що зумовило в даний час їх широке застосування. Слід відмітити, що необхідність паралельного представлення коду кожного із повідомлень в паралельному чи в послідовному форматі вимагає відповідного шинного розширення або забезпечення необхідної пропускнуєї спроможності інформаційного каналу [130].

Значно розширити функціональні можливості засобів перетворення, інфообміну та обробки інформації дозволяє використання спеціалізовано орієнтованих класів кодових систем [132; 133].

Клас непозиційних кодів (Грея, Фібоначчі, римське числення, залишкових класів та Галуа) породжується відповідними базисами та володіє рядом властивостей, що визначають успішне вирішення проблеми створення ефективних сучасних інформаційних технологій. Зокрема, коди Грея та Фібоначчі здебільшого орієнтовані на застосування в перетворювачах форми інформації, римське числення – для відображення, а коди залишків – формування, передачі та обробки дискретних повідомлень. Залишкові коди застосовуються для побудова швидкодіючих обчислювальних машин, які працюють в модульній арифметиці залишкових класів. Проте, труднощі реалізації операцій ділення і порівняння чисел, низька продуктивність програмних процедур прямого та зворотнього перетворень зумовили обмеження їх масового застосування. І тільки перехід від паралельного подання коду повідомлення до послідовного векторного рекурсивного (в системах Галуа) дозволив перейти до вертикальної інформаційної технології та істотно покращити техніко-економічні показники інформаційної системи.

Клас надлишкових завадозахищених кодів (Галуа, деревоподібних та арифметичних) породжується похідними ущільненими вибірковими та комбінованими базисами функцій і сьогодні набув ширшого застосування при завадозахищеній передачі дискретних повідомлень та їх обробці. І тільки кодові системи Галуа в загальній класифікації володіють найширшим спектром властивостей, що дозволило класифікувати їх в одному з випадків до блокових систематичних поліномних циклічних кодів, а в іншому – до неперервних згорткових кодів в залежності від специфіки вирішуваної задачі, обраної кодової системи та планованих властивостей кодів, на використанні яких досягається перевага в порівнянні з іншими кодовими системами.

Основний метод підвищення завадозахищеності при кодуванні Галуа – не багаторазове повторювання, або вибіркове вкраплення провірочних елементів, а розширення  $n$ -розрядного повідомлення  $A_\mu = \{a_i, a_{i+1}, \dots, a_{i+n-1}\}$

шляхом введення додаткових  $l$  символів  $\{a_{i+n}, \dots, a_{i+n+l-1}\}$ , які визначаються контрольними і пов'язані з інформаційними  $\{A_i\}$  рекурсивною функцією взаємозалежності  $a_{i+1} = \sum_{i=1}^{n-1} a_i q_i \bmod p$ , де  $p$  – просте число, основа системи числення,  $q_i$  – вектор зворотних зв'язків. В подальших викладках приймемо  $p = 2$ , а в окремих випадках, де необхідно буде вказати значення  $p \neq 2$  у виразі визначимо літеру  $p$ . Шляхом зміни значення  $l$  згідно визначення Шеннона ступінь завадозахисту можна довести до необхідної величини.

Позитивною якістю синтезу кодових послідовностей Галуа в Булевому базисі є виродження операції модульної суми в логічну функцію “Виключне АБО”, що значно спрощує апаратну та програмну реалізацію технічних засобів, спричиняє до високої степені регулярності обчислювального та алгоритмічного середовища [129]. Рекурсивні властивості кодових послідовностей накладають нові вимоги щодо принципового, структурного та схемотехнічного проектування технічних засобів, орієнтованих на функціонування в кодових системах Галуа. Новий підхід вимагає розробки схемотехнічних рішень на базі конкретного кодового впорядкування послідовності з метою вирішення специфіки задачі. В той же час більшість інформаційних систем функціонують із використанням традиційних кодових систем і при переході до більш ефективних систем кодування виникає потреба кодової та алгоритмічної адаптації, що зумовило необхідність встановлення аналітичних взаємозалежностей між базисами, їхньої первинності та функціоналів взаємоперетворення. Здійснено дослідження основних теоретико-числових базисів, математичні викладки по встановленню їх взаємозалежностей із кодовими системами, що ними породжуються.

Проведений аналіз показує неповноту відомих базисів та систем функцій, зумовлену їхньою функціональною обмеженістю внаслідок природної виродженості (наприклад, Хаара, унітарний, Радемахера), що звужує галузі їх ефективного застосування. В той же час, доповнення кожного із базисів та введення проміжних базисів математичного аналітичного взаємопородження (наприклад, дискретно-фазових функцій Лібова-Крейга) розширює функціональні можливості при реалізації інформаційних технологій та забезпечує перехід до більш ефективних базисів та систем функцій Уолша, Галуа, комбінованих, які забезпечують ряд унікальних системних властивостей. Кодові системи Галуа повністю сумісні з кодовими системами, що набули застосування в даний час (як то двійкова, Грея та т.п.). Сучасний досвід

досліджень і практика підтверджують забезпечення простої технічної, алгоритмічної, програмної реалізації та покриття більшості системних функцій при переході до вертикальної інформаційної технології в кодових системах Галуа.

В результаті аналізу можна підсумувати:

- відомі базиси є функціонально обмежені внаслідок їхньої природної породжуючої первинності, що призводить до обмеження набору системних функцій їх ефективного застосування;

- ряд проміжних кодових систем аналітично не описані базисними функціями, що не дозволяє здійснити проміжні теоретико-числові перетворення і ускладнює технічну реалізацію алгоритмів перетворення форми інформації та цифрової обробки даних;

- базис Галуа володіє необхідною функціональною широтою системного застосування, що потребує його подальшого дослідження, розвитку та практичного впровадження кодових систем Галуа.

Для визначення теоретико-числових основ кодових систем Галуа проведемо дослідження основних впорядкувань базисних функцій та встановимо функціональні взаємозалежності базисних перетворень.

Довільна аналітична функція  $\Phi(t)$ , що є вихідною системою  $p$ -ічних функцій алгебри логіки від  $n$  аргументів в узагальненому випадку представляється на  $[0, p^n)$  як кусочно-постійна з інтервалом постійності, рівним одиниці. Тому в якості базисних вибираються теж кусочно-постійні функції з тим же інтервалом постійності на  $[0, p^n)$ . Існує можливість визначити базис так, що число ненульових членів ряду, який визначає  $\Phi(t)$ , буде скінченним і не перевищуватиме  $p^n$ . Доцільний вибір базису таким чином, щоб кожна із базисних функцій приймала відносно невелике число значень, що спрощує технічну та алгоритмічну реалізацію системи функцій алгебри логіки. Оптимальним є вибір  $p = 2$ , для якого базисні функції приймають значення  $0, \pm 1$ .

Інваріантно до виконуваної системної функції на довільному із етапів подання інформації, повідомлення приймають певні значення в функції часу. Для аналізу та ефективної обробки даних виникає необхідність представлення досліджуваної функції  $\Phi(t)$  в розкладі різноманітними базисними ортогональними системами функцій. Залежно від типу аналізованих сигналів базисні функції повинні бути підібрані таким чином, щоб необхідна точність представлення забезпечувалась при мінімумі членів ряду [130]. Це дозволяє досягнути високих техніко-економічних характеристик при апаратурній реалізації засобів перетворення форми інформації.

Здійснимо аналіз найбільш широко застосовуваних базисів, кодів та кодових системи, що ними породжуються.

Базисними в класі нелінійно-сходимих для представлення аналогових сигналів є функції Чебишева, Лаггера, Ерміта і т.п., а для представлення дискретних повідомлень – функції Фібоначчі. Обмеження широкого практичного впровадження базисних нелінійно-сходимих функцій зумовлена складністю технічної реалізації високоточних генераторів базисних функцій та забезпечення їхньої стабільності.

Широкого застосування в теорії та техніці цифрової обробки сигналів набули лінійно-сходимі функції. Для представлення аналогових сигналів застосовуються негармонічний базис Лежандра та гармонічні базиси, представлені рядом Фур'є (із *sin*- та *cos*- складовими) та гармонічно-частотним базисом [130; 131].

Об'єктивною необхідністю впровадження цифрових засобів формування, перетворення, передачі та обробки повідомлень зумовлений перехід до використання дискретних теоретико-числових базисів, тому важливим є аналіз ефективності їх застосування та властивостей кодів і кодових систем, що ними породжуються.

Проблема аналізу ефективності взаємозв'язку різних дискретних базисів неодноразово порушувалась в галузі цифрової обробки та перетворення сигналів, проте не здійснено охоплювання з єдиних теоретичних узагальнень всієї повноти взаємодії зв'язків різних базисів, особливо найменш повно подано дискретний базис Галуа та кодові системи, що ним породжуються. Не встановлено також повного наскрізного зв'язку між базисами та відповідними їм кодовими системами, що не дало можливості в найбільш повній мірі здійснити міжбазисні та міжсистемні аналітичні переходи.

Здійснено аналіз дискретних теоретико-числових базисів із встановленням їх математичних основ, аналітичних взаємозалежностей та визначення основних властивостей, на підставі яких проведено класифікацію базисів, породжуваних ними кодових систем та базисних матриць теоретико-числових перетворень.

Дискретні лінійно-сходимі теоретико-числові базиси підрозділяються в свою чергу на три групи: дискретно-гармонічні, дискретно-нерегулярні та комбіновані.

В якості необхідних висунемо умови [130; 131]:

- періоду визначеності базисних функцій  $[0, T]$ , де  $T=2\pi$ , або в окремих обумовлених випадках  $[-T/2; T/2]$ ;

- лінійної незалежності функцій  $B(t) = \{b_0(t), b_1(t), \dots, b_n(t), \dots\}$ ;
- для ненормованих функцій (як то Хаара)

$$(h_i, h_j) = \sum_0^T h_i(t) \cdot h_j(t) \cdot \rho(t) \Delta t = \begin{cases} 0 & \text{для } n \neq m \\ 1 & \text{для } n = m \end{cases},$$

де  $\rho(t)$  - значення вагового коефіцієнта;

- при графічному, матричному та базисному поданні будемо реалізувати sign-процедуру заокруглення

$$\text{sign}\rho(t) = \begin{cases} 1 & \text{для } x(t) > 0 \\ -1 & \text{для } x(t) < 0, \\ 0 & \text{для } x(t) = 0 \end{cases}$$

приводячи функцію до нормованого вигляду, тобто для всіх функцій там, де вони не рівні нулю, будемо заокруглювати цілочисельно значення до  $\pm 1$ . Це допоможе формалізувати перехід від реальних функцій, представлених з вагою  $\rho(t)$  до бінарних логічних, придатних для синтезу цифрових обчислювальних пристроїв.

В якості вихідних в первинних перетворювачах, засобах перетворення форми інформації, зокрема в АЦП паралельної архітектури та унітарних процесорах широкого застосування набули нормальні одиничні коди, по іншому - унітарні, або термометричні. Для математичного представлення унітарних кодів використано унітарні функції, що аналітично описуються виразом

$$\text{Uni}(n, \theta, i) = \text{sign}[\sin(2^{n-1} \cdot i) \pi, \theta],$$

де  $i = 0, 1, 2, \dots, n$  - порядковий номер функції,  $\theta$  - параметр часу, тобто час, нормований до інтервалу  $T$ :  $\theta = t/T$ , де  $t$  - поточне значення часу,  $n = \log_2 N$  - порядок набору базисних функцій теоретико-числових перетворень,  $N$  - модуль цілочислених значень системи функцій.

Функції молодших порядків  $n$  повного набору системи функцій є результатом попарного добутку вибіркового функцій вищих порядків  $n+1$  згідно залежності

$$\text{Uni}(n, \theta, i) = \text{Uni}(n+1, \theta, i) \text{Uni}(n+1, \theta, i+2^{n-1}).$$

Недоліком системи унітарних функцій є їхня неортогональність, чим непрямо зумовлена некомпактна (повна) упаковка кодових елементів, що спричиняє значну надлишковість інформаційних потоків, тому і недоцільності застосування при передачі повідомлень. В той же час є ряд прикладів ефективного застосування унітарних кодів в цифрових унітарних кореляційних процесорах.

В літературі відсутні відомості про перехідні системи функцій від унітарних до інших. Це в певній мірі ускладнює вирішення проблеми переходу від Булевої алгебри до багатозначної. Для вирішення даної задачі в якості перехідного вводиться система дискретно-фазових функцій, що є основою коду Лібова-Крейга-Ліппела. Система дискретно-фазових функцій є похідним системи унітарних функцій шляхом розширення періоду  $T$  визначення унітарних функцій в два рази. Формально система унітарних функцій порядку  $n-1$  визначає свій період  $2\pi$  на половині періоду  $\pi$  системи дискретно-фазових функцій порядку  $n$ , тому функції з парними номерами системи дискретно-фазових функцій порядку  $n$  є прямим відображенням системи унітарних функцій порядку  $n-1$  із формальним представленням

$$DF(n, \theta, 2i) = Uni(n-1, \theta, i).$$

Набір унітарних функцій 2-го порядку є набором дискретно-фазових функцій 2-го порядку на періоді  $0 \div \pi$ . Представлення системи дискретно-фазових функцій в розмірності вищих порядків потребує підвищення точності, що досягається додатковим введенням проміжних функцій із фазовим зсувом, рівним половині періоду слідування функцій набору найвищого порядку.

Дискретно-фазові функції порядку  $n$  описуються узагальненим аналітичним виразом

$$DF(n, \theta, i) = \text{sign}[\sin(2^{n-i}\pi, \theta)].$$

Система дискретно-фазових функцій на періоді  $T=2\pi$  відображає фазовий зсув  $\Delta\theta$  sign- процедури над функцією  $\sin 2\pi$ , де

$$\Delta\theta = T/N = 2\pi/N = 2\pi/2^n = \pi/2^{n-1}.$$

Слід відмітити, що функції нижчих порядків є результатом попарного добутку вибірових функцій вищих порядків згідно залежності взаємопородження

$$DF(n, \theta, i) = DF(n+1, \theta, i) DF(n+1, \theta, i+2^{n-1}).$$

З іншого боку дискретно-фазові функції є базою для породження базисів Радемахера, Грея, Уолша. Виходячи з точки зору утворення названих базисів та систем функцій із системи дискретно-фазових функцій, необхідно провести спочатку кілька дискретних тригонометричних перетворень. Оскільки повний дискретно-фазовий базис вміщує всі складові фазового зсуву функції виду  $\text{sign}[\sin(2\pi + \Delta\varphi)]$ , то при значеннях  $\Delta\varphi = \pi/2$  вказана функція трансформується в функцію виду  $\text{sign}[\cos 2\pi]$ .

Набір sin- та cos- складових системи дискретно-фазових функцій аналітично представлено у вигляді:



$$DF(0, \theta, 3) = \begin{cases} \text{sign} [\sin 2\pi\theta] \\ \text{sign} [\cos 2\pi\theta] \end{cases}$$

$$DF(2, \theta, 3) = \text{sign} [\cos 2\pi, \theta]$$

$$DF(0, \theta, 2) = \text{sign} [\sin 4\pi, \theta]$$

$$DF(1, \theta, 2) = \text{sign} [\cos 4\pi, \theta]$$

$$DF(0, \theta, 1) = \text{sign} [\sin 8\pi, \theta].$$

Екстракція sin-складових по кожному із порядків  $n$  набору функцій утворює базис Радемахера, а екстракція cos-складових - відповідно базис Грея.

$$\text{Rad}(n, \theta) = DF(n, \theta, 0),$$

$$\text{Rad}(n, \theta) = \text{sign} [\sin 2^n \pi, \theta].$$

$$\text{Аналогічно } \text{Gry}(n, \theta) = DF(n, \theta, i) = \text{sign} [\cos 2^n \pi, \theta].$$

Тому для прикладу  $n = 3$ :

$$\text{Rad}(0, \theta) = \text{sign} [\sin \pi, \theta] = DF(0, \theta, 0),$$

$$\text{Gry}(0, \theta) = \text{sign} [\cos \pi, \theta] = DF(0, \theta, 0),$$

$$\text{Rad}(1, \theta) = \text{sign} [\sin 2\pi, \theta] = DF(1, \theta, 0),$$

$$\text{Gry}(1, \theta) = \text{sign} [\cos 2\pi, \theta] = DF(1, \theta, 2),$$

$$\text{Rad}(2, \theta) = \text{sign} [\sin 4\pi, \theta] = DF(2, \theta, 0),$$

$$\text{Gry}(2, \theta) = \text{sign} [\cos 4\pi, \theta] = DF(2, \theta, 1),$$

$$\text{Rad}(3, \theta) = \text{sign} [\sin 8\pi, \theta] = DF(3, \theta, 0),$$

$$\text{Gry}(3, \theta) = \text{sign} [\cos 8\pi, \theta] = DF(3, \theta, i).$$

Базис Радемахера є основою двійкової системи числення і сьогодні використовується як основний в більшості засобів обчислювальної техніки та систем керування. Базис Грея - основа для побудови кодів Грея, кодових шкал та перетворювачів форми інформації, завдяки своїм унікальним властивостям набув теж широкого застосування. Базиси Радемахера і Грея доповнюють один одного, оскільки базис Радемахера вміщує набір тільки непарних функцій, тобто  $f(t) = -f(-t)$ , а базис Грея вміщує тільки парні функції, тобто  $f(t) = f(-t)$ . Використання базису Радемахера-Грея дозволяє покращити представлення аналізованого сигналу дискретними функціями. Основним недоліком базису Радемахера-Грея є низька сходимість числового ряду, що не дозволяє ефективно здійснити аналого-цифрове перетворення та обробку високочастотних сигналів. Сходимість можливо покращити шляхом рандомізації вагових розрядів базису Радемахера-Грея, внаслідок чого запропоновано нове впорядкування базисних функцій, яке полягає в рандомізації розрядних функцій в зворотному порядку. Базис названо

рефлексно відображеним Радемахера. В результаті досягнуто одних з кращих показників ймовірнісного розподілу значень сигналу сканування при реалізації АЦП Монте-Карло.

Базисні функції Уолша є повною мультиплікативною групою функцій Радемахера та утворюються почерговим взаємним перемноженням усіх функцій Радемахера. Так як базис Уолша окрім набору дискретно-гармонічних функцій вміщує дискретно-нерегулярні функції, визначимо його до класу дискретно-нерегулярних

До класу дискретно-гармонічних належить також дискретно-частотний базис функцій, ортогональних на періоді по взаємопростому модулю (DT), що породжує коди системи залишкових класів, які ґрунтуються на китайській теоремі про залишки. Для однозначного представлення досліджуваної функції на періоді дослідження  $T$  необхідно, щоб кожна із базисних функцій вміщувала кількість цілих періодів, значення яких для базису є взаємно простими числами.

Дискретно-частотні базисні функції на періоді  $T = M = 2\pi$  синтезуються, виходячи із аналітичного виразу

$$DT(M, \theta, p_i) = \text{sign}[\sin M/p_i, 2\pi, \theta]$$

Підкласом дискретно-гармонічних теоретико-числових базисів є базиси гармонічно-кусочно-визначених функцій.

Основою розрядно-позиційних кодів є теоретико-числовий базис Хаара:

$$\text{Har}(n, \theta, i) = \text{sign}[\sin i2^n \pi, \theta].$$

Слід зауважити неповноту названого базису, що в значній мірі звузило його практичне застосування та можливості математичних взаємоперетворень із іншими базисами. Обмеженість даного базису зумовлена також і тим, що він включає тільки  $\sin$ -складові, якими описуються непарні функції  $f(-t) = -f(t)$ . Для розширення функціональних можливостей базису Хаара вперше введено базис  $\cos$ -складових функцій Хаара, що дозволяє представляти парні функції  $f(-t) = f(t)$ . Повний базис Хаара описується набором  $\sin$ - та  $\cos$ -складових:

$$\text{Har}(n, \theta, i) = \begin{cases} \text{Has}(n, \theta, i) = \text{sign}[\sin i2^n \pi, \theta] \\ \text{Hac}(n, \theta, i) = \text{sign}[\cos i2^n \pi, \theta] \end{cases}$$

де  $i=1, 2, \dots, 2^n$ .

Базис Хаара породжується із базису Радемахера-Грея почерговим маскуванням  $\sin$ - та  $\cos$ -складових на періоді  $2\pi$  базисної функції відповідно до порядку  $n$  визначення  $2^{n+1}\pi$  періоду дослідження  $T = 2^{n+1}\pi$ .

$$\text{Har}(n, \theta, i) = \begin{cases} \text{Has}(n, \theta, i) = \text{Rad}(n, \theta) \cdot 2\pi i \\ \text{Hac}(n, \theta, i) = \text{Gry}(n, \theta) \cdot 2\pi i \end{cases}$$

Функції базису Хаара та розрядно-позиційні коди, що ними породжуються, набули широкого застосування в засобах перетворення форми інформації, зокрема, в якості проміжних кодів при аналого-цифровому перетворенні, в давачах переміщень (лінійних та кутових), для ініціалізації розосереджених елементів обчислювальних систем, зокрема комірок па'мяті та т.п.

Одними із основних системних функцій інформаційних систем є формування, генерування, синхронізація та обмін цифровими повідомленнями, які ґрунтуються на застосуванні число-імпульсних кодів. Досліджено систему функцій, що породжує число-імпульсні коди. Для повного подання базису розглянемо повний набір парних та непарних, відповідно  $\sin$ - та  $\cos$ -складових базисних функцій  $Plk(n, \theta)$

$$Plk(n, \theta) = \begin{cases} Pls(n, \theta) = \text{sign}[\sin 2^n \pi, \theta] \cdot (2n - 1) \cdot 2\pi i \\ Plc(n, \theta) = \text{sign}[\cos 2^n \pi, \theta] \cdot (2n - 1) \cdot 2\pi i \end{cases}$$

де  $i=1, 2, \dots, n/2$  - поточне значення номерів періодів  $2\pi$ .

Виражаючи базис  $Plk(n, \theta)$  через базис  $Har(n, \theta, i)$  отримуємо

$$Plk(n, \theta) = \begin{cases} Pls(n, \theta) = Rad(n, \theta) \cdot (2n - 1) \cdot 2\pi i = \sum_n Har(n, \theta, i) \\ Plc(n, \theta) = Gry(n, \theta) \cdot (2n - 1) \cdot 2\pi i = \sum_n Hac(n, \theta, i) \end{cases}$$

Із аналітичних залежностей слідує, що система число-імпульсних функцій прямим чином походить із повного базису  $Rad(n, \theta) - Gry(n, \theta)$  шляхом вибіркового маскування парних значень поточних періодів  $2\pi$  і виділення непарних значень поточних періодів  $2\pi$  відповідно до порядку  $n$  базисної функції. Отриманий аналітичний опис системи число-імпульсних функцій дозволяє спростити вирішення задачі математичного подання та аналізу запропонованого нового класу інтегрально-імпульсних кодів у базисі Галуа, що набули перспективи широкого практичного застосування в формувачах та перетворювачах форми інформації. Математичний апарат формування число-імпульсних кодових послідовностей, розроблений на основі базису  $Plk(n, \theta)$ , застосовано при проектуванні засобів формування, генерування та дистанційної передачі даних в кодах Галуа.

Таким чином, аналіз дискретно-гармонічних базисів в обсязі наведеної класифікації вичерпаний. Перейдемо до класу дискретно-нерегулярних базисів, із яких базиси Уолша та Галуа в останній час знаходять все ширше застосування при розробці систем контролю та передачі інформації.

**Висновки.** На підставі аналізу теорії дискретних ТЧП, сучасного стану і тенденцій розвитку методів формування, перетворення та цифрової обробки

повідомлень узагальнені і визначені перспективні напрямки розвитку вертикальних інформаційних технологій на основі кодових систем Галуа. Проведено класифікацію кодових систем по ступеню реалізації системних функцій інформаційних технологій. Запропоновано аналітичне узагальнення теоретичних основ дискретних базисів та розширення їх класів шляхом диференціації на дискретно-гармонічні та дискретно-нерегулярні базиси. Додано відомі базиси  $\sin$ - та  $\cos$ - складовими, внаслідок чого розширено функціональні можливості базисів, що представлені тільки парними або непарними системами дискретних функцій. Введено ряд нових впорядкувань базисних функцій, які спрощують взаємобазисні перетворення та проведено їх ідентифікацію відомими кодовими системами. Встановлено повні функціонали взаємоперетворень різних теоретико-числових базисів, що дозволило здійснити формалізацію аналітичних переходів між класифікованими базисами. Отримані теоретичні функціонали є основою для реалізації засобів міжбазисних перетворень дискретних повідомлень, представлених відповідними кодовими системами.

## РОЗДІЛ 18

# АНАЛІЗ ЕФЕКТИВНОСТІ МЕТОДІВ ПЕРЕТВОРЕННЯ ФОРМИ ІНФОРМАЦІЇ В АДИТИВНИХ СИСТЕМАХ ЧИСЛЕННЯ

*Анотація.* В роботі проаналізовано ефективність застосування двійкових методів перетворення форми інформації з надлишковими операціями як способу вирішення проблеми невизначених областей в двійкових методах з лінійними операціями перетворення. Побудовано векторно-розгалужені діаграми, які дозволяють здійснити моделювання, візуалізацію та оцінку процесів ПФІ. Визначено кількісні параметри ефективності кожного з методів та вказано оптимальний.

**Ключові слова:** перетворення форми інформації, двійкове числення, векторно-розгалужені діаграми, моделювання.

**Abstract.** In this paper presents the feasibility of using binary transformation of form information methods with redundant operations as solving the problem of empty areas in the binary method with linear operations. Constructed vector-branching models that reflect the processes flow of TIF for all methods. Determined the quantitative parameters for each of the methods by which the identified the optimal.

**Keywords:** transformation of information form, binary numeral system, vector-branching diagrams, modeling.

**Вступ та постановка завдання.** При перетворенні форми інформації (ПФІ) вирішальним є вибір методу кодування цифрових повідомлень, який в стані забезпечити оптимальні техніко-економічні показники засобів ПФІ. На стадії аналізу та обґрунтування вибору методу ПФІ ефективним є застосування методології моделювання процесів перетворення повідомлень. Візуалізація та можливість кількісної оцінки динамічних показників процесів ПФІ виявились ключовими чинниками в виборі, обґрунтуванні доцільності застосування та розробки методів моделювання.

За результатами аналізу визначено два основні методи моделювання процесів ПФІ: на основі зрівноважування мас на шальковій вазі, а також на основі вимірювання довжин, кожен із яких має проаналізовані в [134] функціональні обмеження.

Мета проведених досліджень полягала в розробці методу моделювання процесів ПФІ з надлишковими операціями в двійковій системі числення на основі векторно-розгалужених діаграм (ВРД), які дозволили виявити причину формування областей невизначеності числових значень діапазону перетворення

та здійснити графічну візуалізацію процесу ПФІ, відобразити процес перетворення для кожного та всіх числових значень квантів діапазону перетворення, надати цілісну інформацію про перебіг процесу, а також здійснити кількісну оцінку властивостей аналізованих методів ПФІ [135].

Практика моделювання процесів ПФІ ґрунтується на здійсненні визначених процедур порівняння, що реалізуються за допомогою компараторів, які дозволяють визначити співвідношення невідомої величини з визначеною системою “еталонних величин” або “мір”, чи то “шкалою”, сформованою в системі одиниць перетворення. Моделювання процесу ПФІ зводиться до послідовних порівнянь невідомого значення величини  $x$  з заданими визначеним чином значеннями системи “мір”, які формуються як результат суми на кожному ітераційному кроці процесу перетворення. За результатами ітераційних порівнянь формується кінцевий результат кількісної оцінки невідомої величини вхідного параметру перетворення.

Запропонована методика моделювання на основі ВРД ґрунтується на застосуванні шкал порядку [136], тому умовою завершення процесу ПФІ є визначення невідомого значення вхідного параметру перетворення  $x$ , що знаходиться у відповідному одиничному проміжку кванту діапазону перетворення. Індикаторний елемент (ІЕ) в моделюванні виконує функцію компаратора, який в межах  $i$ -го кванту значення величини перетворення  $x$  в точці порівняння  $P_{i-1}$  по своєму виходу формує значення 1 ( $\geq$ ), а в точці  $P_i$  – значення 0 ( $<$ ). ІЕ, як компонент порівняння, що формує по виходу два альтернативні результати згідно заданої умови, на ВРД позначено аналогічно до оператора умовних переходів  $\diamond$ .

### **Моделювання процесів ПФІ в двійковій позиційній системі числень**

Алгоритм моделювання адитивного процесу ПФІ довільного невідомого значення  $x$  вхідної величини в код позиційної системи числення за основою  $q$  полягає в здійсненні ітераційних процедур порівняння значення вхідної невідомої величини  $x$  з набором значень еталонних величин (мір), що становлять степеневий ряд:

$$x = a_{n-1}q^{n-1} + a_{n-2}q^{n-2} + \dots + a_iq^i + \dots + a_2q^2 + a_1q^1 + a_0q^0,$$

де  $q$  – основа системи числення;

$a_i$  – розрядні коефіцієнти, що набувають значень  $a_i = \{q-1, 0\}$ .

Проаналізуємо моделі основних методів ПФІ з адитивним зрівноваженням в позиційних системах числень. Початкове перетворення невідомого значення вхідної величини  $x$  як наслідок ітераційного прикладання

ІЕ до алгоритмічно визначених точок діапазону перетворення вхідної величини до  $i$ -го моменту перевищення значення суми позиційних мір  $\sum q^i$  можливе:

- від молодших значень мір  $q^0$  до старших  $q^{i-1}$ ;
- від старших значень мір  $q^{n-1}$  до молодших  $q^{n-i}$ .

Зворотне перетворення після перевищення значенням суми мір  $\sum q^i$  значення невідомої вхідної величини  $x$  можливе:

- від молодших значень мір  $q^0$  до старших значень  $q^{i-2}$ ;
- від старших значень мір  $q^{i-2}$  до молодших значень  $q^0$ .

В табл. 18.1 узагальнено методи адитивного порівняння значення вхідної невідомої величини  $x$  з набором значень еталонних величин (мір)  $\sum q^i$ , значення яких становлять степеневий ряд основи системи числення  $q^i$  [137].

Таблиця 18.1

**Узагальнена класифікація методів адитивного ПФІ  
в позиційних системах числень**

		Методи адитивного ПФІ	
Додавання	від молодших значень мір $q^0$ до старших $q^{i-1}$	1.1	1.2
	від старших значень мір $q^{n-1}$ до молодших $q^{n-i}$	2.1	2.2
<b>Спосіб формування суми мір</b>		від молодших значень мір $q^0$ до старших $q^{i-2}$	від старших значень мір $q^{i-2}$ до молодших $q^0$
		Віднімання	

Проаналізуємо методи і відповідні їм алгоритми ПФІ та побудуємо ВРД і таблиці перебігу ПФІ як моделі процесу адитивного порівняння на прикладі двійкової системи числення за основою  $q=2$ .

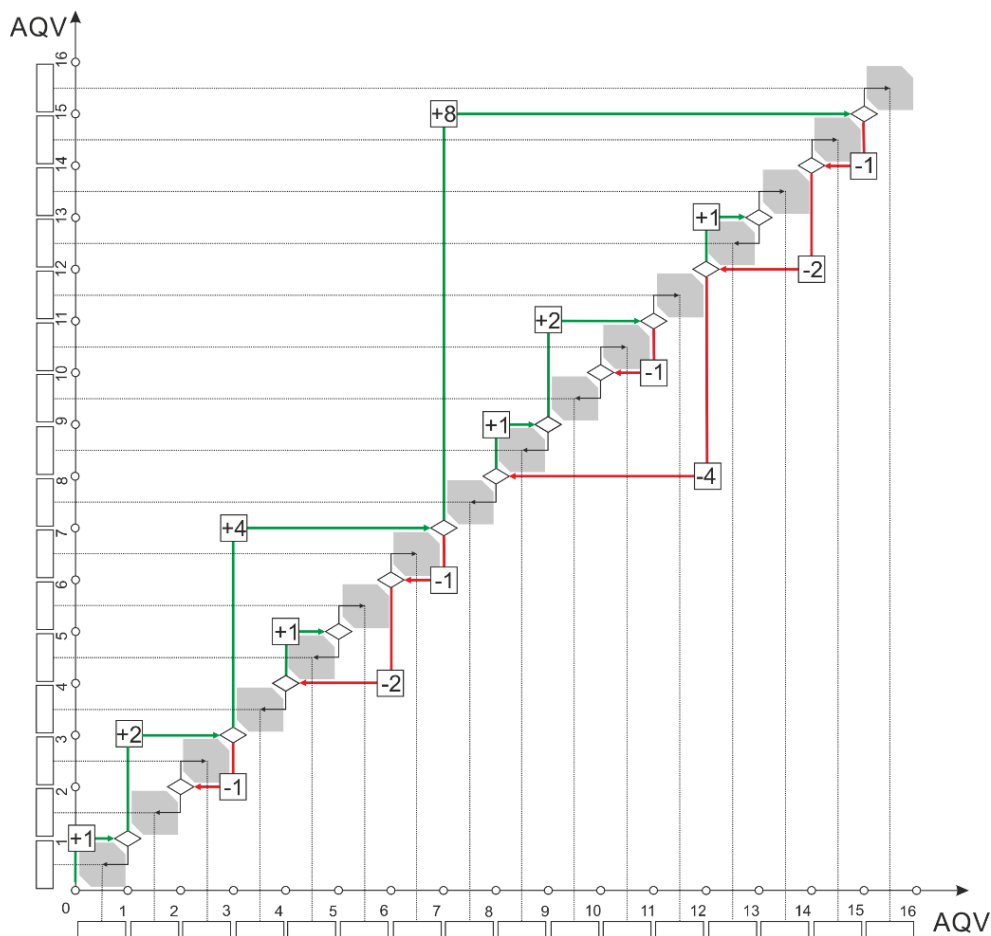
**Метод ПФІ (1.1) із адитивним формуванням двійкової суми мір  $\sum 2^i$   
від молодших значень мір  $2^0$  до старших  $2^{i-1}$  та зворотним  
порівнянням від молодших значень мір  $2^0$  до старших  $2^{i-2}$**

Даний метод передбачає виконання наступних операцій (рис. 18.1):

- 1) додавання позиційних мір здійснюється від молодших значень  $2^0$  до старших  $2^{i-1}$ , ( $2^0, 2^1, \dots, 2^{i-1}$ ) до  $i$ -го моменту порівняння, на якому отримане

значення суми мір  $\sum 2^i$  перевищує невідоме вхідне значенням  $x$ , внаслідок чого ІЕ по своєму виходу формує результат 0 (<);

2) зворотне перетворення значення отриманої суми мір  $\sum 2^i$  виконується шляхом віднімання попередньо встановлених значень позиційних мір в порядку від молодших значень  $2^0$  до старших  $2^{i-2}$  ( $2^0, 2^1, \dots, 2^{i-2}$ ) до моменту отримання такого значення суми мір  $\sum 2^i$ , внаслідок порівняння значення якого із невідомим вхідним значенням  $x$  в точці порівняння ІЕ по своєму виходу формує результат 1 ( $\geq$ );



**Рис. 18.1. Приклад моделювання процесу ПФІ із адитивним формуванням двійкової суми мір  $\sum 2^i$  від молодших значень  $2^0$  до старших  $2^{i-1}$  та зворотним перетворенням від молодших значень мір  $2^0$  до старших  $2^{i-2}$  для діапазону перетворення  $0 \div 2^4 - 1$**

3) якщо значення сформованої суми мір  $\sum 2^i$  потрапляє в одиничний проміжок кванту значення  $x$ , процес ПФІ завершується, в іншому випадку, в залежності від стану виходу ІЕ, повторюються кроки 1 та 2 до виконання умови попадання значення сформованої суми  $\sum 2^i$  в одиничний проміжок кванту визначення невідомого значення  $x$ .



В табл. 18.2 представлено покроковий перебіг процесу ПФІ кожного із числових значень  $x$  діапазону перетворення  $0 \div 2^4 - 1$ , ВРД якого зображена на рис. 18.1. В табл. 18.1 визначено типи операцій (додавання чи віднімання), які необхідно виконати для входження в проміжок кванту визначення кожного із числових значень  $x$  та сумарну кількість ітераційних кроків (рис. 18.2) як для окремих значень  $x$ , так і для повного діапазону перетворення  $0 \div 2^4 - 1$ .

Таблиця 18.2

**Перебіг процесу ПФІ із адитивним формуванням двійкової суми мір  $\sum 2^i$  від молодших значень  $2^0$  до старших  $2^{i-1}$  та наступним порівнянням від молодших значень мір  $2^0$  до старших  $2^{i-2}$**

Ітераційні кроки порівняння/компарування (ПФІ) ( $i=...$ )													значення кванту ПФІ	
$x$	1	2	3	4	5	6	7	8	9	10	знак	$\sum i$		
<b>0</b>	+1										<	1	[0,1)	
<b>1</b>	+1	+2	-1								<	3	[1,2)	
<b>2</b>	+1	+2	-1								$\geq$	3	[2,3)	
<b>3</b>	+1	+2	+4	-1	-2						<	5	[3,4)	
<b>4</b>	+1	+2	+4	-1	-2	+1					<	6	[4,5)	
<b>5</b>	+1	+2	+4	-1	-2	+1					$\geq$	6	[5,6)	
<b>6</b>	+1	+2	+4	-1							$\geq$	4	[6,7)	
<b>7</b>	+1	+2	+4	+8	-1	-2	-4				<	7	[7,8)	
<b>8</b>	+1	+2	+4	+8	-1	-2	-4	+1			<	8	[8,9)	
<b>9</b>	+1	+2	+4	+8	-1	-2	-4	+1	+2	-1	<	10	[9,10)	
<b>10</b>	+1	+2	+4	+8	-1	-2	-4	+1	+2	-1	$\geq$	10	[10,11)	
<b>11</b>	+1	+2	+4	+8	-1	-2	-4	+1	+2		$\geq$	9	[11,12)	
<b>12</b>	+1	+2	+4	+8	-1	-2	+1				<	7	[12,13)	
<b>13</b>	+1	+2	+4	+8	-1	-2	+1				$\geq$	7	[13,14)	
<b>14</b>	+1	+2	+4	+8	-1						$\geq$	5	[14,15)	
<b>15</b>	+1	+2	+4	+8							$\geq$	4	[15,16)	

Сумарна кількість кроків, яка необхідна для визначення всіх значень з даного діапазону, рівна 95.



**Рис. 18.2. Залежність кількості кроків перетворення в функції кількості квантів**

Рис. 18. 2 відображає залежність кількості значень квантів ПФІ в функції кількості кроків перетворення в діапазоні  $0 \div 2^8 - 1$ .

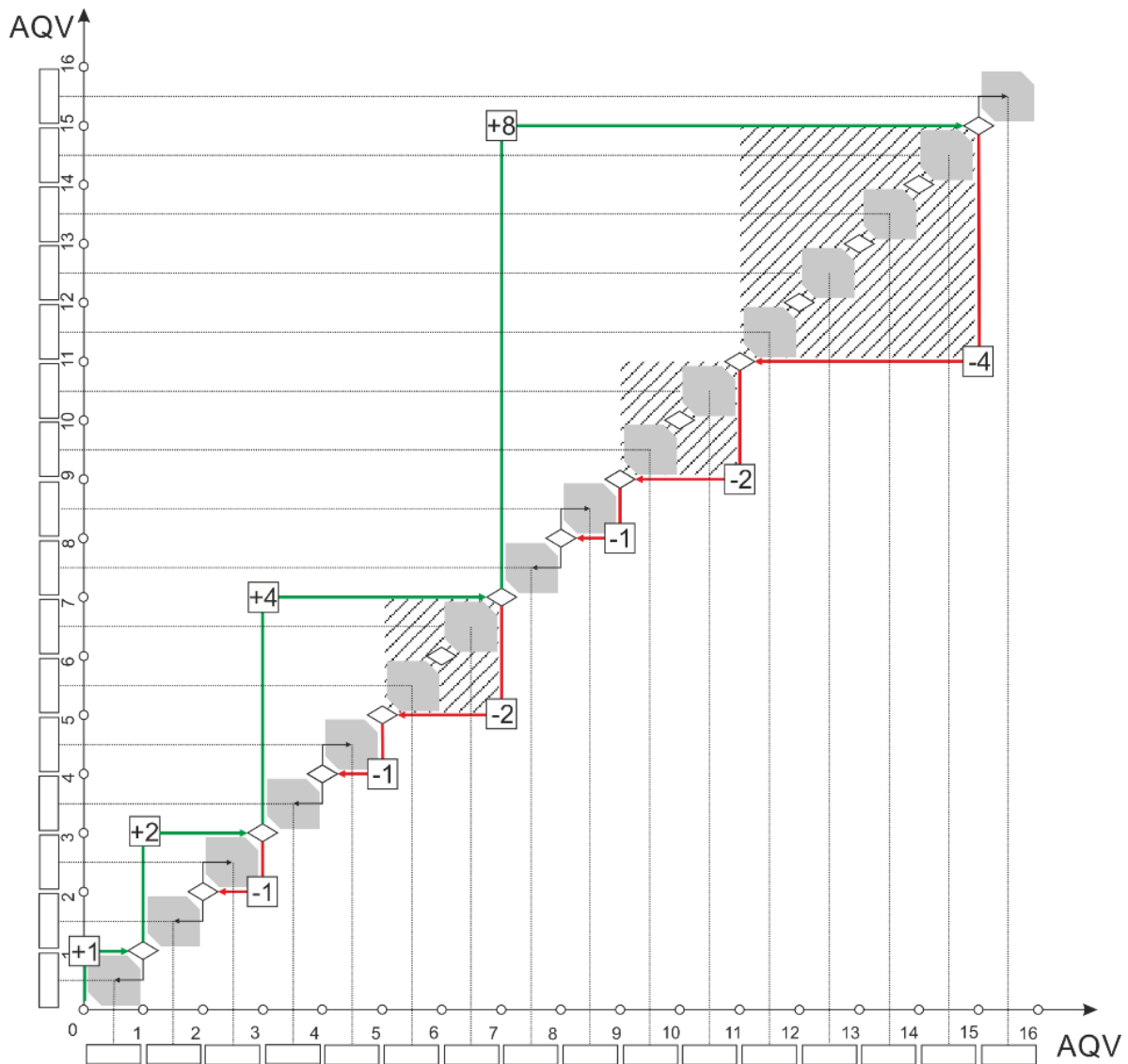
**Метод ПФІ (1.2) із адитивним формуванням двійкової суми мір  $\sum 2^i$  від молодших значень мір  $2^0$  до старших  $2^{i-1}$  та зворотним перетворенням від старших значень мір  $2^{i-2}$  до молодших  $2^0$**

Згідно даного методу виконується послідовність наступних операцій (рис. 18.3):

1) додавання позиційних мір здійснюється від молодших значень  $2^0$  до старших  $2^{i-1}$ ,  $(2^0, 2^1, \dots, 2^{i-1})$ , до  $i$ -го моменту порівняння, в якому отримане значення суми мір  $\sum 2^i$  перевищує невідоме вхідне значенням  $x$ , внаслідок чого ІЕ по своєму виходу формує результат 0 (<);

2) зворотне перетворення значення отриманої суми мір  $\sum 2^i$  виконується шляхом віднімання попередньо встановлених значень позиційних мір в порядку від старших значень  $2^{i-2}$  до молодших  $2^0$ ,  $(2^{i-2}, \dots, 2^1, 2^0)$  до моменту отримання значення суми мір  $\sum 2^i$ , внаслідок порівняння якого із невідомим вхідним значенням  $x$  в точці порівняння ІЕ по своєму виходу формує результат 1 ( $\geq$ );

3) якщо значення сформованої суми мір  $\sum 2^i$  потрапляє в одиничний проміжок кванту значення  $x$ , процес ПФІ завершується, в іншому випадку, в залежності від стану виходу ІЕ, повторюються кроки 1 та 2 до виконання умови попадання значення сформованої суми  $\sum 2^i$  в одиничний проміжок кванту визначення значення  $x$ .



**Рис. 18.3. Приклад моделювання процесу ПФІ із адитивним формуванням двійкової суми мір  $\sum 2^i$  від молодших значень  $2^0$  до старших  $2^{i-1}$  та зворотним перетворенням від старших  $2^{i-2}$  до молодших значень мір  $2^0$  для діапазону перетворення  $0 \div 2^4 - 1$**

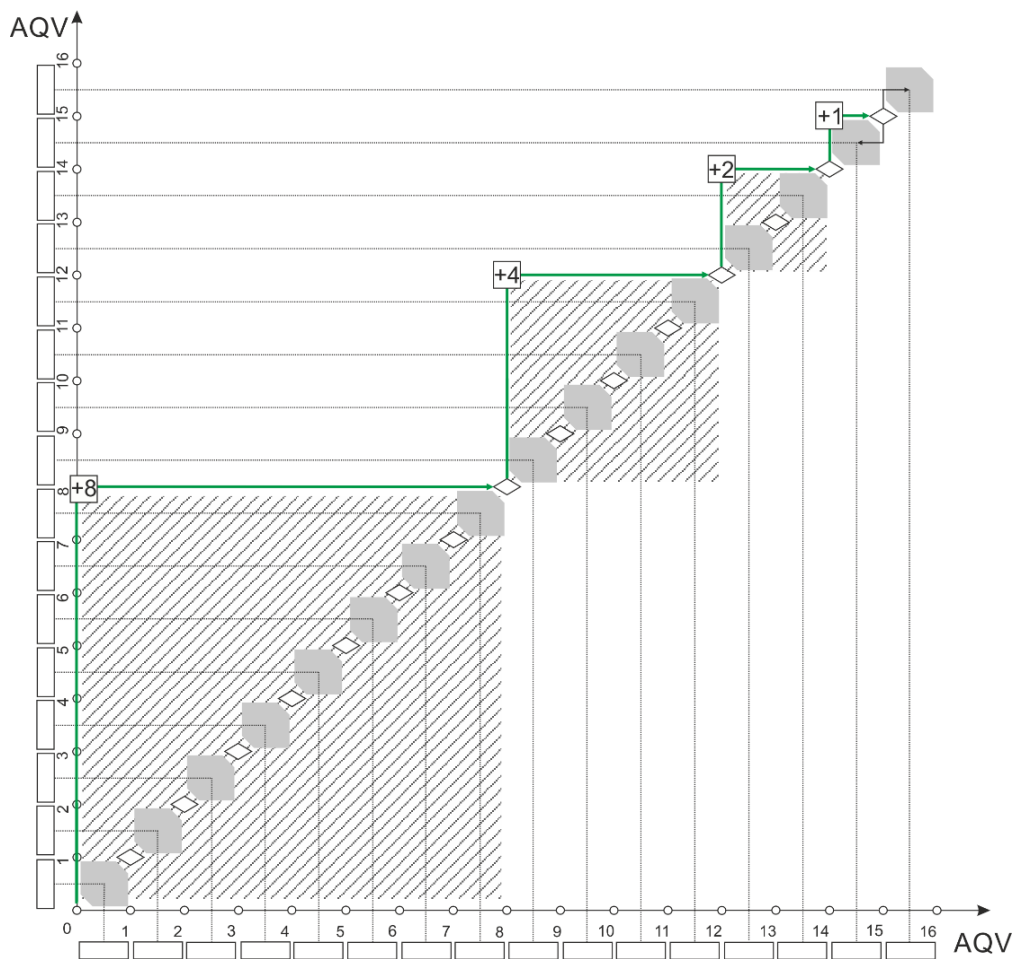
Недоліком даного методу є те, що існують проміжки, в яких, згідно однозначно сформульованих правил виконання алгоритму, неможливо визначити значення невідомої величини  $x$ , що спричинено відсутністю значень позиційних мір в наборі сформованої суми  $\sum 2^i$ , які, з метою зрівноваження можна було б додати чи зняти з метою попадання в значення кванту невідомої величини  $x$ . На рис. 18.3 проміжки невизначеності  $[5 \div 7)$ ,  $[9 \div 11)$  та  $[11 \div 15)$  означено заштрихованими областями. Слід підсумувати, що вказаний принциповий недолік унеможливорює здійснення ПФІ в повному діапазоні перетворення методом, визначеним як 1.2 в класифікаційній табл. 18.1.

**Метод ПФІ (2.1) із адитивним формуванням двійкової суми мір  $\sum 2^i$  від старших  $2^{n-1}$  до молодших значень мір  $2^{n-i}$**

**та зворотним перетворенням від молодших  $2^0$  до старших значень мір  $2^{i-2}$**

Даний метод передбачає необхідність виконання наступних операцій (рис. 18.4):

1) додавання позиційних мір здійснюється від старших значень  $2^{n-1}$  до молодших  $2^{n-i}$ , ( $2^{n-1}, 2^{n-2}, \dots, 2^{n-i}$ ) до  $i$ -го моменту порівняння, на якому отримане значення суми мір  $\sum 2^i$  перевищує невідоме вхідне значення  $x$ , внаслідок чого ІЕ по своєму виходу формує результат 0 (<);



**Рис. 18.4. Приклад моделювання процесу ПФІ із адитивним формуванням двійкової суми мір  $\sum 2^i$  від старших  $2^{n-1}$  до молодших значень мір  $2^{n-i}$  та зворотним перетворенням від молодших  $2^0$  до старших значень мір  $2^{i-2}$**

2) зворотне перетворення значення отриманої суми мір  $\sum 2^i$  виконується шляхом віднімання попередньо встановлених значень позиційних мір в порядку від молодших значень  $2^0$  до старших  $2^{i-2}$ , ( $2^0, 2^1, \dots, 2^{i-2}$ ) до моменту отримання такого значення суми мір  $\sum 2^i$ , внаслідок порівняння якого із невідомим вхідним значенням  $x$  в точці порівняння ІЕ по виходу формує результат 1 ( $\geq$ );

3) якщо значення сформованої суми мір  $\sum 2^i$  потрапляє в одиничний проміжок кванту значення  $x$ , процес ПФІ завершується, в іншому випадку, в залежності від стану виходу ІЕ, повторюються кроки 1 та 2 до виконання умови попадання значення сформованої суми  $\sum 2^i$  в одиничний проміжок кванту визначення невідомого значення  $x$ .

Зворотне зрівноваження, згідно визначених правил алгоритму перетворення є неможливим, тому що на кожному кроку додавання позиційної міри  $2^i$  не можна здійснити зворотне перетворення від молодших значень мір до старших, оскільки на відповідних кроках зрівноваження в складі суми мір  $\sum 2^i$  відсутні значення мір, молодші  $2^i$ . Області невизначеності на рис. 18.4 заштриховано в проміжках  $[0 \div 8)$ ,  $[8 \div 12)$ ,  $[12 \div 14)$ . Вказаний принциповий недолік унеможлиблює здійснення ПФІ в повному діапазоні перетворення методом, визначеним як 2.1 в класифікаційній табл. 18.1.

**Метод ПФІ (2.2) із адитивним формуванням двійкової суми мір  $\sum 2^i$  від старших  $2^{n-1}$  до молодших значень мір  $2^{n-i}$**

**та зворотним перетворенням від старших значень мір  $2^{i-2}$  до молодших  $2^0$**

Згідно даного методу виконується послідовність наступних операцій:

1) додавання позиційних мір здійснюється від старших значень  $2^{n-1}$  до молодших  $2^{n-i}$ , ( $2^{n-1}, 2^{n-2}, \dots, 2^{n-i}$ ) до  $i$ -го моменту порівняння, в якому отримане значення суми мір  $\sum 2^i$  перевищує невідоме вхідне значенням  $x$ , внаслідок чого ІЕ по своєму виходу формує результат 0 (<);

2) зворотне перетворення значення отриманої суми мір  $\sum 2^i$  виконується шляхом віднімання попередньо встановлених значень позиційних мір в порядку від старших значень  $2^{i-2}$  до молодших  $2^0$ , ( $2^{i-2}, 2^{i-3}, \dots, 2^0$ ) до моменту отримання такого значення суми мір  $\sum 2^i$ , внаслідок порівняння якого із невідомим вхідним значенням  $x$  в точці порівняння ІЕ по своєму виходу формує результат 1 ( $\geq$ );

Проте, для даного методу (2.2 в табл. 18.1), аналогічно зображеному на рис. 18.4, серед значень мір, що формують суму  $\sum 2^i$  відсутні значення мір, молодші значень від  $2^{i-2}$  до  $2^0$ , за допомогою яких можна було б здійснити зворотне перетворення до молодших значень, що є принциповим обмеженням, яке унеможлиблює здійснення процесу ПФІ даним методом. Області невизначеності між точками прикладання ІЕ на рис. 18.4 заштриховано в діапазонах  $[0 \div 8)$ ,  $[8 \div 12)$ ,  $[12 \div 14)$ .

Вказаний недолік унеможлиблює здійснення ПФІ в повному діапазоні перетворення методом, визначеним як 2.2 в класифікаційній табл. 18.1.

## Аналіз методів ПФІ із адитивним формуванням двійкової суми мір з надлишковими операціями

Проблему відсутності позиційних мір в складі суми  $\sum 2^i$  для здійснення операцій зворотного перетворення можна вирішити шляхом здійснення надлишкових операцій віднімання старших значень мір  $q^i$  та переходу до молодших значень  $q^{i-1}$ .

Таблиця 18.3

### Класифікація методів адитивного ПФІ з надлишковими операціями порівняння в позиційних системах числень

		методи адитивного ПФІ з надлишковими операціями	
		3.1	3.2
4.1	4.2		
<b>формування суми мір</b>	від молодших значень мір $q^0$ до старших $q^{i-1}$	від старших значень мір $q^{i-1}$ до молодших $q^0$	віднімання

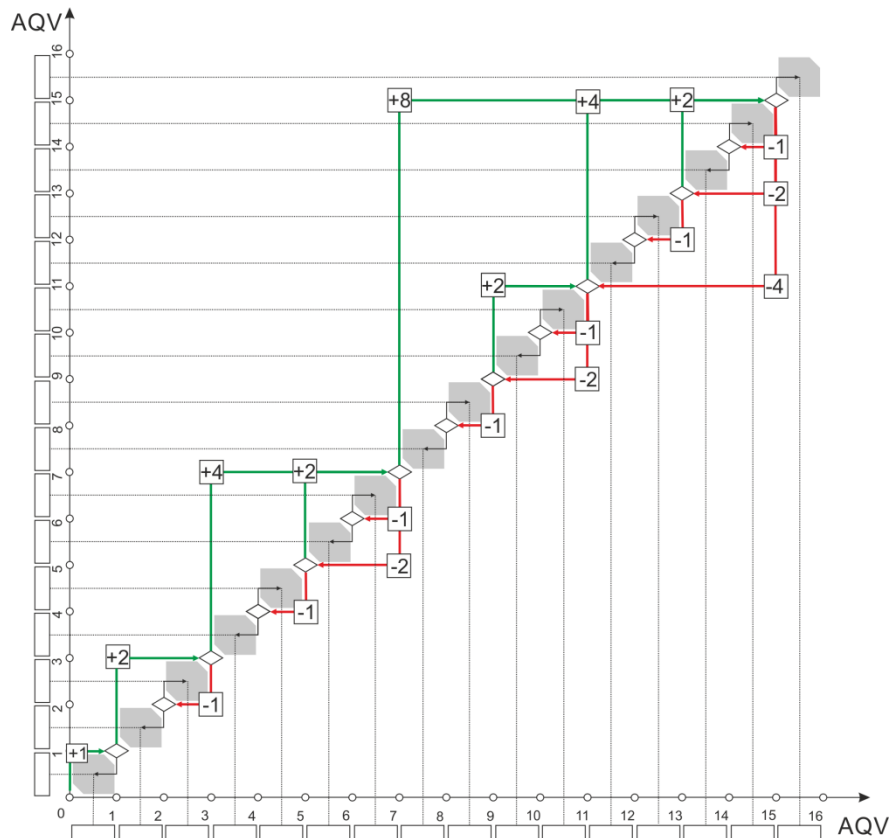
**Аналіз методу ПФІ (3.1) із адитивним формуванням з надлишковими операціями двійкової суми мір  $\sum 2^i$  від молодших  $2^0$  до старших значень мір  $2^{i-1}$  та зворотним порівнянням від молодших  $2^0$  до старших значень мір  $2^{i-2}$**

У даній класифікації метод 3.1 із адитивним формуванням з надлишковими операціями двійкової суми мір  $\sum 2^i$  від молодших  $2^0$  до старших значень мір  $2^{i-1}$  та зворотним порівнянням від молодших  $2^0$  до старших значень мір  $2^{i-2}$  є ідентичним до методу, класифікованому як 1.1 в табл. 18.1.

