

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

*Кваліфікаційна наукова праця
на правах рукопису*

ЧИТУЛЯН ВАДИМ ОЛЕГОВИЧ

УДК 004.75:004.94(043.3)

**ДИСЕРТАЦІЯ
ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ АВТОМАТИЗОВАНОГО
ПРОЕКТУВАННЯ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ ДЛЯ
КОРПОРАТИВНИХ СТРУКТУР**

123 «Комп'ютерна інженерія»

12 «Інформаційні технології»

Подається на здобуття ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____ **В.О. Читулян**

(підпис, ініціали та прізвище здобувача)

Наукові керівники

Герцюк Микола Модестович

доктор філософії (PhD),

Поперешняк Світлана

Володимирівна

кандидат фізико-математичних

наук, доцент

АНОТАЦІЯ

Читулян В. О. Інформаційні технології автоматизованого проектування обчислювальних систем для корпоративних структур. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії з галузі знань 12 «Інформаційні технології» за спеціальністю 123 «Комп'ютерна інженерія» – Державний університет інформаційно-комунікаційних технологій Міністерства освіти і науки України, Київ, 2026.

У дисертаційній роботі розв'язано актуальне науково-прикладне завдання розроблення та дослідження методів та інформаційних технологій автоматизованого проектування обчислювальних систем для корпоративних структур. Робота висвітлює дослідження, спрямовані на усунення фундаментальних прогалин наявних засобів проектування, пов'язаних з відокремленим оцінюванням продуктивності, функціональної стійкості, надійності та безпеки функціонування, відсутністю відкаліброваної оцінки невизначеності прогнозів і значними обсягами натурних вимірювань, що обмежує можливість кількісно обґрунтувати конфігурацію та архітектуру системи ще до її розгортання. Сучасні корпоративні обчислювальні системи розгортаються у хмарних і гібридних середовищах, працюють за змінного навантаження та мають задовольняти жорсткі вимоги до продуктивності, безперервності функціонування й безпеки, унаслідок чого їх проектування перетворюється на неперервний процес кількісного обґрунтування проєктних рішень. Причиною прогалин є самостійний розвиток прогнозування продуктивності, теорії стійкості та забезпечення надійності й безпеки, а наслідком – пізнє виявлення архітектурних помилок, що зумовлює наукову суперечність між зростанням вимог до обчислювальних систем корпоративних структур і можливостями засобів їх автоматизованого проектування.

Об'єктом дослідження є процеси автоматизованого проектування та забезпечення функціонування обчислювальних систем корпоративних структур. Предметом дослідження є методи, моделі та інформаційні технології

автоматизованого проєктування обчислювальних систем корпоративних структур із забезпеченням продуктивності, функціональної стійкості, відмовостійкості та безпеки функціонування.

Метою роботи є підвищення ефективності автоматизованого проєктування обчислювальних систем для корпоративних структур шляхом розроблення сурогатної моделі прогнозування продуктивності, інформаційної технології забезпечення функціональної стійкості та методу й інформаційної технології структурно-параметричного проєктування відмовостійких і захищених розподілених систем. Для досягнення мети розв'язано такі завдання:

1. Проаналізовано сучасний стан автоматизованого проєктування обчислювальних систем для корпоративних структур, виявлено прогалини наявних підходів.
2. Розроблено метод автоматизованого проєктування на основі сурогатної моделі гаусівської процесової регресії.
3. Удосконалено інформаційну технологію забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи.
4. Розвинено метод та інформаційну технологію структурно-параметричного проєктування відмовостійких і захищених розподілених обчислювальних систем.
5. Виконано експериментальну верифікацію запропонованих методів та інформаційних технологій засобами комп'ютерного моделювання.

Методологічну основу дослідження становлять теорія ймовірностей і математична статистика, апарат гаусівської процесової регресії та машинного навчання, теорія масового обслуговування, методи статистичного контролю процесів і дисперсійного аналізу, математичне програмування й теорія графів, багатокритеріальна оптимізація і теорія надійності, методи оцінювання ризиків та аналізу «витрати – вигоди», а також методи комп'ютерного моделювання та обчислювального експерименту. Достовірність результатів підтверджено зіставленням запропонованих методів із коректно визначеними базовими альтернативами на єдиних фізично обґрунтованих синтетичних даних,

детермінованістю й відтворюваністю всіх експериментів із фіксованим початковим значенням генератора та узгодженою системою метрик.