**Аналіз методу ПФІ (3.2) із адитивним формуванням з надлишковими операціями двійкової суми мір  $\sum 2^i$  від молодших значень мір  $2^0$  до старших  $2^{i-1}$  та зворотним перетворенням від старших  $2^{i-2}$  до молодших значень мір  $2^0$**

Як показано на рис. 18.5, згідно методу 2.2 (табл. 18.1) формування суми позиційних мір  $\sum 2^i$  як результату моделювання процесу аплікації ІЕ до пунктів квантів діапазону перетворення, здійснюється ітераційно в зростаючому порядку від молодших розрядних значень мір  $2^0, 2^1, \dots, 2^{i-1}$  до  $i$ -го ітераційного кроку перевищення значенням сформованої суми  $\sum 2^i$  вхідного невідомого значення перетворення  $x$ , в результаті чого на виході ІЕ буде сформовано

значення  $0 (\leq)$ ,  $i$ , як наслідок, здійснено зворотне прикладання мір, розпочинаючи від старших значень  $2^{i-2}$  мір до молодших  $2^0$ , поки сформоване значення суми  $\sum 2^i$  не потрапить в одиничний проміжок кванту перетворення невідомого значення  $x$ . Можливим є ітераційне вкладення в процес ПФІ коротших циклів додавання-віднімання до моменту потрапляння значення сформованої суми  $\sum 2^i$  в межі кванту визначення  $x$ , на рис. 18.5 – в проміжки квантів  $[5 \div 6)$ ,  $[6 \div 7)$ ,  $[13 \div 14)$ ,  $[14 \div 15)$  та ін.



**Рис. 18.5.** Приклад процесу ПФІ із адитивним формуванням суми мір  $\sum 2^i$  від молодших значень  $2^0$  до старших  $2^{i-1}$  та зворотним порівнянням від старших  $2^{i-2}$  до молодших значень мір  $2^0$  для діапазону перетворення  $0 \div 2^4 - 1$

В табл 18.4 представлено покроковий перебіг процесу ПФІ кожного із числових значень  $x$  діапазону перетворення  $0 \div 2^4 - 1$ . Визначено типи операцій (додавання чи віднімання), які необхідно виконати для входження в межі кванту кожного із числових значень  $x$  для діапазону перетворення  $0 \div 2^4 - 1$ . Сумарна кількість кроків, яка необхідна для визначення всіх значень даного діапазону становить 97.

Таблиця 18.4.

Перебіг процесу ПФІ із адитивним формування двійкової суми мір  $\sum 2^i$  від молодших значень  $2^0$  до старших  $2^{i-1}$  та зворотним порівнянням від старших  $2^{i-2}$  до молодших значень мір  $2^0$

Ітераційні кроки порівняння/компарування (ПФІ) ( $i=...$ )														знач-ня		
$x$	1	2	3	4	5	6	7	8	9					знак	$\sum i$	кванту
0	+1													<	1	[0,1)
1	+1	+2	-1											<	3	[1,2)
2	+1	+2	-1											$\geq$	3	[2,3)
3	+1	+2	+4	-2	-1									<	5	[3,4)
4	+1	+2	+4	-2	-1									$\geq$	5	[4,5)
5	+1	+2	+4	-2	+2	-1								<	6	[5,6)
6	+1	+2	+4	-2	+2	-1								$\geq$	6	[6,7)
7	+1	+2	+4	+8	-4	-2	-1							<	7	[7,8)
8	+1	+2	+4	+8	-4	-2	-1							$\geq$	7	[8,9)
9	+1	+2	+4	+8	-4	-2	+2	-1						<	8	[9,10)
10	+1	+2	+4	+8	-4	-2	+2	-1						$\geq$	8	[10,11)
11	+1	+2	+4	+8	-4	+4	-2	-1						<	8	[11,12)
12	+1	+2	+4	+8	-4	+4	-2	-1						$\geq$	8	[12,13)
13	+1	+2	+4	+8	-4	+4	-2	+2	-1					<	9	[13,14)
14	+1	+2	+4	+8	-4	+4	-2	+2	-1					$\geq$	9	[14,15)
15	+1	+2	+4	+8										<	4	[15,16)

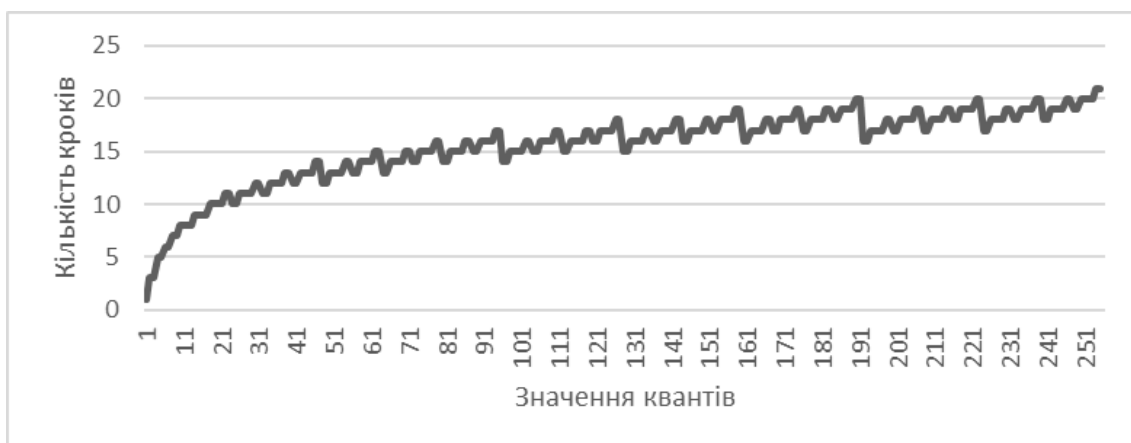


Рис. 18.6. Залежність зміни кількості кроків процесу перетворення в функції кількості квантів

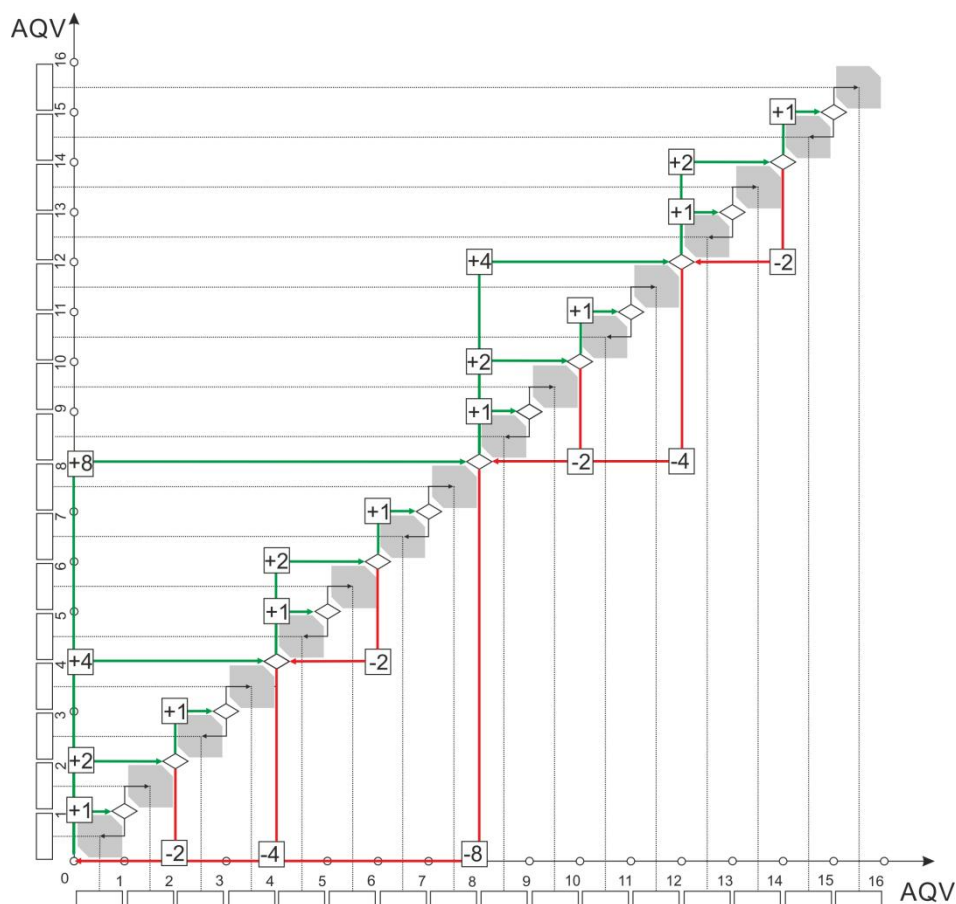


**Аналіз методів ПФІ (4.1, 4.2) із адитивним формування двійкової суми мір з надлишковими операціями  $\sum 2^i$  від старших значень мір  $2^{n-1}$  до молодших  $2^{n-i}$**

Для методу із адитивним формування двійкової суми мір з надлишковими операціями  $\sum 2^i$  від старших значень мір  $2^{n-1}$  до молодших  $2^{n-i}$  існує 2 два можливі варіанти зворотного перетворення:

- від молодших значень мір  $q^0$  до старших  $q^{i-1}$ ;
- від старших значень мір  $q^{i-2}$  до молодших  $q^0$ .

Побудова ВРД для даних двох методів зводиться до одного, оскільки можливо здійснити операції додавання від старшого до молодших значень, зі зняттям/додаванням до моменту знаходження невідомого значення, виключаючи можливості здійснення зворотних перетворень, що складають другу частину алгоритму.



**Рис. 18.7. Приклад моделювання процесу ПФІ із адитивним формування двійкової суми мір  $\sum 2^i$  від старших значень  $2^{n-1}$  до молодших  $2^i$  та зворотним порівнянням від старших значень мір  $2^{i-1}$  до молодших  $2^0$  для діапазону перетворення  $0 \div 2^n - 1$**

Як показано на рис. 18.7, згідно методу 2.3 (табл. 18.1), формування суми позиційних мір  $\sum 2^i$ , як результат ПФІ здійснюється ітераційно в зростаючому порядку, розпочинаючи від старшого значення міри  $2^{n-1}$  діапазону перетворення  $0 \div 2^n - 1$  шляхом покрокового додавання мір  $2^{n-1}, 2^{n-2}, \dots, 2^{i+1}, 2^i$ , до  $i$ -го ітераційного

кроку перевищення значення сформованої суми  $\sum 2^i$  вхідного невідомого значення перетворення  $x$ , в результаті чого на виході ІЕ буде сформовано значення 0 ( $\leq$ ), після чого здійснюється зворотне віднімання мір, розпочинаючи від старших значень  $2^{i-1}$  мір до молодших  $2^0$ , поки невідоме значення  $x$  не потрапить в межі одиничного кванту діапазону перетворення. Можливим є ітераційне вкладення в процес ПФІ коротших циклів додавання-віднімання до моменту потрапляння значення сформованої суми  $\sum 2^i$  в проміжок кванту визначення  $x$ , на рис. 18.7– в проміжки квантів  $[0\div 1)$ ,  $[2\div 3)$ ,  $[3\div 4)$ ,  $[8\div 9)$ ,  $[9\div 10)$  та ін.

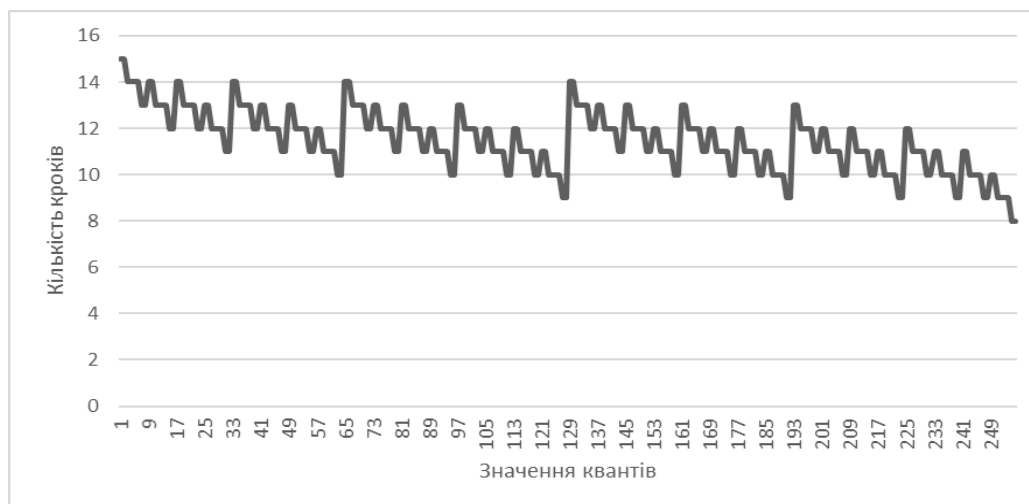
В табл. 18.5 представлено покроковий перебіг процесу ПФІ кожного із числових значень  $x$  діапазону перетворення  $[0\div 2^4-1)$ , зображеного на рис. 18.7. Визначено типи операцій (додавання чи віднімання), які необхідно виконати для потрапляння в межі кванту визначення кожного із числових значень  $x$  та сумарну кількість ітераційних кроків як для окремих значень  $x$ , так і для повного діапазону перетворення  $[0\div 2^4-1)$ . Сумарна кількість кроків, яка необхідна для визначення всіх значень з даного діапазону становить 88.

Таблиця 18.5

**Перебіг процесу ПФІ із адитивним формування двійкової суми мір  $\sum 2^i$  від старших значень  $2^{n-1}$  до молодших  $2^i$  та зворотним порівнянням від старших  $2^{i-1}$  до молодших значень мір  $2^0$**

Ітераційні кроки порівняння/компарування (ПФІ) ( $i=...$ )																знач-ня кванту	
$x$	1	2	3	4	5	6	7	8	9						знак	$\sum i$	
0	+8	-8	+4	-4	+2	-2	+1								<	7	(0,1]
1	+8	-8	+4	-4	+2	-2	+1								$\geq$	7	(1,2]
2	+8	-8	+4	-4	+2	+1									<	6	(2,3]
3	+8	-8	+4	-4	+2	+1									$\geq$	6	(3,4]
4	+8	-8	+4	+2	-2	+1									<	6	(4,5]
5	+8	-8	+4	+2	-2	+1									$\geq$	6	(5,6]
6	+8	-8	+4	+2	+1										<	5	(6,7]
7	+8	-8	+4	+2	+1										$\geq$	5	(7,8]
8	+8	+4	-4	+2	-2	+1									<	6	(8,9]
9	+8	+4	-4	+2	-2	+1									$\geq$	6	(9,10]
10	+8	+4	-4	+2	+1										<	5	(10,11]
11	+8	+4	-4	+2	+1										$\geq$	5	(11,12]
12	+8	+4	+2	-2	+1										<	5	(12,13]
13	+8	+4	+2	-2	+1										$\geq$	5	(13,14]
14	+8	+4	+2	+1											<	4	(14,15]
15	+8	+4	+2	+1											$\geq$	4	[15,16)

Методи позиційного ПФІ володіють основним недоліком, що полягає в надлишковості операцій порівняння, що зумовлено необхідністю здійснення процедур додавання та віднімання тих самих значень мір в процесі перетворення і спричиняє до зменшення швидкодії процесу ПФІ та погіршення динамічних показників, надійності функціонування і збільшення енергоспоживання. На рис. 18.8 зображено залежність кількості кроків в функції кількості квантів.



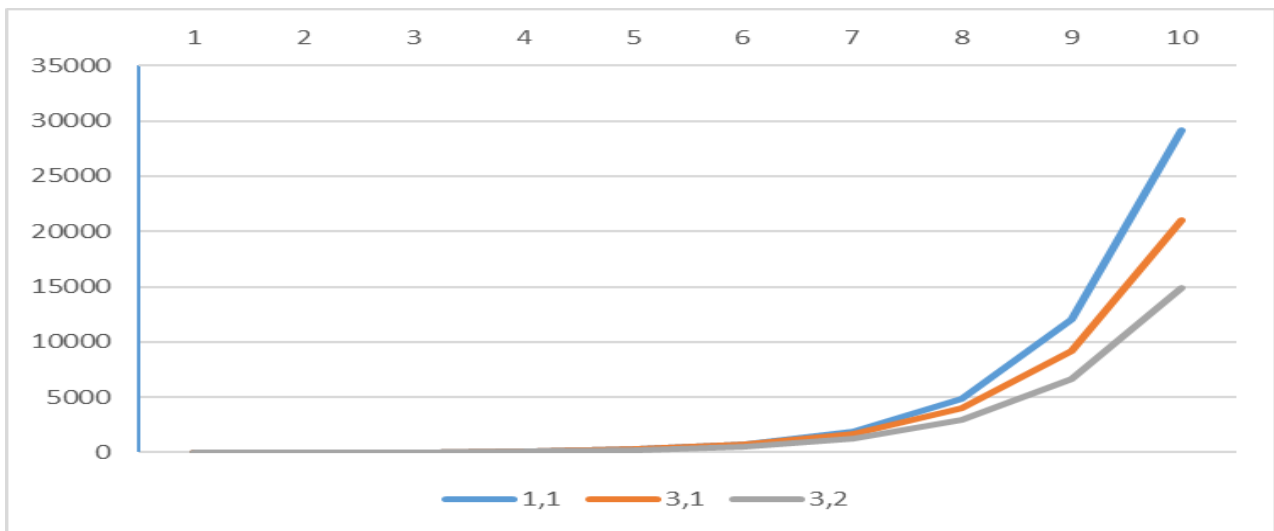
**Рис. 18.8. Залежність зміни кількості кроків в функції кількості квантів  
Порівняння ефективності двійкових методів ПФІ**

Для визначення ефективності проаналізованих методів перетворення здійснимо порівняння суми кількості кроків перетворення для значень формату розрядності 0÷10 (табл. 18.6).

Таблиця 18.6

**Кількість квантів перетворення**

Розряд	Метод 1.1	Метод 3.1	Метод 3.2
<b>1</b>	1	1	1
<b>2</b>	9	9	9
<b>3</b>	31	32	32
<b>4</b>	95	97	88
<b>5</b>	271	266	224
<b>6</b>	735	683	544
<b>7</b>	1919	1676	1280
<b>8</b>	4863	3981	2944
<b>9</b>	12031	9230	6656
<b>10</b>	29183	21007	14848



**Рис. 18.9. Оцінка ефективності двійкових методів ПФІ**

За отриманими результатами можна ствердити, що метод 1.1 володіє вищими техніко-економічними параметрами в порівнянні з методами 3.1 та 3.2.

**Висновки.** За результатами досліджень визначено чотири методи адитивного ПФІ та два методи із надликовими операціями перетворення і, як наслідок, відповідні алгоритми ПФІ в позиційних системах числення, зокрема:

1. з додаванням від молодших значень мір до старших  $+ (q^0 \div q^{i-1})$ , для класу яких можливі наступні два методи зрівноваження після перевищення невідомого значення перетворення  $x$ :

1.1 – із відніманням від молодших значень мір до старших  $- (q^0 \div q^{i-2})$ ;

1.2 – із відніманням від старших значень мір до молодших  $- (q^{i-2} \div q^0)$ ;

2. з додаванням від старших значень мір до молодших  $+ (q^{n-1} \div q^{n-i})$ , для класу яких можливі наступні два методи зворотного порівняння після перевищення невідомого значення  $x$  перетворення:

2.1 – із відніманням від молодших значень мір до старших  $- (q^0 \div q^{i-1})$ ;

2.2 – із відніманням від старших значень мір до молодших  $- (q^{i-2} \div q^0)$ .

Запропонований метод моделювання на базі ВРД дозволив здійснити моделювання процесів ПФІ, визначити властивості та недоліки, які унеможливають реалізацію методів 1.2, 2.1, 2.2 за причини існування невизначеності областей діапазону перетворення. Для вказаних методів запропоновано виконання операцій з надлишковими операціями перетворення від старших значень до молодших. Для кожного з методів визначено параметри, які дозволяють визначити ефективність їх імплементації в засобах ПФІ.

# *Інформаційні технології в економіці, екології, медицині та освіті*

## РОЗДІЛ 19

### МЕТОДИ ТА МОДЕЛІ ПРОГНОЗУВАННЯ МІСЬКОГО ДОБОВОГО ВОДОСПОЖИВАННЯ

*Анотація.* Розглянуто методи та моделі розрахунку прогнозу міського водоспоживання, запропоновано алгоритм прогнозування міського добового водоспоживання, заснований на понятті однотипних днів за попередні роки. Представлений алгоритм був реалізований у вигляді програмного комплексу та пройшов багаторічну перевірку в реальних умовах, яка підтвердила його ефективність. Середня абсолютна процентна похибка добового прогнозу міського водоспоживання на один місяць не перевершує 5 %.

**Ключові слова:** водопостачання, водоспоживання, прогноз, прогнозування, часові ряди, модель, статистичний аналіз, регресія, штучна нейронна мережа.

*Abstract.* The methods and models for forecasting of urban water use have been reviewed, an algorithm for prediction of urban daily water use based on the concept of similar days in previous years has been offered. The provided algorithm was implemented in a software system and passed a long-term check under real-life conditions that confirmed its effectiveness. An average absolute percentage error of daily forecasting of urban water use for one month does not exceed 5%.

**Keywords:** water supply, water consumption, forecast, forecasting, time series, model, statistical analysis, regression, artificial neural network.

**Вступ та постановка задачі.** Прогнозування міського добового попиту на водні ресурси є основою для прийняття оперативних, тактичних і стратегічних рішень комунальними підприємствами (міськими водоканалами), що надають послуги з постачання води населенню, і має вирішальне значення з кількох причин. Наприклад, комунальні служби повинні знати потребу в чистій воді на декілька днів уперед для того, щоб забезпечити цю потребу достатньою роботою як насосних станцій, що перекачують воду, так і належною роботою очисних споруд. Крім того, оптимальне управління роботою насосних станцій, дає можливість економити витрати електроенергії в процесі перекачування води.

Таким чином, облік невизначеності в прогнозах міського водоспоживання дає можливість комунальним підприємствам оптимізувати свої операційні та

інвестиційні рішення. Тому актуальною задачею є розробка методів та моделей прогнозування міського водоспоживання, які б адекватно описували процес добових змін міського споживання води та давали змогу побудувати достовірні прогнози майбутнього добового споживання води у місті.

**Огляд іноземної літератури з короткострокового прогнозування міського водоспоживання.** На сьогоднішній день питання прогнозування міського добового водоспоживання є недостатньо розробленими та мало висвітленими в світі, а в Україні цим питанням зовсім не приділяється ніякої уваги. Тому приведемо огляд іноземної літератури з прогнозування міського водоспоживання з метою пошуку найпопулярніших сучасних методів та моделей прогнозування.

Якщо говорити про літературу з питань прогнозування часових рядів в цілому, то основною тут є відома книга з аналізу часових рядів Дж. Бокса і Г. М. Дженкінса “Аналіз часових рядів, прогноз та управління”. Особливу увагу в ній приділено нестационарним часовим рядам, що містять або стационарний приріст, або періодичні нестационарності. В цю роботу увійшли розділи, що містять основні відомості з кореляційної теорії випадкових процесів, вибір моделі, оцінювання її параметрів і перевірку моделі, а також моделі для сезонних часових рядів.

Іноземна література з прогнозування міського водоспоживання не є широко освітленою і, крім декількох комплексних посібників, представлена в основному невеликими публікаціями.

Книга Девіда Батлера та Файяза Алі Мемона (*Water Demand Management, London: IWA Publishing, 2006*) є комплексним посібником з управління водоспоживанням. Концепцією книги було зібрання повної картини специфіки управління попитом на воду, починаючи від технічних до соціальних і правових аспектів. Тенденціям споживання води і методам прогнозування попиту водоспоживання присвячена перша глава цього посібника, в якій розглянуто методи та техніки прогнозування водоспоживання, такі як статичні методи, методи на основі сценаріїв та методи з використанням штучних нейронних мереж.

Книга Р. Брюса Біллінгса та Клайва В. Джонса (*Forecasting Urban Water Demand, Denver: American Water Works Association, 2008*) надає всі необхідні інструменти для досить точного прогнозування потреб питної води для міста в короткостроковий, середньостроковий і довгостроковий період. Вона розглядає повний спектр сфер впливу на міське водоспоживання населенням, включаючи погоду, клімат, цін на воду, тарифних ставок, і коротко- і довгострокових

програм по збереженню. Книга детально описує всі методи прогнозування водоспоживання, що використовуються водоканалами Сполучених Штатів Америки:

- щорічні прогнози водоспоживання на душу населення;
- щорічні прогнози водоспоживання за основними класами користувачів;
- пікові прогнози на день;
- щомісячні прогнози водоспоживання;
- щоденні прогнози водоспоживання;
- зв'язок прогнозів доходів з прогнозами водоспоживання.

Книга Р. Брюса Біллінгса та Клайва В. Джонса являє собою посібник з вибору найбільш відповідного методу прогнозування, залежно від мети застосування прогнозу, такої як регулювання потужності системи водопостачання і запасів води, нормування, прогнозування доходів, або планування поліпшення системи розподілу. Крім того, в книзі описано, які дані потрібні для прогнозування, і показується, як використовувати такі методи, як підгонка кривої, статистичний регресійний аналіз та інші потужні методики.

За останні роки було опубліковано незначну кількість статей в спеціалізованих і економічних, фінансових та економетричних журналах світу, що стосуються проблем прогнозування водоспоживання.

Так, у публікації Е. А. Донкора, Т. А. Маззучі, Р. Соєра та Д. А. Робертсона [106] зроблено огляд літератури з прогнозування водоспоживання міст, опублікованої від 2000 до 2010 року з метою виявлення методів і моделей, корисних для вирішення проблеми водопостачання конкретного водоканалу. Результати їх дослідження показують, що, хоча широке розмаїття методів і моделей варті уваги, застосування цих моделей на практиці відрізняється, залежно від поведінки змінної (водоспоживання), її періодичності та горизонту прогнозу. В цій роботі зроблено висновок, що в той час як штучні нейронні мережі є більш зручні для короткострокового прогнозування, економетричні моделі, в поєднанні з моделюванням та прогнозуванням на основі сценаріїв, як правило, будуть більш ефективними при їх використанні для прийняття довгострокових стратегічних рішень. А якщо необхідно прийняти рішення, які включають невизначеність у майбутніх прогнозах водопостачання, то більше уваги має приділятися імовірнісним методам прогнозування.

В науковій статті [115] її автори, Юнгуо Л'ю, Хьюберт Савеніє та Цзяньсинь Сюй, пропонують для моделювання та прогнозування водоспоживання в містах відносно нову методику штучної нейронної мережі

(ANN). Результати їх дослідження показали, що модель прогнозування водоспоживання за допомогою штучної нейронної мережі надає ефективний спосіб прогнозування внутрішнього попиту на воду в місті Вейнань у Китаї. Оцінка моделі показала, що коефіцієнти кореляції більше, ніж 90 %, як для даних навчання, так і для даних тестування.

Наукова праця нідерландських вчених Д. Х. Г. Врібурга, Л. К. Рітвелда, Т. Блома, М. Баккера та М. Ван дер Рупа (The Use of An Adaptive Water Demand Prediction Model / M. Bakker, J. H. G. Vreeburg, L.C. Rietveld, T. Blom, M. Van der Roer // Journal of Water Distribution Systems Analysis. – 2012) розглядає побудову адаптивної моделі прогнозування водоспоживання, що була введена в експлуатацію у 1996 році і в даний час прогнозує водоспоживання в Нідерландах. Приведено реальні розрахунки та графіки, детально описано состав моделі прогнозування та проаналізовано її точність та результати її впровадження за 2007-2011 роки, а також виявлено, що ця модель дозволяє впоратися зі швидкими змінами в попиті, викликаними погодними умовами, повільними сезонними коливаннями і навіть соціально-економічними катаклізмами.

Наукова публікація авторів М. Хака, Х. Хаддада, А. Рахмана, М. Хоссейна, Д. Хагара та Г. Кибріа (Long Term Water Demand Forecasting: Use of Monte Carlo Cross Validation for the Best Model Selection / H. Mahmudul, K. Haddad, A. Rahman, M. Hossain, D. Hagare, Golam Kibria // International Journal of Modelling and Simulation. – 2013) з університету Західного Сіднея у Австралії розглядає використання простої, але ефективної процедури перехресної перевірки Монте Карло (MCCV), а також її порівняння з найбільш широко використовуваною звичайною перехресною перевіркою leave-one-out (LOO) для вибору найкращої моделі множинної регресії для прогнозування водоспоживання. Авторами виявлено перевагу методу Монте Карло в можливості знизити ризик перенавчання моделі, коли побудована модель добре пояснює приклади з навчальної вибірки, але відносно погано працює на прикладах, які не брали участі в навчанні. У цій статті методи MCCV і LOO застосовуються на практиці для аналізу даних водоспоживання у житловому секторі Блакитних Гір в Австралії. Результати цих досліджень показують, що метод MCCV здатен вибрати належну модель прогнозування водоспоживання, а також оцінити здатність прогнозування обраної моделі з більш високим ступенем точності.

В статті [125] її автори, М. К. Тивари і Ж. Адамовський, пропонують нову гібридну модель нейронної мережі (WBNN) для прогнозування міського



попиту на водні ресурси на короткий строк (1, 3 і 5 діб, 1 та 2 тижні, а також 1 і 2 місяця). Автори протестували свій метод з використанням даних по місту Монреаль в Канаді. Ефективність методу WBNN автори порівнювали с методом авторегресії та інтегрованої ковзної середньої (ARIMA), моделлю авторегресії та інтегрованої ковзної середньої з екзогенними вхідними змінними (ARIMAX), традиційною нейронною мережею (NN), методом NN на основі wavelet-аналізу (WNN), методом NN на основі bootstrap-техніки (BNN). Модель WBNN була розроблена як комбінація з кількох нейромережевих моделей та збудована з застосуванням бутстраповської передискретизації вейвлетовских часових підрядів замість початкових наборів даних. Результати досліджень показали, що гібридні моделі WBNN и WNN забезпечують значно більш точні результати прогнозування, ніж традиційні моделі NN, BNN, ARIMA та ARIMAX. Авторами також було виявлено, що модель WBNN зменшує невизначеність, пов'язану з прогнозами. При застосуванні моделі WBNN прогнозні довірчі смуги виявилися точнішими і надійнішими, ніж прогнозні довірчі смуги моделі BNN. Як було визначено в цьому дослідженні, використання при вейвлет-аналізі максимальної температури повітря та загальної кількості опадів поліпшує точність прогнозів попиту води.

В науковій публікації Yun Bai, Pu Wang, Chuan Li, Jingjing Xie, Yin Wang [127] пропонується підхід багатомасштабної релевантної векторної регресії (MSRVR) для прогнозування щоденного міського попиту на воду. Підхід використовує стаціонарне вейвлет-перетворення для розкладання часового ряду добових поставок води в різних масштабах. На кожному рівні вейвлет-коефіцієнти використовуються для комп'ютерного навчання моделі за методом релевантної векторної регресії (RVR). Розрахункові коефіцієнти виходів моделі RVR для всіх масштабів використовуються при відновленні результатів прогнозування за допомогою зворотного вейвлет-перетворення. Для того, щоб полегшити прогнозування при підході MSRVR, аналізуються особливості невизначеності добового часового ряду водоспоживання для визначення вхідних змінних моделі RVR. Підхід MSRVR оцінювався при використанні реальних даних, зібраних з двох гідротехнічних споруд, і порівнювався з традиційними методами. Результати досліджень авторів показали, що запропонований метод MSRVR може прогнозувати щоденний попит на воду в містах набагато точніше, ніж традиційні, в термінах нормалізованої середньоквадратичної помилки, коефіцієнта кореляції і середньої абсолютної відсоткової помилки.

На основі зробленого вище огляду літератури можна зробити висновок, що найбільш часто при прогнозуванні міського водоспоживання на короткостроковий період застосовуються моделі на базі нейронних мереж. Так як багато з дослідників підкреслюють наявність кореляції водоспоживання води з погодними умовами та іншими факторами, то саме ці моделі дозволяють впоратися зі швидкими змінами в попиті, викликаними погодними умовами, повільними сезонними коливаннями і навіть соціально-економічними катаклізмами.

В даній роботі буде розглянуто метод, що не застосовує моделі нейронних мереж, та все ж дозволяє отримувати досить точні прогнози добового міського водоспоживання.

**Основна частина.** Необхідність в прогнозуванні міського водоспоживання має місце як для довгострокових періодів (річного та місячного), так і короткострокових (добового і годинного). При цьому основна проблема, пов'язана з точністю прогнозу, виникає при прогнозуванні саме добового водоспоживання, незважаючи на існування основної сезонності в сім днів, а також сезонності в рік. Це пов'язано, перш за все, з наявністю свят, як з фіксованою датою (наприклад, 1 січня чи 8 березня), так і свят, прив'язаних до вихідних днів (наприклад, православні Пасха чи Трійця), а також з переносами робочих днів на інші дні. Тому для підвищення точності добового прогнозу будемо застосовувати ідею однотипних днів за попередні роки.

Часовий ряд добового водоспоживання відноситься до календарних часових рядах. Календарні часові ряди (не обов'язково часові ряди водоспоживання) виникають завжди, коли містять показники, пов'язані з життєдіяльністю людей. Календарні часові ряди добових показників характеризуються тим, що їх значення залежать від наступних факторів:

- ✓ до якого місяця року належать ця доба;
- ✓ який день тижня відповідає цій добі;
- ✓ чи збігаються ця доба зі святковими, передсвятковими або після святковими днями;
- ✓ чи знаходиться ця доба на початку або на прикінці місяця.

Для того, щоб розрізнити доби в календарних часових рядах, введемо поняття регулярних та нерегулярних днів. Під нерегулярними днями будемо розуміти дні, котрі є святковими, передсвятковими або після святковими. Якщо вони припадають на заздалегідь відомі фіксовані дати, то назвемо їх фіксованими нерегулярними днями. До нефіксованих нерегулярних днів будемо відносити дні, на які припадають перенесення днів тижня, або які

збігаються з релігійними святами, або в які трапляються великі аварії і т.п. Регулярними днями будемо називати всі ті дні, які не належать до нерегулярних днів.

Для аналізу календарного часового ряду необхідно мати статистику значень цього ряду за кілька минулих років. Побудова добових прогнозів, заснованих тільки на вивченні деякого невеликого відрізка часового ряду, навряд чи буде успішним, так як значення прогнозу на будь-який день (особливо нерегулярний) в більшій мірі пов'язане з датою цього дня, ніж з найближчими значеннями цього часового ряду. Тому для виділення календарних ефектів необхідно проводити ретроспективний аналіз статистики за кілька років.

### **Алгоритм розрахунку прогнозу добового водоспоживання**

Розглянемо алгоритм розрахунку прогнозу добового водоспоживання, заснований на введеному вище понятті регулярних та нерегулярних днів.

Розрахунок прогнозу добового водоспоживання проводиться відразу на весь місяць і побудований на наступному принципі: спочатку розраховується прогноз середньодобового за місяць водоспоживання на вказаний місяць, а потім вже проводиться розрахунок прогнозу по добам, тобто йде розподіл місячного прогнозу водоспоживання по дням місяця.

Проведені дослідження показали, що часовий ряд добового водоспоживання зручно замінити на часовий ряд індексів вкладу.

Індексом вкладу  $d$ -го дня, що належить  $k$ -му місяцю  $n$ -го року, будемо називати величину  $C(n, k, d)$ :

$$C(n, k, d) = Y(n, k, d) / Y_{av}(n, k), \quad (1)$$

де  $Y(n, k, d)$  – значення добового водоспоживання в  $d$ -й день  $k$ -го місяця  $n$ -го року,  $Y_{av}(n, k)$  – значення середньодобового водоспоживання за  $k$ -й місяць  $n$ -го року.

Вивчення цих індексів показало, що в різних роках можна знайти однотипні дні, у яких індекси вкладу виявляються приблизно рівні. При цьому спосіб знаходження однотипних днів різний для регулярних та нерегулярних днів. Під однотипними регулярними днями будемо розуміти дні з різних років, що належать одному і тому ж місяцю та припадають на один і той же день тижня, при цьому різниця порядкових номерів цих днів у місяці повинна бути мінімальною.

Тепер опишемо метод отримання прогнозу добового водоспоживання. Нехай потрібно зробити прогноз  $Y_{\text{pred}}(n, k, d)$  на  $d$ -й день  $k$ -го місяця  $n$ -го року. Цей прогноз будемо розраховувати по формулі (2), яка випливає з формули (1):

$$Y_{\text{pred}}(n, k, d) = Y_{\text{pred}}(n, k) * C_{\text{pred}}(n, k, d), \quad (2)$$

де  $Y_{\text{pred}}(n, k)$  – прогноз середньодобового за місяць водоспоживання для  $k$ -го місяця  $n$ -го року,  $C_{\text{pred}}(n, k, d)$  – прогноз індекса вкладу  $C(n, k, d)$ .

Місячний прогноз  $Y_{\text{pred}}(n, k)$  може бути розрахований, наприклад по моделі Бокса-Дженкінса ARIMA (1,1,1) (0,1,1) с сезонним лагом 12. Структура моделі Бокса-Дженкінса для прогноза місячного водоспоживання вибиралася таким чином, щоб задовольнялася наступна умова: середня абсолютна процентна помилка (за один рік) прогнозу з попередженням в один місяць не перевищує 1.5%. Досить докладний опис аналізу даної моделі наведено в роботі [112].

Тому залишається провести опис методу знаходження прогнозу індекса вкладу  $C_{\text{pred}}(n, k, d)$ .

Для регулярних днів величина  $C_{\text{pred}}(n, k, d)$  знаходиться ковзним середнім за індексами вкладу однотипних днів, взятих з заданої кількості  $m$  років, попередніх  $n$ -му року. Для цього спочатку необхідно визначити номери однотипних днів.

В  $(n-1)$ -му році день, однотипний  $d$ -му дню  $k$ -го місяця  $n$ -го року, також буде належати  $k$ -му місяцю. Позначимо номер цього дня в місяці через  $d_s(1)$ . Тоді:

$$d_s(1) = \begin{cases} (d + 2), & \text{якщо } (\text{mod}(n, 4) = 0 \text{ и } k \geq 3) \text{ або } (\text{mod}(n-1, 4) = 0 \text{ и } k \leq 2); \\ (d + 1), & \text{в інших випадках.} \end{cases}$$

Позначимо через  $N_{\text{dm}}(k, n)$  – кількість днів в  $k$ -му місяці  $n$ -го року. Зауважимо, що в рідких випадках може виявитися, що  $d_s(1) > N_{\text{dm}}(k, n)$ , але оскільки для прогнозу важливий саме значення індекса вкладу, а не його номер, то уточнювати визначення  $d_s(1)$  коректніше не має великого сенсу.

Позначимо різницю між  $d_s(1)$  і  $d$  через  $h_s(1)$ . Далі рекурентним чином визначимо  $d_s(i)$  – номер однотипного дня в  $(n-i)$ -му році для  $i \geq 2$ .

Нехай вже знайдено номер  $d_s(i-1)$ . Введемо:  $h_s(i-1) = d_s(i-1) - d$ . Тоді  $h_s(i)$  визначається наступним правилом:

$$h_s(i) = h_s(i-1) + \Delta h,$$

де

$$\Delta h = \begin{cases} 2, & \text{якщо } (Ndy(n-i) = 366 \text{ и } k \geq 3) \text{ або } (Ndy(n-i) = 366 \text{ и } k \leq 2); \\ 1, & \text{в інших випадках;} \end{cases}$$

$Ndy(n-i)$  - кількість днів в  $(n-i)$ -му році.

При цьому, якщо  $h_s(i) > 3$ , то тоді  $h_s(i) = h_s(i) - 7$  (тут «7» – це кількість днів в тижні). Остаточо вважаємо:  $d_s(i) = d + h_s(i)$ .

Запропонований спосіб визначення однотипних днів забезпечує максимальну близькість однотипного дня  $d_s(i)$  до початкового дня  $d$ .

Тепер визначемо індекс вкладу  $C_s(i, k, d)$  однотипного дня  $d_s(i)$  для дня  $d$ .

Він визначається за наступним правилом:

$$C_s(i, k, d) = \begin{cases} C(n-1, k, d_s(i)), & \text{якщо } 1 \leq d_s(i) \leq Ndm(n-i, k); \\ \frac{C(n-1, k, Ndm(n-i, k)) \cdot Y(n-1, k, d_s(i) - 7)}{Y(n-i, k, Ndm(n-i, k) - 7)}, & \text{якщо } d_s(i) > Ndm(n-i, k); \\ \frac{C(n-1, k, 1) \cdot Y(n-1, k, d_s(i) + 7)}{Y(n-i, k, 8)}, & \text{якщо } d_s(i) < 1, \quad k \neq 3, 10, 11; \\ C_s(i-1, k, d), & \text{якщо } d_s(i) < 1, \quad k = 3, 10, 11. \end{cases}$$

Таким чином, для фіксованих нерегулярних днів прогноз індекса вкладу дорівнює індексу вкладу  $d$ -го дня  $k$ -го місяця найближчого року, який містить однотипний нерегулярний день. Наприклад, прогноз на 1 січня 2008 року визначається значенням індекса вкладу на 1 січня 2002 року, так як обидва ці дні потрапляють на вівторок.

Остаточо для більшої точності визначаємо для регулярних днів:

$$C_{\text{pred}}(n, k, d) = \frac{1 + \sum_{i=1}^m C_s(i, k, d)}{m+1}, \quad (3)$$

тобто для регулярних днів величина  $C_{\text{pred}}(n, k, d)$  знаходиться ковзним середнім за індексами вкладу однотипних днів, взятих з заданої кількості  $m$  років, попередніх  $n$ -му року.

Для фіксованих нерегулярних днів виявився ефективним наступний алгоритм знаходження прогнозу  $C_{\text{pred}}(n, k, d)$ :

$$C_{\text{pred}}(n, k, d) = C(n(s), k, d), \quad (4)$$

де  $n(s)$  – номер року, що містить однотипний нерегулярний  $d$ -й день  $k$ -го місяця для  $n$ -го року, тобто прогноз індекса вкладу збігається з індексом однотипного нерегулярного дня.

В загальному випадку величина  $n(s)$  визначається з табл. 19.1.

Таблиця 19.1

**Таблиця визначення номеру року, що містить однотипний  
нерегулярний день**

	$\text{mod}(n,4)=0$	$\text{mod}(n,4)=1$	$\text{mod}(n,4)=2$	$\text{mod}(n,4)=3$
$k \leq 2$	$n(s)=n-6$	$n(s)=n-5$	$n(s)=n-11$	$n(s)=n-6$
$k \geq 3$	$n(s)=n-5$	$n(s)=n-11$	$n(s)=n-6$	$n(s)=n-6$

Наприклад, при  $n = 2008$ ,  $k = 1$ ,  $d = 1$ ,  $n(s) = 2002$ , оскільки 1 січня 2008 року і 1 січня 2002 року потрапляють на вівторок.

Для випадку, коли необхідно отримати прогноз на нерегулярні нефіксовані дні, то спочатку обчислюється прогноз на ці дні в припущенні, що ці дні є регулярними, а потім проводиться їх коригування за допомогою поправочних коефіцієнтів, які обчислюються з аналізу аналогічних нерегулярних днів в минулих роках.

Так як добовий прогноз розраховується на основі зпрогнозованого середньодобового за місяць значення, то помилки добового прогнозу, розрахованого на місяць вперед, зростають на помилку прогнозу середньодобового за місяць значення. Для зменшення помилок добового прогнозу виявилось доцільним застосовувати коригування добових прогнозів по отримуваним фактичним значенням добового водоспоживання протягом місяця. Це коригування може здійснюється одним із двох способів: коригуванням середньодобового за місяць значення і коригуванням найближчих прогнозів добових значень за допомогою методу експоненціального згладжування.

Наведений вище метод розрахунку прогнозу добового водоспоживання був програмно реалізований та включений в програмний комплекс "Прогноз", що впроваджено в водоканалах м. С-Петербург і м. Москва (Росія).

Наведемо короткий опис програмного комплексу "Прогноз", призначеного для прогнозування міського водоспоживання, як довгострокового (річного, місячного), так і короткострокового (добового, годинного).

Розрахунок поточного прогнозу міського водоспоживання проводиться послідовно, починаючи з річного і далі по низхідній. При цьому річний прогноз є основою для місячного, місячний - для добового, а добовий - для годинного.

У добовому прогнозі введені режими ретроспективного прогнозу, прогнозу на поточний і прогнозу на наступний місяць. Під ретроспективним прогнозом

розуміється прогноз на період часу, для якого вже є фактичні данні, але вони не застосовуються для побудови прогнозової моделі. Особливість добового прогнозу в тому, що він розраховується відразу на всі дні заданого місяця. Тому протягом місяця повинно запускатися коригування раніше розрахованого прогнозу середньодобового за місяць водоспоживання по фактичним даним, що були знову отримані. Для підвищення точності добового прогнозу потрібно враховувати вплив нерегулярних днів, тобто днів за якими споживання досить сильно відрізняється від сусідніх днів (як правило, це святкові дні, переноси вихідних і т.п.). Список нерегулярних днів повинен бути підготовлений заздалегідь.

Для ілюстрації ефективності запропонованого методу прогнозування на рис. 19.1 і рис. 19.2 наведені графіки ретроспективного прогнозу добового водоспоживання в м. Москва на грудень 2008 року і січень 2009 року. На рис. 19.1 відображено випадок, коли дні 31 грудня і 1 січня розглядаються як регулярні дні, а на рис. 19.2 відображено випадок, коли ці дні вже відзначені як нерегулярні.

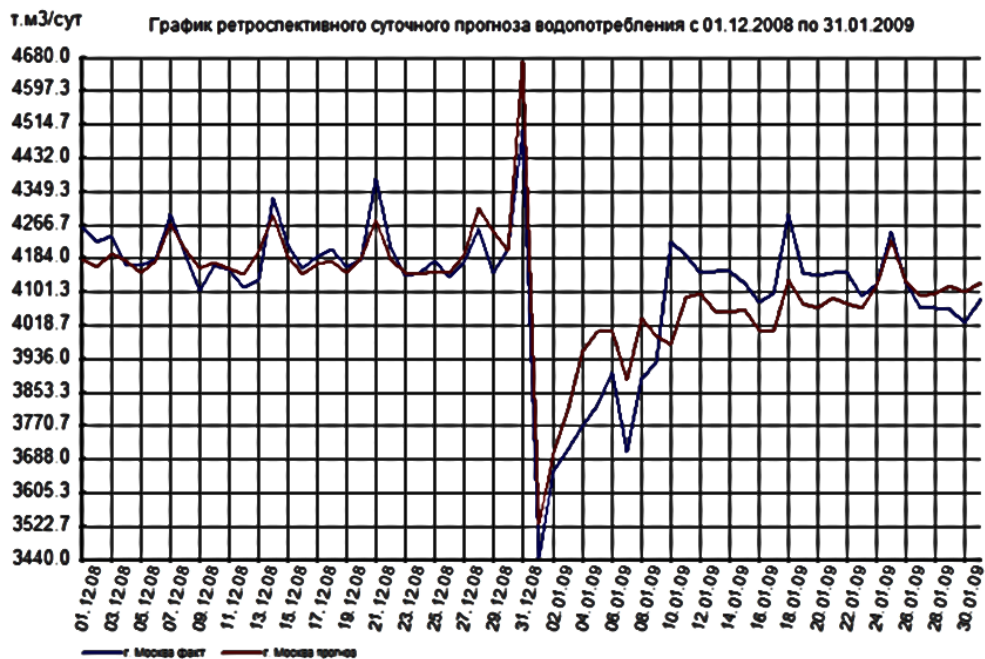


**Рис. 19.1. Графік ретроспективного прогнозу добового водоспоживання для випадку, коли дні 31 грудня і 1 січня розглядаються як регулярні**

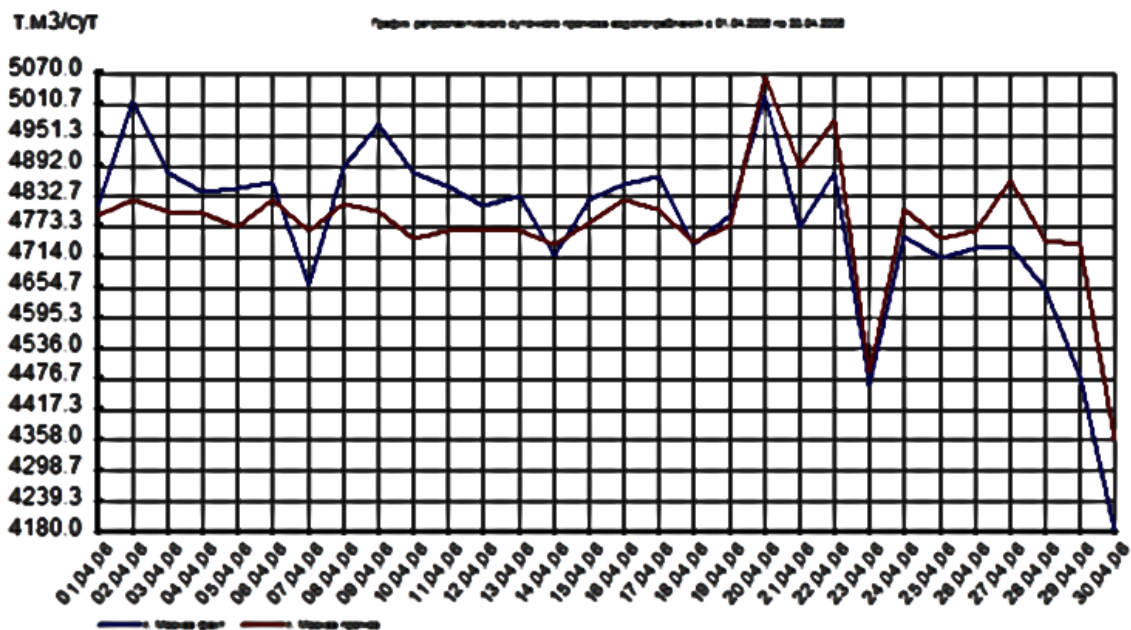
Як видно з рис. 19.1 та рис. 19.2 прогноз на дні 31 грудня і 1 січня у випадку, коли ці розглядаються як нерегулярні дні, значно кращий.

На рис. 19.3 наведено графік ретроспективного прогнозу добового водоспоживання в м. Москва на квітень 2006 року. В цьому році православна Пасха припадала на 23 квітня, тому ця дата при прогнозуванні була визначена як нефіксований нерегулярний день. Також при розрахунку добового прогнозу

на місяць квітень нерегулярним, але фіксованим днем розглядалась дата 30 квітня, як передсвятковий день.



*Рис. 19.2. Графік ретроспективного прогнозу добового водоспоживання для випадку, коли дні 31 грудня і 1 січня розглядаються як нерегулярні*

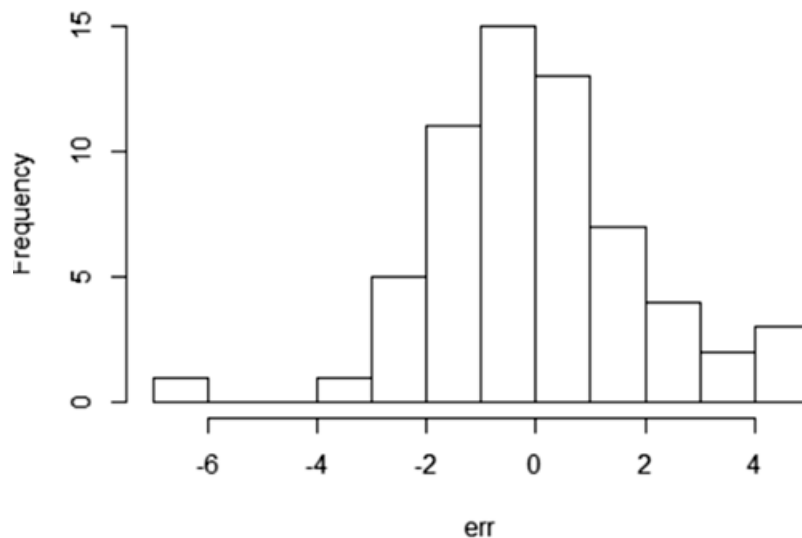


*Рис. 19.3. Графік ретроспективного прогнозу добового водоспоживання в м. Москва на квітень 2006 року*

В цілому, механізм використання поняття нерегулярних днів при добовому прогнозі дозволяє отримувати прогноз з відносною похибкою, що не



перевищує 5 %. Так на рис. 19.4 приведена гістограма похибок прогнозу добового водоспоживання для випадку, коли дні 31 грудня і 1 січня розглядаються як нерегулярні (див. рис. 19.2).



**Рис. 19.4. Гістограма похибок прогнозу добового водоспоживання для випадку, коли дні 31 грудня і 1 січня розглядаються як нерегулярні**

Треба зауважити, що різка зміна температури повітря також впливає на точність добового прогнозу водоспоживання. Це відзначають і інші дослідники [106; 115]. З пониженням температури повітря водоспоживання зростає, а з підвищенням – навпаки знижується. Так на рис. 19.4 видно, що фактичне водоспоживання з 16 по 29 січня 2006 року виявилось більшим ніж прогнозне, і пояснюється це саме різким зниженням температури повітря з 16 по 23 січня 2006 року до  $-27$  градусів за Цельсієм. Аналогічна ситуація доводиться і на період з 4 по 10 лютого 2006 року. При цьому, невеликі коливання температури повітря істотного впливу на точність прогнозування не роблять (див. рис. 19.5 та рис. 19.6). Тому у випадку, коли прогнозуються різкі зміни температури повітря, треба проводити корекцію прогнозу добового водоспоживання. Для цього в програмному комплексі “Прогноз” передбачена можливість ручного коригування розрахованого прогнозу. Для автоматичного коригування прогнозу водоспоживання треба завжди мати надійний прогноз погоди, що досить складно реалізувати як технічно, так і організаційно. Як відомо, зараз надійний прогноз погоди є лише десь на 3 дні.

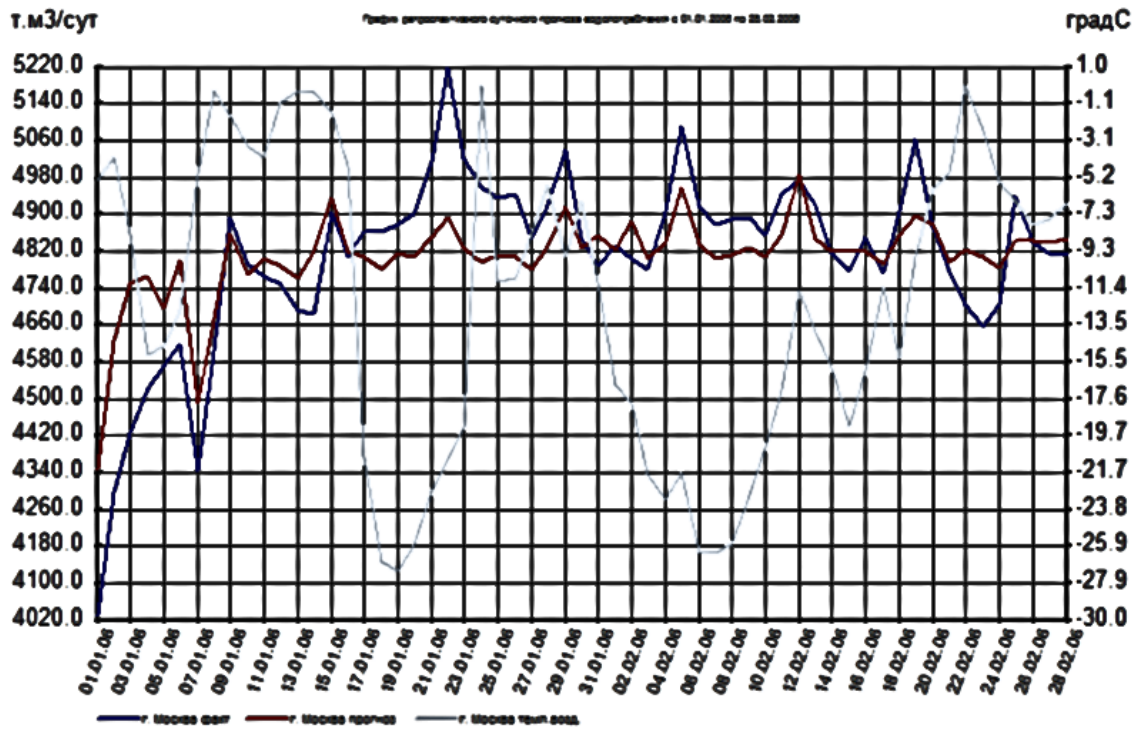


Рис. 19.5. Порівняльний графік фактичного і прогнозного добового водоспоживання та добової температури повітря з 1 січня по 28 лютого 2006 року

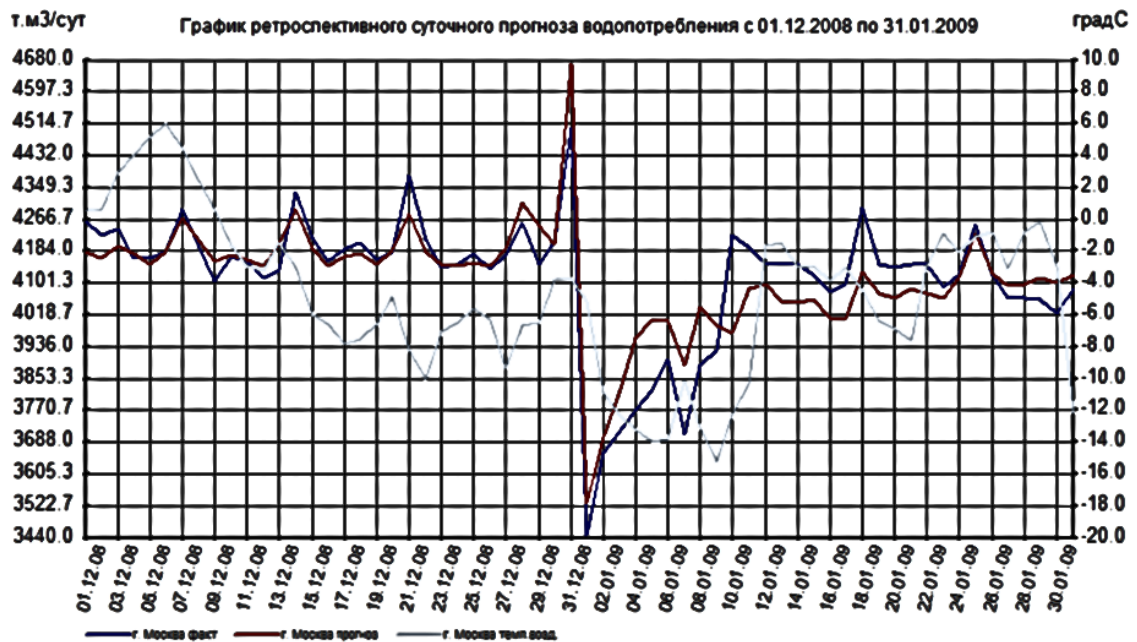


Рис. 19.6. Порівняльний графік фактичного і прогнозного добового водоспоживання та добової температури повітря з 1 грудня 2008 року по 31 січня 2009 року

**Висновки.** Щоденне прогнозування попиту на воду є ефективним заходом для планування роботи міських об'єктів водопостачання. Облік невизначеності в прогнозах міського водоспоживання дає можливість комунальним підприємствам оптимізувати свої операційні та інвестиційні

рішення. Тому актуальною задачею є розробка методів та моделей прогнозування міського водоспоживання, які б адекватно описували процес змін міського споживання води та давали змогу побудувати достовірні прогнози майбутнього споживання води у місті.

Представлено огляд літератури з прогнозування попиту на водні ресурси, яка була опублікована за останній час. На основі зробленого огляду можна зробити висновок, що найбільш часто при прогнозуванні міського водоспоживання на короткостроковий період застосовуються моделі на базі нейронних мереж та комбіновані (гібридні) моделі. Нейромережеві та гібридні моделі, на відміну від класичних, можуть модулювати нелінійні зв'язки, є масштабованими, високо адаптивними та мають множина прикладів застосування. Багато з дослідників підкреслюють наявність кореляції водоспоживання води з погодними умовами та іншими факторами, то саме ці моделі дозволяють впоратися зі швидкими змінами в попиті, викликаними погодними умовами, повільними сезонними коливаннями і навіть соціально-економічними катаклізмами.

Запропоновано метод, що не застосовує моделі нейронних мереж, та все ж дозволяє отримувати досить точні прогнози добового міського водоспоживання. Метод засновано на понятті однотипних днів за попередні роки. Алгоритм методу програмно реалізований та включений в програмний комплекс “Прогноз”, що впроваджено в водоканалах м. С-Петербург і м. Москва (Росія).

Запропонований метод пройшов багаторічну перевірку в реальних умовах застосування, яка підтвердила його ефективність. Середня абсолютна процентна похибка добового прогнозу міського водоспоживання на один місяць не перевершує 5 %.

## РОЗДІЛ 20

# МОДЕЛЮВАННЯ МАКСИМІЗАЦІЇ ПРИБУТКУ НА ОСНОВІ ВИРОБНИЧОЇ ФУНКЦІЇ КОББА-ДУГЛАСА

***Анотація.** Робота присвячена дослідженню екстремуму функції для оптимізації роботи виробничого підприємства за допомогою виробничої функції Кобба-Дугласа. Отримана залежність максимального значення функції прибутку двохресурсного підприємства від показників цієї функції. Аналіз такої залежності дає можливість оцінити значення цих параметрів, при яких функція прибутку досягне максимуму. В роботі проведено моделювання показників двохфакторної моделі виробництва з метою отримання максимізації прибутку, на основі якого була проведена оцінка числових значень всіх параметрів, що входять в функцію прибутку двохресурсного підприємства. Були проведені дослідження чутливості функції прибутку на зміну параметрів, що розглядаються.*

***Ключові слова:** виробнича функція Кобба-Дугласа, двохфакторна модель виробництва, економетрична модель, максимізація прибутку, чутливість.*

***Abstract.** The article is devoted to the research of the extrimum function to optimize the work of an industrial enterprise using the Cobb-Douglas' production function. The dependence of the maximum value of two-resource industrial enterprise profit function on the indexes of the function was obtained. The analysis of such dependence makes it possible to estimate the values of these parameters when the profit function attains a maximum. The simulation of the indexes of the two-factor enterprise pattern to obtain the maximum profit was conducted. Numerical values of the parameters of the two-resource enterprise profit function was estimated. The study of quick response of the profit function on changing the parameters analyzed was performed.*

***Keywords:** Cobb-Douglas' production function, two-factor production models, econometric model, profit maximization, quick response.*

**Постановка проблеми у загальному вигляді.** Одним з напрямків максимізації прибутку виробництва є застосування виробничих функцій. Виробнича функція – це економетрична модель, що відображає залежність показників виробничогосподарської діяльності від факторів, що визначають ці показники. До основних показників можна віднести: обсяг продукції, що випускається, прибуток, рентабельність, собівартість, фондівдача й інші. Факторами для цих показників можуть бути: робоча сила, основні засоби або капітал, земля, продуктивність праці та інші. Виробнича функція дає можливість дослідити для галузей та економіки в цілому показники середньої і граничної ефективності ресурсів робочої сили і основних виробничих фондів,

граничні норми заміщення ресурсів у виробничому процесі. Велику цінність являють собою методи оцінки, що ґрунтуються на теорії виробничої функції і які дозволяють розглядати також проблеми конкурентноспроможності в часовому аспекті, враховувати і структурувати фактори конкурентноспроможності.

До найвідоміших виробничих функцій відноситься функція Кобба-Дугласа, яка визначає залежність між обсягом виробництва, обсягами капіталу та витратами ресурсів праці. Дослідження показали, що багато явищ виробництва добре апроксимуються залежностями такого типу. За допомогою функції Кобба-Дугласа можна прогнозувати обсяги виробництва, оцінювати ефективність виробництва та використання окремих виробничих факторів, визначати можливість взаємозаміни факторів виробництва, оцінювати масштаб виробництва та його вплив на ефективність виробництва, виявляти вплив НТП на процеси виробництва.

Таким чином, застосування функції Кобба-Дугласа щодо планування виробництва дозволяє знаходити обґрунтовані управлінські рішення. Протягом багатьох років економетричні дослідження були спрямовані в основному не на заміну її іншими видами функцій, а на модифікацію виробничої функції, яка могла б адекватніше описати реальні економічні співвідношення.

**Аналіз останніх досліджень і публікацій.** Науковими дослідженнями в сфері застосування математичних моделей при аналізі діяльності складних соціально-економічних систем присвячені роботи вітчизняних та зарубіжних науковців Р. Акофф, К. Бравий, В. Вітлінського, В. Геєця, М. Іванова, А. Камінського, Ю. Лисенка, В. Порохні, О. Разумовського, Л. Сергєєвої, В. Тимохіна, М. Хазова та інших.

На сьогоднішній день існує великий перелік робіт, що присвячені теорії і практиці застосування функції Кобба-Дугласа [13; 41; 63; 89; 92; 26]. Проте вони належним чином не торкаються питання застосування методу дослідження екстремуму функції для оптимізації роботи виробничого підприємства за допомогою виробничої функції Кобба-Дугласа. Вирішенню поставленої задачі і присвячується означена робота.

**Мета роботи:** дослідження виробничої функції Кобба–Дугласа і моделювання показників двохфакторної моделі виробництва з метою отримання максимізації прибутку.

**Виклад основного матеріалу дослідження.** При дослідженні ефективності роботи виробничого підприємства використовують поняття субституційної виробничої функції [13]. Вона відображає функціональний

зв'язок між об'ємом ефективно використаних факторів виробництва (працею та майновим капіталом) і з їх допомогою досягнутим випуском продукції при існуючих технічних і організаційних знаннях. Така функція відображає кількість випущеної продукції і показує, що виробництво може бути збільшене за рахунок підвищення кількісної характеристики одного з факторів, в той час як кількісна характеристика іншого фактору виробництва залишається без змін. В іншому варіанті виробництво залишається без змін при різноманітних кількісних комбінаціях факторів праці та майнового капіталу.

Субституційна виробнича функція має в загальному випадку наступний вигляд:

$$y = y(K, L); y'_x > 0; y''_{xx} < 0 \quad (1)$$

де:

K – кількість виробничого капіталу,

L – кількість виробничих трудових годин або, іншими словами, число виробничих одиниць гуманного капіталу.

Найбільш простою є двохфакторна модель виробничої функції Кобба–Дугласа, за допомогою якої розкривається взаємозв'язок праці (L) и капіталу (K). Ці фактори взаємозамінні и взаємодоповнювальні. Ще в 1928 році американські вчені – економіст П. Дуглас и математик Ч. Кобб – створили макроекономічну модель, яка дозволяє оцінювати вклад різних факторів виробництва в збільшення об'єму виробництва або національного доходу. Ця функція має наступний вигляд:

$$y = AK^\alpha L^\beta \quad (2)$$

В таких випадках в якості виробничої функції зручно використовувати функцію Коба-Дугласа [13]

$$f(x, y) = Ax^\alpha y^\beta \quad (3)$$

Тут значення функції  $f(x, y)$  відображає кількість випущеної продукції, величини  $x$  і  $y$  є кількісними показниками використаних при виробництві ресурсів обох видів, наприклад, по витратам капіталу та праці, що відображають вклад праці і капіталу у виготовлення продукції,  $\alpha$  і  $\beta$  – безрозмірні показники еластичності випуску продукції по затратам кожного з видів ресурсів відповідно,  $A$  – деяка постійна, що переводить одиниці виміру праці і капіталу в одиниці виміру продукту.

Вибір параметрів  $\alpha$  і  $\beta$  здійснюється методами регресивного аналізу. Між цими параметрами є певний зв'язок. Їх сума порівнюється з одиницею. Важлива властивість коефіцієнтів еластичності ресурсів функції Кобба-

Дугласа: сума коефіцієнтів еластичності дорівнює показнику ефекту розширення масштабу. Якщо  $\alpha$  і  $\beta$  в сумі перевищують одиницю, то виробнича функція має зростаючий ефект від масштабу виробництва (це означає, що якщо величини  $x$  і  $y$  збільшуються в деякій пропорції, то величина  $f(x,y)$  зростає в більшій пропорції). Якщо ж їх сума дорівнює одиниці, то говорять про постійний ефект від масштабу виробництва (величина  $f(x,y)$  збільшується в тій же пропорції, що і величини  $x$  і  $y$ ), або рівень ефективності ресурсів не залежить від масштабів виробництва. У випадку, коли сума параметрів  $\alpha$  і  $\beta$  менша, ніж одиниця, то має місце зменшувальний ефект від масштабу виробництва (величина  $f(x,y)$  збільшується в меншій пропорції, ніж величини  $x$  і  $y$ ), або з розширенням масштабів виробництва середні витрати ресурсів в розрахунку на одиницю продукції зменшуються. Зазначимо, що ці властивості не залежать від числових значень  $K$  і  $L$  і зберігаються в кожній точці виробничої функції, тобто коефіцієнти  $\alpha$  і  $\beta$  постійні, які не залежать від об'єму факторів  $K$  і  $L$ .

Коефіцієнти еластичності мають велике значення для аналізу ВФ і характеризують відсоток приросту об'єму випуску продукції при збільшенні витрат ресурсу на 1%. Таким чином, збільшення витрат капіталу на 1 % приводить до зростання випуску продукції на  $\alpha$  відсотків, а зростання витрат праці на 1 % приводить до зростання випуску продукції на  $\beta$  відсотків. Якщо  $\alpha = 0,25$ , то зростання витрат капіталу на 1 % збільшує об'єм виробництва на 0,25 %. Можна допустити, що обидві величини  $\alpha$  і  $\beta$  знаходяться між нулем та одиницею. Вони повинні бути додатними, оскільки збільшення витрат виробничих факторів повинно викликати зростання випуску продукції. В той же час, можливо, вони будуть менше одиниці, оскільки доцільне припущення, що зменшення ефекту від масштабу виробництва приводить до більш повільного зростання випуску продукції, ніж витрати виробничих факторів, якщо інші фактори залишаються постійними.

Із збільшенням кількісного параметру майнового капіталу збільшується і середня продуктивність праці, яка є часткою від ділення величини випуску на величину витраченої праці. Але при цьому зменшується коефіцієнт праці, який визначає середню кількість витраченої праці на кожен одиницю випуску і є, таким чином, оберненою величиною середній продуктивності праці.

Величина майнового капіталу приймається в рамках даного короткочасного аналізу як екзогено задана, тому в моделі і описі не враховується технічний прогрес, а також ефект виробничих потужностей за рахунок інвестицій.

Особливість реальних виробничих процесів полягає в можливості заміщення одного фактора іншим; наприклад, існує абстрактна можливість замінити одиницю виробничого обладнання еквівалентним за об'ємом фондівітдачі кількістю одиниць праці. Але реальне втілення цієї абстрактної можливості – неможливе.

Для випадку двофакторної ВФ числова характеристика ефекту заміни повинна показувати, на яку величину  $dy$  зменшиться об'єм витрат другого ресурсу, якщо збільшити об'єм витрат першого ресурсу на  $dx$ , щоб при цьому об'єм випуску продукції  $Q$  залишався незмінним.

Еластичність випуску продукції по капіталу і праці дорівнює відповідно  $\alpha$  та  $\beta$ , оскільки має місце співвідношення

$$\frac{\partial y / \partial K}{y / K} = \frac{A(\alpha [K^{\alpha-1}] L^{\beta})}{AK^{\alpha-1} L^{\beta}} = \alpha. \quad (4)$$

Аналогічним чином легко показати, що  $(dy/dL)/(y/L)$  дорівнює  $\beta$ .

Фактори виробництва  $\alpha$  і  $\beta$  мають інтерпретацію як прогнозовані долі прибутку, який отримується відповідно за рахунок капіталу і праці. Якщо ринок праці має конкурентний характер, то ставка заробітної плати ( $w$ ) дорівнюватиме граничному продукту праці ( $dy/dL$ ):

$$w = \frac{\partial y}{\partial L} = AK^{\alpha} \beta L^{\beta-1} = \frac{\beta y}{L}. \quad (5)$$

Таким чином, загальна сума заробітної плати ( $wL$ ) дорівнюватиме  $\beta y$ , а доля праці в загальному випуску продукції ( $wL/Y$ ) складе постійну величину  $\beta$ . Аналогічним чином норма прибутку виражається через  $dy/dK$ :

$$\rho = \frac{\partial y}{\partial K} = A\alpha K^{\alpha-1} L^{\beta} = \frac{\alpha y}{K}. \quad (6)$$

і, таким чином, загальний прибуток ( $\rho K$ ) дорівнюватиме  $\alpha y$ , а доля прибутку буде постійною величиною  $\alpha$ . Це означає, що граничний приріст продукції за рахунок приросту кожного ресурсу визначається як добуток коефіцієнта еластичності на середню ефективність ресурсу. Параметр  $A$  у функції Кобба – Дугласа залежить від вибраних одиниць вимірювання  $Y$ ,  $K$ ,  $L$ ; водночас числове значення цього параметра визначається також ефективністю виробничого процесу. У цьому можна переконатись, порівнявши дві виробничі функції, які відрізняються одна від одної лише значенням параметра  $A$ .

Для фіксованих значень  $K$  і  $L$  тієї функції, де більше числове значення параметра  $a$ , відповідає більше значення  $Y$ . Отже, і виробничий процес, який



описується цією функцією, буде ефективнішим. Другі похідні функції Кобба – Дугласа мають такий вигляд:

$$Y_{KK}^* = \frac{\alpha(\alpha-1)Y}{K^2}; \quad Y_{LL}^* = \frac{\beta(\beta-1)Y}{L^2}. \quad (7)$$

Враховуючи, що  $0 < \alpha < 1$  і  $0 < \beta < 1$ ,  $Y_{KK} < 0$  і  $Y_{LL} < 0$  справджується висновок: при збільшенні ресурсів граничний приріст обсягу продукції зменшуватиметься. Якщо обсяг продукції у функції Кобба – Дугласа вважати сталим (const), то можна обчислити граничні норми заміщення ресурсів:

$$h = \frac{Y'_K}{Y'_L} = \frac{\alpha L}{\beta K}. \quad (8)$$

Звідси бачимо, що гранична норма заміщення ресурсів у функції Кобба – Дугласа визначається як добуток співвідношень обсягів ресурсів та їх коефіцієнтів еластичності.

Швидкість зміни норми заміщення ресурсів у зв'язку зі зміною обсягу ресурсів обчислюється так:

$$\frac{\partial h}{\partial L} = \frac{\alpha}{\beta K}; \quad \frac{\partial h}{\partial K} = -\frac{\alpha L}{\beta K^2}. \quad (9)$$

Мірою швидкості зміни  $h$  є еластичність заміщення ресурсів  $K$  і  $L$ , що визначається як відношення зміни обсягу ресурсів до зміни величини  $h$ :

$$h_{\frac{K}{L}} = \frac{h(Lh + K)}{KL \left( h \frac{\partial h}{\partial K} - \frac{\partial h}{\partial L} \right)} = 1. \quad (10)$$

Отже, еластичність заміщення в кожній точці кривої, що характеризує виробничу функцію Кобба – Дугласа, дорівнює одиниці.

Розглянемо тепер поведінку функції під час зміни масштабу виробництва. Для цього припустимо, що витрати кожного ресурсу виробництва збільшилися в  $\lambda$  разів, тоді нове значення  $Y'$  визначатиметься так:

$$Y' = a(\lambda K)^\alpha (\lambda L)^\beta = \lambda^{\alpha+\beta} Y. \quad (11)$$

За припущення, що мета господарської діяльності – максимізація прибутку, можна проілюструвати інші властивості виробничої функції. Запишемо функцію прибутку в такому вигляді:

$$\Pi = bY^{r+1} - wL - rK + \lambda[f(K,L) - Y]. \quad (12)$$

Підприємець вибирає такі значення  $Y$ ,  $L$ ,  $K$  які максимізують прибуток при обмеженнях, що накладаються виробничою функцією. Величини  $b$ ,  $w$ ,  $r$  – параметри функції прибутку,  $\lambda$  – множник Лагранжа. Якщо виробничий процес

у даному співвідношенні подається функцією Кобба – Дугласа, то можна записати умови максимізації прибутку:

$$w = \frac{\lambda\beta Y}{L}; r = \frac{\lambda\alpha Y}{K}; \frac{w}{r} = \frac{\beta K}{\alpha L}, \quad (13)$$

$\lambda = (r+1)P$  при  $r \neq -1$ , де  $P = bY'$ .

Звідси обсяги ресурсів такі:

$$L = \frac{(r+1)P\beta Y}{w}; K = \frac{(r+1)P\alpha Y}{r} \quad (14)$$

У цьому випадку максимальне значення випуску продукції, якщо  $\alpha + \beta \neq 1$ , можна записати так:

$$Y = aK^\alpha L^\beta = a \left[ \frac{(r+1)P\alpha Y}{r} \right]^\alpha \left[ \frac{(r+1)P\beta Y}{w} \right]^\beta. \quad (15)$$

При  $r = 1$  згідно із записаними щойно умовами максимізації (15) дістанемо:

$$K = \frac{w\alpha L}{\beta r}; L = a \left( \frac{w\alpha}{\beta r} \right)^\alpha L^{\alpha+\beta}. \quad (16)$$

Отже, необхідні умови для забезпечення максимізації прибутку дають змогу визначити відповідні витрати робочої сили і основного капіталу. З розширенням масштабів виробництва ефективність витрат ресурсів спадає, що відповідає максимізації прибутку в умовах досконалої конкуренції.

Наведений приклад виробничої функції показує, що ця економетрична модель дає змогу досить широко проаналізувати виробничу діяльність, визначити шляхи її вдосконалення для підвищення ефективності. Обґрунтованість такого аналізу повністю залежить від достовірності економетричної моделі, від того, наскільки вона адекватна реальному процесові.

Виробнича функція дає можливість дослідити для галузей та економіки в цілому показники середньої і граничної ефективності ресурсів робочої сили і основних виробничих фондів, граничні норми заміщення ресурсів у виробничому процесі. Важливе значення має також аналіз на основі функцій (2)

таких відносних показників, як продуктивність праці  $\left(\frac{Y}{L}\right)$ , фондоозброєність

праці  $\left(\frac{K}{L}\right)$ , фондоозброєність продукції  $\left(\frac{K}{Y}\right)$ , фондovіддача  $\left(\frac{Y}{K}\right)$ . Наведені

висновки базуються на виробничій функції, коли сума еластичностей ресурсів

дорівнює одиниці, тобто дохід зростає в такому самому співвідношенні, як і обидва ресурси.

При побудові виробничої функції Кобба – Дугласа параметри  $A$ ,  $\alpha$ ,  $\beta$  можна оцінити за допомогою лінійного регресивного аналізу методом найменших квадратів (МНК):

1) Виробничу функцію Кобба – Дугласа приводять до лінійного виду шляхом логарифмування

$$\ln(y) = \ln(A) + \alpha \ln(K) + \beta \ln(L). \quad (17)$$

2) При використанні МНК мета полягає в мінімізації суми квадратичних відхилень (SSD) між величинами, що спостерігаються  $\ln(y_i)$ , ( $i=1..N$ ;  $N$  – кількість спостережень) і відповідними оцінками  $\ln(\hat{A})$ ,  $\hat{\alpha}$ ,  $\hat{\beta}$ .

$$SSD = \sum_{i=1}^N (\ln(y_i) - \ln(\hat{y}_i))^2 = \sum_{i=1}^N (\ln(y_i) - \ln(\hat{A}) - \hat{\alpha} \ln(K_i) - \hat{\beta} \ln(L_i))^2. \quad (18)$$

Застосування методу дослідження екстремуму функції дозволяє оптимізувати роботу виробничого підприємства, що виготовляє певний вид продукції. В цьому випадку суть оптимізації полягає в тому, що необхідно підібрати витрату такої кількості ресурсів при фіксованих на них цінах, щоб прибуток від реалізації виробленої продукції був максимальним [26].

Отримаємо вираз для максимального значення функції прибутку двохресурсного підприємства. Він дозволить оцінити значення параметрів  $\alpha$ ,  $\beta$ ,  $p_0$ ,  $p_1$ ,  $p_2$ ,  $K$ ,  $L$  при яких значення функції прибутку буде максимальним.

Розглянемо випадок двохресурсного виробництва. Кількість виробленої продукції задамо функцією Коба – Дугласа у вигляді (будемо для зручності вважати, що  $A=1$ )

$$f(x, y) = x^\alpha y^\beta \quad (19)$$

Нехай  $p_0$  – вартість одиниці виробленої продукції, а  $p_1$  і  $p_2$  – вартості одиниці використаних ресурсів першого і другого виду відповідно. Тоді чистий прибуток підприємства визначається функцією прибутку, яка має наступний вигляд:

$$P(x, y) = p_0 x^\alpha y^\beta - p_1 x - p_2 y \quad (20)$$

Оскільки нас цікавить випадок отримання максимального прибутку, дослідимо функцію прибутку на наявність екстремуму. Враховуючи необхідну умову екстремуму, обчислимо частинні похідні першого порядку по змінним  $x$  і  $y$ :

$$P'_x(x, y) = \alpha p_0 x^{\alpha-1} y^\beta - p_1 \quad ; P'_y(x, y) = \beta p_0 x^\alpha y^{\beta-1} - p_2. \quad (21)$$

Для знаходження можливих екстремальних точок порівнюємо похідні до нуля і отримаємо систему наступних рівнянь

$$\alpha p_0 x^{\alpha-1} y^\beta = p_1 ; \beta p_0 x^\alpha y^{\beta-1} = p_2. \quad (22)$$

Систему необхідно розв'язати відносно змінних  $x$  і  $y$ . З першого рівняння виразимо  $y^\beta$ , а друге рівняння перепишемо по іншому

$$y^\beta = \frac{p_1 x^{1-\alpha}}{\alpha p_0}; \quad \frac{\beta p_0 x^\alpha y^\beta}{y} = p_2. \quad (23)$$

Після підстановки першого виразу в друге рівняння, отримаємо вираз для зв'язку між величинами, що виражають кількість використаних ресурсів.

$$y = \frac{\beta p_1 x}{\alpha p_2} \quad (24)$$

Якщо його врахувати в обох рівняннях вихідної системи, маємо наступні співвідношення:

$$x^{1-(\alpha+\beta)} = \frac{\alpha^{1-\beta} p_0}{p_1^{1-\beta} p_2^\beta} * \left(\frac{\beta}{\alpha}\right)^\beta; \quad y^{1-(\alpha+\beta)} = \frac{\beta^{1-\alpha} p_0}{p_2^{1-\alpha} p_1^\alpha} * \left(\frac{\alpha}{\beta}\right)^\alpha \quad (25)$$

Тепер необхідно перевірити виконання достатньої умови існування екстремуму функції прибутку. Обчислимо частинні похідні функції прибутку другого порядку

$$P''_{xx} = -\alpha(1-\alpha)p_0 \frac{y^\beta}{x^{2-\alpha}}; \quad P''_{yy} = -\beta(1-\beta)p_0 \frac{x^\alpha}{y^{2-\beta}}; \quad P''_{xy} = \frac{\alpha\beta p_0}{x^{1-\alpha} y^{1-\beta}}. \quad (26)$$

Далі побудуємо вираз

$$\Delta(x, y) = \alpha\beta(1-\alpha)(1-\beta)p_0^2 \frac{x^\alpha y^\beta}{x^{2-\alpha} y^{2-\beta}} - \left(\frac{\alpha\beta p_0}{x^{1-\alpha} y^{1-\beta}}\right)^2. \quad (27)$$

Після нескладних спрощень він набуває вигляду

$$\Delta(x, y) = \frac{\alpha\beta p_0^2}{x^{2-2\alpha} y^{2-2\beta}} (1 - (\alpha + \beta)). \quad (28)$$

Функція прибутку буде мати екстремум лише у випадку, коли  $\Delta(x, y) > 0$ . З отриманого виразу видно, що це буде відбуватися  $\Delta(x, y) > 0$  тільки у випадку, коли виконується умова  $\alpha + \beta < 1$ .

Таким чином, координати екстремальної точки функції прибутку дорівнюють

$$\tilde{x} = \left(\frac{p_0 \alpha^{1-\beta} \beta^\beta}{p_1^{1-\beta} p_2^\beta}\right)^{\frac{1}{1-(\alpha+\beta)}}; \quad \tilde{y} = \left(\frac{p_0 \beta^{1-\alpha} \alpha^\alpha}{p_1^\alpha p_2^{1-\alpha}}\right)^{\frac{1}{1-(\alpha+\beta)}} \quad (29)$$

Тепер можна знайти екстремальне значення функції прибутку

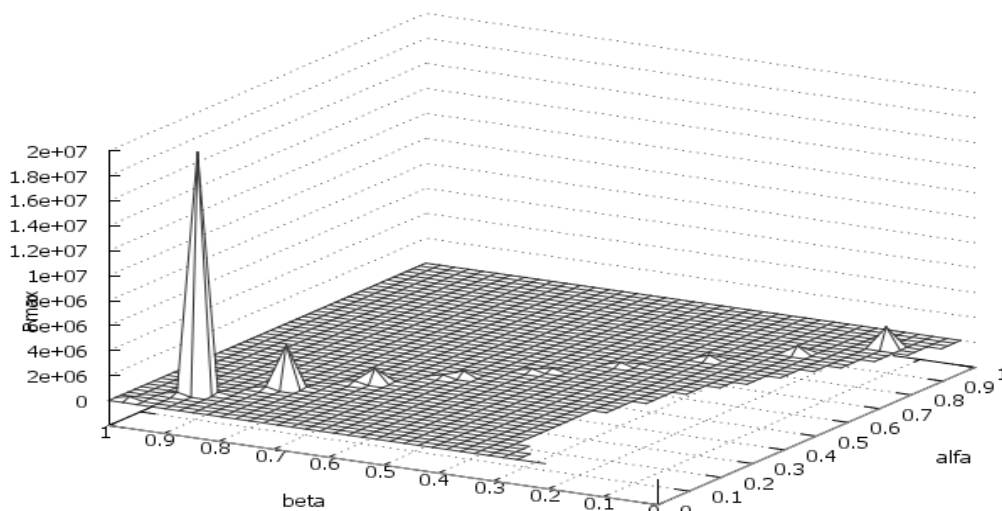
$$P_{\max}(\tilde{x}, \tilde{y}) = p_0 \left(\frac{p_0 \alpha^{1-\beta} \beta^\beta}{p_1^{1-\beta} p_2^\beta}\right)^{\frac{\alpha}{1-(\alpha+\beta)}} \left(\frac{p_0 \beta^{1-\alpha} \alpha^\alpha}{p_1^\alpha p_2^{1-\alpha}}\right)^{\frac{\beta}{1-(\alpha+\beta)}} - p_1 \left(\frac{p_0 \alpha^{1-\beta} \beta^\beta}{p_1^{1-\beta} p_2^\beta}\right)^{\frac{1}{1-(\alpha+\beta)}} - p_2 \left(\frac{p_0 \beta^{1-\alpha} \alpha^\alpha}{p_1^\alpha p_2^{1-\alpha}}\right)^{\frac{1}{1-(\alpha+\beta)}} \quad (30)$$

Отриманий вираз дозволяє обчислювати екстремальне значення функції прибутку виробничого підприємства. Він залежить від вартості продукції виробництва та вартості ресурсів, а також від параметрів  $\alpha$  і  $\beta$ . Представимо

отриманий вираз для наступних значень вартості продукції та ресурсів:  $p_0=2$ ,  $p_1=1$ ,  $p_2=0,5$ . В цьому випадку максимальне значення функції прибутку буде залежити лише від параметрів  $\alpha$  і  $\beta$ . Вираз набуває наступного вигляду

$$P_{\max}(\alpha, \beta) = 2 \left[ 2\alpha^{1-\beta} (2\beta)^\beta \right]^{\frac{\alpha}{1-(\alpha+\beta)}} * \left[ 2\alpha^\alpha (2\beta)^{1-\alpha} \right]^{\frac{\beta}{1-(\alpha+\beta)}} - \left[ 2\alpha^{1-\beta} (2\beta)^\beta \right]^{\frac{1}{1-(\alpha+\beta)}} - 0,5 \left[ 2\alpha^\alpha (2\beta)^{1-\alpha} \right]^{\frac{1}{1-(\alpha+\beta)}} \quad (31)$$

На рис. 20.1 представлена графічна залежність максимального значення функції прибутку  $P_{\max}$  від параметрів  $\alpha$  і  $\beta$ .



$$((2y)^{(1-x)} * x^{x*2} y / (1-(x+y))) - (x^{(1-y)} * (2y)^{y*2} / (1-(x+y))) - 0,5 * ((2y)^{(1-x)} * x^{x*2} / (1-(x+y)))$$

**Рис. 20.1. Залежність максимального значення функції прибутку  $P_{\max}$  від параметрів  $\alpha$  і  $\beta$ .**

З рис. 20.1 видно, що функція  $P_{\max}=P_{\max}(\alpha, \beta)$  є багатоекстремальною. Її локальні максимуми спостерігається в вузькій діагональній смужі області значень параметрів  $\alpha$  і  $\beta$ , для яких обов'язково виконується умова  $\alpha+\beta < 1$ . При певних значеннях параметрів  $\alpha$  і  $\beta$  досягається головний максимум. Наступні локальні максимуму – на порядок менші. Завдяки дискретизації області сингулярності, які утворюються ступенями  $\left(\frac{\alpha}{1-(\alpha+\beta)}\right)$  в (29-31) при  $\alpha+\beta \rightarrow 1$ , на рисунку згладжені.

Нехай  $p_0$  – вартість одиниці виробленої продукції, а  $p_1$  і  $p_2$  – вартості одиниці використаних ресурсів першого і другого виду відповідно. Тоді чистий прибуток підприємства визначається функцією прибутку, яка має наступний вигляд:

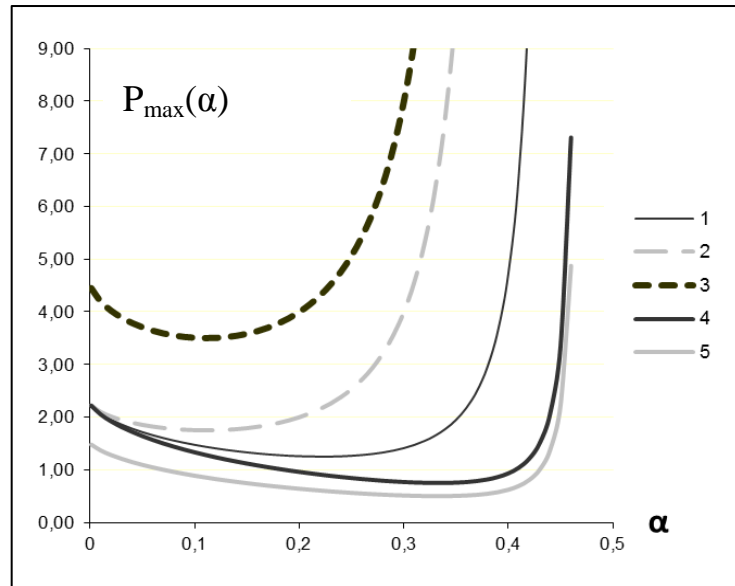
$$P(x, y) = p_0 x^\alpha y^\beta - p_1 x - p_2, \quad (32)$$

і, відповідно до неї функцію максимального прибутку

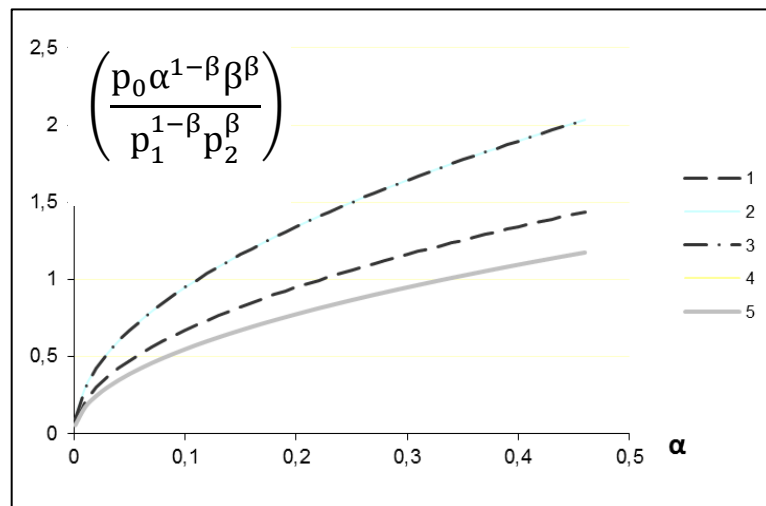
$$P_{\max}(\tilde{x}, \tilde{y}) = p_0 \left( \frac{p_0 \alpha^{1-\beta} \beta^\beta}{p_1^{1-\beta} p_2^\beta} \right)^{\frac{\alpha}{1-(\alpha+\beta)}} \left( \frac{p_0 \beta^{1-\alpha} \alpha^\alpha}{p_1^\alpha p_2^{1-\alpha}} \right)^{\frac{\beta}{1-(\alpha+\beta)}} -$$

$$-p_1 \left( \frac{p_0 \alpha^{1-\beta} \beta^\beta}{p_1^{1-\beta} p_2^\beta} \right)^{\frac{1}{1-(\alpha+\beta)}} - p_2 \left( \frac{p_0 \beta^{1-\alpha} \alpha^\alpha}{p_1^\alpha p_2^{1-\alpha}} \right)^{\frac{1}{1-(\alpha+\beta)}} \quad (33)$$

Дослідимо можливі значення параметрів функції прибутку у випадку  $P(x,y) > 0$ . Як показали дослідження, особливості функції  $P(x,y)$  і умова її додатності накладають ряд обмежень на числові значення параметрів  $\alpha$ ,  $\beta$ ,  $x$ ,  $y$ ,  $p_0$ ,  $p_1$ ,  $p_2$  і на їх співвідношення між собою.



а)

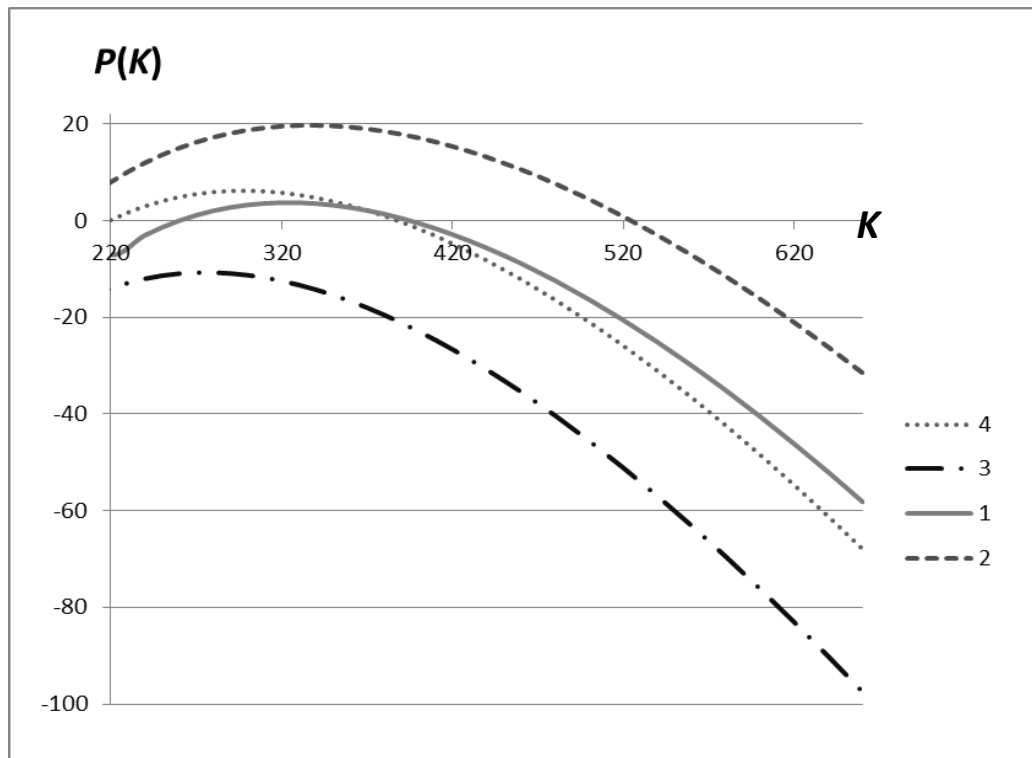


б)

**Рис. 20.2.** Дослідження функції прибутку  $P_{max}(\alpha)$  – а) і  $f(\alpha) = \left( \frac{p_0 \alpha^{1-\beta} \beta^\beta}{p_1^{1-\beta} p_2^\beta} \right)$ , що входить в функцію  $P_{max}(\alpha)$  при  $\beta=0,5$ ;  $p_0=3$ : 1 –  $p_1=1$ ;  $p_2=1$ ; 2 –  $p_1=0,5$ ;  $p_2=1$ ; 3 –  $p_1=1$ ;  $p_2=0,5$ ; 4 –  $p_1=1,5$ ;  $p_2=1$ ; 5 –  $p_1=1$ ;  $p_2=1,5$

Спершу були з'ясовані критичні параметри, що впливають на додатність функції прибутку. Цими параметрами виявилися коефіцієнти еластичності  $\alpha$ ,  $\beta$ .

Функція прибутку має чим більшу додатню ОДЗ, чим більше  $\alpha + \beta \rightarrow 1$ . Тому були обрані числові значення для еластичностей  $\alpha = 0,5$ ;  $\beta = 0,43$ , які вже допускають достатньо вільне варіювання інших значень параметрів  $x$ ,  $y$ ,  $p_0$ ,  $p_1$ ,  $p_2$  у випадку  $P(x, y) > 0$ . На рис. 20.2 показані залежності  $P_{\max}(\alpha)$  (33) при певних значеннях параметрів  $\beta$ ,  $p_0$ ,  $p_1$ ,  $p_2$ . Це дослідження показало, що вид залежності  $P_{\max}(\alpha)$  буде подібним до розглянутих навіть при інших комбінаціях параметрів  $p_0$ ,  $p_1$ ,  $p_2$ .



**Рис. 20.3.** Дослідження чутливості функції прибутку  $P(K)$  при  $L=300$ ;  $\beta = 0,43$ ;  $p_0=3$ ;  $p_2=1$ : 1 –  $\alpha = 0,5$ ;  $p_1=1$  – початкова залежність 2 –  $\alpha = 0,5$ ;  $p_1=0,95$ ;  $\Delta p_1=-5\%$ ;  $\Delta P(K)=402\%$  3 –  $\alpha = 0,5$ ;  $p_1=1,05$ ;  $\Delta p_1=+5\%$ ;  $\Delta P(K)=-202\%$  4 –  $\alpha = 0,505$ ;  $p_1=1,05$ ;  $\Delta p_1=+5\%$ ;  $\Delta \alpha = +5\%$ ;  $\Delta P(K)=66\%$

Були проведені також дослідження чутливості функції прибутку на зміну параметрів, які показані на рис. 20.3. При цьому змінювалися значення капітальних затрат  $K$  (вісь  $x$ ).

Рис. 20.3 ілюструє істотну чутливість і нестійкість до зміни значень початкових параметрів: зміна  $\Delta p_1$  на 5 % призводить до зміни функції прибутку на шість порядків (криві 2,3). Це унеможливує адекватне обчислення прибутковості за навіть незначною невизначеністю в 5 % початкових параметрів і навіть для двохфакторного виробництва. Проте одночасна зміна декількох параметрів може взаємно компенсувати результуюче значення (крива 4).

**Висновки.** Робота присвячена дослідженню екстремуму функції для оптимізації роботи виробничого підприємства за допомогою виробничої функції Кобба-Дугласа. Отримана залежність максимального значення функції прибутку двохресурсного підприємства від показників еластичності випуску продукції  $\alpha$  і  $\beta$ . Аналіз такої залежності дає можливість оцінити значення цих параметрів, при яких функція прибутку досягне максимуму. Дослідження функції прибутку показало, що при деяких значеннях показників еластичності випуску продукції, максимальне значення функції прибутку збільшується на декілька порядків, що є важливим для розрахунку ефективності роботи виробничих підприємств.

В роботі проведено моделювання показників двохфакторної моделі виробництва з метою отримання максимізації прибутку, на основі якого була проведена оцінка числових значень всіх параметрів, що входять в функцію прибутку двохресурсного підприємства. Були проведені дослідження чутливості функції прибутку на зміну параметрів, що розглядаються.



## РОЗДІЛ 21

# МУЛЬТИАГЕНТНАЯ СИСТЕМА НЕЙРОСЕТЕВОЙ ДИАГНОСТИКИ И УДАЛЕННОГО МОНИТОРИНГА ПАЦИЕНТА

***Аннотация.** Разработана мультиагентная система для предоставления дистанционных услуг в телемедицине. Одним из ключевых компонентов системы является нейронная сеть, используемая для постановки диагноза на основе обработки медико-регистрационной информации и изображений. Персонализация веб-ресурса осуществляется путем подстройки контента под предпочтения пользователей. Кластеризация пользователей выполнена самоорганизующейся картой с использованием высокопроизводительных вычислений.*

***Ключевые слова:** медицинская диагностика, нейросетевая классификация, удаленный мониторинг, организационная мультиагентная система, информационный портрет пользователя, персонализация веб-ресурса.*

***Abstract.** Multi-agent system for providing remote services in telemedicine was developed. The diagnosis is made on the basis of processing the medical- registration information and the image with using a neural network. Personalization Web-resource carried out by adjusting the content to the user preferences. The clusterization of users is executed by the self-organizing map using the high-performance computing.*

***Keywords:** medical diagnostics, neural network classification, remote monitoring, organizational multi-agent system, information portrait user, personalization Web resource.*

**Введение и постановка задачи.** Проблемы диагностики объектов или процессов занимают большое место в современном мире. В общем случае, цель диагностики заключается в том, чтобы на основе анализа текущего состояния диагностируемого объекта определить значения выходных переменных, реализация которых позволяет обеспечить желаемое поведение или состояние исследуемого объекта.

На кирпичном производстве объектом контроля качества является кирпич, помещенный на вагонетку. Целью диагностики является выявление наличия дефектов кирпича при приемочных испытаниях партии, основанное на визуальных дефектах, а именно: выявление геометрических дефектов, пережогов, недожогов, а также трещин.

Диагностическими объектами при разбраковке по качеству могут выступать бесшовные трубы общего назначения, применяемые в машиностроении, холодильной промышленности, котельных высокого

давления, в агрессивных средах, для сооружения трубопроводов, конструкций авиапромышленности, судостроении, деталей машин и т.д. В случае обнаружения дефектов труба направляется в зону осмотра для анализа разбраковки по классам качества.

В медицинских диагностических системах существенную роль играют особенности медико-биологической информации, база данных которой зачастую включает высококачественные цифровые фотографии, используемые для предварительного диагноза, динамического наблюдения за текущим состоянием пациента, оценки эффективности лечения;

В системе судейства спортивных соревнований могут складываться противоречия в арбитражной системе. Так, например, для устранения субъективности оценивания выступления спортсменок в художественной гимнастике диагностическая система способна определять точность выполненных групп сложности в случае возникновения спорного момента у судей путем анализа видеопотока, содержащего выступление спортсменки.

Особенностью диагностики представленных объектов является то, что изменение их состояния можно фиксировать на основе анализа изображений этих объектов. В подобных ситуациях кроме простого визуального наблюдения и сравнения обязательными становятся операции подсчета, измерения, статистического анализа полученной информации. Здесь могут помочь интеллектуальные компьютерные системы анализа изображений.

Использование нейронных сетей (НС) для обработки изображений имеет преимущества перед традиционными математическими методами в трех случаях. Во-первых, когда рассматриваемая задача не поддается адекватной формализации, поскольку содержит элементы неопределенности, не формализуемые традиционными математическими методами. Во-вторых, когда рассматриваемая задача формализуема, но пока отсутствует аппарат для ее решения. В-третьих, когда для рассматриваемой, хорошо формализуемой задачи существует соответствующий математический аппарат, но реализация вычислений с его помощью на базе имеющихся вычислительных систем не удовлетворяет требованиям получения решений по времени, размеру, весу, энергопотреблению и др.

Агентно-ориентированная технология (АОТ) базируется на использовании интеллектуальных программных агентов и позволяет увеличить функциональные возможности современных интеллектуальных систем. Одним из важных направлений применения мультиагентных систем (МАС) является разработка концептуальных моделей машинного обучения, позволяющая

объединять информацию из окружения при распознавании данных, классификации, прогнозировании и т. п.

Таким образом, возникает необходимость разработки диагностических/экспертных систем, в которых можно удаленно фиксировать изменение состояния объектов на основе анализа собранной их изображений с учетом особенностей конкретной предметной области, а также предоставлять актуальную информацию по различным аспектам и осуществлять информационное сопровождение пользователей в реальном времени.

Эффективное сочетание АОТ и машинного обучения в реальной среде обусловили возможность построения мультиагентной системы нейросетевой медицинской диагностики и удаленного мониторинга состояния здоровья человека. Так, например, своевременное обнаружение изменений кожного новообразования позволит врачу установить диагноз рака кожи на ранней стадии, что спасет жизнь человеку. Удаленный мониторинг показателей здоровья является результативным способом медицинского сопровождения пациентов, нуждающихся в длительном врачебном наблюдении. Пользователями разрабатываемой системы могут быть пациенты с хроническими заболеваниями при амбулаторном обследовании, при подготовке к операции, на этапе долечивания и реабилитации после острых состояний. Предлагаемая система позволит осуществлять самостоятельный контроль показателей состояния здоровья. В случае необходимости, пациент может удаленно обратиться за помощью или консультацией к специалисту.

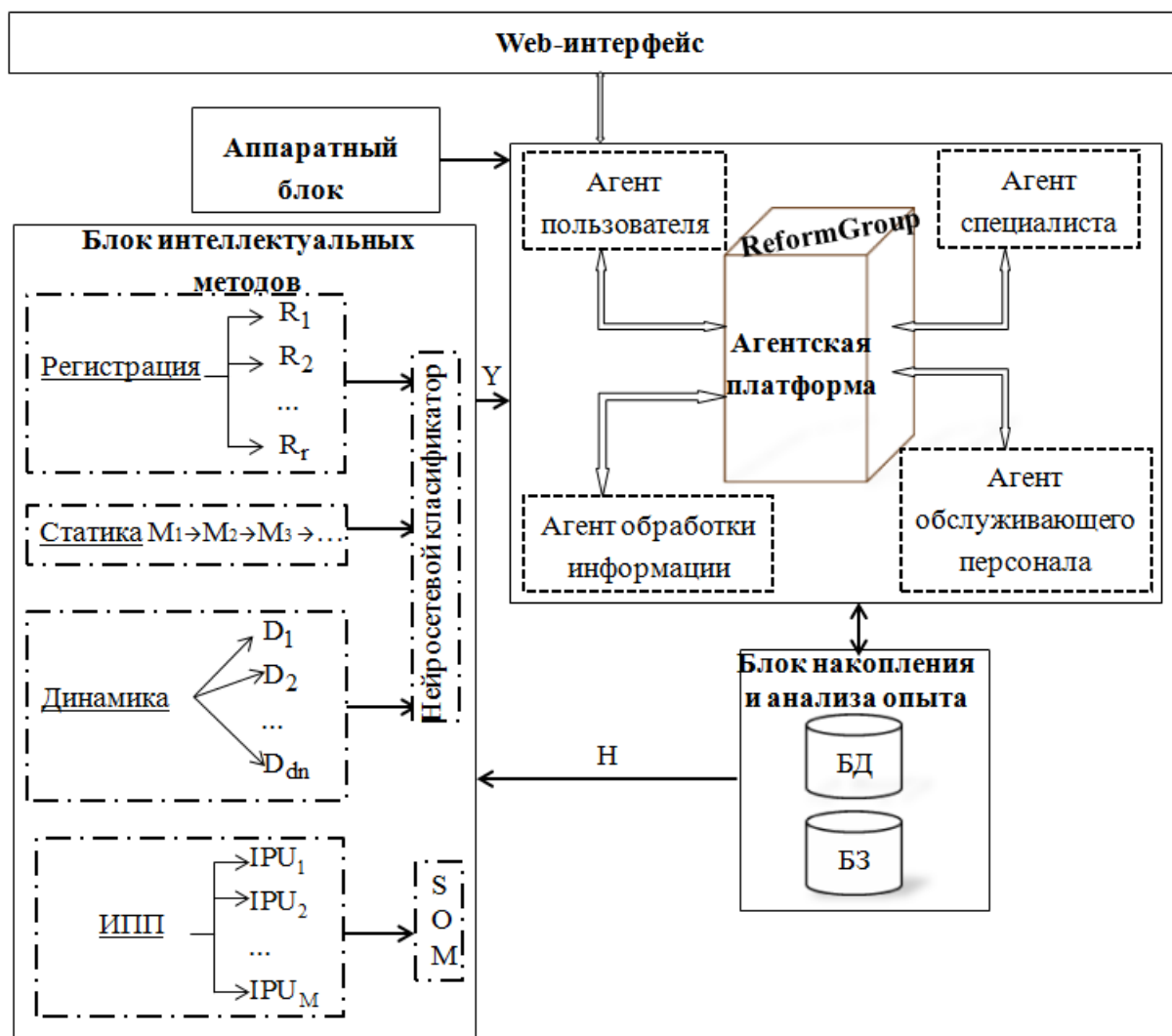
### **Обобщенная модель мультиагентной системы нейросетевой медицинской диагностики и мониторинга**

Основными действующими лицами разработанной системы [102] являются пациенты с хроническими заболеваниями при амбулаторном обследовании, при подготовке к операции, на этапе долечивания и реабилитации после острых состояний и специальные службы (сиделки, медицинский персонал и люди, занимающиеся мониторингом, наблюдатели).

Пациент – человек, который находится за пределами больницы и его физическое состояние требует постоянного контроля, рекомендаций и, возможно, неотложной помощи. Он должен быть зарегистрирован в медицинском учреждении для включения в систему, иметь биодатчики и измерительные приборы, считывающие наиболее важные показатели здоровья и соединяющиеся с мобильным устройством, которое доставляет полученные данные посредством Интернета на основной сервер.

Медицинский персонал – квалифицированные врачи, зарегистрированные в учреждении здравоохранения, которые наблюдают за состоянием здоровья пациента, выписывают рецепты и осуществляют необходимые лечебные манипуляции.

Процесс медицинской диагностики и удаленного мониторинга описывается моделью, которая включает: блок интеллектуальных методов, агентский блок, блок накопления и анализа опыта, Web-интерфейс, а также аппаратный блок (рис. 21.1).



**Рис. 21.1. Обобщенная модель мультиагентной системы нейросетевой медицинской диагностики и мониторинга пациента**

В общем виде принцип функционирования МАС для медицинской диагностики и мониторинга выглядит следующим образом.

Пациент, желающий получить консультацию врача, нуждающийся в услугах медсестры (сиделки) и т. п., после регистрации на медицинском

Web-ресурсе получает уникальный идентификатор и на его мобильное устройство устанавливается специальное приложение.

Датчики аппаратного блока (измерительные приборы, камеры видеонаблюдения, фотокамеры и т. п.) отправляют информацию через Bluetooth к мобильному устройству, которое пересылает полученные данные “Агенту пользователя”  $A_{user}$ , а он агрегирует и переводит показания устройств в форму, пригодную для дальнейшей обработки.

“Агент обработки информации”  $A_{data}$  определяет, какие задания  $D_H$ ,  $D_{Hr}$ ,  $D_{Hst}$ ,  $D_{Hdn}$  или/и  $D_{Hp}$  будут выполняться в блоке интеллектуальных методов, передает/принимает информацию блоку накопления и анализа информации.

В блоке интеллектуальных методов решаются задачи:

- регистрации медико–диагностической информации  $D_{Hr}$ ;
- обработки изображений объектов в состоянии покоя  $D_{Hst}$ ; или
- обработки изображений объектов в состоянии движения  $D_{Hdn}$ ;
- нейросетевой классификации – ставится предварительный диагноз и передается информация “Агенту специалиста”  $A_{doctor}$ ;
- построения информационного портрета пользователя (ИПП);
- категоризации пользователей  $D_{SOM}$  с помощью самоорганизующейся сети Кохонена [120];
- персонализации веб-ресурса  $D_{Hp}$ .