Наукова новизна одержаних результатів полягає в наступному:

1. Вперше розроблено метод автоматизованого проєктування обчислювальних систем на основі сурогатної моделі гаусівської процесової регресії, яка, на відміну від адитивних моделей впливу, апроксимує нелінійну поверхню відгуку у просторі «конфігурація – навантаження» та забезпечує оцінювання невизначеності прогнозу за допомогою відкаліброваних довірчих інтервалів, що дозволяє скоротити обсяг натурних вимірювань та обґрунтовано визначати раціональну конфігурацію системи за змінних умов навантаження.

2. Удосконалено інформаційну технологію забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи, яка, на відміну від відомих реактивних рішень, інтегрує статистичний моніторинг аномалій, прогнозування поведінки методами машинного навчання, оптимізацію інформаційних потоків та оцінювання ризиків у єдиний контур керування, що забезпечує адаптацію системи в режимі реального часу та підтримання критичної функціональності в умовах збурень.

3. Набули подальшого розвитку метод та інформаційна технологія структурно-параметричного проєктування відмовостійких і захищених розподілених обчислювальних систем корпоративних структур, які відрізняються від існуючих використанням результатів методу автоматизованого проєктування на основі сурогатної моделі гаусівської процесової регресії для обґрунтування параметрів конфігурації та результатів інформаційної технології забезпечення функціональної стійкості для врахування вимог до стійкості системи, у поєднанні з багатокритеріальною оптимізацією розподілу ресурсів безпеки з урахуванням живучості, адаптивності та плавної деградації, що дозволяє обирати відмовостійку й захищену архітектуру системи.

Практичне значення одержаних результатів полягає в наступному:

1. Розроблений метод автоматизованого проєктування на основі сурогатної моделі гаусівської процесової регресії реалізовано у вигляді

програмного засобу, який забезпечує апроксимацію поверхні відгуку продуктивності з коефіцієнтом детермінації 0,970 проти 0,381 в адитивній моделі та відкалібровані довірчі інтервали з покриттям 1,00 за очікуваної похибки калібрування 0,055. Метод дозволяє досягти заданого порогу точності за 75 натурних вимірювань замість 720, тобто скоротити їх обсяг на 89,6%, і придатний для вибору мінімально достатньої конфігурації ресурсів корпоративних систем у хмарних і гібридних середовищах.

2. Удосконалену інформаційну технологію забезпечення функціональної стійкості реалізовано у вигляді програмних модулів і панелі моніторингу. Її застосування підвищує індекс функціональної стійкості з 0,917 до 0,996 і значення цільової функції стійкості з 0,481 до 0,681, повністю усуває недообслуговування інформаційних потоків під час відмов (з 6,0% до 0,0%), забезпечує прогнозування навантаження із середньою абсолютною відсотковою похибкою 5,2% і знижує втрати від часу реагування на 53,5%, що дозволяє службі експлуатації підтримувати безперервність функціонування корпоративної інформаційної системи.

3. Розвинені метод та інформаційну технологію структурно-параметричного проектування реалізовано у вигляді програмних модулів і панелі вибору архітектури. Їх застосування дозволяє на етапі проектування підвищити допустиму частку втрати компонентів із 40% до 70% за умови плавної деградації, знизити інтегральний ризик системи з 1,769 до 0,722, тобто на 59%, підвищити індекс безпеки відмовостійкої основи з 0,103 до 0,736 та обрати збалансовану архітектуру за найвищим композитним критерієм 0,896.

Дисертація складається зі вступу, п'яти розділів, висновків, списку використаних джерел та додатків. У першому розділі проаналізовано сучасний стан автоматизованого проектування обчислювальних систем для корпоративних структур і виявлено прогалини наявних підходів. У другому розділі розроблено метод автоматизованого проектування на основі сурогатної моделі гаусівської процесової регресії. У третьому розділі удосконалено інформаційну технологію забезпечення функціональної стійкості адаптивної

корпоративної інформаційної системи. У четвертому розділі розвинено метод та інформаційну технологію структурно-параметричного проєктування відмовостійких і захищених розподілених обчислювальних систем. У п'ятому розділі виконано експериментальну верифікацію засобами комп'ютерного моделювання.