“Агент специалиста”  $A_{doctor}$  собирает информацию от «Агента пользователя»  $A_{user}$ , “Агента обслуживающего персонала”  $A_{nurse}$ , «Агента обработки данных»  $A_{data}$ , назначает лечение пациенту и дает предписания “Агенту обслуживающего персонала”  $A_{nurse}$ .

“Агент обслуживающего персонала”  $A_{nurse}$  обменивается информацией с “Агентом специалиста”  $A_{doctor}$ , “Агентом обработки данных”  $A_{data}$  и посылает распоряжения пациенту, медсестре, сиделке, родственникам и т. д.

Одновременно “Агент пользователя”  $A_{user}$  собирает информацию о пользователе и посылает ее “Агенту обработки данных”  $A_{data}$ . По результатам решения задачи категоризации пользователей  $D_{SOM}$  “Агент пользователя”  $A_{user}$  для каждого кластера вырабатывает оптимальную стратегию, позволяющую учитывать предпочтения пользователей, и настраивает соответствующий контент.

Состав и взаимосвязь компонентов блока интеллектуальных методов  $\Xi$  выражается как преобразование входных значений  $H$  в выходные величины  $Y$ :

$$\Xi \subset H \times Y,$$

где  $H(i) = \{H_D(i), H_P(i)\}$ , ( $i = \overline{1, M}$ ) – матричное представление входных данных:

– медико-диагностические данные измерительных приборов,

– изображения, у которого значения яркости элементов обозначаются  $h_{kj}$ –

( $h_{kj} = \overline{0, 255}$ ,  $k = \overline{1, m}$ ,  $j = \overline{1, n}$ , );

– доступная информация о пользователе медицинского веб-портала;

$Y = \{Y_D, Y_P\}$ , ( $Y \subset \Upsilon$ ) – множество формализованных свойств.

Таким образом, пространство (универсум)  $\Sigma = X \times Y$  включает  $\Xi \subset (H \times Y)$ , это означает, что существует такое подмножество  $H$ , ( $H \subset X$ ) и отношений между ними, на которых строится модель  $\Xi$  ( $\Xi \subset \Sigma$ ).

Для выходных величин  $Y_D$  построено множество заданий, решение которых принадлежит множеству  $D_H = \{D_{Hr}, D_{Hst}, D_{Hdn}\}$ , где  $D_{Hr} = \{R_1, R_2, \dots, R_r\}$  – задача регистрации медико-диагностической информации (температура, артериальное давление, пульс, ...),  $D_{Hst} = \{M_1, M_2, \dots, M_{st}\}$  – задача обработки изображений объектов в состоянии покоя ( $M_1$  – выделение области интереса,  $M_2$  – бинаризация,  $M_3$  – скелетонизация, ...),  $D_{Hdn} = \{D_1, D_2, \dots, D_{dn}\}$  – задача обработки изображений объектов в состоянии движения (прямолинейного, вращательного, поступательного, равноускоренного и других видов). Для выходных величин  $Y_P$  построено множество заданий, решение которых принадлежит множеству  $D_P = \{D_{IPU}, D_{SOM}\}$  – задача адаптации медицинского веб-портала ( $D_{IPU}$  – построение информационного портрета пользователя,  $D_{SOM}$  – кластеризация пользователей с помощью сети Кохонена).

Отображение  $T: H_D \rightarrow Y_D$  позволяет для каждого  $H_D(i)$  найти такое  $Y_j \in Y_D$  ( $j = \overline{1, Q}$ ,  $Q$  – количество классов), которое является решением задачи  $D_H$ .

Значения  $Y_j \in Y_D$  используются для формирования конечного диагноза с помощью нейросетевого классификатора и выработки дальнейшей тактики поведения.

Отображение  $\Pi: H_P \rightarrow Y_P$  позволяет для каждого  $H_P(i)$  получить такое  $Y_j \in Y_P$  ( $j = \overline{1, \Omega}$ ,  $\Omega$  – количество интерфейсов), которое является решением задачи  $D_P$ , полученное в виде распределенной модели агентно-ориентированного медицинского Интернет-ресурса [4].

Значения  $Y_j \in Y_p$  используются для выработки стратегии настройки контента для каждого кластера пользователей с помощью самоорганизующейся карты Кохонена.

Для оценки качества процесса функционирования предлагаемой системы используется совокупность критериев оценки эффективности  $K = \{k_1, k_2, k_3\}$ , позволяющих определить реакцию адаптации сайта [68].

Одним из главных критериев является степень релевантности

$$k_1 : \varpi = \max(\varpi),$$

где  $\varpi$  - степень релевантности отображенной информации.

Другим критерием является время адаптации веб-ресурса

$$k_2 : \tau = \min(\tau^s, \tau^d),$$

где  $\tau^s$  – время статической генерации страниц;  $\tau^d$  – время динамической генерации.

Соотношение множества реально отображенных разделов характеризуются следующими характеристиками:

$$\psi = \{\rho, \eta, e, q, \upsilon\},$$

где  $\rho = \frac{a}{a+c}$  – коэффициент полноты, характеризующий долю релевантных

страниц на веб-ресурсе к общему числу релевантных документов;  $\eta = \frac{a}{a+b}$  –

коэффициент точности, характеризующий долю отображенных релевантных

страниц на веб-ресурсе к общему объему выдачи;  $e = \frac{b}{a+b} = 1 - \eta$  –

коэффициент шума, характеризующий долю отображенных нерелевантных

документов во всем множестве отображенных документов;  $q = \frac{b}{b+f}$  –

коэффициент осадка, характеризующий долю отображенных нерелевантных

документов во всем множестве нерелевантных документов;  $\upsilon = \frac{f}{b+f}$  –

коэффициент специфичности, характеризующий долю неотображенных

нерелевантных документов во всем множестве нерелевантных документов; а-

количество адаптированных релевантных разделов; b – количество

адаптированных нерелевантных разделов; с- количество неадаптированных

релевантных разделов; f – количество неадаптированных нерелевантных

разделов.

Для эффективного функционирования предлагаемой модели необходимо, чтобы система удовлетворяла следующим требованиям:

$$k_3 : \forall (d_i \in D_p) [(\tau < \tau^{\max}) \& (\varpi > \varpi^{\min}) \& (\rho, \eta \rightarrow 1) \& (e, q, v \rightarrow 0)] \Rightarrow R,$$

где  $d_i$  – подзадача общей задачи  $D_p$ .  $\tau^{\max}$  – максимально допустимое время решения задачи;  $\varpi^{\min}$  – минимально допустимая релевантность.

При моделировании полученных теоретических результатов адаптации Интернет-ресурса были систематизированы пользователи медицинского сайта. Так, для пользователя первый раз посетившего ресурс “Клиника Феськова”, отобразится начальная страница, изображенная на рис. 21.2а, которая содержит общую информацию с переходами на все разделы сайта – энциклопедию, статьи, новости по всем тематикам и т.д. А для женщин, имеющих проблемы с зачатием, отобразится страница, представленная на рис. 21.2б.



а

б

Рис. 21.2. Категории посетителей медицинского сайта

Данная модель учитывает предпочтения пользователей и позволяет автоматически настраивать контент медицинского портала в зависимости от состояния внешней среды или индивидуальных предпочтений пользователя.

Агентский блок представлен следующими множествами:

$$MAS = \{A, E, Org\},$$

где  $A = \{A_{\text{user}}, A_{\text{doctor}}, A_{\text{nurse}}, A_{\text{data}}\}$  – множество агентов, функционирующих в среде  $E$ ;  $E$  – среда, представляющая собой программную платформу для выполнения агентов и предоставляющая функциональность для создания/уничтожения агентов, для применения интеллектуальных методов и

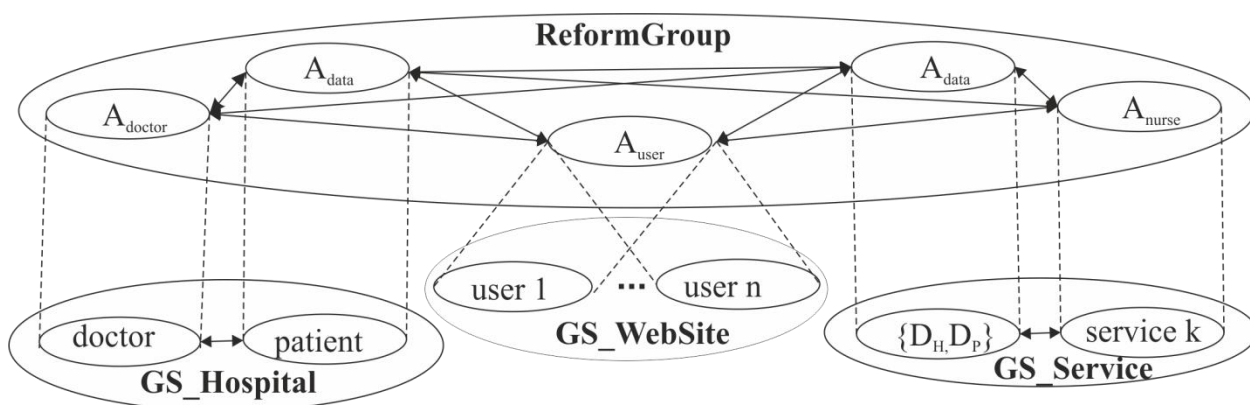


для приема/передачи сообщений; Org – Web-портал службы здравоохранения, построенный на основе организационной модели для взаимодействия агентов.

Организация состоит из структурного и динамического аспектов. Первый включает группы и роли – агенты организованы в связанные между собой группы, а для каждой роли в группе определен набор функций и их отношения. Динамический аспект организации связан с установившимися шаблонами взаимодействия, определенными в ролях.

Для описания структурной части модели используется трехуровневое моделирование Агент-Группа-Роль (АГР), где на верхнем уровне выступает характеристика организации, на среднем – групповые характеристики, а на нижнем – ролевые характеристики.

На рис. 21.3 представлена структура адаптивной организационной модели Org, назначением которой является управление инцидентом – обнаружение и устранение критических состояний здоровья пациента.



**Рис. 21.3. Структура адаптивной организационной модели медицинского Интернет-ресурса**

Маленькие овалы обозначают роли Role, большие овалы – группы Group, сплошные линии обозначают связи между ролями, а пунктирные линии – межгрупповые связи. Структура нижнего уровня зависит от определенной области приложения, в данном случае: GS\_Hospital – структурная группа лечебного учреждения, состоящая из лечащего состава и пациентов; GS\_WebSite – структурная группа пользователей медицинского Web-сайта; GS\_Service – структурная группа, обеспечивающая реализацию интеллектуальных методов и предоставляющая услуги медсестры, сиделки. Другая структура ReformGroup – группа преобразований (верхний уровень) осуществляет адаптацию организации и может быть общей для любого типа

приложения. Агенты этой группы выполняют адаптирующую роль, т. е. контролируют ролевые, групповые и организационные характеристики.

В случае управления инцидентом аспектами будут: мониторинг состояния здоровья, постановка диагноза и медицинская помощь. Для мониторинга используется механизм “градации” состояния пациента, определяющий диапазоны измеряемых значений, таких как: кардиограмма, кровяное давление, уровень сахара, температура тела и т. д., которые разделяются на следующие уровни тяжести: критический (Critical), высокий (Tall), средний (Average), низкий (Lower).

Для описания обмена сообщениями между агентами воспользуемся следующими обозначениями:  $A.send(B, m)$  – агент  $A$  отправляет сообщение  $m$  агенту  $B$ ;  $roleIn(r, g)$  – роль  $r$  определена в группе  $g$ ;  $plays(A, r, g)$  – агент  $A$  играет роль  $r$  в группе  $g$ ;  $GStruct(g, gs)$  –  $g$  является группой, рассматриваемой как экземпляр структуры группы  $gs$ ;  $member(A, g)$  – агент  $A$  является членом группы  $g$ . Структурными аспектами модели АГР являются:

– каждый агент – член (по крайней мере, одной) группы:

$$\forall A : Agent, \exists g : Group, member(A(i), g);$$

– каждый агент играет (хотя бы одну) роль в группе:

$$\forall A : Agent, \forall g : Group \Rightarrow \exists r : Role, plays(A(i), r, g);$$

– агент является членом группы, в которой он исполняет роль:

$$\forall A : Agent, \forall g : Group, \forall r : Role, plays(A(i), r, g) \Rightarrow member(A(i), g);$$

– роль определена в структуре группы:

$$\forall A : Agent, \forall g : Group, \forall r : Role, plays(A(i), r, g) \Rightarrow \exists gs : GroupStructure \wedge GStruct(g, GS) \wedge roleIn(r, GS).$$

Для выражения соответствия между пациентом медицинского учреждения и пользователем медицинского Web-сайта используем выражение:

$$\begin{aligned} &role(patient, GS\_Hospital) \Rightarrow \\ &\Rightarrow role(user, GS\_WebSite). \end{aligned}$$

Это ограничение определяется следующим образом:

$$\forall A : Agent, \forall g : Group,$$

где группы  $\exists g1, g2, g3 : Group$  образуют

$$\begin{aligned} &GroupStruct(g1, GS\_WebSite), \\ &GroupStruct(g2, GS\_Hospital), \text{ и } GroupStruct(g3, GS\_Service) \end{aligned}$$

такие, что

$$\begin{aligned} &plays(A_{user}, user, g1) \Rightarrow plays(A_{data}, expert, g3) \vee \\ &\vee plays(A_{doctor}, doctor, g2) \Rightarrow plays(A_{user}, patient, g2). \end{aligned}$$

Агент консультанта  $A(i)$  Web-сайта активизируется в том случае, если консультант является лечащим врачом в больнице:

$$\text{role}(\text{consult}, \text{GS\_WebSite})$$

требует

$$\text{role}(\text{doctor}, \text{GS\_Hospital}) \vee \text{role}(\text{expert}, \text{GS\_Service}).$$

Его семантика определяется:

$$\text{GStruct}(g2, \text{GS\_Hospital}) \wedge \text{GStruct}(g3, \text{GS\_Service}),$$

$$\text{plays}(A_{\text{doctor}}, \text{doctor}, g2) \vee \text{plays}(A_{\text{data}}, \text{expert}, g3) \Rightarrow \text{plays}(A_{\text{doctor}}, \text{consult}, g1).$$

Агент пользователя  $A_{\text{user}}$  обращается за медицинской услугой путем отправки сообщения агенту консультанта  $A_{\text{doctor}}$

$$\forall A_{\text{user}}, A_{\text{doctor}} : \text{Agent}, \forall m1 : \text{Message},$$

$$A_{\text{user}}.\text{send}(A_{\text{doctor}}, m1) \Rightarrow \exists g1 : \text{Group},$$

$$\text{member}(A_{\text{user}}, g1) \wedge \text{member}(A_{\text{doctor}}, g1).$$

Ответ может быть получен в виде

$$\forall A_{\text{user}}, A_{\text{doctor}}, A_{\text{data}} : \text{Agent}, \forall m2, m3 : \text{Message},$$

$$A_{\text{doctor}}.\text{send}(A_{\text{data}}, m2) \Rightarrow \exists g2, g3 : \text{Group},$$

$$A_{\text{data}}.\text{send}(A_{\text{user}}, m3) \Rightarrow \exists g1, g3 : \text{Group},$$

$$\text{member}(A_{\text{user}}, g1) \wedge \text{member}(A_{\text{doctor}}, g2) \wedge \text{member}(A_{\text{data}}, g3).$$

Все агенты, участвующие в организации, выполняют адаптирующую роль.

Блок накопления и анализа информации включает базу данных (БД) медико-регистрационной информации, цифровых фотографий, информационных портретов пользователей, а также базу знаний (БЗ), содержащую правила вывода для постановки диагноза, для персонализации веб-ресурса и информацию, являющуюся результатом накопленного опыта. Этот блок кроме осуществления одноименных операций осуществляет резервное копирование данных и их воссоздание.

### **Имитационное моделирование мультиагентной системы нейросетевой медицинской диагностики и удаленного мониторинга пациента**

Для моделирования полученных теоретических результатов нейросетевого диагностирования использована компьютерная сеть Fast Ethernet с физической топологией “звезда” со скоростью передачи 100 Мбит, которая состоит из 10 четырехядерных вычислителей Intel Core 2 Quad CPU Q8200 @2.33GHz с видеокартой GeForce GTX 460. Распараллеливание выполнено с

помощью технологий CUDA, OpenMP и MS MPI на языке программирования C++ в операционной системе Microsoft Windows Compute Cluster Server 2003.

Исходные данные для проведения экспериментальных исследований предоставлены Институтом дерматологии и венерологии Академии наук Украины в рамках договора о научно техническом содружестве [69]. Предметом Договора является разработка системы дифференциальной диагностики кожных заболеваний на основе клинических критериев, клинического анализа крови и данных гуморального иммунитета.

Реализована частная постановка задачи первичной диагностики меланомы кожи на основании опроса больного и снимков участков пораженной кожи со следующими ограничениями:

– диагноз пациента может включать не более одного заболевания, пациент болен одним заболеванием (меланома (м), невус (н) и линтиго (л)) или здоров;

– пациент обследуется врачом с момента подозрения на возникновение заболевания;

– подзадача поиска причины заболевания не рассматривается.

В процессе консультации пациент (возможно удаленно) отвечает на вопросы, поставленные врачом. На основании ответов формируется вектор опроса, который вместе с результатом обработки изображения поступает на вход нейронной сети для постановки первичного диагноза. Мобильный агент с заданной периодичностью посылает сообщения, содержащие напоминание о повторном фотографировании или употреблении лечебно-профилактических препаратов, о контрольном посещении врача и т. п.

В ходе проведения экспериментов врачу предлагалось поставить диагноз самостоятельно, затем с использованием разработанной системы (табл. 21.1).

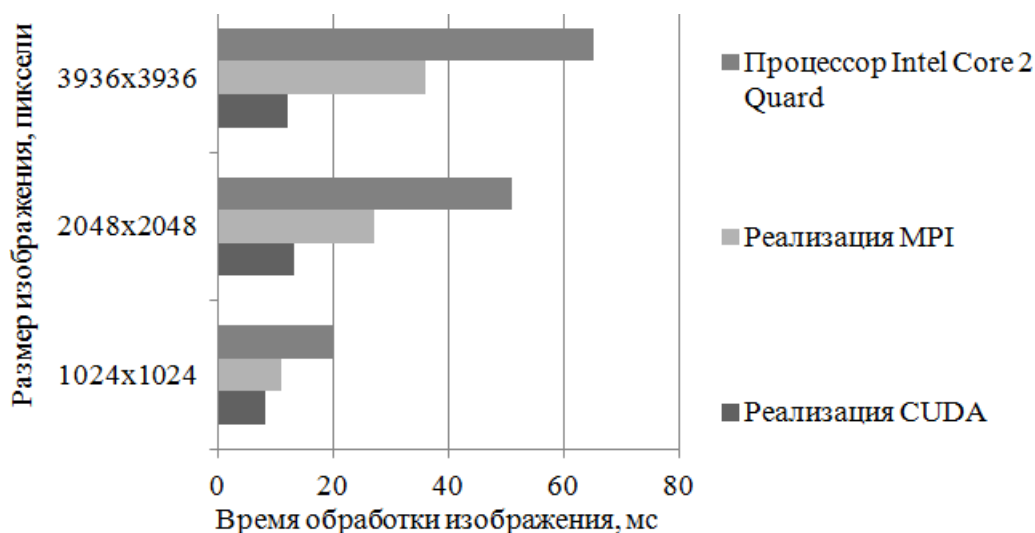
Таблица 21.1

**Результаты тестирования разработанной системы медицинской нейросетевой диагностики заболеваний кожи**

Класс	Кол-во примеров	Результат диагностики					
		без использования системы			с использованием системы		
		поставленный диагноз			поставленный диагноз		
		л	н	м	л	н	м
лентиго	50	40	7	3	44	4	2
невус	50	2	41	7	3	45	2
меланома	50	0	10	40	0	5	45

Самыми трудоемкими подзадачами при постановке диагноза являются обработка изображений большой размерности и обучение нейронной сети. Поэтому для ускорения решения этих задач использовались технологии параллельного и распределенного программирования.

Для обработки изображений в блоке интеллектуальных методов использована NVIDIA NPP-библиотека. Результаты сравнения времени обработки изображений приведены на рис. 21.4.



*Рис. 21.4. Процесс обработки изображения на основе последовательной и параллельной реализаций*

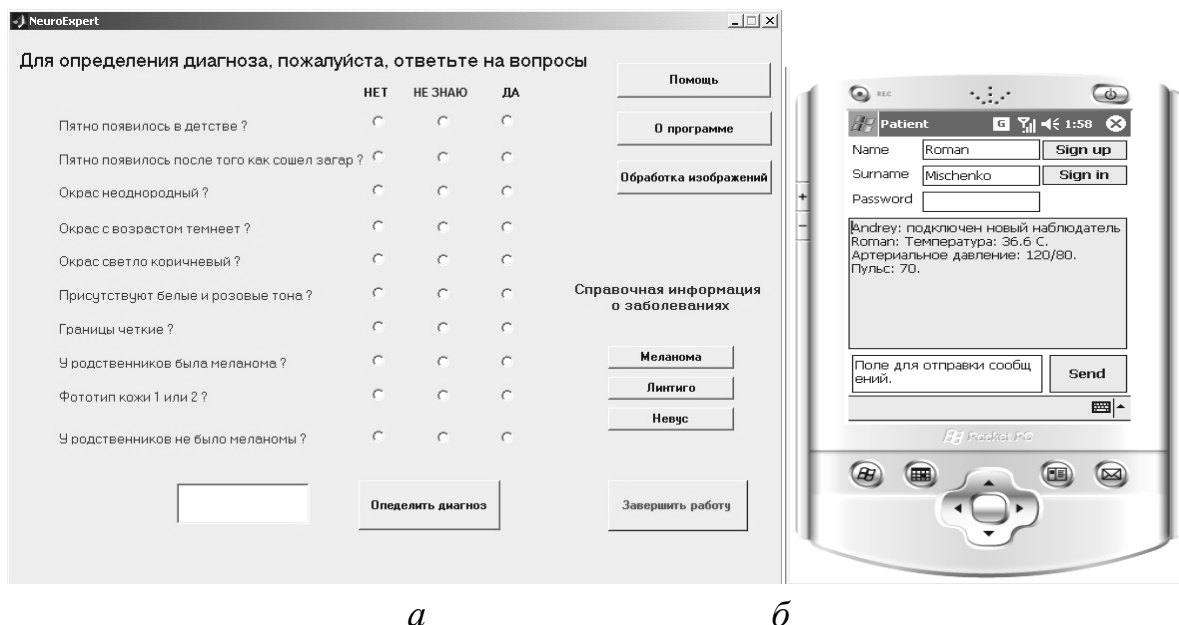
Для анализа процедуры параллельного и распределенного обучения нейронной сети на многопроцессорной системе проведен ряд экспериментов. Время обучения НС в высокопроизводительной среде для решения задачи диагностики меланомы кожи приведено в табл. 21.2.

Таблица 21.2

**Время обучения многослойного персептрона с разным количеством нейронов входного, скрытого и выходного слоев**

Архитектура НС	Количество вычислителей MPI								Количество вычислителей OpenMP		
	1	4	8	12	16	20	24	28	2	4	8
2500-120-28	923	279	202	149	130	120	123	130	504	280	183
4225-245-28	3613	1159	578	397	388	355	338	327	1959	1042	483
10000-455-28	5431	1025	508	377	353	337	322	310	2715	1180	705

Программное обеспечение для мобильного агента, осуществляющего удаленный мониторинг, реализовано на платформе .NET Framework с использованием языка C# для операционной системы Windows и Windows Mobile (рис. 21.5).



**Рис. 21.5. Скриншот интерфейса пациентов для пользователя: а – Windows 7; б – Windows Mobile**

Персональное цифровое программное обеспечение устанавливается на личном цифровом устройстве пациента. Оно взаимодействует с датчиками, приложенными к пациенту, и получает от них информацию для обработки и последующего отправления центральному серверу.

Клинические ошибки, связанные с многообразием проявлений меланомы, несмотря на визуальную локализацию опухоли, даже у опытных дерматологов достигают 25 – 40 %. У клиницистов, не специализирующихся в области дерматологии и онкологии, ошибки достигают 50 %. Кроме того, почти 2/3 врачей не в состоянии диагностировать диспластический невус, являющийся предшественником меланомы, который может малигнизироваться (переродиться в злокачественное образование).

В дерматологическом отделении израильской больницы Хадасса работает лаборатория компьютерного наблюдения за родимыми пятнами. При обращении пациента проводится осмотр кожных покровов для выявления родинок, подлежащих наблюдению. Они фотографируются с помощью цифровой камеры, также проводится измерение диаметра, площади и цвета

родимого пятна. Все эти данные заносятся в компьютерную базу данных и сохраняются. Последующие осмотры проводятся раз в полгода, и их результаты сравниваются с предыдущими измерениями. При необходимости производится биопсия. Предложенная система отличается от израильской тем, что она помогает врачу в постановке диагноза за счет удаленного автоматического контроля изменений кожных образований. Тестирование показало, что разработанная система медицинской диагностики кожных образований позволяет повысить точность диагностики на 9.3 %. Таким образом, разработка моделей ранней клинической диагностики первичной опухоли может улучшить результаты лечения больных меланомой кожи.

Для обработки изображений использован набор изображений размером от 1024x1024 до 3936x3936. Время обработки изображений не зависит от содержимого изображения, а только от количества пикселей. При параллельной реализации с использованием архитектуры CUDA было задействовано разное количество физических потоков от 1 до 112. Проведенные эксперименты показали, что применение графического процессора при использовании 112 потоков значительно ускоряет процесс обработки изображений.

Анализ таблицы 2 показал, что технология MPI позволяет ускорить распределенное обучения НС со структурами 2500-120-28, 4225-245-28, 10000-455-28 соответственно в 7, 11, 17 раз для топологии передач данных «звезда». Однако при значительном увеличении количества процессоров в системе растет время на передачу данных между ними. Поэтому, при достижении определенного прироста производительности, дальнейшее увеличение количества процессоров в системе дает обратный эффект – производительность начинает падать. Стандарт OpenMP позволяет ускорить параллельное обучение НС до 7 раз.

Ускоренная обработка данных позволяет воспользоваться системой медицинской диагностики и удаленного мониторинга состояния здоровья человека в реальном времени.

**Заключение.** Разработана система диагностики меланомы кожи на основе вектора опроса пациента с врачом и изображений исследуемого участка кожи, которая поможет врачу в постановке диагноза на ранней стадии. Предложенная система обладает возможностью дистанционного контроля изменений наблюдаемых кожных образований в реальном времени за счет высокопроизводительных вычислений.

Данная система реализована благодаря предложенной обобщенной модели процесса нейросетевой диагностики и мониторинга, в которой

сочетаются машинное обучение и агентские технологии. Такое сочетание позволило результаты удаленного мониторинга использовать для постановки диагноза. Описанное в модели применение совокупности методов и средств обработки изображений может быть использовано в системах, в которых целесообразно удаленно фиксировать изменение состояния объектов на основе анализа их изображений.

Взаимодействие компонентов предложенной модели осуществлено за счет спроектированной мультиагентной организационной структуры. Динамическое распределение ролей между агентами организационной модели медицинского Интернет-ресурса позволило адаптироваться для управления такими инцидентами, как мониторинг состояния здоровья (контроль изменения размеров кожного образования), постановка диагноза и медицинская помощь (рекомендации о принятии препаратов).

Разработанный метод адаптации Интернет-сервиса учитывает предпочтения пользователей и позволяет автоматически настраивать контент медицинского портала. Внедренный в Клинике профессора Феськова А.М. адаптивный медицинский Интернет-ресурс позволяет предоставлять актуальную информацию по различным аспектам диагностики и лечения бесплодия, осуществлять медицинские консультации в режиме реального времени, осуществлять информационное сопровождение пациентов в процессе лечения и используется на этапах предварительной диагностики.



## РОЗДІЛ 22

# РОЗВИТОК ІТ-ГАЛУЗІ В УКРАЇНІ: СВІТОВІ РЕЙТИНГИ, ТЕНДЕНЦІЇ, ПЕРСПЕКТИВИ

***Анотація.** Розглянуто питання розвитку ІТ-галузі в Україні, як однієї з інвестиційно привабливої галузі, на основі міжнародних рейтингів розвитку інформаційного суспільства. Визначено пріоритетні напрямки розвитку ІТ в Україні. Сучасне інформаційне суспільство базується на розвитку людських ресурсів, а саме їх креативності, яка є основою інноваційних ІТ. Проведено аналіз позицій України в глобальному рейтингу за конкурентоспроможністю талантів.*

***Ключові слова:** ІТ-галузь, інформаційне суспільство, міжнародні індекси, рейтинг України, інновації, людські ресурси, творчий персонал.*

***Abstract.** The problems of the IT industry in Ukraine as one of the attractive investment area, based on international rankings of the Information Society. The priority areas of IT development in Ukraine. The modern information society based on human resource development, and it is their creativity, which is the basis of innovative IT. The analysis of Ukraine's position in the global ranking of competitiveness talents.*

***Keywords:** IT industry, information society, international indices, ratings of Ukraine, innovation, human resources, creative staff.*

**Вступ та постановка завдання.** У період активних процесів євроінтеграції для національної економіки розвиток інформаційного суспільства та поширенні ІТ у всі сфери життєдіяльності суспільства є вкрай актуальним питанням. Інформаційні технології грають сьогодні провідну роль у розвитку інновацій та підвищенні конкурентоспроможності країни, стимулюють розвиток людських ресурсів та процвітання креативної економіки, тим самим сприяючи підвищенню рівня життя суспільства. Звідси випливає тісний зв'язок між розвитком ІТ-галузі і конкурентоспроможністю національної економіки у світі. Зважаючи на це, чітке розуміння сучасних позицій України за рівнем розвитку ІТ у світових рейтингах дає можливість розуміти шляхи поліпшення ситуації, вектори змін для побудови високорозвиненої національної ІТ-галузі, яка буде відповідати світовому рівню розвитку, зокрема рівню розвитку країн-членів ЄС.

Мета – на підставі світових індексів, які характеризують розвиток ІТ-галузі, проаналізувати позиції України у світових рейтингах та визначити пріоритетні напрями її подальшого розвитку.

**Основна частина.** Сучасний світ динамічно розвивається, а невід’ємною, важливою частиною цього розвитку стає інноваційна діяльність. Джерелом фінансування інноваційної діяльності в більшій мірі є інвестиційні ресурси, що сприяють зростанню інновацій або, навпаки, суттєво обмежують можливість стійкого довгострокового розвитку країни. Відсутність інвестиційного забезпечення створює умови для послаблення позицій України на світових ринках і зниження рівня її конкурентоспроможності. Оскільки у світі йде боротьба між країнами за залучення інвестицій, нагальним стає питання щодо створення сприятливого інвестиційного клімату в країні. Левчук Н.І., Фащевська О.М [96] зазначають, що досвід країн ЦСЄ (Центральної та Східної Європи) показує, що державна політика зі створення ефективного інвестиційного середовища передусім має передбачати визначення пріоритетних (цільових) секторів і видів діяльності відповідно до наявних в Україні конкурентних переваг, розвиток яких створить базис для здійснення структурних трансформацій економіки, оптимізації її структурних пропорцій відповідно до тенденцій розвитку світового господарства, і, таким чином, забезпечить економічне зростання у нестабільному глобальному середовищі. Науковці зазначають, що до цільових секторів державної підтримки відносять, як правило, новітні види діяльності і виробництва останніх технологічних укладів [96, с. 170]. У табл. 22.1 представлені пріоритетні сектори розвитку у країнах ЦСЄ.

Таблиця 22.1

### Пріоритетні сектори розвитку країн ЦСЄ

Країна	Сектори
Чехія	Біотехнології, виробництво устаткування, відновлювальні джерела енергії, чисті технології, автомобільну промисловість, електроніку, літакобудування. Секторами з високим потенціалом розвитку визнано: ІКТ, виробництво програмного забезпечення, науки про життя.
Польща	Автомобілебудування, авіаційну промисловість, біотехнології, дослідження та розробки, ІТ, електроніку.
Угорщина	ІКТ, науки про життя та фармацевтику, автомобільну промисловість, туристичну індустрію.
Словаччина	Автомобільну промисловість, електроніку, ІКТ та зв’язок, туристичну індустрію, розвиток мережі сервісних, технологічних і кол-центрів.
Словенія	Стратегічні послуги (регіональні штаб-квартири, логістичні та розподільні центри, загальні сервісні центри), дослідження і розробки, відродження підприємств у галузях деревообробки, виробництво продуктів харчування і напоїв, хімічній і фармацевтичній промисловості, виробництва гумових та пластмасових виробів, виробництві готових металевих виробів, електрообладнання, машини і обладнання, автомобілів.

Орієнтація на інновації та розвиток сектору ІТ забезпечує економічне зростання всіх розвинених і деяких країн, що розвиваються (Японія, Китай, Ізраїль, Малайзія, Індія). Саме уряди цих держав були основними інвесторами у сфері високих технологій, а також здійснювали підтримку даної галузі в формі преференцій, створення прозорих умов для ведення бізнесу, всебічної підтримки малого та середнього бізнесу [6].

Якщо враховувати кадровий, науковий потенціал вітчизняної економіки, рівень привабливості окремих її галузей для вітчизняних та іноземних інвесторів, прогнозований характер розвитку глобальних тенденцій інвестування можна стверджувати, що серед пріоритетних галузей можна виділити ІТ галузь, що підтверджується у Законі України “Про Основні засади розвитку інформаційного суспільства в Україні на 2007-2015 роки”, в якому зазначається, що розвиток інформаційного суспільства в Україні та впровадження новітніх ІКТ в усі сфери суспільного життя і в діяльність органів державної влади та органів місцевого самоврядування визначається одним з пріоритетних напрямів державної політики.

Сучасний світ характеризується швидким розвитком інформаційних технологій, які впливають на всі галузі світової економіки, що сприяє загальному зростанню економічних та соціальних показників організацій. Тому розуміння важливості ІТ для формування стійких конкурентних позицій вкрай необхідно.

Проникнення інформаційних технологій у суспільство дозволяє говорити про поширення та розвиток єдиного інформаційного простору. Застосування ІТ у всіх сферах людського життя є одним із головних факторів підтвердження переходу людства до інформаційного суспільства. Тому поростає необхідність у проведенні регулярної оцінки рівня розвитку інформатизації як в світі, так і в Україні. Оцінкою процесу інформатизації у світі та його впливу на конкурентоспроможність окремих країн займаються різні організації та інститут, серед яких слід виділити Всесвітній Економічний Форум, Всесвітній банк, Організація Об'єднаних Націй, Міжнародний союз електрозв'язку, який є спеціалізованим підрозділом ООН у сфері ІКТ. Для оцінки стану розвитку інформаційного суспільства цими організаціями та інститутами було розроблено значну кількість ІКТ-індексів, кожен з яких ґрунтується на обраних пріоритетах дослідження. Оцінка проводиться шляхом експертного опитування, а також на основі статистичного матеріалу цих країн. Україна також бере участь у цих оцінках й опитуваннях, тому дослідження ступеня розвитку ІС в Україні доцільно проводити саме на основі цих показників з метою визначення місця України у світі.

ІТ сфера, на відміну від традиційної економіки, заснована на людських ресурсах, ІКТ та мережах. Тому серед індексів, що характеризують розвиток ІС в Україні та її позиції серед країн світу, виступають наступні:

- 1) Глобальний індекс конкурентоспроможності (WEF Global Competitiveness Index);
- 2) Індекс технологічної готовності (WEF Technological Readiness Index);
- 3) Індекс мережевої готовності (WEF Networked Readiness Index);
- 4) Індекс електронного уряду ООН (UN e-Government Index).

Стратегією розвитку інформаційного суспільства в Україні [75] визначалися планові показники розвитку показники за даними індексами, які у 2020 році передбачають суттєве підвищення позицій України у міжнародному рейтингу інформатизації, але на сьогоднішній день спостерігається значне відхилення в сторону погіршення фактичних показників від планових (табл. 22.2).

Таблиця 22.2

### Місце України в міжнародних рейтингах розвитку інформаційного суспільства

Найменування індексу	Значення індексу за роками (план), місце в рейтингу			Значення індексу за роками (факт), місце в рейтингу					Відхилення фактичного показника від планового	
	2013	2015	2020	2012	2013	2014	2015	2016	2013	2015
1. Глобальний індекс конкурентоспроможності (WEF Global Competitiveness Index)	70	65	50	73	84	76	79	85	↓ 14	↓ 14
2. Індекс технологічної готовності (WEF Technological Readiness Index)	80	65	50	81	94	85	86	85	↓ 14	↓ 25
3. Індекс мережевої готовності (WEF Networked Readiness Index)	73	65	60	75	73	81	71	64	0	↓ 6
4. Індекс електронного уряду ООН (UN e-Government Index)	65	60	50	68	-	87	-	62	-	-

Індекс глобальної конкурентоспроможності оцінює здатність економік забезпечувати більш високу продуктивність бізнесу, і як наслідок, більш високі темпи економічного зростання й економічного добробуту нації. Індекс формується на основі 12 субіндексів, складається з більш ніж 100 змінних, які становлять 3 основні групи: “Базові умови розвитку”, «Підсилюючі продуктивності (фактори-каталізатори)” та “Інновації та фактори вдосконалення”. Світовий досвід показує, що конкурентоспроможність національної економіки в цілому пов’язана з розвитком ІКТ. Зокрема, за даними Всесвітнього економічного форуму WEF, індекс конкурентоспроможності економіки держави має високий рівень кореляції з індексом розвитку інформаційно-комунікаційних технологій в країні [122].

Вже багато років поспіль рейтинг очолюють Швейцарія та Сінгапур. У 2016 році до десятки найконкурентоспроможніших також увійшли США, Німеччина, Нідерланди, Японія, Гонконг, Фінляндія, Швеція, та Великобританія.

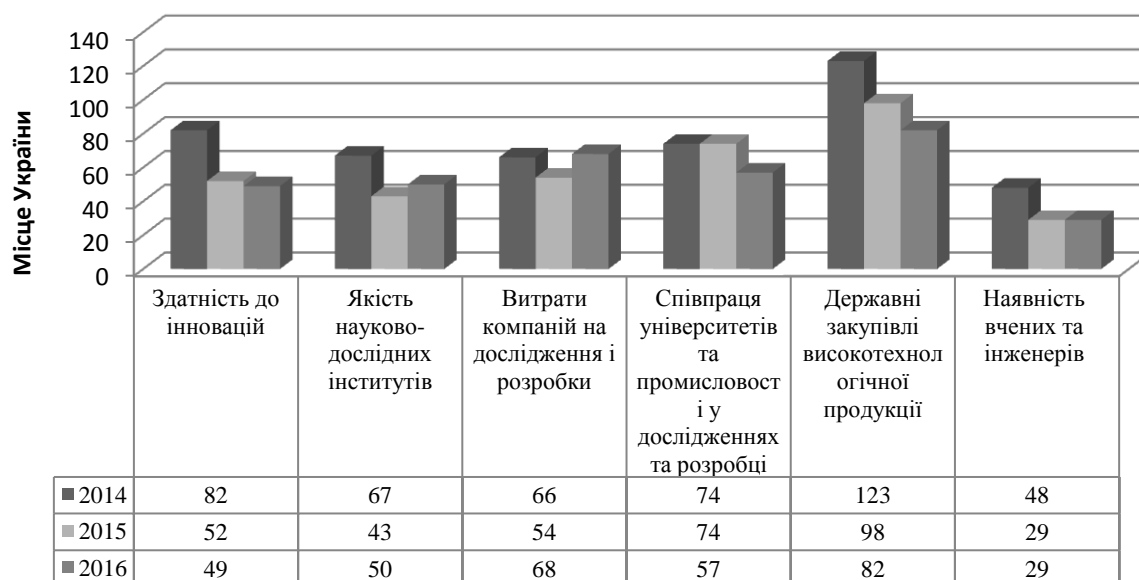
Україна же показує негативну тенденцію та у 2016 році втратила вже 12 позицій у порівнянні з 2012 роком. Щодо відхилення фактичних показників від планових теж спостерігається негативна динаміка: відхилення від планового показника на 14 пунктів і в 2013 році, і в 2015 році.

Згідно з даними дослідження, Україна у 2015 році найбільше втратила позицій за показниками, які характеризують розвиток інфраструктури, макроекономічне середовище та розвиток фінансового ринку країни. А у 2016 році погіршила свої позиції у 7 з 12 основних показників. Найбільше нами втрачено (мінус 17 пунктів) за складовою “Ефективність ринку праці”, по 9 пунктів втрачено за показниками, що характеризують розвиток фінансового ринку країни та охорону здоров’я і початкову освіту. Пасемо задніх ми із розвитком інфраструктури та іншими складовими Індексу глобальної конкурентоспроможності (табл. 22.3).

Причому, за деякими позиціями країна «очолює» списки з кінця, зокрема, у 2015 році – за міцністю банків нам присвоєно 140 місце зі 140 країн світу; за регулюванням фондових бірж – 135, за інфляційними змінами – 134. У 2016 році – за міцністю банків – 138 місце серед 138 країн світу, за регулюванням фондових бірж – 137, ефективність антимонопольної політики – 136 місце, за інфляційними змінами – 136 місце, за якістю доріг – 134 місце. Щодо показників інноваційної діяльності Україна займає досить 52 місце у 2016 році. Місце України по показникам субіндекса “Інновації” серед 138 країн наведено на рис. 22.1.

**Позиції України за показниками Індекса глобальної конкурентоспроможності у світі**

Індекс глобальної конкурентоспроможності (позиції України за основними складовими)	2015 (140 країн)	2016 (138 країн)		Тренд (5 останніх досліджень)
		місце в рейтингу	Індекс	
	79	85	4,0	
<b>БАЗОВІ ВИМОГИ РОЗВИТКУ</b>	101	102	4,0	
Інституції	130	129	3,0	
Інфраструктура	69	75	3,9	
Макроекономічне середовище	134	128	3,2	
Охорона здоров'я та початкова освіта	45	54	6,0	
<b>ПІДСИЛЮЮЧІ ПРОДУКТИВНОСТІ</b>	65	74	4,0	
Вища освіта і професійна підготовка	34	33	5,1	
Ефективність ринку товарів	106	108	4,0	
Ефективність ринку праці	56	73	4,2	
Розвиток фінансового ринку	121	130	3,0	
Технологічна готовність	86	85	3,6	
Розмір ринку	45	47	4,4	
<b>ІННОВАЦІЇ ТА ФАКТОРИ ВДОСКОНАЛЕННЯ</b>	72	73	3,5	
Відповідність бізнесу сучасним вимогам	91	98	3,6	
Інновації	54	52	3,4	

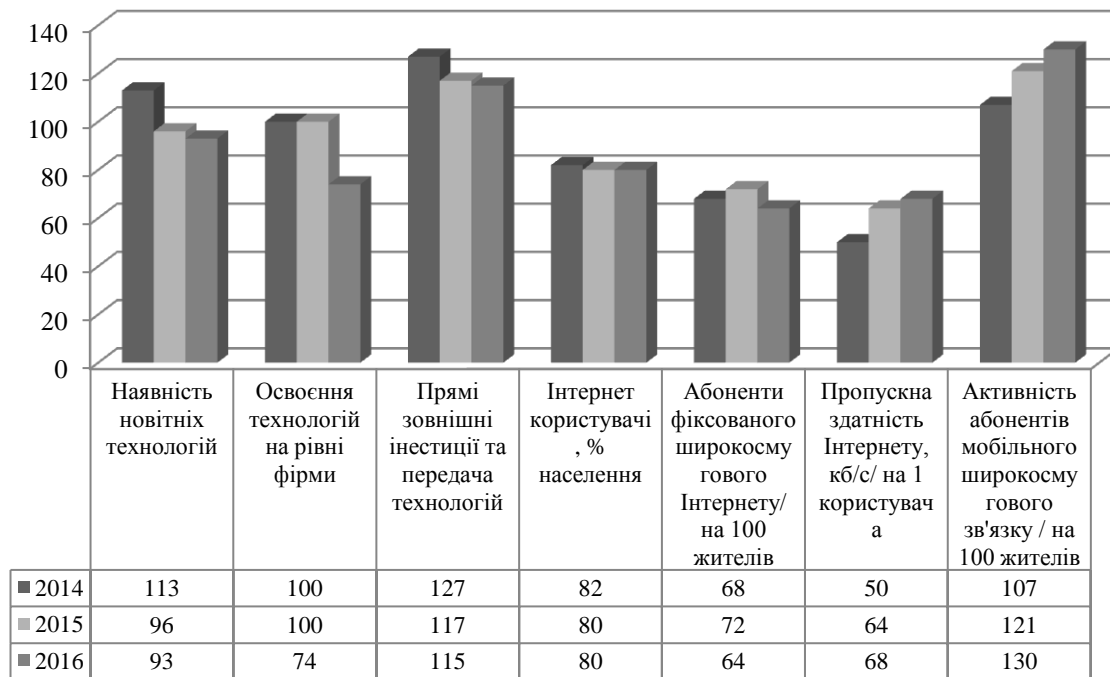


**Рис. 22.1. Позиції України за показниками інновацій у розрізі індексу глобальної конкурентоспроможності за 2014-2016 рр.**

У розрізі субіндекса “Інновації” майже по всіх показниках у 2016 році спостерігається покращення позицій, окрім показників витрат компаній на дослідження і розробки (падіння на 14 позицій у порівнянні з попереднім роком) та якості науково-дослідних інститутів (падіння на 7 позицій у порівнянні з попереднім роком). За наявністю вчених і інженерів Україна займає 29 місце, за потенціалом для створення інновацій – 49 місце, що говорить про наявність значного потенціалу створення інновацій та виходу національної економіки на новий етап розвитку. Покращенню результатів інноваційної діяльності може посприяти вступ України до ЄС, оскільки це дає можливість залучитися до Стратегічного форуму з міжнародного співробітництва в галузі науки і технологій (SFIC). Участь у SFIC дозволяє обмінюватися інформацією про науково-технічне співробітництво, створювати мережі наукових радників та отримувати регулярні консультації щодо визначення загальних пріоритетних напрямів розвитку, об'єднувати інформацію про науково-технічні ресурси. Також ключовим завданням інноваційної стратегії ЄС до 2020 р. є побудова єдиного європейського ринку знань та інновацій з метою створення умов для розвитку ІС і підвищення конкурентоспроможності відносно США та інших країн, тому залучення України до ініціатив країн ЄС дасть змогу поліпшити свої показники у світових рейтингах.

Істотним для нашого дослідження є також більш детальний аналіз субіндексу технологічної готовності, за яким Україна посіла 85 місце у 2016 році та піднялась всього на одну позицію в порівнянні з попереднім роком, але має відхилення на 14 пунктів від планового показника у 2013 році, та на 25 пунктів – у 2015 році. Цей показник акумулює проникнення та використання ІКТ та має в своєму складі ряд показників, що його характеризують. Аналіз показників технологічної готовності наведено на рис. 22.2.

Зміни в бік покращення субіндексу спричинені збільшенням рейтингу за показниками наявності нових технологій та їх трансферу, а також освоєнням технологій на рівні фірми та абонентів фіксованого широкосмугового Інтернету. Але є і погіршення таких показників, як пропускна здатність Інтернету та активність абонентів широкосмугового зв'язку. Нажаль вже не один рік Україна переходить від стадії розвитку, що керується базовими факторами (дешева сировина та некваліфікована робоча сила), до стадії, керованої ефективністю (ефективність ринків та інституцій, ефективність впровадження інноваційних ІКТ, тощо) [122]. Тому необхідно стимулювати розробку, поширення та використання ІКТ у всіх сферах життєдіяльності з метою довготермінової конкурентоспроможності та зростання добробуту населення.



**Рис. 22.2. Позиції України за показниками технологічної готовності за 2014-2016 рр.**

Окрім низького рівня глобальної конкурентоспроможності та, зокрема, технологічної готовності, Україна посіла 64 місце у 2016 році за індексом мережевої готовності (Networked Readiness Index) серед 139 країн, що досліджувались. Цей комплексний показник характеризує рівень розвитку ІКТ в країні і вважається найповнішим і авторитетнішим джерелом міжнародної оцінки впливу ІКТ на конкурентоспроможність і добробут країни. Індекс мережевої готовності оприлюднюється з 2002 року Всесвітнім економічним форумом та міжнародною школою бізнесу INSEAD в межах щорічних доповідей про розвиток інформаційного суспільства в країнах світу. Для розрахунку індексу використовують дані національних міністерств ІКТ, органів електрозв'язку і національних статистичних управлінь. Індекс вимірює рівень розвитку ІКТ за чотирма субіндексами, що розподілені на 10 складових і 53 змінних (рис. 22.3).

Індекс мережевої готовності (WEF Networked Readiness Index)	Середовище (Environment)	Політичне та регуляторне (Political and regulatory)
		Бізнесу та інновацій (Business and innovation)
	Готовність (Readiness)	Інфраструктура (Infrastructure)
		Доступність (Affordability)
		Навички населення (Skills)
	Використання (Usage)	Індивідуальне (Individual)
		Бізнес (Business)
		Державне (Government)
	Вплив (Impact)	Економічний (Economic)
		Соціальний (Social)

**Рис. 22.3. Складові Індексу мережевої готовності (WEF Networked Readiness Index)**



В цілому спостерігається позитивна динаміка змін. Україна пересягнула середину у рейтингу країн по даному показнику у 2016 році та у 2013 році плановий показник відповідав фактичному, але у 2015 році спостерігається відхилення від планового показника на 6 пунктів в негативний бік. Отже рейтинг України залишається все ж таки низьким, що свідчить про недостатнє використання потенціалу інформаційних технологій на всіх рівнях. Фахівці Всесвітнього економічного форуму WEF [123] відзначають, що уряд України не має чіткого плану впровадження і використання ІКТ для підвищення конкурентоспроможності країни. Відсутність упевненості в захисті своїх прав і капіталовкладень, корупція і “зарегульованість” багатьох процесів, а також нестабільність українського ринку, відлякують інвесторів, які могли б розповсюдити в країні кращі практики з розвитку інформаційно-комунікаційних технологій.

Комплексний показник Індекс розвитку електронного уряду (United Nations Global E-government Development Index) оцінює готовність і можливості національних державних структур до використання інформаційно-комунікаційних технологій (ІКТ) для надання громадянам державних послуг. Індекс акумулює дані про рівень розвитку електронного уряду в країні та оцінку тенденцій у використанні ІКТ державними установами за трьома основними складовими, що характеризують стан людського капіталу, ІКТ-інфраструктуру та веб-присутність органів державної влади. Перші два ґрунтуються на офіційних статистичних даних, а останній визначається за результатами обстеження веб-сайтів уряду і п'яти міністерств – фінансів, охорони здоров'я, освіти, праці, соціального забезпечення. Він публікується раз в два роки Департаментом по економічних і соціальних питаннях ООН.

За Індексом розвитку електронного уряду Україна у світі опинилася на 62 сходинці серед 190 країн, продемонструвавши позитивну тенденцію в порівнянні з показником 2012 року – 87 місце серед 190 країн (покращення на 25 пунктів). Вказана динаміка характеризує відповідні зусилля нашої держави на шляху побудови інформаційного суспільства.

Загалом можна зробити висновок, що протягом останніх років Україна у міжнародних рейтингах за основними показниками індексів розвитку інформаційного суспільства утримує позиції середні/нижче середніх, і, до того ж, відстає за темпами розвитку. Але динаміка розвитку протягом останніх років є стабільно позитивною, і це можна вважати сталою тенденцією. Зокрема, є всі підстави стверджувати, що в коротко- й середньостроковій перспективах на теренах нашої держави триватимуть процеси повномасштабної імплементації інформаційних технологій, але для цього ще треба багато чого зробити, зокрема, звернути увагу на активне запровадження інновацій та підготовку висококваліфікованого та творчого персоналу для створення цих інновацій. Вітчизняна практика розбудови інформаційного суспільства залишається

здебільшого фрагментарною, ситуативною і поки що значно більше залежить від ринкової кон'юнктури й суспільних ініціатив, аніж від рішень і діяльності органів державної влади. Невирішеним залишається цілий комплекс проблем нормативно-правового, організаційно-технічного, науково-методичного, інформаційно-аналітичного, ресурсного забезпечення розбудови інформаційного суспільства [21].

Кабінет Міністрів України затвердив Національну систему індикаторів розвитку інформаційного суспільства в Україні [75]. На її основі проведені розрахунки індексу RITS [20, с. 82], що характеризує рівень проникнення інформаційних технологій в життєдіяльність суспільства по регіонах України (табл. 22.4).

Таблиця 22.4

**Індекс RITS рівня проникнення ІТ в життєдіяльність суспільства по регіонах України**

Регіон	Рейтинг за індексом	Індекс RITS	Субіндекс рівня розвитку ІТ	Субіндекс рівня використання ІТ
м. Київ	1	52,76	30,93	21,84
Запорізька	2	41,64	22,77	18,87
Одеська	3	38,93	18,59	20,34
Івано–Франківська	4	38,16	20,41	17,75
Хмельницька	5	36,45	17,83	18,62
Черкаська	6	36,07	19,70	16,37
Донецька	7	35,97	17,56	18,41
Сумська	8	35,43	19,65	15,78
Миколаївська	9	35,39	19,28	16,11
м. Севастополь	10	35,37	20,19	15,18
Рівненська	11	34,72	17,29	17,43
Харківська	12	34,21	16,29	17,92
Кіровоградська	13	34,12	16,76	17,36
Чернігівська	14	33,26	15,80	17,46
Житомирська	15	33,24	17,24	16,01
Полтавська	16	32,98	16,65	16,32
Дніпропетровська	17	32,85	15,98	16,87
Львівська	18	32,68	16,13	16,55
Волинська	19	32,58	17,21	15,37
Вінницька	20	31,95	14,30	17,65
Київська	21	31,95	17,00	14,95
Автономна Республіка Крим	22	31,69	15,46	16,23
Чернівецька	23	31,61	16,40	15,21
Тернопільська	24	31,52	15,12	16,41
Закарпатська	25	30,91	15,69	15,21
Херсонська	26	30,70	15,14	15,56
Луганська	27	30,63	14,82	15,81

Виходячи з аналізу індексу RITS, значення індексу по Києву майже вдвічі перевищує значення по більшості областей. Даний індекс дає змогу зрозуміти на які регіони може орієнтуватися державна політика, за даним показником можна визначати “вузькі місця” регіонів та спрямовувати конкретні напрями розвитку в відповідні регіони. Також треба враховувати, що ця політика повинна включати заходи з залучення населення та підприємців до використання ІТ, оскільки значна частка населення не сприймає ІТ для себе, остерігається інновацій, тощо. Розвиток ІТ-галузі є значним підґрунтям інноваційної діяльності країни, що дає змогу вийти на передові позиції в світі, таким чином, серед компонентів державної політики мають бути представлені цілі та завдання фінансового, економічного, соціального, ресурсного, особистісного розвитку, а не тільки завдання розбудови інформаційного суспільства.

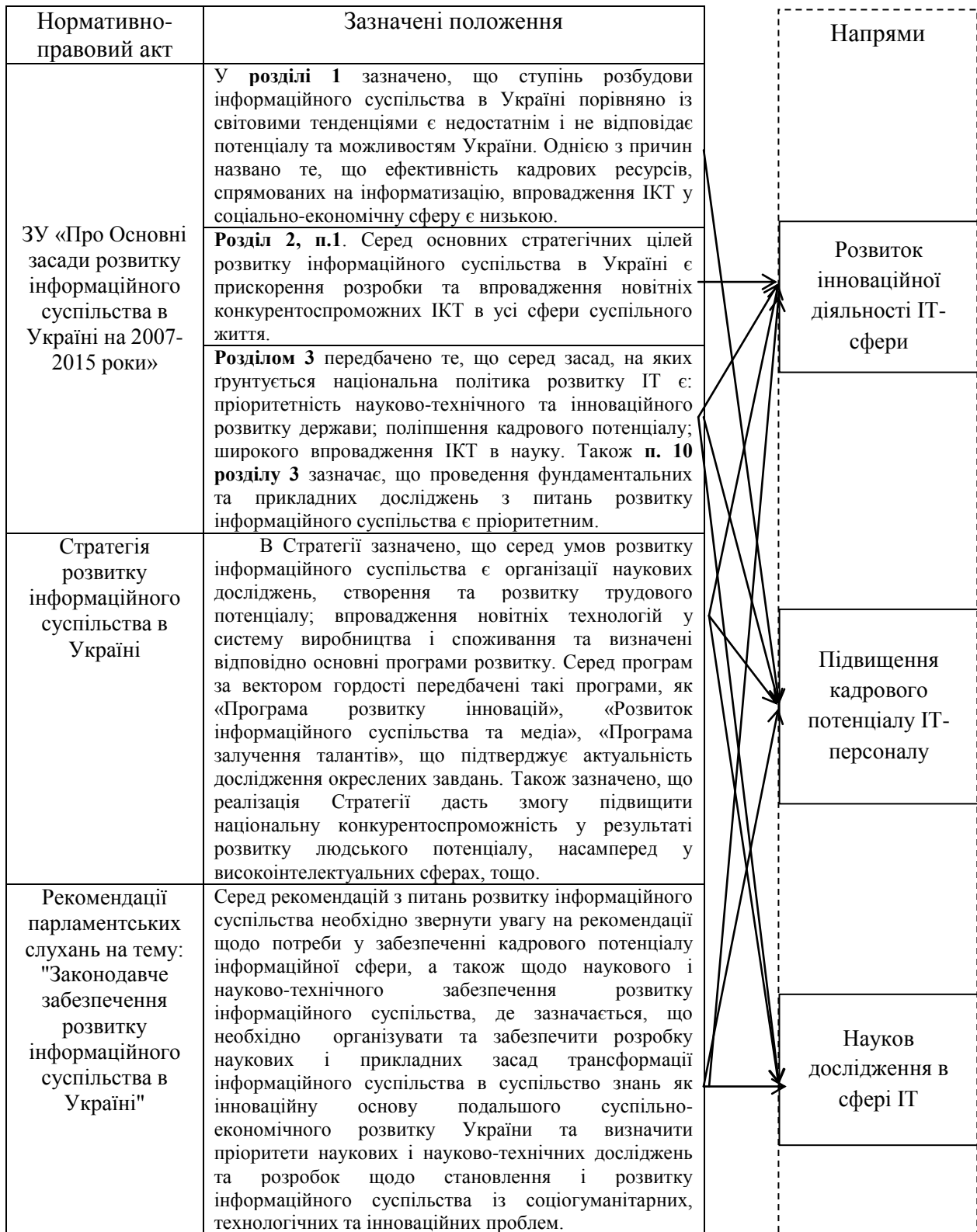
ІТ входить в число найбільш швидкозростаючих галузей сучасної світової економіки. При цьому зростання як національних, так і міжнародних компаній даної галузі найбільшою мірою серед усіх високотехнологічних галузей економіки, забезпечується за рахунок двох ендогенних факторів - інновацій та маркетингу.

Специфіка ІТ-галузі полягає в тому, що основним ресурсом створення інновацій є персонал, а частка матеріальних витрат у загальному обсязі інвестицій, що спрямовуються на дослідження і розробку нових продуктів, незначна, і як наслідок від ефективності науково-дослідної роботи співробітників, їх винахідливості так креативності безпосередньо залежить комерційний успіх компанії і її положення на ринку.

Законом України “Про Основні засади розвитку інформаційного суспільства в Україні на 2007-2015 роки” [74], Стратегією розвитку інформаційного суспільства в Україні до 2020 р. [75], та іншими нормативно-правовими актами визначено необхідність розвитку інформаційного суспільства, орієнтованого на інтереси людей, відкритого для всіх, в якому кожна людина може створювати і накопичувати інформацію та знання, мати до них вільний доступ, користуватися і обмінюватися ними, мати можливість повною мірою реалізувати свій потенціал, сприяти суспільному і особистісному розвитку та підвищувати якість життя.

Вивчення законодавчої сторони інформатизації суспільства України дало змогу зрозуміти актуальність проведення наукових досліджень з питань розвитку кадрового потенціалу ІТ-спеціалістів та інноваційної діяльності в сфері інформаційних технологій, оскільки у більшості нормативних актах серед пріоритетних завдань стоять проведення наукових досліджень в сфері ІТ, прискорення розробки та впровадження новітніх конкурентоспроможних ІКТ в усі

сфери суспільного життя, а також забезпечення висококваліфікованого кадрового потенціалу ІТ-персоналу. Результати даного аналізу наведені на рис. 22.4.



*Рис. 22.4. Пріоритетні напрями розвитку ІТ-галузі*

Виходячи з результатів дослідження, наведених на рис. 22.4, нарощування науково-технічного потенціалу та його зміцнення, сприяння створенню нових товарів і послуг на початку XXI століття стають найважливішими стратегічними завданнями економічної політики будь-якої держави. Зміни в технології та умовах господарювання висунули нові вимоги до функцій, завдань і кваліфікації фахівців з інформаційно-комунікаційних технологій. Підприємства, що працюють в умовах динамічного розвитку економіки, відчувають нагальну потребу в професіоналах в області інформаційних систем, здатних не тільки проектувати і розробляти ці системи, але і творчо підходити до управління їх інфраструктурою. Оскільки формування високо компетентного ІТ-персоналу є ключовим фактором технологічної конкурентоспроможності, необхідно краще розуміти потреби господарюючих суб'єктів в якісних показниках розвитку людських ресурсів даного сектора діяльності. Одним із суттєвих якісних показників розвитку ІТ-персоналу сучасності є творчий потенціал. Важливість талантів в сьогоденному глобалізованому, мобільному, заснованому на знаннях світі є безперечною.

Існують потужні причини того, чому креативність є першочерговою формою економіки у двадцять першому столітті. Перша причина ховається у еволюції потреб людини відповідно піраміди Маслоу. За останні десятиліття потреби людини значно здвинулись від функціональних і практичних до розуміння благополучного життя та особистісної реалізації. Відповідно, оскільки такі потреби набувають актуальності, розвивається і ринок для задоволення цих потреб. З'явилися нові індустрії на основі нових інформаційних технологій, в яких існує нагальна потреба в компетенціях та ідеях, тобто інформаційні технології посприяли зародженню креативної економіки.

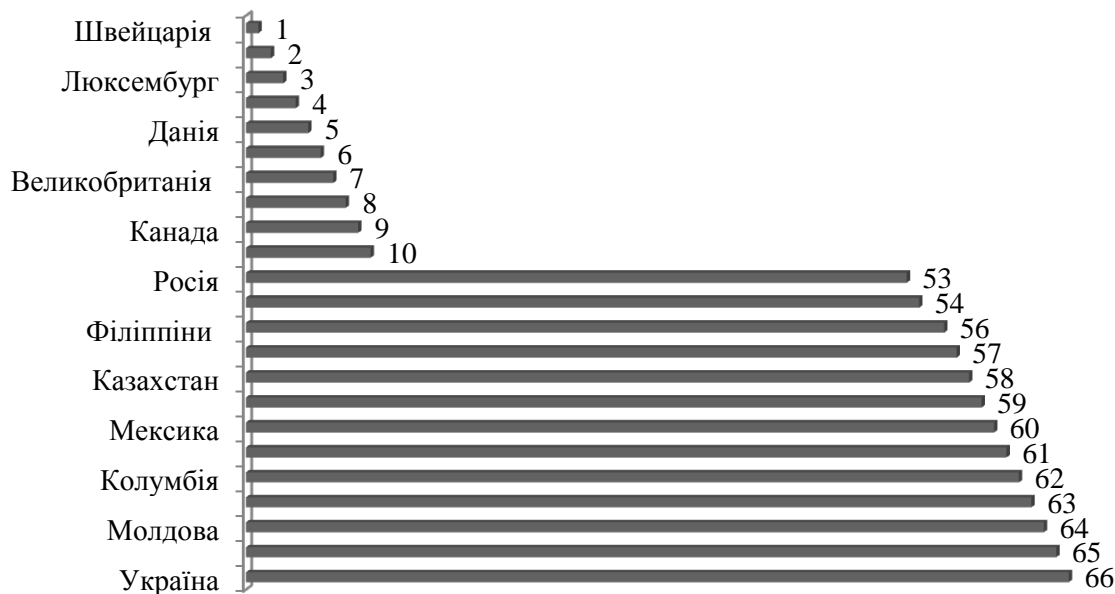
Успіх компаній в ІТ-індустрії часто визначається не передовими технологіями, не волею керівника і навіть не інвестиціями в розробку і маркетинг, а кадровою політикою. Грамотний творчий персонал часом є головною перевагою в конкурентній боротьбі. Особливо це видно на прикладі таких компаній, які спеціалізуються на вирішенні нетипових завдань.

Як вже зазначалось, що ІТ-сфера, на відміну від традиційної економіки, заснована на людських ресурсах. Тому в межах розвитку кадрового потенціалу як пріоритетний напрям розвитку ІТ-галузі необхідно вважати і розвиток талантів. Для оцінки розвитку талантів в світі існує Індекс глобальної конкурентоспроможності талантів. Мета цього індексу – оцінити ресурси і зусилля, які країни витрачають на розвиток талантів, а також якість людського капіталу. Сумарний “індекс талантів” розраховується на основі шести критеріїв: ринкові і нормативні умови на ринку

праці; шанси для кар'єрного зростання; можливості роботодавців залучати таланти з усього світу; здатність утримувати кваліфікований персонал; виробничі навички співробітників і глобальні знання.

Країни, які мають високий показник конкурентоспроможності талантів, також демонструють і високий розвиток ІТ, тому є істотна кореляція між наявністю талантів та інформаційним процвітанням.

Рейтинг за 2015 – 2016 рр. Топ-10 країн світу, за якими був розрахований Індекс глобальної конкурентоспроможності талантів (всього увійшли до рейтингу 109 країн), та країн, що передують Україну представлений на рис. 22.5.



**Рис. 22.5. Рейтинг України за Індексом глобальної конкурентоспроможності талантів за 2015-2016 рр.**

За результатами звітності 2015-2016 року Україна за індексом глобальної конкурентоспроможності талантів зайняла 66 місце серед 109 країн світу та 37 місце серед країн Європи. Передують Україну такі країни, як Молдова, Колумбія, Мексика, Казахстан, Філіппіни, Росія. За даним показником у світовому рейтингу Україна знаходиться нижче середнього рівня, але все ж у 2015-2016 рр. спостерігається покращення показника на 5 пунктів у світовому рейтингу та на 4 пункти у рейтингу країн Європи у порівнянні з 2014 роком (у 2014 р. – 71 місце серед 93 країн світу, 33 місце серед країн Європи). Дані показники свідчать про те, що Україна є аутсайдером серед інших країн та потребує значної уваги щодо змін і розроблення необхідних програм покращення розвитку талантів як на рині країни, так і у бізнес середовищі. Це в свою чергу дасть змогу нашій державі покращити свої позиції у світових рейтингах. Сьогодні ІТ-сфера неможлива без людини, яка є креативною та спроможною генерувати нові ідеї, що сприяють розвитку інновацій в країні та

виводять її на світову арену конкурентоспроможною, як в сфері інформаційних технологій, так і у інших сферах народного господарства.

**Висновок.** Результати аналізу позицій України у світовому рейтингу за глобальними індексами розвитку ІТ-галузі можна зазначити, що на сьогоднішній день спостерігається за більшістю індексів покращення позицій, але від встановлених урядом планових показників Україна значно відстає за всіма індексами. Позитивна динаміка змін ІТ-галузі та зростання конкурентоспроможності зумовлена тим, що світовий ринок диктує умови інтенсивного використання знань, прискорення інформатизації, науково-технологічного та інноваційного розвитку, що сприяє покращення рівня соціально-економічного розвитку, заснованого на процесах інформатизації. У період активних процесів євроінтеграції державна політика з питань розвитку інформаційних технологій повинна узгоджуватися з політикою країн-членів ЄС. У цьому контексті актуальною є проблема вивчення досвіду становлення ІС у країнах-членах ЄС. З погляду на це, в найближчій перспективі Україні необхідно:

1) переглянути чинне законодавство з питань розвитку ІТ-галузі орієнтуючись на досвід країн ЄС;

2) гармонізувати національну систему індикаторів розвитку ІС із відповідними показниками ЄС;

3) стимулювати проведення фундаментальних та наукових досліджень в цій сфері;

4) звернути увагу на розвиток інноваційної діяльності та кадрового потенціалу, зокрема, на створення програм розвитку талантів у сфері ІТ, що є засадою підвищення конкурентоспроможності України у світових рейтингах;

5) здійснити подальший розвиток національної та регіональних інформаційних систем;

6) поширити впровадження інформаційних систем у всі сфери життя суспільства.

Проблеми, які стримують темпи розвитку ІС в Україні, можуть бути вирішені, але євроінтеграція вимагає реформи майже у всіх сферах суспільного життя, які потребують інноваційної діяльності, інформаційного забезпечення, висококваліфікованого креативного персоналу та постійних наукових досліджень, що значною мірою вплине на формування повноцінної ІТ-галузі.

## РОЗДІЛ 23

# ОПРЕДЕЛЕНИЕ СТОИМОСТИ РАЗРАБОТКИ ВНЕДРЯЕМЫХ ТЕХНОЛОГИЙ ЭЛЕКТРОННОГО БИЗНЕСА НА ПРОМЫШЛЕННОМ ПРЕДПРИЯТИИ

*Аннотация.* В работе на основе авторских методических рекомендаций проведены расчеты стоимости разработки внедрения технологий электронного бизнеса на промышленном предприятии. Для упрощения расчетов разработана программа в среде Java.

**Ключевые слова:** программное обеспечение (ПО), технологии электронного бизнеса (ТЭБ), стадии разработки, эффективность.

*Annotation.* In work on the basis of copyright guidelines calculated the cost of the development of the introduction of e-business technologies at an industrial plant. To simplify the calculations, a program in Java.

**Keywords:** software (software), e-business technology (EBT), stage of development, efficiency.

Внедрение прогрессивных технологий на промышленных предприятиях ведет к модернизации элементов производственной системы другими, более современными. Тем не менее любое внедрение требует значительных затрат денежных средств, поэтому руководству необходимо обдуманно подойти к этому вопросу, т.е. принять решение каким образом будет происходить это внедрение. Оно может быть выполнено такими способами, как покупка и обслуживание у сторонней организации и собственная разработка. Вопросы выбора и внедрения у сторонней организации были рассмотрены в работах [78; 12]. Внедрение ПО собственными силами будет рассмотрено в этой работе.

Вопросами оценки программного обеспечения занималось большое количество ученых: Е.В. Ваганова, А.А. Земцов, С.Л. Миньков, А.М. Гудова, С.Ю. Завозюкина, С.Н. Трофимов, Т.Н. Бебриш, Ю.Ю. Якунин. В работах этих ученых рассмотрены методы, модели оценки стоимости программного обеспечения (ПО). Несмотря на большое количество разработок до сих пор не существует единого стандарта по расчету стоимости ПО.

В разделе проведены, по разработанным авторским методическим рекомендациям, расчеты стоимости и эффективности внедрения технологий электронного бизнеса (ТЭБ) [10].

Внедрение ТЭБ на предприятии проводится на основе 5-ти стадий: Техническое задание (ТЗ); Эскизный проект (ЭП); Технический проект (ТП) Рабочий проект (РП) Внедрение (ВН).



## 1. Определение трудоемкости разработки ТЭБ и отдельных стадий.

Выбранная ТЭБ имеет 17690 условных машинных команд и имеет группу сложности – 2. Таким образом, коэффициент сложности,  $T_p = 828$  чел./дней.

Коэффициент сложности:  $K_i = 0,08 + 0,06 = 0,14$ ;  $K_{СЛ} = 1 + 0,14 = 1,14$

Тогда  $T_0 = 828 * 1,14 = 944$  чел./дней.

Поскольку разрабатываемая ТЭБ имеет категорию новизны А и разрабатывается без применения CASE-технологии значения коэффициентов удельных весов составляет  $L_j$ :  $L_1 = 0,12$ ;  $L_2 = 0,21$ ;  $L_3 = 0,32$ ;  $L_4 = 0,26$ ;  $L_5 = 0,09$ .

Поскольку при создании ТЭБ не использовалось новый тип ПК и ТЭБ не создавалась новой среде ОС,  $K_n = 1,00$ . Степень охвата реализуемых функций стандартными программами составляет от 40 до 60%, тогда  $K_t = 0,65$ .

Трудоемкость каждой стадии разработки ТЭБ составляет:

$T_{ТЗ} = 0,12 * 1,00 * 944 = 113$  чел./дней.

$T_{ЕП} = 0,21 * 1,00 * 944 = 198$  чел./дней.

$T_{ТП} = 0,32 * 1,00 * 944 = 302$  чел./дней.

$T_{РП} = 0,26 * 1,00 * 0,65 * 944 = 160$  чел./дней.

$T_{ВН} = 0,09 * 1,00 * 944 = 85$  чел./дней.

Тогда уточнена общая трудоемкость разработки ТЭБ составляет:

$T_{ОБЩ} = 113 + 198 + 302 + 160 + 85 = 858$  чел./дней.

## 2. Определение состава группы исполнителей разработки

$T_{ОБЩ} = 858$  чел.-дней.,  $\Phi_d = 22$  дня,  $D = 6$  месяцев.

Таким образом, количество исполнителей необходимых для разработки

ТЭБ составляет:  $Ч = \frac{858}{22 * 6} = 7$  чел.

В дальнейшем составим штатное расписание сотрудников, табл. 23.1.

Таблица 23.1

### Штатное расписание сотрудников

Наименование должности	Количество, чел.	Заработная плата, грн.
Научный руководитель	1	6500
Главный инженер	1	5100
Инженер-программист	2	4500
Оператор	2	3500
Техник	1	3100

### 3. Расчет и построение сетевого плана-графика выполнения разработки ТЭБ

Построим сетевой план-график выполнения разработки и внедрения ТЭБ. Весь процесс разработки разделим на работе и установим их логическую последовательность, определим исполнителей и продолжительность выполнения работ. Перечень работ и их продолжительность приведены в табл. 23.2.

Таблица 23.2

#### Перечень и продолжительность работ

Код работы	Склад работ	Трудоемкость работ, чел.-дней	Исполнители	Продолжительность работы, дней
1	2	3	4	5
<b>Техническое задание</b>		291		
1-2	Постановка задачи	20	2	10
2-3	Обоснование возможности решения задачи	35	4	9
3-5	Предварительный выбор методов решения задачи	70	5	14
2-4	Определение состава ТЭБ	44	4	11
3-6	Определение состава и структуры информационной базы	32	4	8
4-6	Выбор языка программирования	20	3	7
5-7	Определение стадий, этапов и сроков разработки ТЭБ	50	4	12
7-8	Согласования и утверждения ТЗ	20	4	5
<b>Эскизный проект</b>		238		
8-9	Изучение и разработка ТЗ	24	4	6
6-10	Консультации разработчиков	28	5	6
9-11	Установление перечня и состава информации	43	4	11
10-11	Установление структуры и форм входных и выходных документов	48	5	10
11-12	Определение методов решения задач	59	5	12
12-13	Утверждение ЭП	36	4	9
11-14	Выдача рекомендаций исполняющим	9	2	5
14-15	Определение состава и организации обработки данных	38	4	10
13-14	Решение вопросов по применению ТЭБ	23	3	8
15-16	Анализ и определение форм входных и выходных документов с компоновкой реквизитов	35	4	9
14-16	Организация контроля обрабатываемой информации	41	4	10

Окончание табл. 23.2

1	2	3	4	5
16-17	Подготовка контрольно-наладочного примера	32	3	11
17-18	Окончательное определение конфигурации технических средств	29	4	7
17-19	Контроль за соблюдением требований к составу и оформлению постановки задачи	26	3	9
18-20	Разработка программной документации	29	4	7
19-20	Передача программной документации для включения в ТП	12	2	6
20-21	Согласование и утверждение ТП	17	4	4
<b>Рабочий проект</b>		1019		
21-22	Изучение постановки задачи	92	7	13
22-23	Разработка программы	204	6	34
22-25	Настройка программы	183	5	36
22-26	Описание контрольного примера	82	5	16
23-24	Подготовка руководств программиста и оператора	102	5	20
24-27	Комплексная наладка задач и сдача в опытную эксплуатацию	132	6	22
25-26	Корректировка программ и документации по результатам комплексной отладки	71	5	14
26-27	Подготовка документации РП к рассмотрению	61	5	12
27-28	Размножение рабочей документации	31	3	10
28-29	Согласования и утверждения РП	61	5	12
<b>Внедрение</b>		371		
29-30	Проверка алгоритма и программ решения задач	74	6	12
30-31	Проверка технологического процесса обработки данных	74	5	15
30-32	Опытная эксплуатация задач	67	5	13
31-33	Корректировка технической документации по результатам опытной эксплуатации	63	5	13
32-33	Подготовка документации	56	5	11
33-34	Сдача ТЭБ в промышленную эксплуатацию	37	4	9

#### 4. Расчет затрат на разработку ТЭБ

**Расчет стоимости материалов.** Результаты расчетов стоимости материалов подано представлены в табл. 23.3

Таблица 23.3

##### Расчет стоимости материалов

Наименование материала	Цена за ед., грн.	Количество, шт.	Сумма, грн.
Бумага	40,00	2	80,00
Ручка	3,50	15	52,50
Карандаш	2,00	15	30,00
USB - накопитель	85,00	1	85,00
Картридж для принтера	700,00	1	700,00
Папка для документации	10,00	1	10,00
Всего			957,50
Транспортно-заготовительные расходы, 15%			143,63
Всего			1101,13

**Расчет стоимости специального оборудования.** Рассчитаем стоимость специального оборудования, табл. 23.4.

Таблица 23.4

##### Стоимость специального оборудования

Наименование оборудования	Мощность, кВт	Количество	Цена за ед., грн.	Сумма грн.
Ноутбук Dell Inspiron 3737	0,4	3	7019,00	21057,00
Принтер Samsung ML-2545	0,3	1	1255,00	1255,00
Сканер Epson Perfection V33	0,25	1	1161,00	1161,00
Всего				<b>23473,00</b>
Транспортно-заготовительные расходы, 15%				<b>3520,95</b>
Всего				<b>26993,95</b>

Теперь определим расходы, связанные с использованием компьютерного оборудования при разработке ТЭБ.

Стоимость одной машино-часа представляет собой отношение годовой себестоимости одной машино-часа фонда времени работы техники. При пятидневной рабочей неделе с двумя выходными днями общее число рабочих дней в 2016 году составляет – 250, норма рабочего времени в неделю – 40 часов. Соответственно норма рабочего времени за год будет составлять –2000 час.

Среднемесячный номинальный фонд времени равен  $\Phi_H = \frac{2000}{12} = 166$  час.

Таким образом, полезный фонд времени работы оборудования с учетом простоев и других факторов составляет:  $\Phi_{II} = 166 * 6 * (1 - \frac{1}{100}) = 986$  год.

Рассчитаем затраты на электроэнергию,  $C_K = 1,13$  грн.:

- для компьютера  $P_1 = 1,13 * 0,4 * 986 = 445,67$  грн.

- для принтера  $P_1 = 1,13 * 0,3 * 986 = 334,25$  грн.

- для сканера  $P_1 = 1,13 * 0,25 * 986 = 278,55$  грн.

Расходы на техобслуживание техники составляют 25% от ее стоимости, таким образом:

- для компьютера  $P_2 = 7019,00 * 0,25 = 1754,75$  грн.

- для принтера  $P_2 = 1255,00 * 0,25 = 313,75$  грн.

- для сканера  $P_2 = 1161,00 * 0,25 = 290,25$  грн.

По санитарным нормам площадь на одно рабочее место пользователей ПЭВМ с жидкокристаллическим монитором должно составлять  $4,5 \text{ м}^2$ . Поскольку количество ПЭВМ, необходимое для выполнения работ по созданию ПО, равняется 3, площадь помещения должна составлять  $13,50 \text{ м}^2$ .

Расходы на аренду помещения определяются как произведение стоимости аренды 1-го квадратного метра на количество квадратных метров. В нашем случае стоимость аренды  $1 \text{ м}^2$  площади помещения составляет 100 грн., Площадь офисного помещения составляет  $40 \text{ м}^2$ , то, то  $P_3 = 100 * 40 = 4000$  грн.

Налог на имущество составляет 2% от стоимости техники, так:

- для компьютера  $P_4 = 7019,00 * 0,02 = 140,38$  грн.

- для принтера  $P_4 = 1255,00 * 0,02 = 25,10$  грн.

- для сканера  $P_4 = 1161,00 * 0,02 = 23,22$  грн.

Теперь найдем общую сумму текущих расходов для каждого вида оборудования

- для компьютера  $P_T = 445,67 + 1754,75 + 4000 + 140,38 = 6340,80$  грн.

- для принтера  $P_T = 334,25 + 313,75 + 25,10 = 673,10$  грн.

- для сканера  $P_T = 278,55 + 290,25 + 23,22 = 313,47$  грн.

Далее определяем износ оборудования, который связан с эксплуатацией оборудования и помещения.

- для компьютера  $H_a = 10\%$ ,  $P_5 = \frac{10 * 6 * 7019,00}{12 * 100} = 350,95$  грн.

- для принтера  $H_a = 12\%$ ,  $P_5 = \frac{12 * 6 * 1255,00}{12 * 100} = 75,30$  грн.

- для сканера  $H_a = 12\%$ ,  $P_5 = \frac{2 * 6 * 1161,00}{12 * 100} = 100,00$  грн.

Амортизация помещения за год составляет 2% от годовой аренды помещения, то  $P_6 = P_3 * 2\% = 4000 * 0,02 = 80$  грн.

Таким образом, износ оборудования составляет:

- для компьютера,  $P_{ОП} = 350,95 + 80 = 430,95$  грн.

- для принтера,  $P_{ОП} = 75,30$  грн.

- для сканера,  $P_{ОП} = 69,66$  грн.

The screenshot shows a software window titled 'Эксплуатационные расходы' (Operational expenses) with a sub-tab 'Оценка конкурентноспособности' (Competitiveness assessment). The main table lists materials with columns for '№', 'Наименование' (Name), 'Цена' (Price), 'Кол-во' (Quantity), and 'кВт' (kWh). Below the table are buttons for 'Добавить материал' (Add material), 'Удалить материал' (Delete material), and 'Расчитать' (Calculate). A summary table shows 'Наимен. затрат' (Name of expenses) and 'Сумма, грн.' (Sum, UAH). At the bottom, there are input fields for various cost parameters.

№	Наименование	Цена	Кол-во	кВт
1	Ноутбук Dell Inspiron 3737	7 019	5	0,4
2	Принтер Samsung ML-2545	1 255	1	0,3
3	Сканер Epson Perfection V33	1 161	1	0,25

№	Наименование	Цена	Кол-во	Наимен. затрат	Сумма, грн.
1	Бумага	40	2	Затраты на электр.	1058.5139399999998
2	Ручка	3,5	20	Затраты на тех. обл.	9377.75
3	Карандаш	2	15	Затраты на аренду	4000.0
4	Флешка	85	1	Затраты на материа...	975.0
5	Картридж для принтера	700	1	Налог на имущество	188.7
6	Папка для документац...	10	1	Амортизац. расходы	6717.5
				Затраты на спец обо...	37511.0
				Основная зараб. пла...	101600.0
				Дополн. зараб. плата	17272.0

Стоимость кВт*ч	1.13	Площадь помещ..	40
Затраты на тех. обслужив...	0.25	Аренда 1 м2	100
Норма амортизац. исч.	10		
Наценка торговых организ...	0.15		
Транспортные рас...	0.03		

**Рис. 23.1. Расчет затрат на разработку ТЭБ**

Определим стоимость одной машино-часа, подставив полученные значения текущих расходов и износа в формулу, представленную выше, получим:

- для компьютера,  $C = \frac{1,2 * (6340,80 + 430,95)}{986} = 8,27$  грн.

- для принтера,  $C = \frac{1,2 * (673,10 + 75,30)}{986} = 0,91$  грн.

$$\text{ - для сканера, } C = \frac{1,2 * (313,47 + 69,66)}{986} = 0,47 \text{ грн.}$$

**Расчет заработной платы исполнителей.** Рассчитаем заработную плату исполнителей, входные данные для расчета.

$$P_{\text{ОЗП}} = 6500 * 6 + 5100 * 6 + 4500 * 2 * 6 + 3500 * 2 * 6 + 3100 * 6 = 184200 \text{ грн.}$$

$$P_{\text{ДЗП}} = (184200 * 17\%) / 100\% = 31314 \text{ грн.}$$

$$P_{\text{ОТЧ}} = (184200 + 31314) * 0,26 = 56033,64 \text{ грн.}$$

**Определение себестоимости, договорной и продажной цены.**

Определим себестоимость ТЭБ для этого рассчитаем накладные расходы.

$N_{\text{накл}}$  – норматив накладных расходов составляет 15%.

$$P_{\text{накл}} = (184200 * 15\%) / 100\% = 27630,00 \text{ грн.}$$

Таким образом, себестоимость составляет:

$$C_{\text{ТЕБ}} = 1101,13 + 26993,95 + 184200 + 31314 + 56033,64 + 27630 = 299177,64 \text{ грн.}$$

Определим нормативную прибыль  $R_{\text{н}}$ :

$$P_{\text{н}} = (299177,64 * 15) / 100 = 44876,65 \text{ грн.}$$

Выполним расчет договорной цены разрабатываемой ТЭБ

$$C_{\text{ДОГ}} = 299177,64 + 44876,65 = 344054,29 \text{ грн.}$$

Все расчеты представлены в табл. 23.5.

Таблица 23.5

**Расчет стоимости договорной цены ТЭБ**

№ п/п	Наименование статей расходов	Сумма, грн.
1	Расходы на материалы	1101,13
2	Специальное оборудование для научных и экспериментальных работ	26993,95
3	Основная заработная плата исполнителей	184200,00
4	Дополнительная заработная плата исполнителей	31314,00
5	Отчисления на социальные нужды	56033,64
6	Накладные расходы	27630,00
7	Себестоимость	299177,64
8	Нормативная прибыль	44876,65
9	Договорная цена	344054,29

Рассчитаем продажную цену ТЭБ:

$$C_{\text{ПР}} = 344054,29 + 0,15 * 344054,29 + 0,03 * 344054,29 = 405984,06 \text{ грн.}$$

## 5. Расчет эксплуатационных затрат связанных с использованием нового ПО

Расходы, связанные с эксплуатацией ТЭБ, рассчитываются по формуле, так как количество часов работы ПО в день составляет 5 ч., а количество рабочих дней в году в среднем составляет 250 дней, то

$$T_{\text{МВ}} = 250 * 5 = 1250 \text{ машино-чел./час.}$$

Тогда эксплуатационные расходы на 1 час машинного времени ПК, грн. / машино-чел. составляют:

$$T_{\text{ч}} = (0,4 + 0,3 + 0,25) * 1,13 = 1,07 \text{ грн./машино-час.}$$

$T_{\text{с}}$  – срок службы ТЭБ - 4 года.

Расчет машинного времени ПК приведен в табл. 23.6.

Таблица 23.6

**Расчет машинного времени ПК**

Наименование показателя	Количество часов работы в день	Количество рабочих дней 2016 году	Значение показателя
Машинное время ПК	5	250	1250

Таким образом, эксплуатационные расходы составляют:

$$E = 1250 * 1,07 + \frac{405984,06}{4} = 1337,5 + 101496,02 = 102833,52 \text{ грн.}$$

## 6. Определение экономической эффективности ТЭБ

Эффективность – это характеристика системы с точки зрения соотношения затрат и результатов функционирования системы. К основным показателям экономической эффективности относятся: экономический эффект, коэффициент экономической эффективности капитальных вложений, срок окупаемости капитальных вложений и др.

Экономический эффект – это результат внедрения какого-либо мероприятия, выраженный в стоимостной форме, в виде экономии от его осуществления.

Товар-конкурент имеет следующие показатели: объем памяти – 9 Мбт.; время обработки данных – 0,6 с.; время выполнения типовой операции – 6 мин.; надежность – 2; срок службы - 4 года; договорная цена – 477 011 грн.

## 7. Годовой экономический эффект от производства ТЭБ

$E_{\text{н}}$  – нормативный коэффициент эффективности капиталовложений

$$Ц_{\text{дог}} = C_{\text{ПО}} + П_{\text{н}} = C_{\text{ПО}} + 0,15 \cdot C_{\text{ПО}} = C_{\text{ПО}} \cdot (1 + 0,15) = 1,15 \cdot C_{\text{ПО}} \Rightarrow$$

$$\Rightarrow C_{\text{ПО}} = \frac{Ц_{\text{дог}}}{1,15}$$



$$C_{\text{ПО1}} = \frac{477011}{1,15} = 414792,20 \text{ грн.}, C_{\text{ПО2}} = 299177,64 \text{ грн.}$$

Таким образом, если принять, что при разработке базового ПО использовалось компьютерное оборудование одного вида, то капитальные вложения равны:

$$З_1 = 414792,20 + (0,15 * 26993,95) = 414792,20 + 4049,09 = 418841,27 \text{ грн.}$$

$$З_2 = 299177,64 + (0,15 * 26993,95) = 299177,64 + 4049,09 = 303226,73 \text{ грн.}$$

А годовой экономический эффект от производства новой ТЭБ равен:

$$\mathcal{E} = (418841,27 - 303226,73) * 1 = 115614,53 \text{ грн.}$$

Тогда показатель прироста прибыли в расчетном году будет составлять:

$$\Delta\Pi = [(344054,29 - 299177,64) - (477011 - 414792)] * 1 = [44876,65 - 62218,83] * 1 = -17342,18 \text{ грн.}$$

Полученное значение показывает, что производство новой ТЭБ дешевле на 17342,18 чем производство базового ПО.

### **8. Годовой экономический эффект от использования ТЭБ**

Определим себестоимость единицы работ, которая включает в себя расходы на печать одного листа (для базового 2,5 грн. / л., для новой 2 грн. / л.),

Зарботная плата оператора, работающего с программой будет равна 3500,00 грн., Тогда почасовая зарботная плата составляет:

$$3500 / (22 * 8) = 19,88 \text{ грн./час.}$$

Время выполнения типовой операции при использовании новой ТЭБ равен 4 мин., на базовом ПО 6 мин., из этого следует, что в час при использовании новой ТЭБ оператор может выполнить 15 операций, а на базовом ПО 10. Таким образом, получим стоимость одной типовой операции для новой ТЭБ  $19,88 / 15 = 1,33$  грн., для базового ПО  $19,88 / 10 = 1,99$  грн.

Тогда себестоимость составляет:

$$C_1 = 1,13 \cdot (0,4 + 0,3) + 2,5 + 1,99 = 5,28 \text{ грн.}$$

$$C_2 = 1,13 \cdot (0,4 + 0,3) + 2 + 1,33 = 4,12 \text{ грн.}$$

Количество операций в год, выполняются с помощью нового и базового ПО, рассчитаем, зная количество операций, выполняемых в час. В час с помощью нового ПО выполняется 15 операций, поскольку рабочий день оператора равен 8 часам, следовательно за день выполняется:  $15 \cdot 8 = 120$  операций. Зная, что количество рабочих дней в году в среднем составляет 250, можем рассчитать количество операций, выполняемых в год:  $120 \cdot 250 = 30000$  операций.

Таким же образом сделаем расчеты для базового ПО. В час с помощью базового ПО выполняется 10 операций, следовательно, за день выполняется:  $10 \cdot 8 = 80$  операций и за год соответственно:  $80 \cdot 250 = 20000$  операций.

В нашем случае удельные капитальные вложения в ПО равны затратам на специальное оборудование, то есть  $K_{PI} = P_{CO} = 26993,95$  грн. Принимаем, что при разработке базового и нового ПО использовалось компьютерное оборудование одного вида, то есть  $K_{PI} = K_{P2} = 26993,95$  грн. Теперь рассчитаем приведенные затраты на единицу работ:

$$Z_1 = 5,28 + 0,15 * \frac{26993,95}{20000} = 5,48 \text{ грн.}$$

$$Z_2 = 4,12 + 0,15 * \frac{26993,95}{30000} = 4,26 \text{ грн.}$$

А годовой экономический эффект от производства новой ТЭБ равна:

$$\mathcal{E} = (5,48 - 2,26) * 30000 = 36600 \text{ грн.}$$

#### *8.1. Расчет экономии затрат на оплату машинного времени*

Стоимость часа работы  $i$ -го ресурса вычислительного комплекса включает затраты на работы компьютера и принтера, а также стоимость бумаги, следовательно:  $C_{MI} = 1,13 * (0,4 + 0,3) + 2 = 2,79$  грн.

Время обработки данных для базового и нового ПО соответственно равны 0,6 и 0,45 с.

Таким образом,  $dC_{MI} = 2,79 * (0,6 - 0,45) * 30000 = 12555$  грн.

#### *8.2. Расчет условной годовой экономии затрат на оплату труда работников*

В новой ТЭБ на подготовку и обработку данных идет 15 мин. ( $Q_2 = 4$  чел./час.), в базовом в следствии низкой скорости 17 мин. ( $Q_1 = 3,5$  чел./час.)

$$dC_{ij} = \frac{3500 * 12}{(4,0 - 3,5) * 2000} * 30000 = \frac{42000}{1000} * 30000 = 1260000 \text{ грн.}$$

#### *8.3. Расчет относительной годовой экономии затрат на материалы*

Относительная годовая экономия затрат на материалы при внедрении новой ТЭБ и составляет:  $dC_M = (5,28 - 4,12) * 30000 = 34800$  грн.

#### *8.4. Расчет общего экономического эффекта от использования ТЭБ*

Приведенные удельные затраты на базовый и новый ПО равны соответственно 418841,27 грн. и 303226,73 грн.

Годовой объем работ, произведенных с помощью базового ПО и новой ТЭБ, равны соответственно 20000 и 30000 натур. ед.

Срок службы новой ТЭБ и базового ПО совпадают и равны 4 года.

Годовые средние удельные эксплуатационные затраты пользователя при эксплуатации единицы старого ПО и новой ТЭБ рассчитаем как произведение годового объема работ, произведенных с помощью ПО, и стоимость одной типовой операции, следовательно:

$$U_1 = 20000 \cdot 1,99 = 39800 \text{ грн.}, U_2 = 30000 \cdot 1,33 = 39900 \text{ грн.}$$

$$K_1 = \frac{26\,993,95}{20\,000} = 1,35 \text{ грн.}, K_2 = \frac{26\,993,95}{30\,000} = 0,90 \text{ грн.}$$

Общий экономический эффект от производства и эксплуатации новой ТЭБ:

$$\begin{aligned} \mathcal{E}_{\text{общ}} &= \left[ \left( 418\,841,27 \cdot \frac{30\,000}{20\,000} \cdot \frac{(4 + 0,15)}{(4 + 0,15)} - \frac{(39900 - 39900) - 0,15 \cdot (1,35 - 0,9)}{(4 + 0,15)} \right) - \right. \\ &- 303226,73 \cdot 1 = \left[ \left( 418841,27 \cdot 1,5 \cdot 1 - \frac{(-100) - 0,07}{4,15} \right) - 303226,73 \right] \cdot 1 = \\ &= [(418841,27 \cdot 1,5 \cdot 1 + 24,11) - 303226,73] \cdot 1 = [628286,02 - 303226,73] \cdot 1 = \\ &= 325059,29 \text{ грн} \end{aligned}$$

Должность	Экономия
Научный руководит...	2340000.0
Головний інженер	1836000.0
Інженер-програміст	1620000.0
Оператор	1260000.0
Технік	1116000.0

Годовой экономич. эффект произ...	-162625.59
Прирост прибыли в расчетном году	24393.84
Годовой экон. эфф. от использ. ПО	3517813.33
Экономия на оплату маш. верм.	12559.5
Экономия затрат на материалы	3515000.0
Общий экономический эффект	-139348.59
Коэффициент экономической эф...	1.94

Рис. 23.2. Расчет экономического эффекта от использования ТЭБ

### 8.5. Расчётный коэффициент экономической эффективности

Годовой прирост прибыли определяется на основе разницы между ценой и себестоимостью единицы ТЭБ.

$$E_p = \frac{405984,06 - 299177,64}{26993,95} = \frac{106806,42}{26993,95} = 3,96 \text{ грн.}$$

Так как расчетный коэффициент экономической эффективности превышает нормативное значение 0,15, следовательно, производство и внедрение ТЭБ является эффективным.

### 9. Оценка конкурентоспособности ТЭБ

В качестве эталона будет выступать ПО со следующими параметрами: объем памяти - 6 Мбайт, время обработки данных - 0,1 с., надежность - 1.

Значения параметров сравниваемых ТЭБ представим в табл. 23.7.

Таблица 23.7

**Значения параметров сравниваемых ТЭБ**

Наименование параметра	Вес, параметра	Значение параметра			$\frac{П_Б}{П_Е}$	$\frac{П_Н}{П_Е}$	$\beta * \frac{П_Б}{П_Е}$	$\beta * \frac{П_Н}{П_Е}$
		$П_Б$	$П_Н$	$П_Е$				
Объем памяти	0,2	9	7	6	0,67	0,86	0,134	0,172
Время обработки данных	0,3	0,6	0,45	0,1	0,17	0,22	0,051	0,066
Количество отказов	0,5	2	1	1	0,5	1	0,25	0,5
<b>Всего</b>							0,435	0,738

Следовательно,  $K_{ЭК} = \frac{0,74}{0,43} = 1,72$

Полученное значение коэффициента эквивалентности больше 1, а значит разработанная ТЭБ является технически прогрессивной.

Далее рассчитаем коэффициент изменения функциональных возможностей новой ТЭБ.

$$K_{ФВ} = \frac{25}{19} = 1,32$$

Параметры ТЭБ представлены в табл. 23.8.

Коэффициент функциональных возможностей превышает 1, т.е. новая ТЭБ превышает по своим функциональным возможностям старое ПО в 1,3 раза.

### Параметры ТЕБ

Наименование показателя	Бальная оценка базового ПО	Бальная оценка новой ТЭБ
Объем памяти	3	4
Функциональные возможности	4	5
Быстродействие	3	4
Удобность интерфейса	4	5
Степень стимулированности	2	3
Продуктивность работы	3	4
Всего	19	25

Выполним оценку конкурентоспособности ТЭБ используя интегральный коэффициент.

$$K_{ц} = \frac{344054,29}{477011,00} = 0,72$$

$$K_{и} = 1,72 * 1,32 * \frac{1}{0,72} = 3,16$$

$K_{и} > 1$ , а значит новая ТЭБ более конкурентоспособна нежели старое ПО.

Эксплуатационные расходы
Оценка конкурентоспособности

Экономический эффект
Исполнители
Расчет затрат
Сохранить проект
Трудоёмкость

Коэффициент старого ПО

Коэффициент нового ПО  Оценка

Наименование показателя	Бальная оценка базового ПО	Бальная оценка нового ПО
Объем памяти	3	4
Функциональные возможности	4	5
Быстродействие	4	4
Удобство интерфейса	3	5
Степень утомляемости	2	3
Производительность труда	3	4

Оценка базового ПО

Оценка нового ПО

Коэффициенты	Значение
Коэффициент эквивалентности	1.7209302325581395
Коэффициент изменения функц. возможнос...	1.3157894736842106
Коэффициент соответствия нормативам	1.0
Коэффициент цены потребителя	4.978205696539108
Интегральный коэффициент конкурентнос...	0.4548590441972172

**Рис. 23.3. Оценка конкурентоспособности ТЭБ**

Все рассчитанные коэффициенты внесены в табл. 23.9.

Таблица 23.9

**Рассчитанные коэффициенты**

<b>Коэффициент</b>	<b>Значение</b>
Коэффициент эквивалентности	1,72
Коэффициент изменения функциональных возможностей	1,32
Коэффициент соответствия нормативам	1,00
Коэффициент цены потребителя	0,72
Интегральный коэффициент конкурентоспособности	3,16

**Анализ полученных результатов.** Расчётный коэффициент экономической эффективности составляет 3,96, что значительно превышает нормативное значение 0,15. Коэффициент функциональных возможностей ТЭБ больше старого ПО в 1,3 раз. Полученное значение коэффициента эквивалентности составляет 1,72. Конкурентоспособность ТЭБ 3,16.

Все полученные значения коэффициентов являются однонаправленными и превышают нормативное значение, а это означает, что разрабатываемая ТЭБ является эффективной, прогрессивной и конкурентоспособной.

Таким образом, разрабатываемая ТЭБ является экономически выгодной.

## РОЗДІЛ 24

# ПОДХОД К ПОСТРОЕНИЮ ИНФОРМАЦИОННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ЭФФЕКТИВНОСТЬЮ ПРЕДПРИЯТИЯ, ВХОДЯЩЕГО В ХОЛДИНГ

***Аннотация.** Предложена структура EPM – системы предприятия, входящего в холдинг. Рассмотрен процесс разработки ключевых показателей деятельности холдинга до бюджетных показателей предприятий, которые входят в данный холдинг. Каскадирование предполагает декомпозицию ключевых показателей холдинга на стратегическом периоде на ключевые показатели деятельности предприятий по годам. Годовые показатели далее декомпозируются до показателей бизнес-процессов операционной, финансовой и инвестиционной деятельности, а затем на их основании формируются показатели бюджетов предприятия.*

***Ключевые слова:** стратегическое планирование, ключевые показатели деятельности, стратегическая карта, каскадирование, система управления эффективностью предприятия, EPM-система*

***Abstract.** The structure of EPM-system of the company which is part of the holding, was offered. The process of developing from key indicators of activity for the holding up to the budget indicator of enterprises that are part of the holding was considered. Cascading involves the decomposition of key holding's indicators at the strategic period on key performance indicators of the companies over the years. Annual figures are further decomposed to indicators of business processes for the operating, financing and investing activities, and then enterprise budgets are formed based on them.*

***Keywords:** strategic planning, key performance indicators, strategic map, cascading, the system of the corporate performance management, EPM-system*

**Введение и постановка задачи.** Стратегическое управление является сложным процессом, который требует сбора, обработки и анализа множества информации, данных разной природы, требует решения множества задач. И естественно, все это приводит к необходимости автоматизации этого процесса. Однако сложность и слабая формализуемость многих задач стратегического управления не позволяет применять стандартные технологии, математические методы и модели, и, как следствие, разработать стандартные программные решения. Рассмотрим одну из основных проблем стратегического планирования – реализуемость стратегических планов. Стратегические планы должны быть реализованы за счет реализации годовых, оперативных планов и бюджетов [66].

На практике затруднительно определить мероприятия по достижению плановых значений стратегических показателей деятельности предприятия и конкретных исполнителей. Это связано с тем, что эти плановые показатели представляют собой агрегированные, интегративные показатели, они вычисляются на основе множества других показателей, характеризующих разные направления деятельности предприятия. Только немногие стратегические показатели могут быть однозначно отнесены к определённым центрам ответственности (исполнителям и/или подразделениям предприятия). Поэтому возникает “разрыв” между стратегическим управлением и оперативным.

В настоящее время перспективным направлением совершенствования системы стратегического управления является внедрение бизнес-ориентированного подхода, который базируется на системе ключевых показателей эффективности (KPI – Key Performance Indicators) [40].

Крупные производители платформ для разработки корпоративных информационных систем реализуют функции управления эффективностью в системах EPM (Enterprise Performance Management), например, Oracle EPM, Oracle Hyperion Planning, IBM Cognos Disclosure Management, SAP EPM, IBM Cognos FPM и т.д. Так как требования к таким программным системам постоянно возрастают, поэтому компании совершенствуют свои разработки за счет включения дополнительных функциональных модулей, изменяют алгоритмы решения управленческих задач и т.д. [66].

На основе анализа функционала современных EPM-систем и предлагаемых программных решений, можно сделать вывод, что остаётся актуальной проблема интеграции системы стратегического управления в системы текущего, оперативного управления и бюджетирования. В рамках решения этой проблемы рассматривают задачу каскадирования KPI.

Чаще всего под каскадированием понимается построение стратегических карт дерева целей для структурных подразделений предприятия. Каскадирование стратегической карты осуществляется по двум направлениям: горизонтально (вовлечение подразделений на одном иерархическом уровне); вертикально (вовлечение других уровней руководства). Результатом являются созданные стратегические карты и KPI для различных организационных подразделений и различных уровней организационной иерархии [40; 11]. На основе проведённого анализа проблем каскадирования были сделаны выводы, что не до конца проработан вопрос отнесения того или иного стратегического показателя предприятия к определённому подразделению (или



центру ответственности, или бизнес-процессу), нет методологии определения плановых значений показателей эффективности подразделений и ключевых показателей деятельности сотрудников по интервалам стратегического периода, как взаимосвязанной системы. Трудность здесь заключается в том, что системы годового планирования, бюджетирования, управленческого и бухгалтерского учёта построены на других принципах, нежели стратегическое планирование, т.е. другая классификация показателей и ответственных за выполнение, другие цели и измерители [11; 65]. В основном при годовом планировании используют показатели бюджетов за предыдущие периоды, а стратегические планы, разрабатываемые руководством, носят декларативный характер. Поэтому необходимо разработать такую технологию каскадирования ключевых показателей деятельности, которая будет осуществлять декомпозицию стратегических показателей до годовых показателей и показателей бюджетов. Результатом такой технологии будет система показателей эффективности (деятельности) для всех уровней управления предприятия, т.е. как в разрезе бизнес-процессов, подразделений и исполнителей, так и во временном аспекте. Эта технология должна быть реализована в виде отдельного модуля ЕРМ-системы. Разрабатываемый модуль должен быть интегрирован с системой оперативного учёта и системой бюджетирования.

**Основная часть.** Рассмотрим задачу формирования стратегических планов для холдинга. Задача каскадирования стратегических КРІ холдинга будет состоять в том, чтобы определить стратегические значения КРІ для всех предприятий, входящих в холдинг, и годовые их значения.

### **1. Описание холдинговой структуры**

Холдинговая структура представляет собой иерархию, на верхнем уровне которой находится головная компания (ГК) или материнская компания, на нижнем уровне – предприятия, которые могут быть дочерними и/или аффилированными. Каждое предприятие имеет функционально-линейную структуру, т.е. состоит из структурных подразделений, разделённых по функциональному признаку. Головная компания занимается вопросами стратегического управления, разрабатывает стратегические цели и программы развития всего холдинга. Предприятия, входящие в холдинг также разрабатывают свои стратегические планы, которые должны согласовываться с программой развития холдинга.

Холдинговая структура управления характеризуется следующим набором термов:

$$HS = \langle SG, EN, VC, HC, MH \rangle,$$

где  $SG = \{SG_g\}$ ,  $g \in G^H$  – множество стратегических целей холдинга,  $G^H$  – множество индексов целей холдинга,  $|SG| = G$ ;

$EN = \{EN_i\}$ ,  $i \in M^H$  – множество предприятий, которые входят в холдинг,  $M^H$  – множество индексов предприятий,  $|M^H| = m$ , т.е.  $m$  – количество предприятий в холдинге, не считая головную компанию;

$VC = \{VC_k\}$ ,  $k \in V^H$ ;  $HC = \{HC_k\}$ ,  $k \in H^H$  – вертикальные и горизонтальные связи внутри холдинга соответственно; горизонтальные связи показывают бизнес-отношения между предприятиями, а вертикальные – отношения ГК с предприятиями.  $V^H$  и  $H^H$  множества индексов вертикальных и горизонтальных связей внутри холдинга соответственно;

$MH = \{MH_f\}$ ,  $f \in L^H$  – управленческие воздействия ГК на предприятия – это методы и механизмы управления в холдинге (включая систему отчётности предприятий для ГК);  $L^H$  – множество индексов управленческих воздействий.

Система показателей деятельности предприятий может входить как подмножество множества управленческих воздействий холдинга  $MH$ .

Аналогично описанию всего холдинга, каждое предприятие холдинга характеризуется следующим набором термов:

$$EN = \langle Z^E, SS, VS, HS, MS \rangle,$$

где  $Z^E = \{Z_i^E\}$ ,  $i \in N^E$  – множество стратегических целей предприятия,  $N^E$  – множество индексов общих целей предприятия (множество  $Z^E$  разбивается на подмножества *целей по перспективам*);

$SS = \{SS_k\}$ ,  $k \in K^E$  – множество структурных подразделений предприятия,  $K^E$  – множество индексов структурных подразделений предприятия,  $|K^E| = E$ ;  $E$  – количество структурных подразделений, структурные подразделения выделяются в зависимости от утверждённой на каждом предприятии организационной структуры, например, это могут быть центры финансовой ответственности (ЦФО);

$VS = \{VS_k\}$ ,  $k \in V^E$ ,  $HS = \{HS_k\}$ ,  $k \in H^E$  – соответственно множества вертикальных и горизонтальных связей внутри предприятия (аналогично

описанию холдинга);  $V^E$ ,  $H^E$  – множества индексов вертикальных и горизонтальных связей соответственно;

$MS = \{MS_l\}, l \in L^E$  – множество управленческих воздействий руководства предприятия (главного офиса) на структурные подразделения предприятия (структурные подразделения – это хозяйственные единицы, центры финансовой ответственности и т.п.),  $L^E$  – множество индексов управленческих воздействий главного офиса предприятия на структурные подразделения.

Рассмотрим процессы формирования стратегических планов на уровне головной компании и на уровне предприятий.

Будем считать, что ГК определяет набор общих целей всего холдинга. Используя методологию ССП (система сбалансированных показателей), строится стратегическая карта холдинга, выделяются основные ключевые показатели – КРІ, для которых определяют стратегические значения [40]. Эти значения должны быть достигнуты всем холдингом на конец стратегического периода, что и будет означать достижение общих стратегических целей. Эти ключевые показатели являются агрегированным представлением других показателей деятельности предприятий, входящих в холдинг.

Необходимо разработать такие стратегические, тактические и годовые планы предприятия, выполнение которых позволит всему холдингу достичь поставленных стратегических целей.

## **2. Технология каскадирования показателей деятельности холдинга**

ЭТАП 1. Стратегическое планирование на уровне головной компании холдинга.

1.1. Головная компания холдинга формирует стратегические цели развития всего холдинга на определённый долгосрочный период (так называемые общие стратегические цели холдинга). Пусть имеется множество всех возможных стратегических целей развития предприятия  $SG^Z$ . Стратегический менеджер (или руководитель предприятия, или другое лицо, ответственное за стратегическое развитие предприятия) формирует стратегии на долгосрочный плановый период – множество  $SG$ ,  $SG \subset SG^Z$ . Предварительно проводится анализ деятельности предприятия и анализ перспектив, прогнозов и т.д. Здесь используется SWOT-анализ, PEST-анализ и другой инструментарий стратегического анализа.

1.2. Общие стратегические цели холдинга  $SG$  декомпозируются на стратегические цели по перспективам: финансы, клиенты, бизнес-процессы и

персонал (согласно ССП). Каждой цели  $SG_g, g \in G^H$ , ставится в соответствие множества целей по перспективам:  $Z^{HF} = \{Z_g^{HF}\}, g \in G^{HF}$  – финансы,  $|G^{HF}| = Q^F$ ;  $Z^{HK} = \{Z_g^{HK}\}, g \in G^{HK}$  – клиенты,  $|G^{HK}| = Q^K$ ;  $Z^{HBP} = \{Z_g^{BP}\}, g \in G^{HBP}$  – внутренние бизнес-процессы,  $|G^{HBP}| = Q^B$ ;  $Z^{HP} = \{Z_g^{HP}\}, g \in G^{HP}$  – персонал/развитие;  $|G^{HP}| = Q^P$ , где  $G^{HF}, G^{HK}, G^{HBP}, G^{HP}$  – множества индексов целей по перспективам финансы, клиенты, бизнес-процессы и персонал, а их мощности равны соответственно  $Q^F, Q^K, Q^B, Q^P$ .

Общие цели  $SG = \{SG_g\}, g \in G^H$ , цели по перспективам  $Z^H = \{Z^{HF}, Z^{HK}, Z^{HBP}, Z^{HP}\}$  и связи между целями формируют стратегическую карту холдинга  $\overline{SM^H}$ . Связи между общими целями и целями по перспективам, а также связи между целями по перспективам описываются следующими матрицами связностей:

$C^{HF} = \{c_{ig}^{HF}, i = 1, \dots, Q^F; g = 1, \dots, G\}$  – матрица связностей целей по перспективе “финансы” с общими целями:

$$C^{HF} = \begin{pmatrix} c_{11}^{HF} & c_{12}^{HF} & \dots & c_{1g}^{HF} & \dots & c_{1G}^{HF} \\ c_{21}^{HF} & c_{22}^{HF} & \dots & c_{2g}^{HF} & \dots & c_{2G}^{HF} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ c_{i1}^{HF} & c_{i2}^{HF} & \dots & c_{ig}^{HF} & \dots & c_{iG}^{HF} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ c_{Q^F1}^{HF} & c_{Q^F2}^{HF} & \dots & c_{Q^Fg}^{HF} & \dots & c_{Q^FG}^{HF} \end{pmatrix}, \text{ где } c_{ig}^{HF} = \begin{cases} 1, \text{ связь существует,} \\ 0, \text{ связь отсутствует.} \end{cases}$$

Если, например,  $c_{21}^{HF} = 1$ , то это означает, что вторая цель по финансам имеет связь с первой общей целью, т.е. для достижения первой общей цели необходимо достижение второй цели по финансам (помимо достижения других целей).