Ключові слова: автоматизоване проєктування, обчислювальна система, корпоративна структура, сурогатна модель, гаусівська процесова регресія, продуктивність, довірчі інтервали, функціональна стійкість, машинне навчання, оптимізація інформаційних потоків, оцінювання ризиків, відмовостійкість, надійність, безпека функціонування, живучість, плавна деградація, багатокритеріальна оптимізація, розподілені обчислювальні системи, комп'ютерне моделювання, інформаційні технології.

SUMMARY

Chytulian V. O. Information Technologies for Computer-Aided Design of Computing Systems for Corporate Structures. – Qualifying scientific work as a manuscript.

Dissertation for the degree of Doctor of Philosophy in the field of knowledge 12 "Information Technologies" in specialty 123 "Computer Engineering" – State University of Information and Communication Technologies of the Ministry of Education and Science of Ukraine, Kyiv, 2026.

The dissertation is devoted to improving the efficiency of computer-aided design of computing systems for corporate structures while simultaneously accounting for performance, functional stability, reliability, and operational security. Modern corporate computing systems are deployed in cloud and hybrid environments, operate under variable workloads, and must satisfy strict requirements for performance, continuity of operation, and security, which turns their design into a continuous process of quantitative justification of design decisions. The common shortcoming of existing computer-aided design tools is that these properties are assessed by separate models at different stages of the life cycle: performance prediction relies on additive influence

models without a calibrated estimate of uncertainty, functional stability is ensured reactively, and a fault-tolerant secure architecture is justified qualitatively. The cause is the independent development of the corresponding research areas, and the consequence is the late detection of architectural errors and the inability to justify a design decision quantitatively before deployment, which gives rise to a scientific contradiction between the growing requirements for corporate computing systems and the capabilities of the tools for their computer-aided design.

The object of research is the processes of computer-aided design and operation assurance of computing systems for corporate structures. The subject of research is the methods, models, and information technologies of computer-aided design of computing systems for corporate structures that ensure performance, functional stability, fault tolerance, and operational security.

The aim of the work is to improve the efficiency of computer-aided design of computing systems for corporate structures through the development of a surrogate performance-prediction model, an information technology for ensuring functional stability, and a method and information technology of structural-parametric design of fault-tolerant and secure distributed systems. To achieve this aim, the following tasks were solved: the current state of computer-aided design of computing systems for corporate structures was analyzed, and the gaps of existing approaches were identified; a computer-aided design method based on a Gaussian process regression surrogate model was developed; the information technology for ensuring the functional stability of an adaptive corporate information system was improved; the method and information technology of structural-parametric design of fault-tolerant and secure distributed computing systems were further developed; and the proposed methods and information technologies were experimentally verified by means of computer modeling.

The methodological basis of the research consists of probability theory and mathematical statistics, the apparatus of Gaussian process regression and machine learning, queueing theory, methods of statistical process control and analysis of variance, mathematical programming and graph theory, multi-criteria optimization and

reliability theory, methods of risk assessment and cost-benefit analysis, as well as methods of simulation modeling and computational experiment. The validity of the results was confirmed by comparing the proposed methods with correctly defined baseline alternatives on the same physically grounded synthetic data, by the determinism and reproducibility of all experiments with fixed generator seed values, and by a consistent system of metrics.

The dissertation presents the following new scientific results: 1. For the first time, a method of computer-aided design of computing systems based on a Gaussian process regression surrogate model has been developed; unlike additive influence models, the model approximates the nonlinear response surface in the "configuration – workload" space and provides an estimate of prediction uncertainty by means of calibrated confidence intervals, which makes it possible to reduce the volume of full-scale measurements and to determine a rational system configuration under varying workload conditions in a substantiated way. 2. The information technology for ensuring the functional stability of an adaptive corporate information system has been improved; unlike known reactive solutions, it integrates statistical anomaly monitoring, behavior forecasting by machine learning methods, information-flow optimization, and risk assessment into a single control loop, which ensures real-time adaptation of the system and the preservation of critical functionality under disturbances. 3. The method and information technology of structural-parametric design of fault-tolerant and secure distributed computing systems for corporate structures have been further developed; they differ from existing ones in using the results of the computer-aided design method based on a Gaussian process regression surrogate model to substantiate configuration parameters and the results of the information technology for ensuring functional stability to account for system stability requirements, combined with multi-criteria optimization of security-resource allocation with account for survivability, adaptivity, and graceful degradation, which makes it possible to select a fault-tolerant and secure system architecture.