$C^{HK} = \{c_{ig}^{HK}, i = 1, \dots, Q^K; g = 1, \dots, G\}$  – матрица связностей целей по перспективе «клиенты» с общими целями,

$C^{HBP} = \{c_{ig}^{HBP}, i = 1, \dots, Q^B; g = 1, \dots, G\}$  – матрица связностей целей по перспективе «бизнес-процессы» с общими целями,

$C^{HP} = \{c_{ig}^{HP}, i = 1, \dots, Q^P; g = 1, \dots, G\}$  – матрица связностей целей по перспективе «клиенты» с общими целями,

$$C^{HK} = \begin{pmatrix} c_{11}^{HK} & \dots & c_{1G}^{HK} \\ \dots & \dots & \dots \\ c_{Q^{K1}}^{HK} & \dots & c_{Q^{KG}}^{HK} \end{pmatrix}, C^{HBP} = \begin{pmatrix} c_{11}^{HBP} & \dots & c_{1G}^{HBP} \\ \dots & \dots & \dots \\ c_{Q^{B1}}^{HBP} & \dots & c_{Q^{BG}}^{HBP} \end{pmatrix}, C^{HP} = \begin{pmatrix} c_{11}^{HP} & \dots & c_{1G}^{HP} \\ \dots & \dots & \dots \\ c_{Q^{P1}}^{HP} & \dots & c_{Q^{PG}}^{HP} \end{pmatrix}.$$

Также формируются матрицы связностей между целями по перспективам:

$P^{FK} = \{p_{ij}^{FK}, i = 1, \dots, Q^F; j = 1, \dots, Q^K\}$  – матрица связностей целей по перспективе «финансы» с целями по перспективе “клиенты”,

$P^{FBP} = \{p_{ij}^{FBP}, i = 1, \dots, Q^F; j = 1, \dots, Q^B\}$  – матрица связностей целей по перспективе «финансы» с целями по перспективе “бизнес-процессы”,

$P^{FP} = \{p_{ij}^{FP}, i = 1, \dots, Q^F; j = 1, \dots, Q^P\}$  – матрица связностей целей по перспективе «финансы» с целями по перспективе “персонал”,

$P^{KB} = \{p_{ij}^{KB}, i = 1, \dots, Q^K; j = 1, \dots, Q^B\}$  – матрица связностей целей по перспективе «клиенты» с целями по перспективе “бизнес-процессы”,

$P^{KP} = \{p_{ij}^{KP}, i = 1, \dots, Q^K; j = 1, \dots, Q^P\}$  – матрица связностей целей по перспективе «клиенты» с целями по перспективе “персонал”,

$P^{BP} = \{p_{ij}^{BP}, i = 1, \dots, Q^B; j = 1, \dots, Q^P\}$  – матрица связностей целей по перспективе «бизнес-процессы» с целями по перспективе “персонал”,

Тогда стратегическую карту можно представить, как набор термов:

$$\overline{SM^H} = \langle SG, Z^H, CZ^H \rangle, \text{ где } CZ^H - \text{ множество матриц связностей, } CZ^H = \{C^{HF}, C^{HK}, C^{HBP}, C^{HP}, P^{FK}, P^{FBP}, P^{FP}, P^{KB}, P^{KP}, P^{BP}\}.$$

1.3. Проведение когнитивного анализа стратегической карты холдинга  $\overline{SM^H}$  для определения “силы влияния” целей по перспективам на общие цели и степени связности целей по перспективам. В результате строится новая стратегическая карта  $SM^H$  холдинга (убираются “лишние”, менее значимые или дублирующие цели и связи). В матрицы связностей вносятся изменения.

Применение на этом этапе экспертного оценивания важности достижения целей по перспективам для реализации общих целей даст возможность исключить менее значимые цели по перспективам.

1.4. Для каждой стратегической цели по перспективе определяются ключевые факторы успеха (КФУ). КФУ группируются по множествам:

$$KSF^{HF} = \{KSF_i^{HF}\}, i \in K^{HF} - \text{ множество КФУ для целей по перспективе “финансы”,}$$

$KSF^{HK} = \{KSF_i^{HK}\}, i \in K^{HK}$  – множество КФУ для целей по перспективе “клиенты”,

$KSF^{HBP} = \{KSF_i^{HBP}\}, i \in K^{HBP}$  – множество КФУ для целей по перспективе “бизнес-процессы”,

$KSF^{HP} = \{KSF_i^{HP}\}, i \in K^{HP}$  – множество КФУ для целей по перспективе “персонал”.

Множество всех КФУ состоит из подмножеств КФУ по перспективам:

$$KSF^H = \{KSF^{HF}, KSF^{HK}, KSF^{HBP}, KSF^{HP}\}.$$

Каждый фактор успеха однозначно связан с целью по перспективе, т.е. реализация цели по перспективе зависит от набора ключевых факторов успеха. Например, для каждой цели по перспективе “финансы” можно записать следующую зависимость

$$Z_g^{HF}(KSF_{gi}^{HF}, i=1, \dots, K_g^F), g \in G^{HF},$$

где  $K_g^F$  – количество факторов успеха для  $g$ -ой цели по перспективе “финансы”.

Также по каждой перспективе можно представить взаимосвязь КФУ и целей в виде матриц связностей:

$SF^{HF} = \{s_{ij}^F, i=1, \dots, Q^F, j=1, \dots, K^F\}$  – по перспективе “финансы”, вид этой матрицы:

$$SF^{HF} = \begin{pmatrix} s_{11}^F & s_{12}^F & \dots & s_{1j}^F & \dots & s_{1K^F}^F \\ s_{21}^F & s_{22}^F & \dots & s_{2g}^F & \dots & s_{1K^F}^F \\ \dots & \dots & \dots & \dots & \dots & \dots \\ s_{i1}^F & s_{i2}^F & \dots & s_{ij}^F & \dots & s_{1K^F}^F \\ \dots & \dots & \dots & \dots & \dots & \dots \\ s_{Q^F1}^F & s_{Q^F2}^F & \dots & s_{Q^Fj}^F & \dots & s_{Q^F K^F}^F \end{pmatrix},$$

$K^F$  – количество КФУ для целей по перспективе “финансы”,

$SF^{HK} = \{s_{ij}^K, i=1, \dots, Q^{KJ}, j=1, \dots, SK^K\}$  – по перспективе “клиенты”,

$SF^{HB} = \{s_{ij}^B, i=1, \dots, Q^B, j=1, \dots, SK^B\}$  – по перспективе “бизнес-процессы”,

$SF^{HP} = \{s_{ij}^P, i=1, \dots, Q^P, j=1, \dots, SK^P\}$  – по перспективе “персонал”.

$SF^H = \{SF^{HF}, SF^{HK}, SF^{HB}, SF^{HP}\}$  – множество всех матриц связи КФУ и целей по перспективам.

1.5. Построение когнитивной карты КФУ для анализа взаимосвязей. В результате получим множество КФУ со связями их со стратегическими целями по перспективам.

1.6. Определение на основе КФУ стратегических ключевых показателей

$$S^H = \{S^{HF}; S^K; S^{HBP}; S^{HP}\},$$

где  $S^{HF} = \{S_i^{HF}, i \in M^{HF}\}$  – финансовые показатели холдинга,  $|M^{HF}| = M^F$ ;

$S^{HK} = \{S_i^{HK}, i \in M^{HK}\}$  – показатели, характеризующие отношения с клиентами,  $|M^{HK}| = M^K$ ;

$S^{HBP} = \{S_i^{HBP}, i \in M^{HBP}\}$  – внутренние бизнес-процессы,  $|M^{HBP}| = M^B$ ;

$S^{HP} = \{S_i^{HP}, i \in M^P\}$  – развитие и обучение персонала,  $|M^{HP}| = M^P$ ;

где  $M^{HF}$ ,  $M^{HK}$ ,  $M^{HBP}$ ,  $M^{HP}$  – множества показателей, характеризующие финансы, клиентов, бизнес-процессы и персонал соответственно, а их мощность соответственно равны  $M^F$ ,  $M^K$ ,  $M^B$ ,  $M^P$ .

1.7. Определение плановых значений KPI  $S^H = \{S^{HF}; S^K; S^{HBP}; S^{HP}\}$  на стратегический период в виде интервальных оценок (минимально и максимально достижимые значения).

В результате построена стратегическая карта холдинга

$$SM_T^H = \langle SG, Z^H, CZ^H, KSF^H, SF^H, \{S^{HF}; S^K; S^{HBP}; S^{HP}\} \rangle,$$

которая состоит из следующих компонент:

- общие стратегические цели холдинга  $SG$ ;
- стратегические цели холдинга по перспективам  $Z^H = \{Z_H^F, Z_H^K, Z_H^{BP}, Z_H^P\}$ ;
- связи между целями в виде множества матриц связностей  $CZ^H$ ;
- ключевые факторы успеха  $KSF^H$  по перспективам;
- стратегические ключевые показатели по перспективам  $S^H = \{S^{HF}; S^K; S^{HBP}; S^{HP}\}$ ;
- связи между КФУ и целями по перспективам  $SF^H$ .

1.8. Определение приоритетов стратегических KPI по перспективам на основе метода анализа иерархий. Определяются приоритеты KPI относительно стратегических целей холдинга, влияние предприятий на успешность выполнения KPI (или степень влияния деятельности предприятия на ключевой показатель холдинга). В результате можно будет выделить те KPI, которые

однозначно можно отнести к определённым предприятиям и «закрепить» их за предприятием, т.е. предприятие будет являться ответственным за достижение плановых показателей “закрепленных” за ним КРІ (это те КРІ, которые имеют наибольший приоритет для предприятия). Те КРІ, которых нельзя однозначно отнести к тому или иному предприятию, необходимо закрепить за несколькими предприятиями и установить для каждого плановые значения на стратегический период, суммарные их значения дадут плановое значение по всему холдингу.

Предлагается в основу технологии распределения одного КРІ по показателям предприятия применить конкурсные механизмы из теории активных систем.

На уровне холдинга осуществляется анализ КФУ: какие предприятия, входящие в холдинг, могут “отвечать” за отдельные КФУ и, как следствие, за реализацию плановых значений КРІ.

КФУ холдинга можно разделить на 3 класса.

1 класс – КФУ, которые можно отнести к определённым предприятиям однозначно. Например, факторы успеха по направлениям деятельности холдинга, по которым работают отдельные предприятия холдинга.

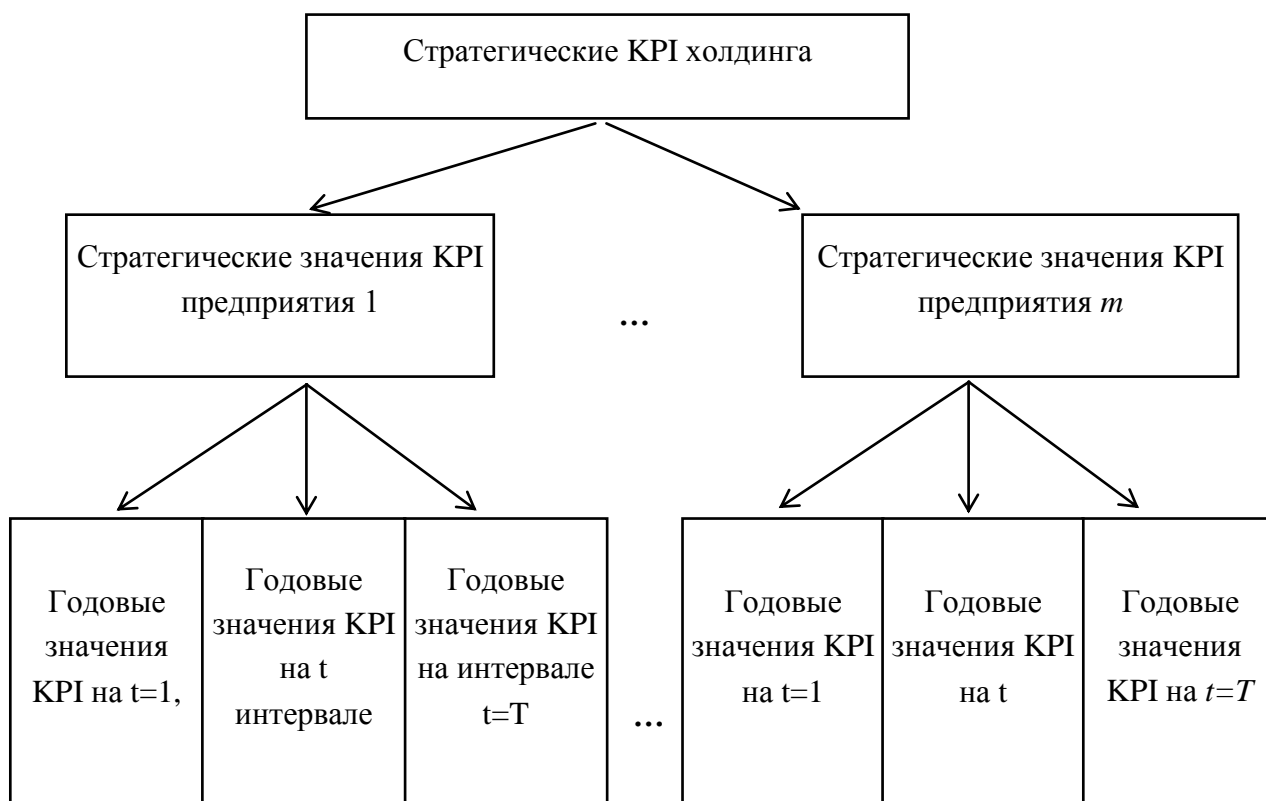
2 класс – КФУ, за которые “отвечают” несколько предприятий, т.е. только совместные их усилия позволят достигнуть плановых показателей КРІ, полученные на основе этих КФУ.

3 класс – КФУ, которые являются синергическим результатом работы всех предприятия, всего холдинга, например, повышение узнаваемости бренда холдинга (который относится ко всем предприятий).

ЭТАП 2. Декомпозиция стратегической карты холдинга на стратегические карты предприятий, входящих в холдинг, и каскадирование КРІ.

Сформируем задачу декомпозиции стратегической карты холдинга на стратегические карты предприятий холдинга. Дано: стратегическая карта холдинга  $SM_T^H$ . Определяются множество КРІ для каждого предприятия, их нормативные значения (рис. 24.1).





**Рис. 24.1. Иерархия системы сбалансированных показателей холдинга**

Предлагается для первого года стратегического периода определять плановые значения KPI, на второй и третий год определять интервальные оценки плановых значений, на последующие года строить функции принадлежности для KPI.

Соответственно будет построены бюджеты этих предприятий по интервалам – по годам планового стратегического периода:

1) первый год – бюджет на основе каскадирования установленных плановых значений KPI,

2) на второй и третий год строится тактический бюджет на основе каскадирования интервальных оценок KPI, на последующие – стратегический бюджет, который содержит бюджетные показатели, полученные за счет каскадирования KPI, описываются функцией принадлежности.

### **3. Процесс стратегического планирования на уровне предприятия, входящего в холдинг.**

Предприятие рассматривается как четырёхуровневая система: Центр (руководство предприятия) – бизнес-процессы – структурные подразделения – персонал. Такое разделение по уровням соответствует процессному подходу в

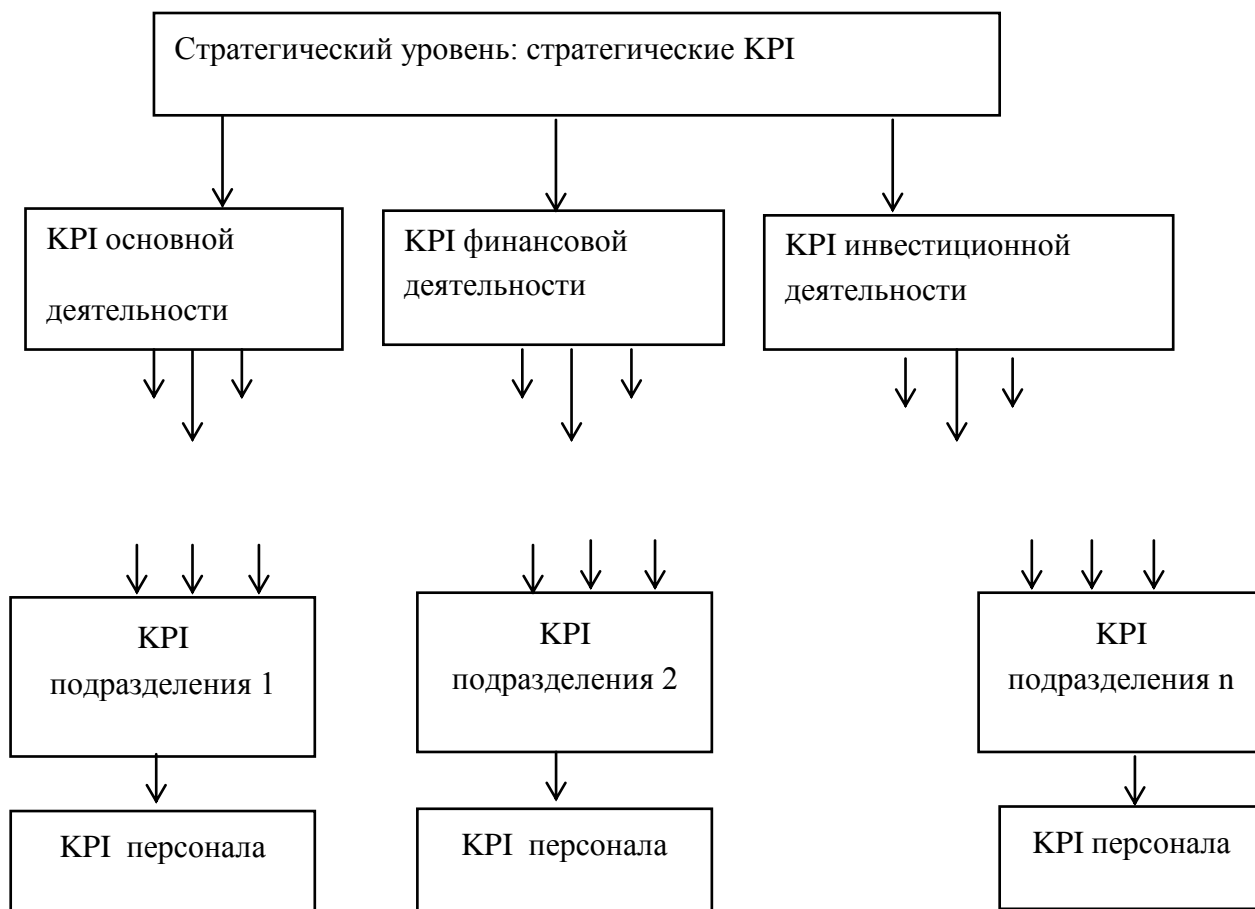
управлении. Система управления рассматривается как совокупность управления бизнес-процессами:

– управленческие бизнес-процессы, которые образуют верхний уровень. Здесь формируются стратегические цели, строится стратегическая карта, определяются ключевые показатели на стратегический период (при рассмотрении холдинговой структуры эти показатели определяются ГК);

– основные и вспомогательные бизнес-процессы, которые охватывают текущую деятельность предприятия. Основные бизнес-процессы осуществляются в рамках операционной деятельности, вспомогательные – обеспечивают непрерывность основных процессов, в качестве обеспечивающих процессов будем рассматривать бизнес-процессы инвестиционной и финансовой деятельности предприятия.

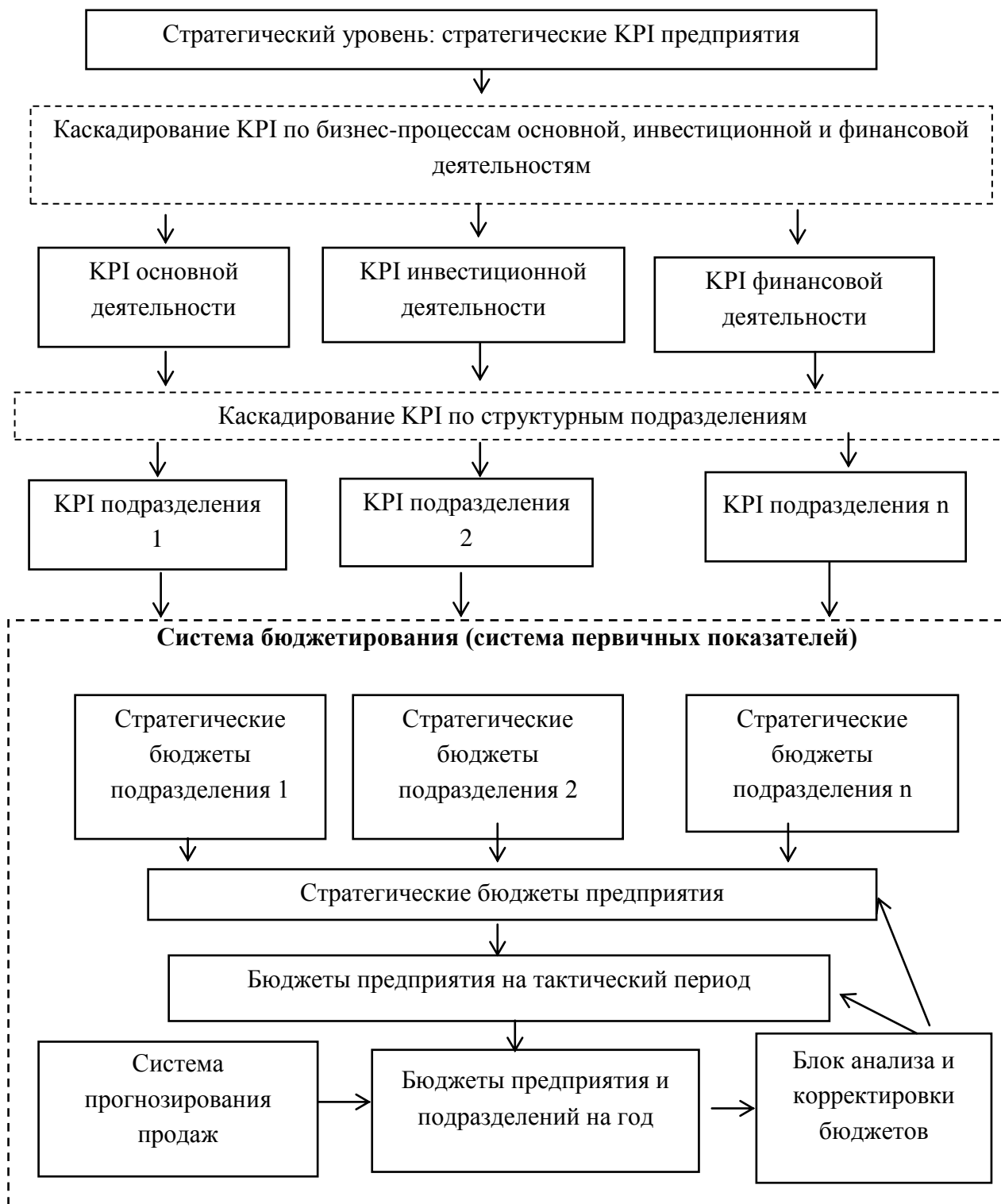
На третьем уровне располагаются структурные подразделения, персонал которых являются исполнителями процедур операционных бизнес-процессов.

Тогда система ключевых показателей должна охватывать все уровни управления (рис. 24.2).



*Рис. 24.2. Иерархия системы KPI*

Процесс построения такой иерархической системы будем называть каскадированием. КРІ основной, инвестиционной и финансовой деятельности – это итоговые индикаторы соответствующих бизнес-процессов. На рис. 24.3 показана иерархия показателей деятельности от стратегических КРІ до показателей бюджета.



*Рис. 24.3. Каскадирование КРІ до показателей бюджетов предприятия*

Таким образом, необходимо провести каскадирование стратегических КРІ предприятия до стратегических КРІ подразделений, а затем до показателей бюджетов, как основных плановых документов годовой и операционной отчётности. В качестве структурных подразделений будем рассматривать ЦФО – центры финансовой ответственности, для которых составляются бюджеты. Эти ЦФО в свою очередь могут состоять из множества подразделений, но в работе определение КРІ для них не рассматривается. Ограничиваемся определением КРІ для подразделений, которые имеют свои бюджеты и определённую финансовую отчётность. Тогда система бюджетирования представлена не с точки зрения функционального содержания (бюджет продаж, бюджет себестоимости и т.п.), а как совокупность бюджетов предприятия и бюджетов его отдельных структурных подразделений. Переход от стратегических КРІ видов деятельности до КРІ подразделений осуществляется через декомпозицию КРІ на первичные показатели, т.е. показатели, которые отражаются в управленческой отчётности и бюджетах.

Таким образом, возникает следующая задача декомпозиции: дано множество стратегических показателей – КРІ по видам деятельности. Необходимо осуществить декомпозицию их по процедурам соответствующих бизнес-процессов и выразить их в метриках первичных показателей. Далее определить их стратегические значения. В результате каскадирования предлагается сформировать следующую систему бюджетов на 3 подпериода:

**1-й подпериод:** численные значения бюджетных показателей по всем бюджетам, т.е. бюджеты формируются как годовые плановые документы;

**2-й подпериод:** формируются бюджеты, в которых значения бюджетных показателей представлено интервально (используется для вычисления функция принадлежности);

**3-й подпериод:** бюджеты представляют собой прогноз показателей бюджетов.

На каждом из периодов будут решены задачи каскадирования показателей разными методами. Однако предварительно необходимо решить задачу определения значений стратегических КРІ на каждый период, как целевые установки.

#### **4. Структура ЕРМ-системы предприятия**

Структура ЕРМ-системы предприятия, входящего в холдинг, представлена на рис. 24.4. [66]. Предлагается внести в аналитическую инфраструктуру модуль, который бы реализовывал технологию каскадирования.



**Рис. 24.4. Технологическая структура EPM системы предприятия**

Модули каскадирования объединяются в отдельную подсистему, что позволит вносить коррективы в алгоритмы, которые реализованы в этих модулях. В результате работы подсистемы стратегического планирования будет сформирован стратегический план, который затем подлежит декомпозиции до годовых планов с помощью реализации модулей подсистемы каскадирования.

Систему бюджетирования необходимо рассматривать на двух уровнях: на стратегическом и оперативном.

На стадии разработки информационного и программного обеспечения для связи с другими информационно-аналитическими системами используют ETL – технологии. ETL (Extract, Transform, Load) – это системы корпоративного класса, которые применяются, чтобы привести к одним справочникам и загрузить данные из нескольких разных учётных систем в Datawarehouse (хранилище данных) и EPM-систему [65].

**Заключение.** Предложена структура EPM–системы предприятия, которая дополнена модулем каскадирования KPI. В рамках этого модуля реализуется технология декомпозиции стратегических показателей деятельности до показателей бизнес-процессов основной, инвестиционной и финансовой деятельности и показателей бюджетов. EPM-система является частью корпоративной информационной системы предприятия.

## РОЗДІЛ 25

### ФОРМАЛІЗАЦІЯ ЗНАНЬ ПРО ПРОЦЕСИ ПІДГОТОВКИ ВИХІДНИХ ДАНИХ ПРИ ТЕСТУВАННІ

*Анотація.* В роботі запропонований підхід до формалізованого опису процесу визначення рівня знань студентів шляхом тестування. Враховуючи складність опису звичайними математичними моделями нечіткої, неповної і різномірної інформації, запропоновано для визначення результатів тестування використання методів штучного інтелекту, а саме використання нечітких мір і множин. Еталонні та поточні результати тестування представляються у вигляді нечітких LR-інтервалів, а процес логічного виводу на формалізованих знаннях організовується за допомогою неоднорідної функціональної мережі.

*Ключові слова:* тестування, формалізований опис, знання, нечіткі множини, LR-інтервал, початкова умова, функція приналежності, ознака, еталонні значення, поточні значення, функціональна модель.

*Abstract.* The paper presents a formal description process to determine the level of students' knowledge by testing. Vrahovuchy complexity of conventional mathematical models describing fuzzy, incomplete and heterogeneous information by applying the test results to determine the use of artificial intelligence methods, such as using weights and fuzzy sets. Reference and current test results are represented as LR-fuzzy intervals, and the process of inference on formalized knowledge is organized with the help of functional heterogeneous network.

*Keywords:* testing, formal description, knowledge, fuzzy sets, LR-interval, initial condition, membership function, feature, reference value, current value, functional model.

**Вступ і постановка завдання.** В теперішній час для оцінки рівня знань тих, хто навчається, все більше використовуються тести: тестування в середній школі, зовнішнє незалежне оцінювання, перевірка знань студентів за кожною темою, допуск до лабораторної роботи, проведення екзамену і т.д. При цьому, загалом, використовуються наступні типи тестів: множинний або альтернативний вибір, встановлення відповідності або послідовності, вільне формулювання відповіді, доповнення та ін.

Аналізуючи зазначені типи тестів, в них можна виділити дві складові результатів – кількісні і якісні (семантичні). Перевіряючому необхідно порівняти результати тестування з еталонними (експертними) значеннями. При цьому, як кількісні так і семантичні відповіді можуть мати дуже широкі межі відхилення від еталонних. Результати тестування не завжди можна

описати звичайними математичними методами. Таким чином виникає проблема якісної формалізації результатів тестування.

Для формалізації процесу підготовки вихідних даних при оцінці рівня знань студентів доцільно використовувати методи штучного інтелекту, а саме методи, засновані на використанні нечітких мір та множин. До таких мір відносяться міри можливості, необхідності, базисної ймовірності [22].

Нечітка множина – множина з нечіткими границями, коли перехід від належності елементів множині до не належності їх множині відбувається не різко, а поступово. В класичній логіці елемент  $X$  із відповідної предметної області належить або не належить деякій множині  $M$ . Характеристична функція належності елемента множині приймає тільки два значення: 1, коли  $X$  дійсно належить  $M$ , і 0, коли  $X$  не належить множині  $M$ .

В нечітких множинах елемент  $X$  належить множині  $M$  з відповідним ступенем належності, який приймає значення в інтервалі від 0 до 1. Даний математичний апарат застосовується при розпізнаванні образів в інформаційних системах. Використання безперервної шкали оцінок дозволяє приймати рішення щодо наявності (відсутності) події на основі неточної, неповної й різномірної інформації, суперечливості знань і даних.

З погляду формалізованого подання для забезпечення наступної алгоритмізації та програмної реалізації, завдання автоматизації процесу підготовки вихідних даних при тестуванні студентів включає:

- змістовний опис процесу тестування студентів, самих тестів та еталонних відповідей на поставленні питання;
- формалізацію знань про процеси тестування;
- організацію логічного виводу на формалізованих структурах знань.

Перелік розв'язуваних при цьому завдань впливає з аналізу змісту процесу організації навчання, що полягає, з одного боку, у розпізнаванні варіантів дій того, якого навчають, з іншого боку – у рішенні приватних завдань, кожна з яких впливає на управління процесом тестування.

*Ціллю та завданнями даної роботи є розробка формально-логічного апарату для опису знань про процеси тестування, який дозволить якісно визначати рівень знань студентів в умовах неточності, невизначеності та різномірності вхідної інформації.*

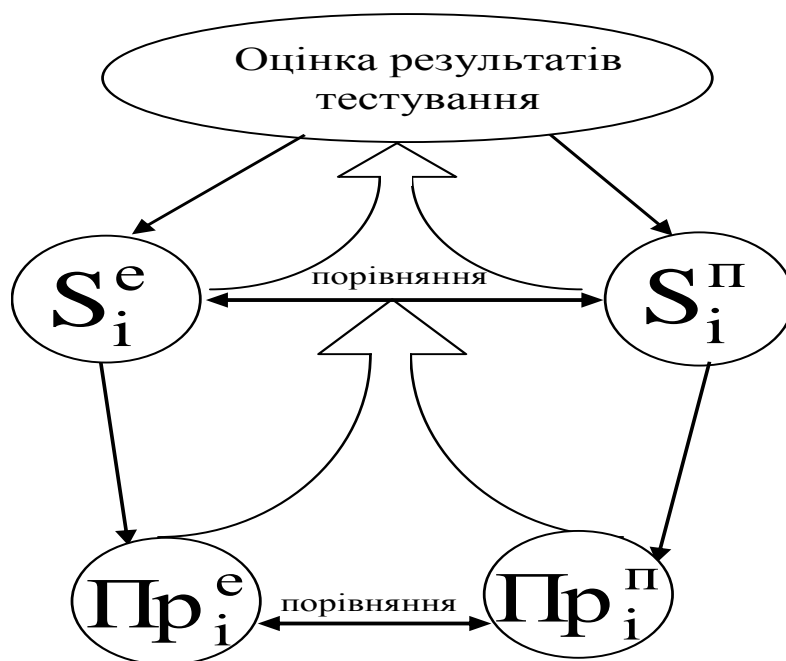
**Основна частина.** Процес оцінки варіантів відповідей в процесі тестування може бути зведений, з одного боку, до розпізнавання й оцінки множини подій, що відбуваються в різні моменти часу  $t_i$ ; визначенню їх кількісних і якісних характеристик; порівнянню результатів з тими, що були



виділені на етапі розробки тестів. При цьому під подією будемо розуміти миттєву зміну деяких характеристик процесу навчання в результаті дій того, якого навчають, викладача, впливу зовнішніх факторів або природної еволюції. Множина приватних завдань процесу тестування після декомпозиції формально можна представити множиною початкових умов (ПУ).

Під початковою умовою (ПУ) будемо розуміти формальний опис події в процесі навчання, що визначає множину необхідних і достатніх умов для реалізації процедур управління навчальним процесом [27]. Тоді, якщо вважати, що кожне ПУ (підмножина ПУ) є формальне подання деякої події (групи подій), то множиною ПУ можна визначити один з варіантів відповідей того, якого навчають, виділених на етапі підготовки тестів. При такому підході під розпізнаванням подій можна мати на увазі інтерпретацію ПУ в процесі тестування.

При оцінці результатів процесу тестування множина ПУ (рис. 25.1) інтерпретується на системі поточних ознак  $\{Pr_j^p\}$ , де  $j=1, \dots, J$  – номер ознаки,  $J$  – кількість ознак, що надходять від основних джерел інформації (тих, яких навчають, викладача).



*Рис. 25.1. Ілюстрація до порядку оцінки результатів тестування.*

При цьому поточна оцінка процесу (результатів) тестування формально може бути зведена до визначення відповідності між:

з одного боку – подіями (ПУ), виділеними на етапі розробки тестів  $S_i^e$ , ( $i = 1, \dots, I$  – номер події (ПУ),  $I$  – кількість подій (ПУ)), які формально можуть бути представлені системою еталонних значень ознак  $\{Pr_i^e\}$ , сформованої викладачами;

з іншого боку – подіями  $S_i^p$ , що відбивають поточний стан процесу тестування й значеннями, що інтерпретуються, поточних  $\{Pr_i^p\}$  ознак, що надходять в результаті відповідей на тести.

Аналіз змісту процесу тестування дозволив виділити наступні типи розв'язуваних при цьому завдань: логіко-аналітичні, розрахункові, пошукові. Кожний з типів завдань можна привести до відповідного класу ПУ. Тоді метод інтерпретації відповідного ПУ буде визначатися методом рішення певних завдань. З огляду на характер цих завдань і особливості їх рішення, а, отже, і інтерпретації ПУ, пропонується ввести наступну класифікацію ПУ [27]:

функціональні ПУ – забезпечують рішення завдань логіко-аналітичного характеру (на кожне питання той, хто навчається, повинен запропонувати свою відповідь: дописати слово (словосполучення), знак, формулу та ін.);

розрахункові ПУ – забезпечують проведення розрахунків для визначення кількісних характеристик розпізнаних ситуацій (необхідно провести обчислення, порівняти отриманий результат та запропоновані відповіді);

пошукові ПУ – забезпечують організацію пошуку й вибірки необхідної інформації з бази даних (знань) відповідно до деяких критеріїв або характеристик (кожне питання супроводжується готовими варіантами відповідей, з яких необхідно вибрати одну або декілька правильних).

Формально ПУ можна представити кортежем:

$$N_k = \langle S_k, T_k, Pr_k, \bar{P}_k, R_k \rangle,$$

де  $S_k$  – семантичний опис ПУ;  $T_k$  – тип ПУ;  $Pr_k$  – процедура визначення істинності (розпізнавання, означування) ПУ;  $\bar{P}_k$  – множина вихідних даних для означування ПУ;  $R_k$  – результат означування ПУ. У свою чергу, результатами означування ПУ можуть бути: тип розпізнаної події; значення функції, отриманої в результаті чисельного розрахунку, результати пошуку необхідної інформації.

Слід відмітити, що організація логічного виводу для різних початкових умов має свої особливості. Так означення пошукових ПУ полягає в

присвоюванні результатів пошуку або узагальнення інформації. Означення розрахункових ПУ передбачає запуск процедур, які реалізують деякі алгоритми обчислень. Результатом означення функціональних початкових умов є отримання одного із двох значень “істина” або “хибність”.

Формулювання й постановка цілей перевірки рівня знань є одним основних етапів цільового планування й управління в навчанні. Всю множину цілей процесу навчання (надалі – цілей) можна розділити на основні, допоміжні й проміжні [29].

Основні цілі визначають шляхи вирішення завдань оцінки рівня знань студента.

Допоміжні – це такі цілі, формулювання яких не входить безпосередньо в опис процесу тестування, але їх виконання є необхідною умовою для реалізації системи навчання або забезпечення нормального ходу процесу тестування.

Проміжні цілі виникають у ході підготовки й виконання дій для досягнення основних цілей і виражають необхідні або достатні умови одержання необхідних результатів навчання.

Процес тестування при відповідній декомпозиції може бути розбитий на логіко-аналітичні, розрахункові та пошукові завдання.

Формалізацію логіко-аналітичних завдань процесу тестування будемо здійснювати за рахунок інтерпретації відповідних початкових умов. При цьому процес інтерпретації зведемо до вирішення задачі розпізнавання в її розширеному розумінні. Суть такого розпізнавання полягає у віднесенні подій, що розпізнаються, до деякого класу еталонних подій відповідного алфавіту, сформованого на етапі підготовки тестів, і подальшого визначення кількісних та якісних характеристик цих подій експертами.

Оцінка істинності функціональних ПУ може бути розбита на наступні етапи:

- 1) інтерпретація значень поточних ознак;
- 2) інтерпретація подій, складених на основі описів ознак;
- 3) визначення кількісних і якісних характеристик подій і варіантів дій при тестуванні.

Перший етап передбачає визначення ступеня близькості еталонного та поточного опису ознак. Це дозволяє врахувати інформаційну цінність ознак при описанні конкретної ситуації. Ця процедура в нашому випадку різна для кількісних і якісних ознак. Для кількісних ознак – це ступінь близькості еталонного та поточного розподілу їх значень, для якісних – ступінь узгодженості еталонного і поточного розподілів базисної ймовірності [28].

Оцінка ступеня близькості кількісних ознак можна визначити наступним чином:

$$v(\delta, \xi) = \frac{V(\delta, \xi) + N(\delta, \xi)}{2},$$

де:  $\delta$  – базовий (еталонний) розподіл можливих значень ознаки;

$\xi$  – поточні ознаки;

$V(\delta, \xi)$  та  $N(\delta, \xi)$  – міри можливості і необхідності відповідно, які використовуються для оцінки ступеня узгодженості твердження  $\delta$  відносно  $\xi$ .

Оцінка ступеня близькості якісних ознак можна визначити наступним чином:

$$\pi(\delta, \xi) = 1 - m_0,$$

де:  $m_0 = \sum_{E_i \cap E_j} m_1(E_i)m_2(E_j)$  – ступінь неузгодженості розподілу базисних ймовірностей.

Інтерпретація подій (другий етап) припускає згортку приватних значень ступеня близькості поточного та еталонного розподілів значень ознак, отриманих на попередньому етапі [29; 64]:

$$\begin{cases} v_{A \cap B}(x_m) = \min\{v_A(x_m), v_B(x_m)\} \\ v_{A \cup B}(x_m) = \max\{v_A(x_m), v_B(x_m)\} \end{cases}$$

де  $v_A(x_m)$ ,  $v_B(x_m)$  – приватні значення ступеня близькості еталонного та поточного значень ознаки  $x_m$ .

Під класом будемо розуміти довільну сукупність подій (об'єктів), що характеризуються яких-небудь набором загальних властивостей або ознак [29].

Під алфавітом класів  $A_r = \{k_1^r, k_2^r, \dots, k_n^r\}$  будемо розуміти непересічні підмножини класів, що описують ту чи іншу подію в процесі навчання, де  $r = 1, \dots, R$  – порядковий номер алфавіту;  $R$  – кількість необхідних алфавітів для оцінки результатів процесу навчання;  $n$  – кількість класів в алфавіті.

Моделі розпізнавання класів можуть включати допоміжні мережі з проміжними алфавітами, які описуються своїм набором ознак, що не перетинаються з ознаками, які безпосередньо використовуються в описі алфавіту верхнього рівня. Кожний проміжний алфавіт для алфавіту більш

вищого рівня вважається складовою якісною ознакою. Розподіл базисної ймовірності, отриманий від значень оцінок подібності класів проміжного алфавіту буде вважатися поточним розподілом даної якісної ознаки.

По відношенню до початкових умов, які містять первинний розподіл значень ознак, розподіл проміжнових алфавітів є вторинними, так як для їх отримання необхідно провести додаткові обчислення.

Оцінка ступеня близькості  $\Phi(k_n^r)$  аналізованого події до  $n$ -го класу  $r$ -го алфавіту визначається за найбільшим значенням функції приналежності інтерпретованих подій для кожного з класів [5]:

$$\Phi(k_n^r) = \max_{\forall i \in I_n} \left[ \min_{\forall k \in K_n} \{v_n^r(x_{ik})\}, \min_{\forall s \in S_n} \{\pi_n^r(x_{is})\} \right],$$

де  $n$  – порядковий номер класу у відповідному алфавіті подій;  $M_n$  – кількість ознак або груп ознак, об'єднаних диз'юнктивній вершиною класу;  $k$  – порядковий номер кількісного ознаки;  $K_n$  – кількість номерів кількісних ознак  $n$ -го класу;  $s$  – порядковий номер якісного ознаки;  $S_n$  – кількість номерів кількісних ознак  $n$ -го класу;  $i$  – порядковий номер групи ознак;  $I_n$  – кількість груп ознак для опису  $n$ -го класу.

Для прийняття рішення про клас події будемо використовувати правило найбільшого значення функції можливості віднесення подій до класів  $r$ -го алфавіту [64]:

$$k_n^* = \arg \max_{\forall k_n} (\Phi(k_n^r) / \Phi(k_n^r) > \delta_{\text{пор}}),$$

де  $k_n^*$  – клас з максимальним значенням заходи можливості;  $\delta_{\text{пор}}$  – порогове значення, вибирається в залежності від розрізнення класів алфавіту.

Третій етап оцінки істинності ПУ припускає реалізацію процедур пошуку і розрахунку з метою визначення кількісних та якісних характеристик розпізнаних подій. Процедура пошуку полягає у формуванні множини відображень значень ознак в базі даних (пошук відповідних інформаційних полів по ключу і значень знайдених входжень):

$$G : \{ \text{Пр}_i \rightarrow \Pi_{N_{\text{ПЗ}}}^{N_{\text{Б}}} (v_{x_i}) \},$$

де  $G$  – множина відображень значень ознак в базі знань (даних) (БЗ);  $\Pi_{N_{ПЗ}}^{N_B}(V_{x_i})$  – поле запису  $x_i$ -го об'єкта в бібліотеці БЗ;  $N_{ПЗ}$  – ім'я поля запису;  $N_B$  – ім'я бібліотеки БЗ;  $V_{x_i}$  – кортеж опису  $x_i$ -го об'єкта в БЗ.

Інтерпретація розрахункових ПУ припускає організацію обчислювального процесу для реалізації алгоритмів, заснованих на відомих математичних методах і методиках. Аналогічно можуть реалізовуватися і процедури пошуку інформації в базі знань (даних) – для інтерпретації пошукових ПУ.

З огляду на вимоги до формалізації різнорідних, неточних і неповних знань, будемо формальний опис класів здійснювати методами, заснованими на положеннях теорії нечітких мір і множин. При цьому, опис деякого класу подій може бути представлено у вигляді формули на мові булевої алгебри [29; 64]:

$$k = \bigcup_{i=1}^I \bigcap_{j=1}^J X_{ij},$$

де  $X_{ij}$  – ознака, що характеризує деякий властивість події;  $i$  – порядковий номер групи ознак;  $I$  – кількість груп ознак;  $j$  – порядковий номер ознаки;  $J$  – кількість ознак у групі.

Всю сукупність ознак об'єктів (ситуацій) розділимо на кількісні та якісні. При цьому під кількісними розуміються ознаки, для яких можливе введення метрики на множина допустимих значень, що дозволяють дати кількісну оцінку тому чи іншому значенню ознаки.

Під якісними ознаками розуміються ознаки, що визначають семантичне опис властивостей ситуацій, що склалися в проблемній області. До якісних ознак належать, наприклад, ознаки, що описують тип тих, яких навчають, тип, форму, зміст джерел інформації та ін.

У загальному випадку формалізований опис цілей тестування може включати деяку множину формул, пов'язаних між собою логічними операціями диз'юнкції і кон'юнкції.

Логічна послідовність досягнення цілей визначається відносинами між ними. Ці відносини можуть бути розділені на відносини підпорядкування, передування і дії [64].

Відносини підпорядкування цілей тестування  $x$  і  $y$  (позначимо  $xPy$ ) визначають необхідні і достатні умови досягнення мети. Формально відносини підпорядкування можна представити у вигляді:

$$xPy \equiv (x \in \Phi_y) \vee (\neg x \in \Phi_y) \vee (\neg x \wedge \neg y \rightarrow \neg M \uparrow_y),$$

де  $\Phi_y$  – множина складових мети тестування  $y$ ;  $M \uparrow$  – оператор “можливо в майбутньому”, а останній диз’юнктивні член висловлює необхідність  $x$  для досягнення  $y$ .

Відносини передування ( $xTy$ ) визначають послідовність досягнення вищих цілей системи при досягненні цілей нижнього рівня протягом деякого проміжку часу.

Відношення передування визначається наступним чином: мета  $x$  передує цілі  $y$ , якщо існує така ситуація  $\alpha$  і момент часу  $t$  для яких: якщо  $x$  істинно, то  $y$  помилково в цій ситуації і в цей момент часу. Формально це може бути виражене твердженням:

$$xTy \equiv \exists \alpha_1 \exists t_1 \exists \alpha_2 \exists t_2 [x(\alpha_1, t_1) \wedge \neg y(\alpha_1, t_1) \wedge y(\alpha_2, t_2) \wedge (t_2 > t_1)],$$

де  $x(\alpha, t)$ ,  $y(\alpha, t)$  – позначають значення істинності формул  $x$ ,  $y$  в ситуації  $\alpha$  і момент часу  $t$ .

Відносини дії ( $xDy$ ) характеризують дії системи з множини  $D$ , що визначають її перехід з одного стану в інший і є необхідними умовами такого переходу.

Всі типи відносин між цілями є відносинами суворого часткового порядку і мають властивості:

- 1) антирефлексивності –  $(xRx)$ ,
- 2) антисиметричних –  $(xRy) \rightarrow \neg(yRx)$ ,
- 3) транзитивність –  $(xRy) \vee (yRz) \rightarrow (xRz)$ ,

де  $R$  – одне з розглянутих відносин.

Розглянемо засоби формалізації і правила визначення ознак інформації для вирішення логіко-аналітичних завдань навчання.

Визначення варіантів дій при тестуванні може бути зведене до оцінки відповідності між виділеною на етапі розробки тестів множиною еталонних відповідей  $S_i^e$  і реальними відповідями  $S_i^n$ . Рішення даної задачі на початковому етапі передбачає визначення ступеня відповідності еталонних та поточних значень ознак.

Для цього необхідно:

1) формально представити значення еталонної та поточної ознакової інформації;

2) визначити правила порівняння (відповідності) значень еталонних та поточних ознак для розпізнавання правильних відповідей, виділених на етапі розробки тестів.

Для формального представлення знань про закономірності прояви значень поточної ознакової інформації пропонується використовувати метод формалізації, заснований на використанні положень нечітких мір і множин [22].

З огляду на прийнятий підхід до визначення класів подій, доцільно ввести деяку функцію, яка характеризує інформативність ознак.

В якості такої функції для кількісних ознак виберемо функцію приналежності  $\mu_n^r(x_m)$  з областю визначення на інтервалі  $[0, 1]$  і характеризує закономірності прояви значень ознаки  $x_m$  при описі  $n$ -го класу  $r$ -го алфавіту учасників процесу тестування.

Для формального опису кількісних ознак обґрунтовано використання нечітких LR-інтервалів:

$$\mu_n^r(x_m) = \langle A, B, C, D \rangle,$$

де  $B, C$  – відповідно нижнє і верхнє модальні значення ознаки (межі ядра нечіткого множини з найбільш можливим значенням ознаки);  $A, D$  – лівий і правий коефіцієнти нечіткості (найменше та найбільше значення ознаки).

Існує декілька видів представлення функції приналежності. В даній роботі запропоновано використання LR-інтервалів трапецоїдальної форми.

Аналітично функцію  $\mu_n^r(x_m)$  можна записати у вигляді [64]:

$$\mu_n^r(x_m) = \begin{cases} 0, & \text{если } E_m \leq A; \\ \frac{E_m - A}{B - A}, & \text{если } A < E_m < B; \\ 1, & \text{если } B < E_m < C; \\ -\frac{E_m - D}{D - C}, & \text{если } C < E_m < D; \\ 0, & \text{если } E_m \geq D, \end{cases} .$$

де  $E_m$  – оцінка значення ознаки  $x_m$ .



Формалізований опис якісних ознак може бути представлено функцією можливості  $\pi_n^r(x_m)$  з областю визначення на інтервалі  $[0, 1]$ , що визначає ступінь можливості віднесення об'єкта (події) до  $n$ -го класу  $r$ -го алфавіту за значенням ознаки  $x_m$ .

Формалізація ознак за допомогою LR-інтервалів має наступні переваги [22]: можливість одночасного представлення даних зі стохастичною та не стохастичною невизначеністю в рамках єдиного формально-логічного апарату; простота з погляду користувача; висока виразність з погляду описових можливостей; зручність наступної обробки на рівні програмних модулів.

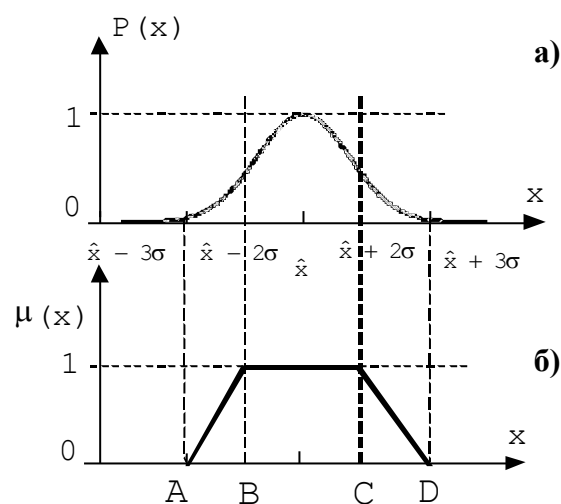
Кількісні ознаки характеризуються стохастичною невизначеністю (реалізація методів стохастичного оцінювання) і не стохастичною невизначеністю (поняття природної мови про значення кількісної ознаки).

Будемо вважати, що результати крапкового статистичного оцінювання закономірностей прояву значення ознаки знань студента розподілені по нормальному закону (рис. 25.2).

Тоді параметри нечіткого LR-інтервалу можна обчислити таким чином:

$$B = \hat{x} - 2\delta_{\hat{x}}; C = \hat{x} + 2\delta_{\hat{x}}; A = D = \hat{x} - \delta_{\hat{x}}$$

де  $\hat{x}$  – поточне значення ознаки об'єкту;  $\delta_{\hat{x}}$  – середньоквадратичне відхилення засобу вимірювання ознаки об'єкту.



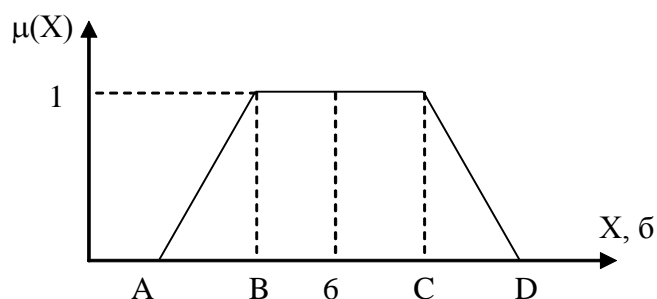
**Рис. 25.2. Опис проявів значень ознак за допомогою:**  
*а) нормального закону, б) нечіткого LR-інтервалу*

Оцінку значень кількісної ознаки з нестохастичною невизначеністю можна представити у виді нечіткої змінної, котра задається набором з 3-х компонентів [29]:

$$\langle Id, X, R(Y, X) \rangle,$$

де  $Id$  – ім'я ознаки;  $X$  – множина припустимих значень ознаки;  $R(Y, X)$  – нечітка множина, визначена на множині  $X$  та нечітке обмеження, що представляє собою, числове значення ознаки  $X$ , обумовлене лінгвістичним значенням  $Y$ .

Наприклад, для ознаки “уміння студента” з множиною припустимих значень  $X = [1, 12]$ , числовим значенням  $X = 6$  і лінгвістичним значенням  $Y =$  “біля” нечітка множина може мати наступний вид (рис. 25.3):



*Рис. 25.3. Представлення у вигляді LR-інтервалу значення ознаки "біля 6"*

Параметри LR-інтервалів визначати таким чином:

$$A = k_A(Y) \cdot |x|$$

$$B = x + k_B(Y) \cdot |x|$$

$$C = x + k_C(Y) \cdot |x|$$

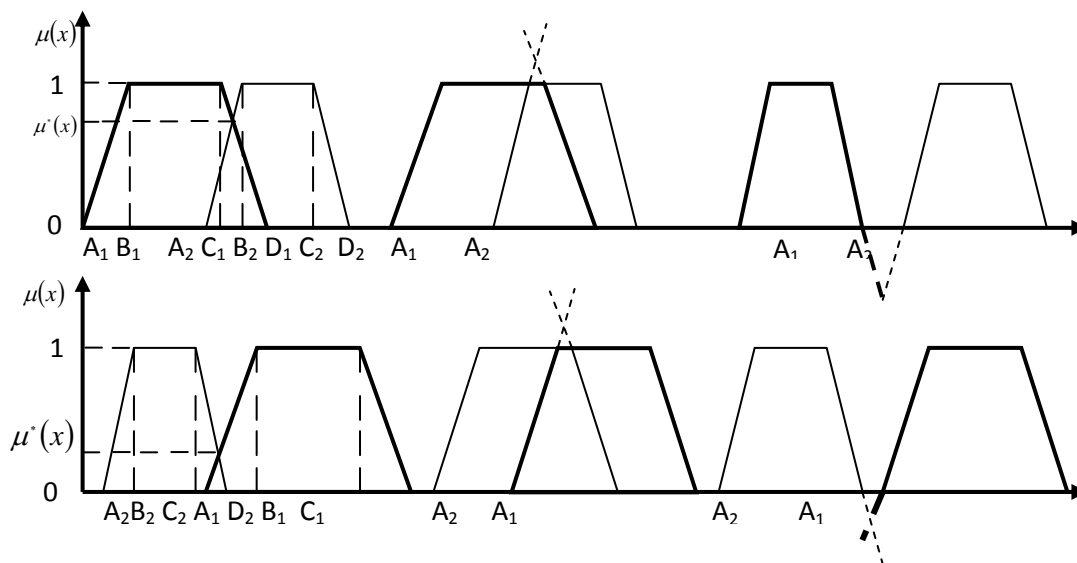
$$D = k_D(Y) \cdot |x|$$

де  $|x|$  – потужність множини  $X$ ;  $k_A, k_B, k_C, k_D$  – числа з інтервалу  $[-1,1]$ , що характеризують процентну частину  $|x|$  для кожного лінгвістичного значення  $Y$ .

Таким чином, формалізація кількісних даних про значення ознак полягає в побудові нечітких LR-інтервалів множини припустимих значень ознак, використовуючи при цьому статистичні дані (стохастична невизначеність) і знання експертів (нестохастична невизначеність).

Для прийняття рішення про близькість об'єкту до одного з класів, необхідно провести перевірку еталонних й поточних розподілів значень ознак та визначити їх ступінь близькості  $\mu^*(x)$ .

LR-інтервал є рівнобіною трапецією. Тому можливі два різних положення еталонної та поточної ознаки відносно одна одної: у першому випадку у всіх можливих взаємних положеннях розподілів  $A_2 > A_1$ ; у другому випадку у всіх можливих взаємних положеннях  $A_2 < A_1$  (рис. 25.4):



**Рис. 25.4. Можливі варіанти перетинання еталонного й поточного розподілів деякої ознаки**

При знаходженні точки перетинання двох розподілів необхідно в першому випадку розглянути прями  $A_1D_1$  і  $A_2B_2$ , у другому випадку –  $C_2D_2$  і  $A_1B_1$  і вирішити для них системи рівнянь.

Для першого випадку:

$$\begin{cases} Y_1 = -\frac{x_1 - d_1}{d_1 - c_1} \\ Y_2 = \frac{x_2 - a_2}{b_2 - a_2} \end{cases} \Rightarrow \begin{cases} Y_1(d_1 - c_1) = d_1 - x_1 \\ Y_2(b_2 - a_2) = x_2 - a_2 \end{cases}$$

Оскільки значення функції в точці перетину однакове, то можна записати  $Y_1 = Y_2 = Y$ . Тоді:

$$\begin{aligned}
Y(b_2 - a_2) + a_2 &= d_1 + Y(d_1 - c_1) \\
Y(b_2 - a_2 + d_1 - c_1) &= d_1 - a_2 \quad . \\
Y &= \frac{d_1 - a_2}{(b_2 - a_2 + d_1 - c_1)}
\end{aligned}$$

Аналогічно для другого випадку:

$$\begin{cases} Y_1 = \frac{x_1 - a_1}{b_1 - a} \\ Y_2 = \frac{x_2 - d_2}{d_2 - c_2} \end{cases} \Rightarrow \begin{cases} Y_1(b_1 - a_1) = x_1 - a_1 \\ Y_2(d_2 - c_2) = d_2 - x_2 \end{cases} .$$

$$\begin{aligned}
Y(b_1 - a_1) + a_1 &= -Y(d_2 - c_2) + d_2 \\
Y(b_2 - a_1 + d_1 - c_1) &= c_2 - a_1 \quad . \\
Y &= \frac{c_2 - a_1}{b_2 - a_1 + d_1 - c_1}
\end{aligned}$$

Якщо  $Y \leq 0$ , то оцінці ступеня відповідності ознаки еталонному слід присвоїти значення, рівне нулю. В інших випадках – значенню  $Y$ .

Таким чином, правило визначення  $\mu^*(x)$  можна записати у вигляді:

$$\mu^*(x) = \begin{cases} 1, Y \geq 1 \\ 0, Y \leq 0 \\ \frac{c_2 - a_1}{b_2 - a_1 + d_1 - c_1}, a_1 > a_2 \quad . \\ \frac{d_1 - a_2}{b_2 - a_2 + d_1 - c_1}, a_2 > a_1 \end{cases}$$

Отримані правила визначення  $\mu(x)$  дозволяють визначити ступінь близькості еталонного й поточного розподілів значень ознаки.

Для формалізованого опису процесу тестування будемо використовувати наступну множину ознак [64]:

$$X = \{ NW, T, T_d, MO, MO_d, R, R_d, O, O_d \},$$

де  $NW$  – кількість питань в  $TV_{kn}$  тесті;  $T$  – час, припустимий на тестування в  $TV_{kn}$  тесті;  $T_d$  – результати порівняння припустимого й поточного часу на тестування в  $TV_{kn}$  тесті (“так”, “ні”),  $MO$  – кількість помилок, припустима в  $TV$

$_{kn}$  тесті,  $MO_d$  – результати порівняння припустимої й фактичної кількості помилок (“так”, “ні”) в  $TV_{kn}$  тесті,  $R$  – припустима кількість повторень питання в  $TV_{kn}$  тесті;  $R_d$  – результати порівняння припустимої й фактичної кількості повторень питання в  $TV_{kn}$  тесті;  $O$  – оцінка за  $TV_{kn}$  тест у заданій метриці;  $O_d$  – результати порівняння оцінки із припустимої для  $TV_{kn}$  тесту.

Формалізований опис еталонних розподілів значень кількісних і якісних ознак для опису результатів тестування можна представити у вигляді:

$$\Phi_n^r(X_m) = \langle \{\mu_n^r(X_m)\}, \{\pi_n^r(X_m)\} \rangle,$$

де  $\Phi_n^r(X_m)$  – множина еталонних розподілів значень ознак  $X_m$ ,  $m=1, \dots, M$  – номер ознаки (групи ознак),  $M$  – кількість ознак, використовуваних для опису типів тих, яких навчають;  $\mu_n^r(X_m) = \langle A, B, C, D \rangle$  – розподіл значень кількісних ознак при описі  $n$ -го класу  $g$ -го алфавіту, представлені у вигляді нечіткого LR-інтервалу;  $\pi_n^r(X_m)$  – розподіл функцій приналежності об’єкта (події) до  $n$  – му класу  $g$ -го алфавіту за значенням якісної ознаки  $X_m$ .

Використовуючи множину поточних ознак  $X_j = \{NW_j, T_j, T_{dj}, MO_j, MO_{dj}, R_j, R_{dj}, O_j, O_{dj}\}$ , що характеризують для кожного  $j$ -го учасника процесу навчання тест  $TV_{kn}$ ,  $j = \overline{1, J}$ , де  $J$  – кількість тих, яких навчають у групі, і правила подання значень ознак у вигляді нечітких множин, одержимо формалізований опис у вигляді функцій описів кількісних  $\mu(X_j)$  і якісних  $\pi(X_j)$  поточних ознак [64].

В роботі [64] запропоновано підхід до формування аксіом щодо визначення результатів тестування та використання неоднорідної ієрархічної функціональної мережі для організації логічного виводу, яка дозволяє ефективно і наглядно представити результати тестування на основі сформульованих правил.

**Висновки.** Використання положень теорії нечітких мір і множин, а саме нечітких LR-інтервалів дозволить врахувати нечіткість, неповноту і різномірність інформації під час тестування та приймати якісні рішення щодо рівня знань студентів.

Розроблений математичний опис визначення функції приналежності при різних значеннях розподілів поточних ознак дасть змогу автоматизувати процес тестування.

**Комп'ютеризовані технології та системи  
видавничо-поліграфічних виробництв та електронних  
мультимедійних видань**

**РОЗДІЛ 26  
ДОСЛІДЖЕННЯ ТЕХНОЛОГІЇ ПОБУДОВИ ТРИВИМІРНИХ  
МОДЕЛЕЙ**

***Анотація.** Розглянуто теоретичні та практичні проблеми побудови тривимірних моделей реальних об'єктів і приведено оригінальне вирішення науково-технічної задачі вдосконалення доступної технології тривимірного сканування та технології створення тривимірних моделей за допомогою цифрового фотоапарату. Зроблено аналіз та описано необхідні умови для створення якісних 3D моделей, враховуючи результати проведених досліджень.*

***Ключові слова:** 3D технології, безконтактне сканування, віртуальна модель, фотограмметрія*

***Abstract.** Considered theoretical and practical problems of building three-dimensional models of real objects and proposed the original solution of scientific and technical problem of improving the available three-dimensional scanning technology and technologies of creation three-dimensional models using a digital photo camera. Have been analyzed and described the necessary conditions to create quality 3D models, taking into account results of the conducted research.*

***Keywords:** 3D technology, noncontact scanning, virtual model, photogrammetry.*

**Вступ і постановка задачі.** Сучасне інформаційне суспільство потребує новітніх розробок, альтернативі минулого століття. На допомогу приходять 3D-технології. Технологія тривимірного друку поки що є екзотикою не тільки для України, але й для багатьох інших країн. Тим не менш, фахівці запевняють, що адаптація до цих пристроїв пройде дуже швидко. Більше того, вже сьогодні 3D принтери активно використовуються в медицині, авіації, аерокосмонавтиці і, навіть, харчовій індустрії.

Для друку необхідна модель, на сьогодні вже існує множина шляхів її отримання. Але найбільш популярними є моделювання об'єктів вручну в програмах-редакторах і сканування необхідних об'єктів за допомогою 3D сканера з подальшою обробкою в тих самих редакторах. У різний час переважав то один, то інший спосіб. Це залежало від появи на ринку нових

технологій оцифрування, алгоритмів візуалізації, нового програмного забезпечення, що дозволяє простіше й ефективніше обробляти 3D об'єкти.

Зараз можна говорити, що сканування та моделювання однаково популярні, однак ця рівність поступово зменшується, зважаючи на постійне вдосконалення технологій тривимірного сканування. До того ж, середньостатистичний користувач не володіє навичками 3D моделювання, для цього потрібно мати теоретичну й практичну базу знань та досвід. Тому для полегшення завдання та економії часу можна користуватись методом тривимірного сканування об'єктів. Проблема лише в тому, що для створення якісної 3D моделі необхідне використання професійних 3D сканерів, вартість яких є досить високою. Проте можливо самостійно створити простий, бюджетний аналог 3D сканера. Використовуючи такий вид створення тривимірних моделей не можна розраховувати на дуже високу якість та точність, але якщо детальність не є критичною, то бюджетне 3D сканування може стати вирішенням проблеми. Для максимального покращення якості сканованої моделі та успішного прототипування важливо звертати увагу на технічні характеристики і відповідність складових сканера, контролювати усі важливі фактори.

Ще одним проривом в області комп'ютерного моделювання є розробка 3D моделей за допомогою статичних фотографій, використовуючи цифрову камеру або будь-яку камеру телефону для захоплення об'єктів реального світу, як повністю текстурованих 3D моделей. Цей метод ідеально підходить для об'єктів складних форм, тоді коли потрібна фотографічна схожість з об'єктом, а також для створення 3D моделей людей при мінімальних матеріальних та часових затратах.

Основним завданням даної роботи було дослідження технологій 3D друку та доступних методів виготовлення комп'ютерних 3D моделей реальних об'єктів. На основі проведених досліджень реалізовано різні методи виготовлення прототипів та в подальшому отримано фізичні об'єкти засобами 3D друку. Використовуючи дані проведених досліджень, зроблено висновки щодо оптимальних технічних характеристик пристроїв та середовищ, за умови яких процес створення 3D моделей буде якомога якіснішим.

### **3D технології як складова сучасного науково-технічного прогресу в світі та в Україні**

3D друк – це технологія адитивного виробництва, що перетворює віртуальну тривимірну модель у її фізичну копію. Використовуючи різноманітні матеріали та техніки друку можна отримати якісний прототип.

Процес друку починається з подачі підготовленої моделі у спеціалізовану програму, що називається слайсером (від англ. – to slice – різати). У цій програмі об'єкт буквально розрізається на шари, які потім накладаються один на одного за допомогою певного матеріалу, що плавиться або розм'якшується. Після затвердіння матеріалу отримуємо готову тривимірну реальну модель. Такий процес відрізняє 3D друк від традиційних технік механічної обробки, більшість яких заснована на видаленні матеріалу шляхом його обрізання або свердління (субтрактивні процеси). Найпопулярнішим матеріалом є пластик (PLA, ABS). Він найзручніший у використанні та найдешевший. Також можна використовувати метал, фотополімер, нейлон та інші матеріали, що можуть бути у рідкому чи твердому стані. Зараз можливе, навіть, використання живих клітин, до прикладу кісткової матерії та епітелію, а також харчових продуктів, як шоколад. Існує велика кількість технологій 3D друку. Вони різняться можливостями у використанні матеріалів, розмірів об'єкту, якістю друку, автоматизацією процесу.

3D моделювання – процес ручного створення тривимірної математичної моделі сцени і об'єктів в ній за допомогою спеціалізованих програмних засобів. 3D модель може бути представлена у вигляді програмного коду або відображена на дисплеї, а також за допомогою двовимірного зображення, що створюється рендерингом (візуалізацією), тобто відбувається побудова проекції відповідно до обраної фізичної моделі. Після цього отримане зображення можна виводити на пристрій виведення – принтер або дисплей. 3D моделі будуються набором точок в 3D просторі, поєднаних між собою різноманітними геометричними об'єктами, як трикутники, лінії тощо.

Існує чимало програм для тривимірного моделювання та редагування, за допомогою яких можна створювати 3D моделі будь-якої точності та з будь-якими характеристиками. Моделювання складних конструкцій вимагає навчання і практики, але протягом останніх кількох років було створено багато простих, зручних у використанні програм. Усі вони підходять і для моделювання і для подальшого 3D друку. Зазвичай в програмах є функції підготовки моделі до друку або існують плагіни, що дозволяють виконати таке завдання. Перед тим, як приступити до етапу моделювання, зроблено огляд програм 3D моделювання, щоб обрати найбільш відповідну. За даними сайту [i.Materialise](http://i.Materialise.com), найпопулярнішим редактором 3D є програма Blender [126]. Це безкоштовне ПЗ з відкритим кодом, що працює на платформах Windows, Mac і Linux та має велику кількість прихильників завдяки професійній команді розробників, конкуруючи з найбільшими комерційними аналогами [124].



Програма може функціонувати навіть на комп'ютерах з дуже слабкими конфігураціями, аж до нетбуків. [101]. Усі найбільш популярні програми мають свої переваги та недоліки. Варто також виділити такі програми, як 3Ds Max, Maya, ZBrush. Вони добре підходять для створення моделей із подальшою подачею на 3D друк.

Редактор 3D Studio Max орієнтований на архітектурну візуалізацію, тут присутня велика кількість інструментів, необхідних при моделюванні найрізноманітніших архітектурних проектів – від заготовок дверей і вікон різних форм до рослинності, сходів і огорож [101].

Maya використовують для роботи не тільки любителі моделювання тривимірної графіки, а й професійні компанії, такі як DreamWorks, Pixar, Walt Disney – і це далеко не повний список прихильників даного редактора. Дана програма затребувана саме завдяки можливості реалізувати всі етапи створення 3D. Програма також добре справляється із завданням підготовки моделей до друку та експортуванням моделей.

ZBrush – програма для цифрового живопису та моделювання. Вона має 30 початкових пензлів для малювання, а моделі створюються шляхом створення 3D-сітки, після чого за допомогою пензлів додаються об'єм і текстура. Моделі ZBrush можуть бути неймовірно деталізованими, але існують безкоштовні плагіни, що дозволяють перетворити їх в роздільну здатність, придатну для використання на споживацькому 3D принтері. Є багато доступних для навчання ресурсів, що допомагають на початку роботи із ZBrush, у тому числі на офіційному сайті від розробників [124]. А от програма SolidWorks має проблеми із імпортуванням STL файлів (файлів для подачі на 3D принтер). Тож для редагування завантаженої моделі, краще використовувати іншу програму.

Іншим способом фіксування фізичних об'єктів є 3D сканування. За допомогою цієї технології можна оцифрувати об'єкти реального світу та створювати їх віртуальні копії. 3D сканер аналізує інформацію та збирає інформацію щодо форми та, якщо можливо, кольору об'єкту. Такі сканери можуть створюватись за різними технологіями, це зазвичай залежить від призначення сканера, фізичних властивостей сканованого об'єкту та вимог до вихідної моделі. Суть технології сканування полягає у визначенні просторових координат точок об'єкта та створенні хмари таких точок. Процес реалізується за допомогою вимірювання відстані до всіх точок, що визначаються за допомогою фазового або імпульсного безвідбивного далекоміра. Вимірювання проводяться з дуже високою швидкістю – тисячі, сотні тисяч, а часом і мільйони вимірювань в секунду [7].

Полегшити процес 3D моделювання можна завдяки фотограмметричному методу безконтактних тривимірних вимірювань. Даний підхід дає змогу визначати просторові координати об'єкта по його зображенням, зробленим з різних ракурсів.

Такий вид 3D моделювання підходить тоді, коли потрібна фотографічна схожість з об'єктом, а також для створення 3D моделі людини при мінімальних матеріальних та часових затратах.

### **Створення віртуальної моделі методом 3D безконтактного сканування та її фізичного прототипу засобами 3D друку**

Для реалізації роботи обрано метод безконтактного сканування. У безконтактних сканерів є два типи сканування: пасивний і активний. Пасивні сканери – ті, що використовують відображення наявного світла у видимому діапазоні спектра та на основі його відображення від об'єкту проводять аналіз. Активний 3D сканер має здатність генерувати свій хвильовий сигнал: направлений світловий, лазерний, звуковий, інфрачервоний. Принцип даного методу ґрунтується на вимірюванні відстані від сканера до точок об'єкта сканування за методом триангуляції. Випромінювач та детектор (камера) знаходяться на різних рівнях, а промінь проектується під певним кутом щодо детектора. Таким чином, виходить трикутник, основу якого утворюють випромінювач і камера, а вершиною є точка на поверхні об'єкту. По зсуву відбиття від цієї точки, яке формується на сенсорі камери можна обчислити кут між проєктованим та відбитим променями, а знаючи кут і довжину основи, можна обчислити відстань до точки об'єкта.

Основним програмним забезпеченням послуговувала програма DAVID Laserscanner, що дозволяє сканувати та у фоновому режимі оцифровувати об'єкти за допомогою ручного лазерного модуля, калібраційного кута та камери. Програма генерує 3D дані в режимі реального часу та показує їх на екрані комп'ютера. Існують різні версії програми, які підходять для різних технологій сканування. Існує готовий пакет DAVID Laserscanner Starter Kit, в комплект якого входять: калібраційні аркуші у чотирьох розмірах, панелі для кріплення аркушів, веб-камера Logitech C615 з високою роздільною здатністю, підставка для камери, червоний лазер (650 нм) зі змінним фокусом і батареєю та необхідне програмне забезпечення.

Етапи створення віртуальної моделі методом 3D сканування та її фізичного прототипу засобами 3D друку:

1. Підготовка об'єкту до сканування. Технології сканування мають свої обмеження. Зазвичай ці обмеження стосуються саме об'єктів сканування, адже оптичні технології не підтримують обробку прозорих, дзеркальних та блискучих об'єктів. Це пояснюється тим, що хвиля (у даному випадку, лазерна

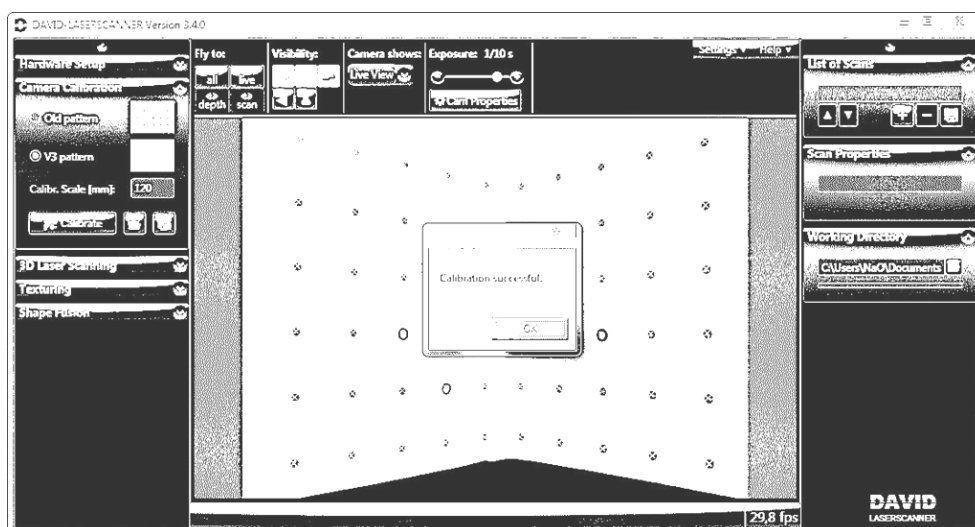
лінія) повинна відбиватись від об'єкту, щоб камера могла її зафіксувати, обробити інформацію щодо відстані та побудувати об'єкт на комп'ютері.

2. Калібрування камери. Калібрування камери відбувається з використанням спеціальних калібрувальних аркушів, що розміщаються під кутом 90 градусів відносно один одного. Вид таких аркушів залежить від програмного забезпечення, що використовується для сканування. Використано DAVID Laserscanner 3.4.0.3008, в комплекті із програмою подаються шаблони калібраційних поверхонь під формати A3 та A4 у файлових форматах PDF та CDR, що готові до друку. Формат аркушів вибирається залежно від розміру сканованого об'єкту. Висота сканованого об'єкту повинна бути в 1,5-2 рази меншою за висоту калібраційного кута.

Обрано формат A4 та розміщено аркуші на попередньо підготовлені дерев'яні панелі, які з'єднуються під прямим кутом. Кут 90 градусів потрібен для правильного заломлення лазерної лінії, що проектуватиметься на об'єкт, який розташовано перед кутом. Також потрібно слідкувати за тим, щоб роздруковані аркуші залишались гладкими та рівними, а на поверхні не було нічого глянцевого, щоб не створювати зайві перешкоди для сканування.

Камера повинна захоплювати калібраційні маркери, в процесі калібрування маркери повинні бути чітко видимими. Якщо це не так – можна увімкнути вікно налаштувань камери та змінити значення експозиції, яскравості, світла, налаштування діафрагми тощо.

На аркушах вказана довжина лінії Scale, яку потрібно виміряти та внести значення у програму. Якщо процес калібрування пройшов успішно, програма дає відповідне повідомлення "Calibration successful" та позначає маркери зеленими хрестами (рис. 26.1). Після калібрації важливо, щоб камера більше не змінювала своє положення.



*Рис. 26.1. Повідомлення успішного проведення калібрації*

3. Сканування об'єкту. Для сканування обраним методом використовується лазерний модуль, що проектує не точку, а лінію. Використано два червоні лазери: один придбаний за \$8 та один, створений самостійно за допомогою лазерної указки та блоку розгортки. Блок розгортки – оптична лінза, яка є скляним циліндром, на який потрапляє промінь лазера під кутом 90 градусів та заломлює його, таким чином утворюючи лінію. В разі необхідності, кут заломлення можна вирахувати залежно від діаметра лінзи.

В якості детектора – пристрою, що калібрується та фіксує відбиття лазера використовувалась веб-камера Logitech C270 з автофокусом. Роздільна здатність матриці – 3мП. Тип матриці – CMOS. Кадрова частота відео – 30 кадрів у секунду. Інтерфейс – USB 2.0.

Під час сканування зображення камери повинно бути дуже темним. Для цього варто мінімізувати світло у середовищі сканування та змінити відповідні налаштування камери так, щоб видно було лише лазерну лінію.

Розташовувати об'єкт потрібно так, щоб він знаходився в куті, перед калібраційними аркушами. Довкола об'єкту повинні залишатись не перекриті калібраційні маркери. При необхідності, об'єкт можна розмістити на підставку, бажано чорну, щоб вона поглинала світло, а не відбивала, передаючи тим самим непотрібні дані.

Коли запущено процес сканування та камера фіксує дані, ми бачимо об'єкт у різних кольорах. Це – карта глибин. Як відомо, довжина хвилі видимого спектру червоного кольору є найбільшою, а фіолетового – найменшою. Тут об'єкт фарбується за тим самим принципом: частини, що найближче до камери – фіолетові, що найдалше – червоні. Інші кольори мають проміжні значення.

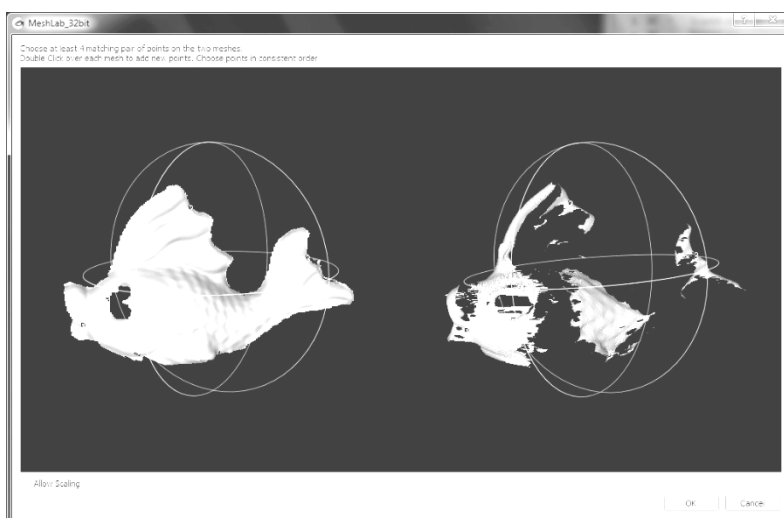
Сканування об'єкту потрібно проводити з різних напрямів огляду. Тобто, обробивши одну сторону об'єкту, варто зберегти отримане зображення, повернути об'єкт та почати сканування знову. Чим більше буде сканованих сторін – тим краще для створення подальшої моделі. Скановані сторони повинні накладатись одне на одного частинами. Тобто, кожен наступний скан повинен захоплювати частину попереднього скану. Це потрібно для наступного етапу склеювання моделі.

Кут між площиною світла лазера та камерою повинен бути не надто малим, інакше заломлення лінії буде недостатнє та програма видасть відповідне повідомлення: Intersection angle too low (Кут перетину замалий). Також у програми є інші повідомлення про помилку: Laser line not detected (too short/too weak?) – коли лазерна лінія є або закороткою, або затонкою, або відсутньою

взагалі. Laser line not detected on the left (right) side – якщо лазерна лінія не розрізняється на лівій (правій) стороні калібраційного кута. Тобто, лазерна лінія посунулась в сторону так, що її не видно на двох сторонах або ж лінія проектується не горизонтально, а з поворотом. Також це може означати, що об'єкт перекриває калібраційні маркери зліва (справа) і камера не може зафіксувати лазер. Тож варто звертати на це увагу та контролювати процес сканування.

4. Вирівнювання та злиття сканованих частин для отримання повної моделі. Вирівнювання сканів – це процес підстановки сканованих частин на відповідні їм місця, щоб після цього їх з'єднати у єдину модель. Для цього процесу використана програми DAVID Laserscanner та Meshlab. Meshlab – це програма для обробки неструктурованих тривимірних форм. Система орієнтована на обробку тривимірних моделей, що отримані шляхом 3D сканування та має інструменти для редагування, чистки, виправлення, перевірки, рендерингу та перетворення обробленої моделі у необхідний формат.

У програмі DAVID Laserscanner виконана чистка+ сканів від великих зайвих елементів та вирівнювання автоматичними засобами, а скани, які програма не змогла самостійно обробити, вирівнювались вручну в Meshlab. У DAVID Laserscanner також є можливість ручної підстановки, але вона розрахована на повністю ручне керування, а у Meshlab скани можна вирівняти по точках стику (рис. 26.2). Обираємо вже розташовану на своєму місці частину моделі та додаємо до неї іншу, відзначаючи точками основні місця, що співпадають. Таким чином збільшується вірогідність правильної підстановки.



**Рис. 26.2. З'єднання сканів за точками стику у Meshlab**

Саме склеювання моделі виконувалось у програмі DAVID Laserscanner кнопкою “Fuse”, тобто “сплавлення”. Програма зливає вирівняні частини в одну тривимірну модель, інтерполуючи місця з розривами чи різкими переходами. Після цього модель експортується у формат OBJ та переходить до наступного етапу – редагування.

5. Обробка, “чистка” моделі. Практично у всіх випадках, після сканування та склеювання модель потребує етапу обробки. На цьому етапі можна усунути діри, згладити поверхню, спростити загальну структуру або усунути зайві елементи. Усе це виконується у програмах тривимірного моделювання. Для реалізації проекту обрано редактор Blender, так як він знаходиться у вільному доступі. Інший плюс Blender в тому, що він адаптований для 3D друку. У ньому є плагін для тесту моделі на придатність до друку, а також імпорт в формати STL і OBJ. У Blender є багато інших плагінів і розширень, що дозволяють налаштувати його більш гнучко. Крім того, одиниця виміру в ньому дорівнює 1мм, тобто  $1.00000 = 1\text{мм}$ . Отже, 0.0010 – це 1 мікрон. Фактично, точність друку моделі обмежується можливостями 3D принтера.

6. Підготовка моделі до 3D друку. Об’єкт не може бути одразу поданий на друк після моделювання. Кожна модель, що подається на 3D принтер повинна пройти попередню підготовку. Підготовка моделі виконувалась також у програмі Blender, використовуючи спеціальний плагін та експортувала модель у необхідний формат. Програми-слайсери працюють з форматом файлу STL (stereolithography) [104]. Тому зберігати модель для друку потрібно саме в цьому форматі.

7. 3D друк. На цьому етапі модель спершу проходить підготовку у програмі-слайсері, де вона розбивається на шари, які під час друку утворює матеріал. Для друку даної моделі використовувалась програма Simplify3D.

На основі проведених досліджень зроблено висновки щодо оптимальних технічних характеристик складових сканера та середовища сканування. А саме:

- Мінімальна кількість освітлення в кімнаті під час сканування. Це важливо, особливо, якщо використовується не надто потужний, а отже, не надто яскравий лазер. Для того, щоб камера-детектор краще розпізнавала проектовану лазерну лінію, потрібно надати їй контрастні умови, щоб інша частина зображення була темною (навіть, чорною), тобто мінімізувати освітлення середовища.

- Веб-камера з роздільною здатністю  $640 \times 480$  та компенсацією шумів. Необхідна роздільна здатність обчислюється залежно від розміру об’єкта. При роздільній здатності  $640 \times 480$  об’єкт встановлюється так, що покриває

приблизно 300×350 пікселів. Піксель – це одна 3D точка. Таким чином, ми отримуємо більше 100 тисяч 3D точок, при чому лише з однієї сторони об'єкта. Тож, 640×480 – достатня роздільна здатність для сканування звичайних об'єктів. Насправді, важливішою характеристикою для камер є компенсація шумів.

- Ручний фокус та якісні лінзи камери. Також важливі характеристики. Можна використовувати камери із автофокусом, але значно зручніше, коли фокус є ручним. Тоді можна налаштувати камеру так, як краще для того чи іншого середовища. Щодо лінз, то достатньо замість пластмасових лінз застосувати скляні. Це покращить результат сканування.

- Лазер з потужністю 5mW та 30° чи 60° модулем, що генерує лінію. Потужність лазера, як вже було зазначено вище, залежить від середовища сканування. Якщо кімната не достатньо темна, то лазер повинен бути більш потужним. Також ця умова потрібна при скануванні темних об'єктів (як відомо, чорний не відбиває, а поглинає світло). Лазерне сканування також можливо виконувати, використовуючи лазерну указку, тобто її проекція буде у вигляді точки. Проте, для прискорення процесу замість лазерної точки використовують лазерну лінію. Вона досягається завдяки блоку розгортки (циліндрична лінза) чи системи механічної розгортки (дзеркало, що обертається). Кут модуля залежить від самого об'єкту, що сканується. У більшості випадків, кут повинен бути не менше 30 градусів, так як при меншому значенні кута, лінія, що проектується буде замалою для покриття об'єкту.

- Лазерна лінія повинна бути якомога тонша, особливо, якщо об'єкт сканування невеликих розмірів. Чим тонша лінія, тим менше вона надає шумів, що заважають успішному скануванню. А отже, поверхня моделі буде більш гладкою та деталізованою.

- Аркуші для калібрації, розташовані під кутом рівно 90° та з чітко надрукованими маркерами калібрації.

- Триангуляційний кут (кут нахилу лазера відносно об'єкту) – близько 30°. Якщо сканувати під меншим кутом, то перелом лазерної лінії буде занадто малим для розпізнання поверхні сканованого об'єкту.

### **Дослідження та створення 3D моделі за допомогою цифрового апарату**

Суть фотограмметричного методу полягає в отриманні вимірювань з декількох знімків, що перекриваються, для створення 3D моделей об'єктів або сцен.

Для реалізації експериментальної частини було обрано спосіб повороту камери навколо об'єкта та поворот об'єкта навколо своєї осі, оскільки такий

спосіб є доступним і не потребує спеціального дорогого обладнання. Зйомка проводилась за допомогою камери Sony Cyber-Shot DSC-W290.

Для дослідницької частини попередньо було обрано 3 об'єкти, різні за формою, текстурою, кольором та розміром. Використовуючи результати проведених досліджень, створено повноцінну 3D модель. Процес створення можна поділити на такі етапи:

*Етап 1. Підготовка об'єкту до фотозйомки.*

Не всі об'єкти можуть відобразитись у процесі створення 3D моделі методом перекриття фотографій. Проблеми виникають у прозорих, напівпрозорих, блискучих або дзеркальних об'єктів, адже камера не може чітко зафіксувати їх точок. В такому випадку необхідно замаскувати поверхню за допомогою матової фарби, нанесеної на об'єкт. Якщо об'єкт є однотонним, мало текстурованим або простої форми, програмі буде важко розпізнати з якої сторони об'єкт було знято. Для цього необхідно нанести спеціальні позначки (маркери) на предмет зйомки (рис. 26.3), які в надалі будуть відзначені у програмі та пронумеровані.



**Рис. 26. 3. Нанесення маркерів на об'єкт**

*Етап 2. Підготовка фотозони.*

Для забезпечення високої різкості фото, необхідно зафіксувати цифровий апарат за допомогою штативу. Освітлення є одним з ключових факторів якісної фотографії, тому наступним кроком є створення лайтбокса. Він має білий однотонний фон, а також змішує і розсіює направлене світло, який формується світловими приладами, створюючи безтіньовий малюнок предмета (рис. 26.4).





*Рис. 26.4. Фотозона*

### *Етап 3. Калібрування фотоапарата.*

Фотоапарат потрібно розмістити таким чином, щоб об'єкт займав як можна більшу область у кадрі. При зйомці невеликих за розміром об'єктів найкраще знімати на камеру, яка оснащена макрооб'єктивом, який забезпечить високу чіткість та деталізацію зображення. У випадку відсутності макрооб'єктиву, можна використовувати об'єктив з фіксованою фокусною відстанню. Перед початком зйомки необхідно відключити автоматичний режим балансу білого, щоб на усіх знімках він був однаковим.

Необхідно обрати режим – пріоритет діафрагми (A). Діафрагму закрити ближче до мінімального значення, від f5.6 до f11 (залежно від об'єкта), щоб уникнути втрати різкості фото. Чим менша діафрагма, тим буде більша глибина простору. Відповідно витримка буде довшою. Світлочутливість ISO налаштовується на мінімальне значення (50 – 100 ISO) для отримання найбільш якісного фотознімку.

### *Етап 4. Фотозйомка.*

Зйомка може проводитися в двох режимах:

1. Здійснюється шляхом обертання моделі (вручну або за допомогою спеціального обертового столика) – таку зйомку найзручніше проводити в лайтбоксі, де є необхідне дифузне світло.

2. Здійснюється шляхом переміщення фотографа з фотоапаратом навколо моделі.

В обох режимах орієнтовно фото робляться через кожні 10 градусів, тобто на одній висоті робимо 30–36 знімків. У разі складних моделей необхідно повторити зйомку з більш високої, або більш низької позиції, щоб охопити всю модель повністю.

## *Етап 5. Обробка фото.*

Не рекомендується обрізати фото, змінювати їх розмір і роздільну здатність будь-якими редакторами. На даному етапі маскуються лишні точки, які не відносяться до об'єкту. Зробити це можна в програмі Adobe Photoshop, або у програмах, в яких створюється 3D модель (наприклад, PhotoScan).

## *Етап 6. Моделювання 3D об'єкту.*

6.1. У програмі PhotoScan завантажуються фотографії, в меню обираємо “Вирівняти фотографії”. Програма знаходить спільні точки фотографій і по них визначає такі параметри камер як стан, орієнтацію, фокусну відстань, параметри дисторсії та інше. На цьому етапі отримуємо розріджену хмару спільних точок в 3D просторі моделі і дані про стан і орієнтації камер.

Даний результат служить для візуальної оцінки якості вирівнювання фотографій та подальшого створення 3D моделі на основі розрідженої хмари точок. При отриманні хмари точок необхідно зайві точки прибрати інструментом для видалення.

6.2. Тепер потрібно створити щільну хмару точок. Для цього обираємо пункт меню “Створити щільну хмару точок”. При завершенні цього етапу контури моделі добре проглядаються, і можна знову видалити зайві точки.

6.3. Переходимо до створення моделі і вибираємо пункт меню “Створити модель”. В результаті отримуємо тривимірну полігональну модель, що описує форму об'єкта, на підставі щільної хмари точок.

Після побудови моделі іноді потрібне її редагування. У програмі Photoscan можуть бути внесені такі зміни як видалення ізольованих компонентів моделі, заповнення отворів і так далі.

3D модель, яка отримана методом сканування зазвичай має надлишкову полігональність. Зменшення полігональності можна зробити за допомогою Autodesk 3Ds Max.

Перейшовши в каркасний режим відображення, у програмі можна помітити завищений показник кількості вершин у 3D моделі у декілька разів. Для цього викликаємо модифікатор ProOptimiser у стеці модифікаторів.

Натискаємо на кнопку Calculate, після чого буде розраховано кількість вершин в моделі. Зменшити їх в процентному співвідношенні, де 100% – це поточна кількість вершин в моделі. Потрібно зменшувати кількість полігонів доти, поки це не буде впливати на форму об'єкта.

У даному випадку було зменшено до 20% від загальної кількості вершин. При меншому відсотку була помітна деформація об'єкту. Для подальшої роботи рекомендовано перевести 3D модель в Editable Poly.

6.4. Наступний пункт меню – “Створення текстури”. Після обробки отримуємо готову модель, яку можна експортувати для переглядів за допомогою Adobe Acrobat, або Adobe Reader. Так само можна експортувати для додаткової обробки моделі в 3D редакторах в форматі .obj, зокрема, в програмі 3Ds Max.

Ще одним варіантом текстурування об’єкта є робота у програмі Autodesk 3Ds Max. Для цього попередньо екпортуємо 3D модель з програми PhotoScan у форматі .obj та окремим .jpeg файлом текстуру.

Для цього відкриваємо редактор матеріалів: Rendering – Material Editor (або натиснувши англійську клавішу “M”). Далі обираємо тип матеріалу Vitmar, у який завантажуюмо файл зі збереженою нами раніше текстурою та переносимо його на об’єкт.

## 2. Корекція моделі.

За потреби підчищаємо модель у програмі Autodesk 3Ds Max. У даному випадку, через складну форму об’єкту, було змодельовано зайві полігони. У режимі EditPoly – Polygon інструментом виділення було обрано непотрібні полігони та видалено. Окремі полігони захоплювали як область правильно так і не правильно змодельованих точок. Для цього потрібно розрізати полігон інструментом “Cut”, що знаходиться на панелі інструментів у розділі “Modeling”, та видалити лише непотрібну частину полігону.

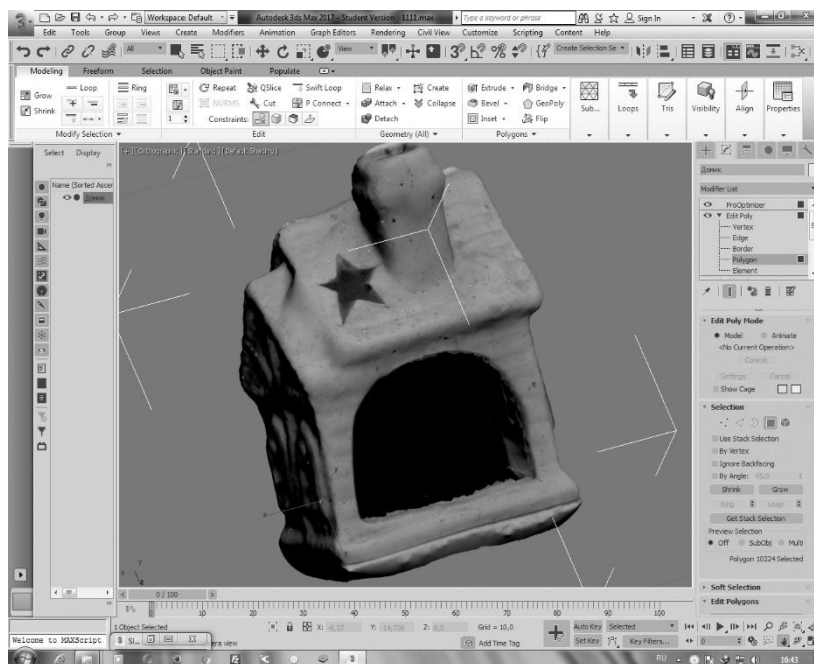


Рис. 26.5. Корекція 3D моделі

Результатом роботи є створення 3D моделі за допомогою цифрового апарата, яку надалі можна використовувати як в електронному вигляді так і підготувати файл до 3D друку. Для цього було обрано фотограмметричний метод.

Даний метод дає змогу визначати просторові координати об'єкта по його зображеннях, зроблених з різних ракурсів. В результаті перекриття фотографій у програмі PhotoScan створена тривимірна модель з максимально реалістичним ефектом.

На основі проведених досліджень було зроблено висновки щодо оптимальних технічних характеристик приладу та умов фотозйомки:

1. Не використовувати прозорі, напівпрозорі, блискучі або дзеркальні об'єкти, або ж покривати ці об'єкти матовою фарбою і в подальшому накладати текстуру у програмному забезпеченні. У випадку мало текстурованих об'єктів необхідно прикріпити на них спеціальні позначки.

2. Об'єкт має знаходитись на однотонному тлі.

3. Необхідно забезпечити рівномірне дифузійне освітлення, за допомогою фотозони – лайбокса, створивши безтіньове зображення предмету. Якщо об'єкт фотографується у природних умовах, потрібно це робити у хмарну погоду.

4. Для збільшення різкості фото, потрібно зафіксувати фотоапарат за допомогою штативу. Об'єкт має займати як можна більшу область у кадрі.

5. Камера має бути з макрооб'єктивом або об'єктивом з фіксованою фокусною відстанню.

6. Включити ручний режим балансу білого. Виставити пріоритет діафрагми (A). Діафрагму закрити ближче до мінімального значення, в межах від f5.6 до f11 (залежно від об'єкта). Відповідно витримка буде довшою. Світлочутливість ISO налаштовується на мінімальне значення (50 – 108.0 ISO).

7. Об'єкт фотографується з поворотом в кожні 10 градусів (30–36 знімків). У разі складних моделей необхідно повторити зйомку з більш високою або більш низькою позицією, щоб охопити всю модель повністю.

8. Не рекомендується обрізати фото, змінювати їх розмір і роздільну здатність будь-якими редакторами.

9. Зменшити полігональність об'єкту приблизно до 20 % від загальної кількості вершин, залежно від моделі. Зменшувати кількість полігонів доти, доки це не вплине на зовнішній вигляд об'єкту.

**Висновки.** Проведено дослідження технологій 3D друку та доступних методів виготовлення комп'ютерних 3D моделей реальних об'єктів.

Реалізовано різні методи виготовлення прототипів та в подальшому отримано фізичні об'єкти засобами 3D друку.

Результатом роботи є самостійна розробка апаратного забезпечення для тривимірного сканування, яке послугувало основою для розробки методики фіксації фізичних об'єктів із розрахунком на подальшу обробку зображення у програмному середовищі 3D моделювання та тривимірним друком за допомогою 3D принтера.

Для виконання проекту по розробці 3D сканера, обрано метод безконтактного активного лазерного сканування. При цьому використовується лінійний лазерний модуль, який проектується на об'єкт. За допомогою спеціального калібраційного кута (вид калібраційного кута обирається у відповідності із програмним забезпеченням для сканування), калібрується веб-камера, яка надалі фіксує відбиті лазерні промені та за допомогою спеціалізованої програми DAVID-Laserscanner, що розраховує 3D координати з перетину площини лазера та видимих променів з камери, будується копія фізичного об'єкту у вигляді віртуальної тривимірної моделі.

На основі проведених досліджень зроблено висновки щодо оптимальних технічних характеристик складових сканера та середовища сканування.

А також розроблена методологія створення 3D моделі за допомогою цифрового апарату, яку в подальшому можна використовувати як в електронному вигляді так і підготувати файл до 3D друку. Даний метод дає змогу визначати просторові координати об'єкта по його зображеннях, зроблених з різних ракурсів. В результаті перекриття фотографій у програмі PhotoScan створена тривимірна модель з максимально реалістичним ефектом.

Використовуючи дані проведених досліджень, зроблено аналіз та описано необхідні умови для створення якісних 3D моделей.

## РОЗДІЛ 27

# МЕТОДЫ ПОВЫШЕНИЯ КАЧЕСТВА КЛАССИФИКАЦИИ СИМВОЛОВ ПРИ ОБРАБОТКЕ ТЕКСТОВЫХ ИЗОБРАЖЕНИЙ

*Аннотация.* В работе рассматриваются методы классификации, применяемые при сжатии файла с битональным изображением текста, полученным сканированием или цифровым фотографированием. Особое внимание обращается на используемые при этом меры различия двух изображений символов, выделенных из изображения текста. Эти меры различия позволяют с той или иной степенью уверенности считать символы на сравниваемых изображениях или совпадающими, или различными. Для известных на сегодняшний день алгоритмов классификации, включая хорошо известный алгоритм JB2, приведены количественные характеристики классификации – число классов, получаемых этими алгоритмами для изображения стандартной страницы текста. Чем меньше это число, тем качество классификации считается выше, так как дает лучшее сжатие файла с изображением текста. Рассмотрены также методы ускорения алгоритмов, классифицирующих изображения символов, и повышения удобочитаемости восстановленного после сжатия изображения текста.

**Ключевые слова:** изображение текста, методы классификации, сжатие данных.

*Abstract.* Methods are in-process examined classifications, applied at the compression of file with the bi-tonal image of photo-typograph got scan-out or digital photographing. The special attention applies on the in-use here measures of distinction of two images of characters, abstracted from the image of text. These measures of distinction allow with one or another degree of confidence to count characters on the compared images or consilient, or different. For the algorithms of classification known to date, including the known algorithm of JB2 well, quantitative descriptions of classification – number of classes, got these algorithms for the image of standard page of text are resulted. What less than it is a number, quality of classification is considered that higher, because gives the best compression of file with the image of text. The methods of acceleration of algorithms, classifying the images of characters, and increases of easy-to-readness of the text recovered after the compression of image are considered also.

**Keywords:** image of text, methods of classification, compression of data.

**Введение и постановка задачи.** Высокие результаты, которые демонстрирует алгоритмом JB2 (формат DjVu), объясняются тем, что он использует классификацию символов. Вообще идея сжатия информации с

помощью классификации очень проста и идеально подходит для сжатия изображений текста.

Пусть необходимо сжать некую информацию, которую можно разбить каким-то образом на элементы. Если эти элементы информации объединить в классы так, чтобы в каждом классе находились тождественные (pattern matching) или почти тождественные (soft pattern matching) элементы, то нет нужды хранить все элементы информации – достаточно хранить только по одному элементу из каждого класса. Совокупность таких элементов – представителей классов – называется словарем. Кроме того для восстановления информации нужно еще иметь таблицу, называемую «картой размещения классов», которая для каждого класса указывает, в каком месте исходной информации находятся его элементы.

Ясно, что степень сжатия информации с помощью классификации тем выше, чем меньше классов образуется при классификации и чем больше элементов в каждом классе.

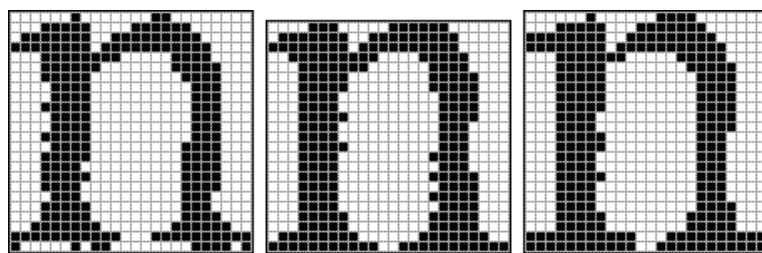
В случае сжатия изображения бинарного (далее черно-белого) текста естественным элементом информации является изображение отдельного символа (буквы, цифры, знака препинания и т.п.). Выделение символов не представляет собой особо трудную задачу. Во всех известных алгоритмах, включая алгоритм JB2, символы выделяются как связные области, состоящие из черных точек.

Следует заметить, что при этом некоторые грамматические символы распадаются на части (например, буква “ё” дает три символа), а некоторые (например, сочетания вида “fh”) объединяются в один. Кроме того метод непригоден для текстов с псевдо рукописным шрифтом. Сжатие таких текстов алгоритмом JB2 и другими катастрофически низкое.

Однако не это представляет собой главную трудность при классификации уже разделенных символов.

Цель настоящей статьи – дать обзор идей и методов, на которых основаны эти алгоритмы, и провести сравнение достигнутых с их помощью результатов по сжатию изображений текста.

На рис. 27.1, взятом из работы [32], представлены три случайно выбранные изображения буквы “n” из различных 257, входящих в изображение страницы текста формата A4, при разрешении сканирования 300 dpi.



*Рис. 27.1. Влияние шумов печати и сканирования на изображение символа «п»*

Легко верится, и это действительно так, что на странице не найдется ни одной пары символов “п”, полностью совпадающих друг с другом. То же, за редким исключением, относится и к другим символам, даже точкам. Причиной этого явления являются шумы (то есть случайные искажения), возникающие при печати страницы и ее последующем сканировании. Шумы печати в основном вызваны диффузией краски, жидкой или твердой, вдоль хаотически расположенных капилляров бумаги. Шумы сканирования – несовпадением контуров символа с матрицей сканера, подобно тому, как прямая наклонная линия на экране монитора отображается “ступеньками”.

Человеку легко заметить, что все три изображения, приведенные на рис. 27.1, представляют собой букву “п”. Однако пока не существует алгоритма, который мог бы установить тождественность этих символов с той же достоверностью, что и человек. Это и есть главная трудность, не позволяющая разбить изображения символов на классы, так чтобы одновременно выполнялись два условия:

**Условие 1.** В каждом классе находятся изображения только одного и того же символа;

**Условие 2.** Все изображения какого-либо символа находятся в одном классе.

Все алгоритмы классификации являются тем или иным компромиссом между этими условиями, причем условие 1 должно выполняться достаточно жестко, иначе в восстановленном тексте будут перепутаны символы, чем иногда грешит алгоритм JВ2. Например, иногда путает между собой буквы «b» и «h».

Соблюдение условия 1 влечет за собой ужесточение алгоритма сравнения изображений символов, так что условие 2, практически, невыполнимо. Это приводит к появлению значительно большего числа классов, чем количество символов, изображенных на странице, так как практически все символы дают по несколько классов своих изображений. Чем больше при классификации



образуется классов, тем больше словарь и (логарифмически) больше карта размещения классов. Как следствие, понижается степень сжатия. И хотя алгоритм JB2 и другие используют те или иные методы дополнительного сжатия словаря и карты, эффективность алгоритма в целом определяется качеством классификации, то есть количеством получившихся классов, которое в идеале (условие 2) должно совпадать с количеством символов, присутствующих в тексте, чье изображение сжимается.

Таблица 27.1

**Количество классов при рассматриваемых методах классификации**

Разрешение сканирования (dpi)	Количество классов в исходном изображении	Количество классов после основной классификации $\varepsilon_{opt} = 6\%$	Количество классов после второй классификации $\varepsilon_{opt} = 6\%$	Количество классов после классификации алгоритмом JB2
600 dpi	3558	197	72	314
500 dpi	3557	137	72	259
400 dpi	3557	130	71	199
300 dpi	3545	122	95	235
200 dpi	3890	237	148	451

В табл. 27.1 из работы [32] для различных разрешений сканирования показано количество классов, полученных предлагаемым алгоритмом классификации (предпоследний столбец) и алгоритмом JB2 (последний столбец). Первый столбец показывает разрешение, использованное при сканировании одной и той же страницы формата А4 с черно-белым текстом, набранным шрифтом Times New Roman, 12 pt. Второй столбец – количество классов при тождественной классификации, то есть классов, состоящих из полностью совпадающих изображений символов.

Из таблицы следует несомненное превосходство алгоритма ИЛЛ – будем для краткости называть так алгоритм, предложенный в упомянутой работе [32], – над алгоритмом JB2. Словарь ИЛЛ получается почти в три раза короче, чем словарь JB2. Отсюда вытекает и превосходство в степени сжатия той же страницы, что показывает следующая таблица, относящаяся к той же странице, что и табл. 27.1.

Не слишком существенное различие между коэффициентами сжатия, продемонстрированными алгоритмами ИЛЛ и JB2, объясняется тем, что авторы

алгоритма ИЛЛ интересовались только классификацией выделенных изображений символов и не оптимизировали алгоритм дополнительного сжатия словаря и карты размещения классов (использовался универсальный алгоритм без потерь 7z).

Кроме того табл. 27.2 показывает, что применение лучшего для сжатия размытых изображений алгоритма JPEG 2000 мало что дает при сжатии изображения черно-белого текста без иллюстраций.

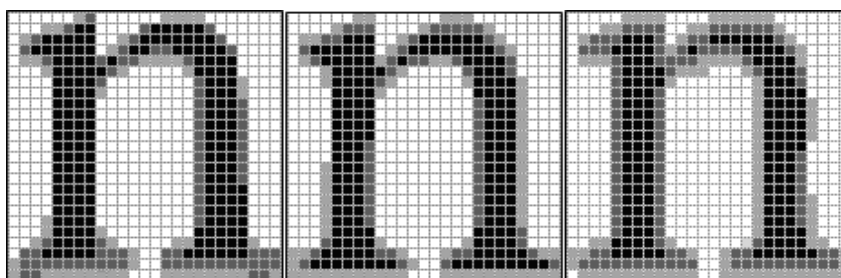
Таблица 27.2

**Сравнительная степень сжатия изображения текста**

Разрешение сканирования (dpi)	200	300	400	500	600
Исходный размер файла (kb)	505,3	1080,2	2003,9	3111,2	4498,0
<b>Методы</b>	<b>Коэффициент сжатия</b>				
JPEG 2000	3,8	3,74	3,76	3,75	3,75
JBIG2	8,2	11,2	16,7	20,9	25,1
JB2	52,6	124,1	202,4	272,9	330,7
ИЛЛ	62,3	135,0	250,4	353,5	436,7

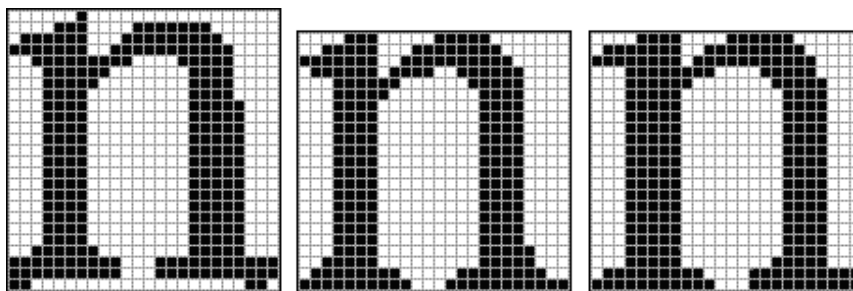
**Дополнительные меры повышения качества классификации символов.**

Своей высокой эффективностью алгоритм ИЛЛ обязан не только удачно выбранной и легко вычисляемой мере отличия [32], но и в большей степени тому, что он использует процедуру статистического усреднения. Это означает, что после проведения классификации для каждого класса находится усредненное изображение. Процедура усреднения состоит в наложении друг на друга всех изображений класса, совмещая их “центры тяжести”, и вычисления среднего значения яркости для каждой точки с последующим округлением. На рис. 27.2 иллюстрируется процесс усреднения для трех классов изображений одного и того же символа “п”. Черные точки означают, что среднее значение яркости в них равно 0, серые – что среднее значение меньше или равно 0,5, а светлые – больше 0,5.



*Рис. 27.2. Суммарные изображения трех различных классов символа «п»*

При округлении черные и серые точки превращаются в черные, а светлые – в белые. Результат показан на рис. 27.3.