The practical significance of the obtained results is as follows: 1. The developed computer-aided design method based on a Gaussian process regression surrogate

model is implemented as a software tool that approximates the performance response surface with a coefficient of determination of 0.970 versus 0.381 for the additive model and provides calibrated confidence intervals with a coverage of 1.00 at an expected calibration error of 0.055; the method reaches the specified accuracy threshold with 75 full-scale measurements instead of 720, that is, reduces their volume by 89.6%, and is suitable for selecting the minimally sufficient resource configuration of corporate systems in cloud and hybrid environments. 2. The improved information technology for ensuring functional stability is implemented as software modules and a monitoring panel; its application raises the functional-stability index from 0.917 to 0.996 and the stability objective function from 0.481 to 0.681, completely eliminates the under-servicing of information flows during failures (from 6.0% to 0.0%), provides load forecasting with a mean absolute percentage error of 5.2%, and reduces response-time losses by 53.5%. 3. The further developed method and information technology of structural-parametric design are implemented as software modules and an architecture-selection panel; their application makes it possible at the design stage to raise the admissible share of component loss from 40% to 70% under graceful degradation, reduce the integral risk of the system from 1.769 to 0.722, that is, by 59%, increase the security index of the fault-tolerant baseline from 0.103 to 0.736, and select a balanced architecture by the highest composite criterion of 0.896. The results of the dissertation research have been implemented at the Institute of Software Systems of the National Academy of Sciences of Ukraine, as well as in the educational process at the State University of Information and Communication Technologies, a fact confirmed by implementation certificates and at Byte Orchard Consulting LLC.

The dissertation consists of an introduction, five chapters, conclusions, a list of references, and appendices. The first chapter analyzes the current state of computer-aided design of computing systems for corporate structures and identifies the gaps of existing approaches. The second chapter develops the computer-aided design method based on a Gaussian process regression surrogate model. The third chapter improves the information technology for ensuring the functional stability of an adaptive corporate information system. The fourth chapter further develops the method and

information technology of structural-parametric design of fault-tolerant and secure distributed computing systems. The fifth chapter presents the experimental verification by means of computer modeling.

Keywords: computer-aided design, computing system, corporate structure, surrogate model, Gaussian process regression, performance, confidence intervals, functional stability, machine learning, information-flow optimization, risk assessment, fault tolerance, reliability, operational security, survivability, graceful degradation, multi-criteria optimization, distributed computing systems, computer modeling, information technologies.

Список опублікованих праць за темою дисертації

Наукові праці, у яких опубліковані основні наукові результати дисертації:

1. Жебка В. В., Читулян В. О. Модель на основі гаусівської регресії для прогнозування продуктивності корпоративних веб-систем за конфігурованих умов навантаження. Телекомунікаційні та інформаційні технології. 2026. № 2 (91). С. 215–222. DOI: 10.31673/2412-4338.2026.029118.

Особистий внесок здобувача: розроблено сурогатну модель прогнозування продуктивності на основі гаусівської процесової регресії, запропоновано конструювання ознак на основі моделі черги масового обслуговування, виконано програмну реалізацію та обчислювальні експерименти, оцінено відкаліброваність довірчих інтервалів і скорочення обсягу натурних вимірювань; співавтором здійснено постановку задачі та наукове консультування щодо вибору показників продуктивності.

2. Ананченко О. Є., Миронюк П. Я., Нестеренко К. С., Читулян В. О. Розробка адаптивної корпоративної освітньої інформаційної системи з використанням методів машинного навчання. Телекомунікаційні та інформаційні технології. 2025. № 4 (89). С. 189–195. DOI: 10.31673/2412-4338.2025.048922.

Особистий внесок здобувача: запропоновано математичну модель функціональної стійкості адаптивної корпоративної інформаційної системи з цільовою функцією безпеки, ризику й адаптивності, статистичний моніторинг аномалій за правилом трьох сигм, модель прогнозування поведінки користувачів методами машинного навчання та модель оптимізації інформаційних потоків; співавторами здійснено аналіз вимог до корпоративної освітньої платформи, огляд наявних рішень і загальне наукове керівництво дослідженням.