*Рис. 27.3. Изображения средних символов в трех классах символа «n»*

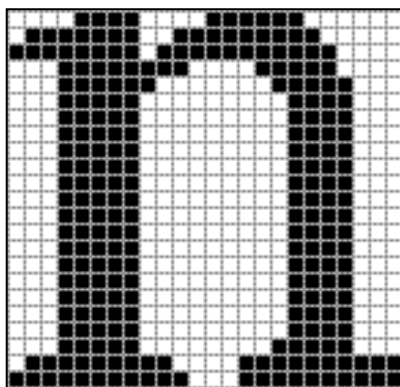
Совокупность полученных средних изображений в алгоритме ИЛЛ используется как промежуточный словарь, то есть все исходные изображения символов более не используются. Легко видеть (см. рис. 27.3), что благодаря процедуре усреднения искажения, вызванные шумами печати и сканирования, заметно меньше, чем в исходных изображениях (см. рис. 27.1).

Однако в полученном промежуточном словаре усредненных изображений по-прежнему присутствуют разные изображения одних и тех же символов, хотя и в значительно меньшем количестве, чем до классификации. Их количество для тестовой страницы указано в третьем столбце табл. 27.1.

Качество классификации можно улучшить, применив еще одну классификацию, призванную объединить в один класс изображения одного и того же символа. Для этой классификации, как и для предыдущей, используется нечувствительная к контурным шумам мера отличия  $\mathcal{E}$  с тем же порогом  $\mathcal{E}_{opt}$ , но алгоритм классификации выбирается другим.

Дело в том, что теперь разные изображения одного того же символа очень близки друг к другу – случайная компонента изображений в значительной мере подавлена. Поэтому существенно меньше опасность спутать изображения двух разных символов.

После получения новых классов, изображения в каждом из них снова усредняются, и получившийся набор изображений представляет собой окончательный словарь. На рис. 27.4 показано теперь единственное изображение символа “n”, вошедшее в словарь. Эффективность повторной классификации можно проследить, сравнивая третий и четвертый столбцы табл. 27.1.



*Рис. 27.4. Представитель класса изображения символа «п», который попадает в словарь символов*

Сравнивая рис. 27.4 с рис. 27.1, можно утверждать, что контурные шумы практически отсутствуют и внешний вид символа значительно улучшился, по сравнению с вариантами его изображения в исходном тексте. Таким образом, качество восстановленного изображения текста в алгоритме ИЛЛ значительно выше исходного. В принципе так и должно быть, потому что значительная часть избыточной информация, которую устраняют алгоритмы сжатия изображений текста, состоит из случайных шумов печати и сканирования.

#### **Методы ускорения классификации символов**

В алгоритме ИЛЛ предварительная классификация проводится методом «просеивания», также с помощью очень быстро вычисляемой дополнительной меры различия. Изображению каждого символа сопоставляется трехмерный вектор, и мера различия между двумя изображениями символов – расстояние между соответствующими векторами, нормированными так, чтобы порог не зависел от размера шрифта и разрешения сканирования.

Вектор параметров каждого изображения символа, на сравнении которых проводится классификация, состоит из следующих величин:  $H$  – высота символа, то есть высота прямоугольника, в который вписано изображение символа;  $W$  – аналогично определенная ширина символа;  $P$  – периметр изображения символа, подсчитанный в пикселях вдоль внешних и внутренних его границ.

При сравнении размеров двух символов вычисляется разность их физических размеров, например, для высоты – величина

$$\Delta H = |H_1 - H_2| \frac{100}{res(dpi)},$$

где  $H_1$ ,  $H_2$  – высоты первого и второго сравниваемых символов в пикселях, и  $ras(dpi)$  – разрешение сканирования, измеряемое в точках на дюйм.

Таким образом, разность высот выражается в сотых частях дюйма. Аналогично вычисляется разность ширин  $\Delta W$  сравниваемых символов.

Переход к физическим линейным размерам оправдан тем, что на отклонения высоты и ширины изображений одного и того же символа в основном влияют шумы печати, а не сканирования. Последние могут изменить линейный размер не более, чем на 1 пиксель. Экспериментально установлено, что отклонение в линейных размерах, вызванное расплыванием краски, находится в пределах 0.01 дюйма. Поэтому линейные размеры символов вне зависимости от разрешения и размера шрифта можно считать значительно отличающимися, если  $\Delta H > 1$  и  $\Delta W > 1$ .

Длина периметра, определяемая как число граничных пикселей в изображении символа, мало искажается шумами печати и сканирования. В основном она выражает индивидуальные особенности символов, то есть гарантированно различает, например, такие буквы, как “n” и “r”. Поэтому для их сравнения вводится в рассмотрение безразмерная, то есть не зависящая ни от разрешения сканирования, ни от размера шрифта величина

$$\Delta P = \frac{|P_1 - P_2|}{\sqrt{P_1 P_2}} 100\% ,$$

где  $P_1$  и  $P_2$  – периметры сравниваемых символов.

Периметры символов считаются далекими, если  $\Delta P > 10\%$ . Если этому сопутствует значительное различие в линейных размерах:  $\Delta H > 1$  и  $\Delta W > 1$ , то можно с уверенностью утверждать, что сравниваемые изображения относятся к разным символам. Такая классификация, с одной стороны, имеет небольшое число классов, каждый из которых содержит изображения близких по начертанию символов. С другой стороны, выполнено условие 2, то есть все изображения одного и того же символа попадают в один и тот же класс.

#### **Дополнительное сжатие словаря и карты размещения классов.**

Размер графического файла, который сжат каким-либо алгоритмом, основанным на классификации, грубо говоря, состоит из размера словаря, и размера соответствующей карты размещения классов. Этот размер можно уменьшить, если дополнительно сжать каким-либо стандартным алгоритмом сжатия без потерь, например, алгоритмами RLE, Хаффмана, LZW и другими [3; 97]. Или их комбинациями. И хотя при создании стандартного формата

сжатия изображений текста на эти вопросы требуется обратить самое серьезное внимание, здесь мы на этом останавливаться не будем. Отметим только, что размер словаря во много раз превышает размер отвечающей ему карты (для одной страницы текста) и представляет собой набор изображений символов, для сжатия которых методы компрессии без потерь не являются самыми эффективными.

Рассмотрим предложенный в работе [33] алгоритм – будем называть его алгоритмом ИЛЛ2 – который сжимает словарь, рассматривая его не как последовательность бит, а как изображение, составленное из отдельных изображений входящих в него символов. Идея этого алгоритма состоит в следующем.

Если представить себе строку в изображении текста как прямоугольник с горизонтальным основанием, охватывающий строку, то вертикальным элементом строки называется пересечение этого прямоугольника с любой вертикальной линией шириной в один пиксель. На рис. 27.5 из работы [33] показано разбиение изображения буквы “е” на вертикальные элементы строки.



*Рис. 27.5. Изображение символа «е» и составляющие его вертикальные элементы строки*

Таким образом, все символы в строке, включая пробелы, можно представить как объединение ее вертикальных элементов одного и того же размера.

Алгоритм ИЛЛ2 рассматривает словарь как изображение строки, составленной из изображений всех входящих в него символов. (Фрагмент такого словаря показан на рис. 27.6.). Эта строка разбивается на вертикальные элементы, совокупность которых классифицируется отдельным способом. В результате получается новый словарь, состоящий из вертикальных элементов строки, и карта размещения соответствующих ему классов. Смысл этих процедур заключается в том, что новый словарь занимает намного меньше

места, чем старый, а дополнительная карта размещения классов невелика по размеру.



**Рис. 27.6. Изображение фрагмента словаря для изображения текста 300 dpi: вверху – словарь до сжатия; внизу – восстановленный словарь**

Внизу, на рис. 27.6 показан словарь символов, восстановленный с помощью словаря вертикальных элементов строки и соответствующей ему карты расположения классов. Некоторое незначительное ухудшение качества заметно только при большом увеличении, которое используется на этом рисунке.

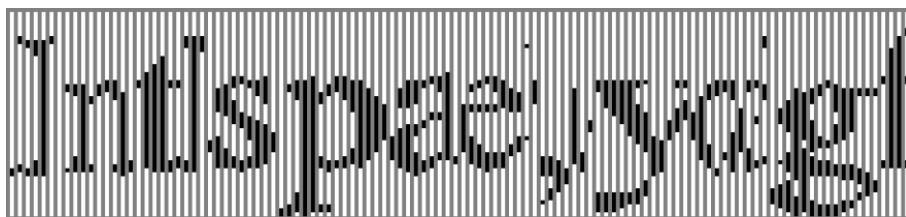
Таким образом, алгоритм ИЛЛ2 работает в два этапа, как это схематически изображено на рис. 27.7. На первом этапе (переход с первого уровня на второй) он совпадает с алгоритмом ИЛЛ, и отличается от последнего наличием еще одного этапа (переходом со второго уровня на третий), на котором сжимается словарь, полученный на первом этапе.



**Рис. 27.7. Схема двухэтапной обработки изображения текста алгоритмом ИЛЛ2**

Фрагмент словаря вертикальных элементов строки для изображения текста с разрешением 300 dpi представлен на рис. 27.8 (серые полосы введены для

наглядности как разделители между представителями отдельных классов вертикальных элементов строки).



*Рис. 27.8. Фрагмент словаря вертикальных элементов строки, полученный после автоматической классификации*

Перейдем к описанию алгоритма классификации вертикальных элементов строки, который базируется на автоматической классификации (алгоритме  $k$ -средних [3; 31]).

Вертикальные элементы строки, совокупность которых обозначим через  $x$ , рассматриваются как векторы с бинарными координатами (0 – черный цвет, 1 – белый). Единственным параметром классификации является  $k$  – число классов.

Некоторым образом из множества  $x$  выбирается элементов  $e_1, \dots, e_k$  – центров класса (в первом приближении). Все элементы изображения разбиваются на  $k$  классов  $S_1, S_2, \dots, S_k$  по правилу: для каждого элемента  $x$  находится ближайший в евклидовой метрике центр класса, после чего в один класс объединяются все элементы, имеющие один и тот же ближайший центр класса. Далее в каждом классе находятся новые центры класса (во втором приближении) путем усреднения элементов в каждом классе:

$$e_i = \frac{1}{N_i} \sum_{x \in S_i} x,$$

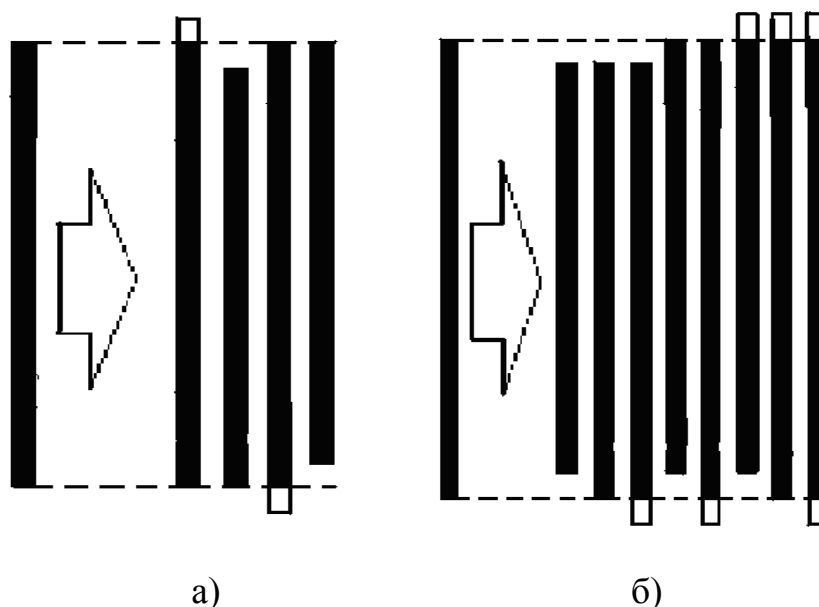
где  $N_i$  – число элементов в классе  $S_i$ , и  $i = 1, 2, \dots, k$ , и последующим округлением до 0 или 1).

После этого процедура повторяется, исходя из новых центров классов. Алгоритм заканчивает работу, когда центры классов перестают изменяться. Их совокупность – словарь вертикальных элементов строки.

Автоматическая классификация – очень быстрый алгоритм, дающий классификацию высокого качества, если правильно выбраны число классов  $k$  и набор центров первого приближения. В алгоритме ИЛЛ2 и то, и другое



определяется с помощью алгоритма просеивания, с мерой отличия, учитывающей контурный характер искажений, вносимых шумами печати и сканирования.



**Рис. 27.9. Возможные варианты близких связанных компонент вертикальных элементов строки: а)  $\varepsilon = 1$ ; б)  $\varepsilon = 2$**

Эта мера отличия  $\varepsilon$  двух вертикальных элементов строки равна бесконечности во всех случаях кроме тех, когда каждой связанной компоненте множества черных пикселей одного из них взаимно однозначно отвечает компонента другого так, что эти компоненты могут отличаться на один пиксель только на своих концах. Возможные варианты показаны на рис. 27.9. Во всех этих случаях мера отличия  $\varepsilon$  равна максимальному по каждой компоненте числу несовпадений. Иначе говоря,  $\varepsilon = 0$ , если вертикальные элементы строки полностью совпадают,  $\varepsilon = 1$ , если все пары отвечающих друг другу компонент совпадают кроме некоторой, отличающихся на один пиксель на одном из концов, верхнем или нижнем, и  $\varepsilon = 2$ , если найдутся пары компонент, отличающихся на один пиксель и на нижнем, и на верхнем концах. Во всех остальных случаях  $\varepsilon_{opt} = \infty$ .

Выбор такой меры отличия объясняется тем, что контурные шумы печати и сканирования могут, как правило, изменить черную компоненту вертикального элемента строки в ту или иную сторону лишь на один приграничный пиксель.

После разбиения всей совокупности вертикальных элементов на классы методом просеивания с порогом  $\varepsilon_{opt} = \varepsilon_{max} = 2$  в качестве центров нулевого приближения для алгоритма автоматической классификации выбираются средние в каждом из полученных классов за исключением тех классов, которые содержат 1–2 элемента.

Последнее объясняется тем, что шумы печати и сканирования порождают новые, как правило, многокомпонентные вертикальные элементы, смежные с вертикальными границами символов. А также сильно искажают соседние элементы, принадлежащие изображению символа. (См., например, рис. 27.1.) Такие элементы при просеивании дают классы, состоящие из одного-двух элементов (для стандартной страницы текста) и не выбираются в качестве центров класса. При автоматической классификации благодаря проводимому на каждой итерации усреднению, они не оказывают никакого влияния на конечный результат. Это обстоятельство оправдывает выбор автоматической классификации.

Отметим также, что предварительное применение алгоритма просеивания значительно улучшает сходимость автоматической классификации по сравнению со случайным выбором центров первого приближения и, следовательно, сокращает вычислительное время. Это объясняется тем, что уже на первой итерации центры нулевого приближения хорошо аппроксимируют члены своего класса с точностью, не меньшей, чем  $\sqrt{2n}$ , где  $n$  – число черных компонент.

Размер файла, который сжат каким-либо алгоритмом, основанным на классификации, в основном состоит из размера словаря и размера карты размещения классов. Причем при классификации выделенных изображений символов размер словаря весьма значителен. Поэтому предложенный в рассматриваемой работе алгоритм автоматической классификации вертикальных элементов строки, уменьшающий размер словаря, обеспечивает более высокую степень компрессии исходного изображения текста.

В табл. 27.3 для тестовой страницы А4, шрифт Times New Roman 12 pt представлены объемы словаря символов, словаря вертикальных элементов строки, а также общего объема словаря вертикальных элементов строки и соответствующей ему карты расположения классов. Параллельно приведены те же характеристики после дополнительного сжатия без потерь с использованием алгоритма 7z, который является модификацией словарного метода компрессии LZ77–LZMA.

Отметим, что полученный на втором этапе алгоритма ИЛЛ2 словарь вертикальных элементов строки в 20–30 раз меньше словаря символов, полученного на первом этапе. Однако после дополнительного сжатия результат не такой впечатляющий – в 2 – 4 раза (хотя это тоже очень хороший результат). Это объясняется тем, что словарь вертикальных элементов практически не содержит избыточную информацию, так что «дожимать» по сравнению с исходным словарем символов фактически нечего.

Карта расположения классов вертикальных элементов строки, по сути, аналогична обычному тексту, скажем, в ASCII кодировке, и при дополнительном сжатии уменьшается всего в 2–3 раза. Это в значительной мере уменьшает эффективность второго этапа алгоритма ИЛЛ2.

Таблица 27.3

### Объем словаря символов

Разрешение сканирования (dpi)	200	300	400	500	600
Размер словаря символов/ после дополнительного сжатия(kb)	34,8/4,4	46,4/5,0	67,7/4,7	94,6/5,4	140,2/6,7
Размер словаря вертикальных элементов строки / после дополнительного сжатия (kb)	1,2/1,0	2,5/1,5	4,4/2,1	7,3/3,0	10,8/3,8
Словарь вертикальных элементов строки + карта размещения их классов / после дополнительного сжатия (kb)	7,3/3,0	8,7/3,4	10,1/4,2	13,7/5,2	18,8/6,6

Таблица 27.4

### Выигрыш в сжатии изображения текста в результате компрессии словаря символов на втором этапе обработки

Разрешение изображения текста (dpi)	200	300	400	500	600
Коэффициент сжатия словаря символов (ИЛЛ = 1-й этап алгоритма ИЛЛ2)	7,9	9,28	14,4	17,51	20,92
Коэффициент сжатия словаря вертикальных элементов вместе с соответствующей ему картой размещения классов (II этап алгоритма ИЛЛ2)	11,6	13,64	16,11	18,19	21,24

Как показывает табл. 27.4, выигрыш в сжатии, полученный на втором этапе, существенен при разрешениях 200 и 300 dpi, что является неплохим результатом, так как эти разрешения наиболее популярны. При 400 dpi

выигрыш не столь заметен и практически отсутствует при больших разрешениях. Это объясняется тем, что с повышением разрешения увеличивается число вертикальных элементов строки на каждый символ, и карта расположения классов вертикальных элементов строки возрастает. Так как она сжимается слабо, то суммарный объем словаря вертикальных элементов строки и соответствующей ему карты с ростом разрешения сжимается все хуже и хуже.

Таблица 27.5

**Выигрыш в степени сжатия двухэтапного алгоритма в сравнении с алгоритмом JB2 (DjVu)**

Разрешение изображения текста (dpi)	200	300	400	500	600
Алгоритм ИЛЛ (или 1-й этап алгоритма ИЛЛ2)	62,38	135	250,48	353,54	436,7
Алгоритм ИЛЛ2	74,3	166,18	267,18	361,76	445,34
Преимуществов сжатии алгоритма ИЛЛ2 над алгоритмом ИЛЛ в (%)	<b>16%</b>	<b>19%</b>	<b>6%</b>	<b>3%</b>	<b>2%</b>
Алгоритм JB2 (DjVu)	52,63	124,16	202,41	272,91	330,73
Преимущество в сжатии ИЛЛ2 над алгоритмом JB2 в (%)	<b>29%</b>	<b>25%</b>	<b>24%</b>	<b>24,5%</b>	<b>25,5%</b>

В табл. 27.5 приведены значения коэффициентов сжатия изображения стандартной страницы текста алгоритмом ИЛЛ (или, что то же самое, после 1-го этапа алгоритма ИЛЛ2), алгоритмом ИЛЛ2 и алгоритмом JB2, а также преимущество в сжатии алгоритма ИЛЛ2 над алгоритмами ИЛЛ и JB2 в процентном соотношении.

Главный вывод, который можно сделать из этой таблицы, такой. Алгоритм ИЛЛ2 благодаря второму этапу повышает степень сжатия алгоритма ИЛЛ в основном при разрешениях сканирования в 200 и 300 dpi и, таким образом обеспечивает сжатие примерно на 25% лучшее, чем алгоритм JB2 при всех разрешениях сканирования в диапазоне 200 – 600 dpi.

**Выводы.** В настоящее время самыми мощными алгоритмами сжатия двухцветного изображения текста являются те, которые используют классификацию выделенных символов. При этом конечный результат – степень сжатия изображения – больше всего зависит от качества классификации, то есть количества полученных классов при неизменном условии, что в каждый класс входят изображения только одного символа. Еще один аспект при обсуждении любого метода сжатия – качество восстановленного изображения по сравнению с исходным сжимаемым оригиналом. Обычно, чем выше степень сжатия, тем

это качество хуже. Методы сжатия сканированных изображений текста, основанные на классификации выделенных символов, позволяют получать восстановленные изображения символов более высокого качества, чем оригинальные. Причем, чем лучше проведена классификация, тем больше сжатие и тем лучше качество изображения символов. Этот парадоксальный факт объясняется очень просто. Подходящая статистическая обработка этого класса позволяет избавиться от искажений, внесенных при печати и сканировании и имеющих случайный характер.

Описанные выше меры отличия, учитывающие контурный характер шумов печати и сканирования, являются достаточно сложными. Но все же есть уверенность, что они будут существенно улучшены, так чтобы проводимая с их помощью классификация выделенных изображений была близка к идеальной.

## Перелік використаної літератури

1. Азаров С. С. Современные модели провайдинга / С. С. Азаров, В. А. Хорошко. – К. : Изд-во «Полиграф Консалтинг», 2006. – 98 с.
2. Азаров С. С. Типизация моделей технологических решений провайдинга Интернет / С. С. Азаров, Е. Б. Азаров, М. М. Дивизинюк, В. А. Хорошко. – Севастополь : СНУЯЭиП, 2005. – 112 с.
3. Айвазян С. А. Прикладная статистика: Классификация и снижение размерности [Текст] / С. А. Айвазян, В. М. Бухштабер, И. С. Енюков и др. – М. : Финансы и статистика, 1989. – 607 с.
4. Аксак Н. Г. Мультиагентная модель адаптации медицинского Интернет-ресурса на основе информационного портрета пользователя / Н. Г. Аксак, С. А. Коргут, И. В. Новосельцев // Системи обробки інформації. – Х. : ХУПС, 2013. – Вип. 2 (109). – С. 242–246.
5. Алексієв В. О. Управління розвитком транспортних систем (автоматика, телематика та мехатроніка на автомобільному транспорті) ВНЗ / В. О. Алексієв. – Харків : ХНАДУ, 2008. – 268 с.
6. Балдин К. В. Управленческие решения: учебник / К. В. Балдин, С. Н. Воробьев, В. Б. Уткин. – 4-е изд. – М. : Издательско-торговая корпорация «Дашков и К», 2014. – 496 с.
7. Балшикеева К. С. Современное геодезическое обеспечение внедрения трехмерного лазерного сканирования в строительстве [Текст] / К. С. Балшикеева // Молодой ученый. – 2015. – №11. – С. 264–266.
8. Будников С. А. Полумарковская модель сложного конфликта радиоэлектронных систем [Текст] / С. А. Будников // V Межд. конф. «Методы и средства управления технологическими процессами», Саранск, 19–21 ноября 2009 года. – Режим доступа : <http://fetmag.mrsu.ru/2009-2/>.
9. Букашкин С. А. Применение цифровых процессоров обработки сигналов – магистральный путь развития современных средств телекоммуникаций / С. А. Букашкин, Г. Ф. Лисицын, В. Г. Миронов // Труды 3-ей Международной конференции "Цифровая обработка сигналов и ее применение". – М. : Инсвязьиздат, 2000. – 315 с.
10. Вильхивская О. В. Методика планирования внедрения технологий электронного бизнеса на предприятиях машиностроительной отрасли / О. В. Вильхивская, Н. А. Брынза // Научные исследования: от теории к

практике: материалы III Междунар. науч. - практ. конф. (30.04.2015 г.). – Чебоксары : ЦНС «Интерактив плюс», 2015. – Т. 2. – С. 207–215.

11. Вихров А. А. Интеграция систем KPI/BSC и бюджетирования в единый инструмент управления / А. А. Вихров, П. А. Лекомцев // Управление компанией. – 2005. – № 10 (53). – С. 15. – Режим доступа : [http://iteam.ru/publications/finances/section\\_12/article\\_2634](http://iteam.ru/publications/finances/section_12/article_2634). – Дата обращения: 15 ноября 2016.

12. Вільхівська О. В. Методичний підхід до вибору технологій електронного бізнесу на підприємстві / О. В. Вільхівська // Системи обробки інформації : збірник наукових праць. – Х. : Харківський університет Повітряних Сил імені Івана Кожедуба, 2016. – Вип. 5 (142). – С. 228.

13. Вітлінський В. В. Моделювання економіки : навч. посіб. / В. В. Вітлінський. – К. : КНЕУ, 2003. – 408 с.

14. Галицкий А. В. Защита информации в сети – анализ технологий и синтез решений / А. В. Галицкий, С. Д. Рябко, В. Ф. Шаньгин. – М. : ДМК Пресс, 2004. – 616 с.

15. Герасименко В. А. Защита информации в автоматизированных системах обработки данных. В 2-х кн. Кн. 1 / В. А. Герасименко. – М., 1994. – 400 с.

16. Гнеденко Б. В. Введение в теорию массового обслуживания / Б. В. Гнеденко, И. Н. Коваленко. – Изд. 3-е, испр. и доп. – М. : Комкнига, 2005. – 400 с.

17. Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. – М. : Техносфера, 2005. – 1072 с.

18. Грушо А. А. Теоретические основы защиты информации / А. А. Грушо, Е. Е. Тимонина. – М. : Яхтсмен, 1996. – 192 с.

19. Деклараційний патент України на винахід № 56943А, Україна, МПК H03G5/08. Фільтр нижніх частот для підвищення стійкості і точності кутового автосупроводження літальних апаратів / О. В. Коломійцев, Г. В. Альошин, Д. П. Пашков – № 2002129790; заяв. 06.12.2002; опубл. 15.05.2003; Бюл. № 5. – 4 с.

20. Доповідь про стан інформатизації та розвиток інформаційного суспільства в Україні за 2014 рік [Електронний ресурс] // Державне агентство з питань науки, інновацій та інформатизації в Україні. – Режим доступу : <http://www.dknii.gov.ua/content/shchorichna-dopovid-pro-rozvytok-informaciynogo-suspilstva>.

21. Дубов Д. В. Інформаційне суспільство в Україні: глобальні виклики та національні можливості : аналітична доповідь / Д. В. Дубов, М. А. Ожеван, С. Л. Гнатюк. – К. : НІСД, 2010. – 29 с.
22. Дюбуа Д. Теория возможностей. Приложения к представлению знаний в информатике / Д. Дюбуа, А. Прад // Перевод с фр. – М. : Радио и связь, 1990. – 288 с.
23. Евсеев С. П. Методология оценивания безопасности информационных технологий автоматизированных банковских систем Украины / С. П. Евсеев // *Ukrainian Scientific Journal of Information Security*, 2016. – vol. 22, issue 3. – pp. 297–309.
24. Єжова Л. Ф. Управління інформаційною безпекою. В 2-х томах / Л. Ф. Єжова, І. О. Мачалін, Я. В. Невойт, В. О. Хорошко. – К. : Вид. ДУІКТ, 2010. – 201 с.
25. Жуков В. Г. Модель нарушителя прав доступа в автоматизированной системе / В. Г. Жуков, М. Н. Жукова, А. П. Стефаров // Программные продукты и системы. – 2012. – № 2. – С. 75 – 78.
26. Засядько А. А. Дослідження ефективності роботи виробничого підприємства / А. А. Засядько, С. С. Королюк // Матеріали International Scientific-Practical Conference Economic Development Strategy in Terms of European Integration., Kaunas, 2016. – 341 p.
27. Затхей В. А. Формалізація знань в системі дистанційного навчання / В. А. Затхей, І. П. Ковріжних // Системи обробки інформації. – Х. : Харківський університет Повітряних Сил імені Івана Кожедуба, 2012. – Вип. 8(106) – С. 242.
28. Затхей В. А. Использование функциональных сетей для означивания начальных условий в открытых экспертных системах / В. А. Затхей, С. Є. Селезньов // Збірник наукових праць / ХВУ. – Х., 2000. – Вип. 4(30). – С. 126–131.
29. Затхей В. А. Формально-логический аппарат представления знаний о процессах управления обучением в экспертных обучающих системах / В. А. Затхей, Н. В. Шаронова, И. Е. Лещенко // «АСУ и приборы автоматизации» № 130 / М-во образования и науки Украины, Харьковский национальный университет радиоэлектроники. – Х., 2005. – С. 52–56.
30. Зегджа Д. П. Основы безопасности информационных систем / Д. П. Зегджа, А. М. Ивашко. – М. : Горячая линия – Телеком, 2000. – 452 с.



31. Иванов В. Г. Сжатие изображения текста на основе статистического анализа и классификации вертикальных элементов строки [Текст] / В. Г. Иванов, Ю. В. Ломоносов, М. Г. Любарский // Восточно-Европейский журнал передовых технологий. – Харьков, 2014. – № 4/2 (70). – С. 4–15.
32. Иванов В. Г. Сжатие изображения текста на основе выделения символов и их классификации [Текст] / В. Г. Иванов, М. Г. Любарский, Ю. В. Ломоносов // Проблемы управления и информатики. – 2010. – № 6. – С. 111–122.
33. Иванов В. Г. Сжатие изображения текста на основе формирования и классификации вертикальных элементов строки в графическом словаре символьных данных [Текст] / В. Г. Иванов, М. Г. Любарский, Ю. В. Ломоносов // Проблемы управления и информатики. – 2011. – № 5. – С. 98–109.
34. Иванов В. Г. Формальное описание дискретных преобразований Хаара // Проблемы управления и информатики. – 2003. – № 5. – С. 68–75.
35. Иванов В. Г. Сокращение содержательной избыточности изображений на основе классификации объектов и фона / В. Г. Иванов, М. Г. Любарский, Ю. В. Ломоносов // Проблемы управления и информатики. – Київ, 2007. – № 3. – С. 93–102.
36. Информационные технологии и защита информации в информационно-коммуникационных системах : коллективная монография / С. П. Евсеев, М. Ю. Лосев, С. В. Минухин и др. ; под ред. В. С. Пономаренко. – Х. : Вид-во ТОВ «Щедра садиба плюс», 2015. – 486 с.
37. Информационные технологии и системы в управлении, образовании, науке : коллективная монография / А. В. Коломийцев и др. ; под ред. В. С. Пономаренко. – Х. : Цифрова друкарня № 1, 2013. – 278 с.
38. Информационные технологии и системы в управлении, образовании, науке : коллективная монография / С. П. Евсеев, М. Ю. Лосев, С. В. Минухин и др. ; под ред. В. С. Пономаренко. – Х. : Вид-во ТОВ «Цифрова друкарня № 1». – 2013. – 278 с.
39. Камер Дуглас Э. Сети TCP/IP, том 1. Принципы, протоколы и структура / Камер Дуглас Э. – М.: Издательский дом "Вильямс", 2003. – 445с.
40. Каплан Р. Стратегические карты. Трансформация нематериальных активов в материальные результаты / Р. Каплан, Д. П. Нортон. – М. : ЗАО «Олимп-Бизнес», 2005. – 512 с.

41. Кейнс Дж. М. Общая теория занятости, процента и денег / Дж. М. Кейнс. – М. : Гелиос АРВ, 2002. – 352 с.
42. Коваленко А. А. Подходы к оптимизации распределения задач управления по компонентам компьютерной системы, образующей систему управления объектом критического применения / А. А. Коваленко // Наука у техніка Повітряних Сил Збройних Сил України: науково-технічний журнал. – Х. : ХУ ПС, 2014. – Вип. 2(15). – С. 158–160.
43. Коваленко А. А. Подходы к синтезу информационной структуры системы управления объектом критического применения / А. А. Коваленко // Системы обработки информации : сборник научных трудов. – Х. : ХУ ВС, 2014. – Вип. 1 (117). – С. 180–184.
44. Коваленко А. А. Подходы к синтезу технической структуры компьютерной системы, образующей систему управления объектом критического применения / А. А. Коваленко // Сборник научных трудов Харьковского университета Воздушных Сил. – Х. : ХУ ВС, 2014. – Вип. 1 (38). – С. 116–119.
45. Коваленко А. В. Метод количественной оценки рисков разработки программного обеспечения / А. А. Смирнов, А. В. Коваленко, Н. Н. Якименко, А. П. Доренский // Збірник наукових праць Харківського університету Повітряних Сил. – Харків : ХУПС, 2016. – Вип. 2 (47). – С. 128–133.
46. Коломійцев О. В. Адаптована структура приймально-передавальної частки вимірювальних каналів для синтезу лазерної інформаційно-вимірювальної системи з використанням частотно-часового методу / О. В. Коломійцев // Системи обробки інформації. – Х. : ХУ ПС, 2011. – Вип. 4 (94). – С. 28–31.
47. Компьютерное зрение. Современный подход / Д. Форсайт, Д. Понс ; пер. с англ. – М. : Вильямс, 2004. – 928 с.
48. Корнеев В. Будущее высокопроизводительных вычислительных систем // Открытые системы. – М., 2003. – № 5. – С. 10–17.
49. Косяков А. Системная инженерия. Принципы и практика / под ред. В. К. Батоврина. – М. : ДМК Пресс, 2014. – 624 с.
50. Кунт М. Методы кодирования изображений второго поколения / М. Кунт, А. Икономопулос, М. Кошер // ТИИЭР. – 1985. – Т. 73, № 4. – С. 59–86.
51. Кучук Г. А. Концептуальний підхід до синтезу структури інформаційно-телекомунікаційної мережі / Г. А. Кучук, І. В. Рубан,

О. П. Давікоза // Системи обробки інформації : збірник наукових праць. – Х. : ХУ ПС, 2013. – Вип. 7 (114). – С. 106–112.

52. Кучук Г. А. Синтез стратифікованої інформаційної структури інтеграційної компоненти гетеро-генної складової Єдиної АСУ Збройними Силами України / Г. А. Кучук, О. П. Давікоза // Наука і техніка Повітряних Сил Збройних Сил України: науково-технічний журнал. – Х. : ХУ ПС, 2013. – № 3 (12). – С. 154–158.

53. Ленков С. В. Методы и средства защиты информации. В 2-х томах / С. В. Ленков, Д. А. Перегудов, В. А. Хорошко. – К. : Арий, 2008. – 344 с.

54. Листровой С. В. Модель и подход к планированию распределения ресурсов в гетерогенных Грид-системах / С. В. Листровой, С. В. Минухин // Международный научно-технический журнал "Проблемы управления и информатики". – 2012. – № 5. – С. 120–133.

55. Литвиненко К. В. Полумарковский гиперслучайный подход к оценке рисков систем // Збірник наукових праць. – ОДАТРЯ, 2014. – № 1 (4). – С. 77–80.

56. Лосев Ю. И. Автоматизация в сетях с коммутацией пакетов / Ю. И. Лосев, М. Ю. Лосев, Ф. К. Яковец. – К. : «Техніка», 1994. – 212 с.

57. Маракова И. И. Синтез и исследование методов верификации объектов электронного документооборота / И. И. Маракова, Л. А. Кузнецова, А. А. Сыропятов // Захист інформації. – 2008. – № 2. – С. 50–65.

58. Маракова І. І. Технологія цифрових водяних меток з головними покриваючими повідомленнями в нагляді бінарних зображень // Правове, нормативне та метрологічне забезпечення систем захисту інформації в Україні. – Науково-технічний збірник. – К. : НДЦ „Тезис” НТУУ „КПІ”, 2003. – Вип. 7. – С. 53–58.

59. Милославская Н. Г. Интрасети: доступ в Internet, защита : учебное пособие для вузов / Н. Г. Милославская, А. И.-Толстой. – М. : ЮНИТИ – ДАНА, 2000. – 527 с.

60. Минухин С. В. Имитационная модель и ее программная реализация планирования ресурсов Грид-системы / С. В. Минухин, С. В. Знахур // Проблеми програмування. – 2012. – № 2-3. Спеціальний випуск. – С. 133–142.

61. Минухин С. В. Исследование и анализ основанных на методе покрытия алгоритмов планирования для гетерогенной распределенной системы с использованием сценарного подхода / С. В. Минухин, С. В. Знахур //

Информационные технологии и системы в управлении, образовании, науке : монография. – Х. : Цифрова друкарня, 2013. – С. 39–54.

62. Минухин С. В. Модели и методы решения задач планирования в распределенных вычислительных системах : монография / С. В. Минухин. – Х. : Изд-во ООО "Щедрая усадьба плюс", 2014. – 324 с.

63. Міхайленко В. М. Математичний аналіз для економістів. / В. М. Міхайленко, Н. Д. Федоренко–К. : Вид-во ЄУФІМБ 2002. – 297 с.

64. Моделі визначення компетентностей у системі дистанційного навчання : монографія / В. П. Степанов, І. О. Борозенець, В. П. Бурдаєв та ін. ; за заг. ред. В. П. Степанова. – Х. : Вид. ХНЕУ, 2013. – 224 с.

65. Мокрышев А. Основные функции ETL-систем / А. Мокрышев // Хабрахабр: Разработка, 19 января 2015. – Режим доступа : <https://habrahabr.ru/post/248231>. – Дата обращения: 18 ноября 2016.

66. Москаленко В. В. Структура системы Enterprise Performance Management с учётом технологии каскадирования ключевых показателей деятельности / В. В. Москаленко, Н. Г. Фонта // Вісник Національного технічного університету «ХПІ». (Серія «Системний аналіз, управління та інформаційні технології»). – Харків: НТУ «ХПІ», 2016. – № 45. – С. 34–40.

67. Ніконов О. Я. Розроблення та впровадження інтернет-технологій для підвищення ефективності використання транспортних засобів / О. Я. Ніконов, В. О. Алексієв, В. Ю. Улько, Г. І. Середіна // Вісник СевНТУ. – 2013. – Вип. 142. – С. 69–72.

68. Новосельцев И. В. Показатели эффективности нейросетевой системы медицинской диагностики / И. В. Новосельцев, Н. Г. Аксак // Системи управління, навігації та зв'язку. – Київ, 2010. – Вип. 4 (16). – С. 142–149.

69. Новосельцев И. В. Система первичной диагностики меланомы кожи / И. В. Новосельцев, Н. Г. Аксак // Бионика интеллекта : научн.-техн. журнал. – 2010. – № 3 (74). – С. 94–98.

70. Нурдинов Р. А. Подходы и методы обоснования целесообразности выбора средств защиты информации [Электронный ресурс] / Р. А. Нурдинов, Т. Н. Батова // Современные проблемы науки и образования. – 2013. – № 2. – Режим доступа к ресурсу : <http://elibrary.ru/item.asp?id=21285749>.

71. Палагин А. В. Реконфигурируемые вычислительные системы: Основы и приложения / А. В. Палагин, В. Н. Опанасенко. – К. : Просвіта, 2006. – 280 с.

72. Патент на корисну модель № 43725, Україна, МПК H04 Q 1/453. Модифікований селектор подовжніх мод / О. В. Коломійцев, Г. В. Альошин, В. В. Белімов та ін. – № u200903693; заяв. 15.04.2009; опубл. 25.08.2009; Бюл. № 16. – 6 с.

73. Петров А. А. Компьютерная безопасность. Криптографические методы защиты информации / А. А. Петров. – М. : ДМК, 2000. – 448 с.

74. Про основні засади розвитку інформаційного суспільства в Україні на 2007-2015 роки: Закон України №537-V від 9 січня 2007 року // Відомості Верховної Ради України (ВВР). – 2007. – № 12. – ст. 102

75. Про схвалення Стратегії розвитку інформаційного суспільства в Україні: Розпорядження Кабінету Міністрів №386-р від 15 травня 2013 року. [Електронний ресурс]. – Режим доступу : <http://zakon2.rada.gov.ua/laws/show/386-2013-p>.

76. Проект Закону України “Про основні засади забезпечення кібербезпеки України” від 19.06.2015 № 2126а. – [Електронний ресурс]. – Режим доступу : [http://w1.c1.rada.gov.ua/pls/zweb2/webproc4\\_1?pf3511=55657](http://w1.c1.rada.gov.ua/pls/zweb2/webproc4_1?pf3511=55657).

77. Проект Стратегії забезпечення кібернетичної безпеки України. – [Електронний ресурс]. – Режим доступу : [http://www.niss.gov.ua/public/File/2013\\_nauk\\_an\\_rozrobku/kiberstrateg.pdf](http://www.niss.gov.ua/public/File/2013_nauk_an_rozrobku/kiberstrateg.pdf).

78. Пушкар О. І. Оцінка вартості розробки та впровадження технологій електронного бізнесу на підприємстві / О. І. Пушкар, О. В. Вільхівська // Бізнес-інформ. – Харків, 2016. – № 12. – С. 38–44.

79. Риз Дж. Облачные вычисления : пер. с англ. – СПб. : БХВ-Петербург, 2011. – 288 с.

80. Ричардсон Ян. Видеокодирование. H. 264 и MPEG-4 – стандарты нового поколения : пер. с англ. – М. : Техносфера, 2005. – 368 с.

81. Різник В. В. Синтез оптимальних комбінаторних систем. – Львів : Вища школа, 1989. – 165 с.

82. Різник О. Я. Метод побудови завадостійких кодів на основі ідеальних кільцевих відношень / О. Я. Різник, О. В. Повшук, Ю. Є. Кинаш, Б. І. Балич // Тези Міжнародної наукової конференції “Інтелектуальні системи прийняття рішень та проблеми обчислювального інтелекту” (ISDMCI'2015), 25–28 травня 2015 р., Залізний Порт. – С. 209–211.

83. Різник О. Я. Авторський захист графічних зображень за допомогою числових в'язанок / О. Я. Різник, Б. І. Балич, Д. Ю. Скрибайло-Леськів // Вісник

НУ "Львівська політехніка" "Комп'ютерні науки та інформаційні технології". – 2008. – С. 62–64.

84. Різник О. Я. Дослідження фільтрів для обробки зображень з використанням ідеальних кільцевих в'язанок / О. Я. Різник, А. Єрьоменко // Вісник НУ"ЛП" Комп'ютерні науки та інформаційні технології. – Львів : Видавництво НУ "Львівська політехніка", 2011. – № 719. – С. 13–19.

85. Різник О. Я. Інформаційні технології оптимізації структур технічних систем на основі нееквідистантних конфігурацій / О. Я. Різник, Ю. Є. Кинаш, Н. О. Кустра // Тези Міжнародної наукової конференції Присвяченої пам'яті професора Рогальського Ф.Б. "Інтелектуальні системи прийняття рішень та проблеми обчислювального інтелекту" (ISDMCI'2014), 28–31 травня 2014 р. – Залізний Порт, 2014. – С. 157–160.

86. РС БС ИББС – 2.2-2009. Обеспечение информационной безопасности организаций банковской системы Российской Федерации. Методика оценки рисков нарушения информационной безопасности – [Электронный ресурс]. – Режим доступа к ресурсу : [www.cbr.ru/credit/gubzi\\_docs/st22\\_09.pdf](http://www.cbr.ru/credit/gubzi_docs/st22_09.pdf).

87. Руденко О. Г. Козволюционирующие нейронные сети прямого распространения / О. Г. Руденко, А. А. Бессонов // Проблемы управления и информатики. – 2016. – № 5. – С. 63–73.

88. Сверхбольшие интегральные схемы и современная обработка сигналов: пер. с англ. / под ред. : С. Гуна, Х. Уайтхадса, Т. Кайлата. – М. : Радио и связь, 1989. – 472 с.

89. Теорія оптимальних рішень. Моделювання та керування в умовах невизначеності : зб. наук. пр. / ред. : Т. П. Мар'янович ; НАН України. Ін-т кібернетики ім. В. М. Глушкова. – К., 2000. – 174 с.

90. Федоренко Ю. С. Кластеризация данных на основе нейронного газа и марковских алгоритмов / Ю. С. Федоренко // Молодежный научно-технический вестник МГТУ им. Н. Э. Баумана. Электрон. журн. – 2014. – № 8. – Режим доступа : <http://sntbul.bmstu.ru/doc/730611.html>.

91. Филимонов А. Ю. Протоколы Интернета. – СПб. : БХВ-Петербург, 2008. – 528с.

92. Хома І. Б. Економіко-математичні методи аналізу діяльності підприємств / І. Б. Хома, В. В. Турко ; Нац. ун-т « Львів. Політехніка». – Л. : Вид-во Нац. ун-ту «Львів. Політехніка», 2008. – 328 с.

93. Хоффман Л. Современные методы защиты информации / Л. Хоффман. – М. : Сов. радио, 1980. – 264 с.
94. Чалая Л. Э. Оценивание пертинентности лингвистических дескрипторов в системах информационного поиска документов [Текст] / Л. Э. Чалая, Ю. Ю. Харитонова // Восточно-европейский журнал передовых технологий. – 2015. – № 1/9 (73). – С. 46–53.
95. Чалая Л. Э. Метод двухэтапной классификации электронных текстов [Текст] // Л. Э. Чалая, С. Г. Удовенко, Е. С. Кушвид // Бионіка інтелекту. – 2016. – № 2 (87). – С. 16–23.
96. Шинкарук Л. В. Структурні трансформації в економіці України: динаміка, суперечності та вплив на економічний розвиток : наукова доповідь / Л. В. Шинкарук, І. А. Бевз, І. В. Барановська та ін. ; за ред. чл.-кор. НАН України Л. В. Шинкарук ; НАН України, ДУ «Ін-т екон. та прогноз. НАН України». – К., 2015. – 304 с.
97. Шлезингер М. И. Математические средства обработки изображений [Текст] / М. И. Шлезингер. – Киев : Наукова думка, 1983. – 200 с.
98. Юдін О. К. Методологія побудови класифікатора загроз державним інформаційним ресурсам / О. К. Юдін, С. С. Бучик, А. В. Чунарьова, О. І. Варченко // Наукоємні технології. – 2014. – № 2 (22). – С. 200-210.
99. Ярочкин В. И. Безопасность информационных систем / В. И. Ярочкин. – М. : Ось-89, 1996. – 320 с.
100. Ярочкин В. И. Служба безопасности коммерческого предприятия / В. И. Ярочкин. – М. : Ось-89, 1995. – 144 с.
101. 3D редакторы, плюсы и минусы / Хабрахабр [Электронный ресурс]. – Режим доступа : <http://habrahabr.ru/post/136350/>.
102. Axak N. G. Development of multi-agent system of neural network diagnostics and remote monitoring of patient / N. G. Axak // Eastern-European Journal of Enterprise Technologies. – 2016. – 4/9 ( 82 ) – P. 4-11.
103. Blobel B. (2006). Advanced and secure architectural \* Encyclopedia of Healthcare Information Systems MEDICAL INFORMATION SCIENCE REFERENCE. – New York, 2008. – 231p.
104. Brian Evans. Practical 3D Printers: The Science and Art of 3D Printing / Evans Brian. – New York : Apress, 2012. – 332 p.

105. Coello C. A. Multi-objective optimization of trusses using genetic algorithms / C. A. Coello, A. D. Christiansen // *Computers & Structures*. – 2000. – 75. – P. 647–660.
106. Donkor E. Urban Water Demand Forecasting: Review of Methods and Models / Donkor E., Mazzuchi T., Soyer R., Roberson A. // *Journal of Water Resources Planning and Management*. – 2014. – № 140 (2). – P. 146–159.
107. Fibich P. Model of grid scheduling problem / P. Fibich, L. Matyska, H. Rudová // *Exploring Planning and Scheduling for Web Services, Grid and Autonomic Computing*, AAAI Press, 2005. – P. 17–24.
108. Furoo S. An enhanced self-organizing incremental neural network for online unsupervised learning / S. Furoo, T. Ogura, O. Hasegawa // *Neural Networks*. – 2007. – № 6. – pp. 893-903.
109. Furoo S. An incremental network for on-line unsupervised classification and topology learning / S. Furoo, O. Hasegawa // *Neural Networks*. – 2006. – Vol. 19, № 1. – pp. 90-106.
110. Garcia-Pedrajas N. Covnet: A cooperative coevolutionary model for evolving artificial neural networks / N. Garcia-Pedrajas, C. Hervas-Martinez, J. Munoz-Perez // *IEEE Transactions on Neural Networks*. – 2003. – 14 (3) . – P. 575–596.
111. Holland J. *Adaptation in Natural and Artificial Systems* 2nd edn. – Cambridge, MIT Press. – 1992. – 228 P.
112. Horielova K. A. Long-term and medium-term forecasting of water consumption of large cities / K. A. Horielova, V. M. Zadachyn // *Збірник наукових праць “Системи обробки інформації”*. – Харків : Вид. ХУПС, 2016. – Вип. № 1 (138). – С. 76-80.
113. Hryshchuk R. The synergetic approach for providing bank information security: the problem formulation // R. Hryshchuk, S. Yevseiev / *Безпека інформації*. – 2016. – № 22 (1). – С. 64–74. – doi:10.18372/2225-5036.22.10456.
114. Idris F. Review of Image and Video Indexing Techniques / F. Idris, S. Panchanathan // *Journal of Visual Communication and Image Representation*, 1997. – Vol. 8. – pp. 53–73.
115. Junguo L. Forecast of Water Demand in Weinan City in China using WDF-ANN model / L. Junguo, H. Savenije, J. Xu // *Physics and Chemistry of the Earth*. – 2003. – № 28. – P. 219–224.
116. Kudriashov V. 'Experimental Evaluation of Opportunity to Improve the Resolution of the Acoustic Maps'. In: Kountchev R. and Nakamatsu K. (eds.), *New*



Approaches in Intelligent Image Analysis, Intelligent Systems Reference Library 108, pp. 353-373. Springer International Publishing Switzerland 2016. DOI: 10.1007/978-3-319-32192-9\_11, SJR: 0.154.

117. Moulin P., O'Sullivan. Information-theoretic Analysis of Watermarking // Proc. of the International Conference on Acoustic, Speech and Signal Processing. – 2000. – Vol. 6. – P. 3630–3633.

118. MySQL. Оптимизация производительности, 2-е издание / Шварц Б., Зайцев П., Ткаченко В., Заводны Дж., Ленц А., Беллинг Д. ; пер. з англ. – СПб. : Символ-Плюс, 2010. – 832 с.

119. Semi-Markov risk models for finance, insurance and reliability [Electronic resource] / J. Jacques, M. Raimondo. - Electronic text data. - Boston, Ma : Springer Science + Business Media LLC, 2007.

120. Shklovets A. V. Visualization of High Dimensional Data Using Two Dimensional Self Organizing Piecewise Smooth Kohonen Maps /A. V. Shklovets and N. G. Ахак //ISSN 1060 992X, Optical Memory and Neural Networks (Information Optics), 2012. – Vol. 21, No. 4. – pp. 227–232.

121. Technical Papers from AT&T Labs: Электронный ресурс. – Режим доступа : <http://dvvuzone.org/techpapers/index.html>.

122. The Global Competitiveness Report 2012–2016 // [Электронный ресурс]. – Режим доступа : <https://www.weforum.org/>.

123. The Global Information Technology Report 2012 -2016 // [Электронный ресурс]. – Режим доступа : <https://www.weforum.org/>.

124. The Ultimate Guide to 3D Printing London: MagBook, 2014. – 148 p.

125. Tiwari M. K. Urban water demand forecasting and uncertainty assessment using ensemble wavelet-bootstrap-neural network models / M. K. Tiwari, J. Adamowski // Water Resources Research. – 2013. – vol. 49. – P. 6486–6507.

126. Top 25: Most Popular 3D Modeling Software for 3D Printing [Электронный ресурс] – Режим доступа: <https://i.materialise.com/blog/top-25-most-popular-3d-modeling-design-software-for-3d-printing>.

127. Yun Bai A multi-scale relevance vector regression approach for daily urban water demand forecasting / Yun Bai, Pu Wang, Chuan Li, Jingjing Xie, Yin Wang // Journal of Hydrology. – 2014. – vol. 517. – P. 236–245.

128. Zitzler E. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach / E. Zitzler, L. Thiele // IEEE Trans. on Evolutionary Computation. – 1999. – 3, N 4. – P. 257–271.

129. Petryshyn L. B. Theory of digital data processing in the ICT. In monography *Advances in ICT for Business, Industry and Public Sector*. Springer International Publishing Switzerland. 2015. –pp. 157-170.

130. Kehtarnavaz N. *Real-Time Digital Signal Processing*. Elsevier Inc. 2011. – 320 p.

131. Morgera S. *Digital Signal Processing: Applications to Communications and Algebraic Coding Theories*. Elsevier Inc. 2012. – 254 p.

132. Petryshyn L. Pelech-Pilichowski T. On a Property of Phase Correlation and Possibilities to Reduce the Walsh Function System // In monography: *Advances in Business ICT: New Ideas from Ongoing Research*. / ed-s Tomasz Pelech-Pilichowski, Maria Mach-Krol, Celina M. Olszak. *Studies in Computational Intelligence 658*. Springer International Publishing AG 2017, Cham, Switzerland. — P. 125–135.

133. Petryshyn L. Applying of the Walsh Functions Systems in Navigation Digital Data Processing. 4th International Conference *Methods and Systems of Navigation and Motion Control (MSNMC-2016)*. October 18-20, 2016. – P. 300-304.

134. Стахов А. П. Введение в алгоритмическую теорию измерения / А. П. Стахов. – М. : Советское Радио, 1977. – 288 с.

135. Петришин М. Л. Застосування векторно-розгалужуючих схем в моделюванні процесів ПФІ // II Міжнародна конференція “Комп’ютерна алгебра та інформаційні технології” : тези доповідей, 21-26 серпня 2016. / Одеський національний університет імені І. І. Мечникова. – Одеса, 2016. – 60с.

136. Брянский Л. Н. Метрология. Шкалы, эталоны, практика. – М. : ВНИИФТРИ, 2004. – 222 с.

137. Петришин Л. Б. Моделювання субтрактивно-адитивного способу перетворення форми інформації / Л. Б Петришин // Математичний вісник НТШ ; ISSN 1812 - 6774. – 2012. – т. 9 – P. 246–268. — Bibliogr. – P. 266–268.

## **Наукове видання**

*Аксак Наталія Георгіївна, Алексієв Олег Павлович, Алексієв Володимир Олександрович, Альошин Геннадій Васильович, Безсонов Олександр Олександрович, Бринза Наталя Олександрівна, Вільхівська Ольга Володимирівна, Дудикевич Валерій Богданович, Євсєєв Сергій Петрович, Задачин Віктор Михайлович, Засядько Аліна Анатоліївна, Затхей Володимир Анатолійович, Іванов Володимир Георгійович, Ізонін Іван Вікторович, Коваленко Андрій Анатолійович, Коваленко Олександр Володимирович, Коломійцев Олексій Володимирович, Королюк Сергій Степанович, Король Ольга Григорівна, Коц Григорій Павлович, Кузнецова Людмила Анатоліївна, Ломоносов Юрій Вячеславович, Лосєв Міхаїл Юрійович, Лотошинська Наталія Дмитрівна, Маракова Ірина Ільїнічна, Мельник Маргарита Олександрівна, Микитин Галина Василівна, Мінухін Сергій Володимирович, Моїсеєнко Аліна Сергіївна, Москаленко Валентина Володимирівна, Нікітін Геннадій Дмитрович, Петришин Любомир Богданович, Петришин Михайло Любомирович, Різник Олег Яремович, Ровінський Віктор Анатолійович, Руденко Олег Григорійович, Смірнов Олексій Анатолійович, Тєслєнко Олег Володимирович, Удовєнко Сергій Григорович, Фонта Наталія Григорівна, Хорошко Володимир Олексійович, Хохлачова Юлія Євгенівна, Чала Лариса Ернєстівна*

# **ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ: ПРОБЛЕМИ ТА ПЕРСПЕКТИВИ**

## **Монографія**

*За загальною редакцією д-ра екон. наук, професора В. С. Пономарєнка*

Підписано до друку 30.03.2017 р. Формат 60x84 1/16.  
Папір офсетний. Друк ризографічний. Ум. друк. арк. 26,0.  
Гарнітура Times New Roman. Наклад 300 прим. Зам. № 0330/8-17.

Видавець: Рожко Сергій Григорович  
Свідоцтво про внесення суб'єкта видавничої справи до державного реєстру видавців,  
виготовлювачів і розповсюджувачів видавничої продукції

ДК № 4924 від 24.06.2015 р.

Адреса для листування: а/с 11437, м. Харків, 61171

Надруковано з готових оригінал-макетів у друкарні ФОП Петров В. В.  
Єдиний державний реєстр юридичних осіб та фізичних осіб-підприємців.

Запис № 2400000000106167 від 08.01.2009 р.

61144, м. Харків, вул. Гв. Широнінців, 79в, к. 137, тел. (057) 78-17-137.

e-mail:bookfabrik@mail.ua