3. Гангало І. М., Читутлян В. О. Кіберфізичні системи екологічного моніторингу: концептуальні засади захисту критичної інфраструктури в умовах війни. Кібербезпека: освіта, наука, техніка. 2025. № 4 (28). С. 688–700. DOI: 10.28925/2663-4023.2025.28.843.

Особистий внесок здобувача: розроблено математичний апарат оцінювання захищеності розподіленої системи – цільову функцію безпеки з обмеженнями, моделі ймовірності успішної атаки, живучості, адаптивності, плавної деградації та інтегрального ризику; співавтором здійснено аналіз загроз об'єктам критичної інфраструктури екологічного моніторингу в умовах війни та характеристику предметної області.

4. Читутлян В. О., Олейніков І. А., Ананченко О. Є., Дзюба В. В. Web-програмування і мобільна розробка: UX/UI-рішення для захисту персональних даних користувачів. Кібербезпека: освіта, наука, техніка. 2025. Спецвипуск № 2 (30). С. 543–554. DOI: 10.28925/2663-4023.2025.30.992.

Особистий внесок здобувача: запропоновано концепцію вбудовування принципів захисту персональних даних у користувацький шлях веб- і мобільних застосунків та критерії її оцінювання; співавторами здійснено аналіз практик веб- і мобільної розробки та участь в обговоренні результатів.

5. Читутлян В. О., Поперешняк С. В. Інформаційні технології автоматизованого проєктування обчислювальних систем для корпоративних структур: сучасний стан та перспективи. Кібербезпека: освіта, наука, техніка. 2026. Т. 2, № 34. С. 831–846. DOI: 10.28925/2663-4023.2026.34.

Особистий внесок здобувача: виконано аналіз і систематизацію підходів до прогнозування продуктивності, забезпечення функціональної стійкості та проєктування відмовостійких і захищених систем, виявлено прогалини та сформульовано напрями досліджень; співавтором визначено мету огляду та здійснено загальне редагування.

6. Hanhalo I., Chytulian V., Zhebka V., Bebeshko B., Khorolska K. Adaptive approach to ensuring the functional stability of corporate educational platforms under dynamic cyber threats. Cybersecurity Providing in Information and Telecommunication Systems (CPITS 2025) : proc. of the Workshop. CEUR Workshop Proceedings. 2025. Vol. 3991. P. 481–491. URL: <https://ceur-ws.org/Vol-3991/paper34.pdf> (Scopus).

Особистий внесок здобувача: розроблено моделі залишкового ризику залежно від витрат на безпеку, рівня стійкості з пороговим ефектом і втрат від часу реагування, а також виконано сценарне моделювання; співавторами здійснено аналіз загроз корпоративним освітнім платформам, огляд підходів до оцінювання кіберризиків і постановку задачі.

7. Chytulian V., Kolodiuk A., Oleinikov I., Zhebka V., Balatska V. Formation of a conceptual model for cyber-physical monitoring of critical infrastructure environmental objects. Cybersecurity Providing in Information and Telecommunication Systems II (CPITS-II 2025) : proc. of the Workshop. CEUR Workshop Proceedings. 2025. Vol. 4145. P. 330–341. URL: <https://ceur-ws.org/Vol-4145/paper23.pdf> (Scopus).

Особистий внесок здобувача: сформовано концептуальну модель кіберфізичного моніторингу та її математичний апарат – цільову функцію безпеки, моделі ймовірності атаки, живучості, адаптивності й плавної деградації, і багаторівневу архітектуру захисту; співавторами здійснено аналіз сенсорного рівня та засобів захищеної передачі даних, загроз в умовах воєнного стану та наукове консультування.

Наукові праці, які засвідчують апробацію матеріалів дисертації:

1. Читулян В. О., Золотухіна О. А. Система тест-менеджменту як інструмент для керування тестуванням. Mathematics and Mathematical Simulation in a Modern Technical University : зб. тез доп. I Міжнар. наук.-практ. конф. студентів та молодих вчених (Луцьк, 30 листоп. 2022 р.). Луцьк : ДонНТУ, 2022. С. 10.
2. Читулян В. О. Побудова моделі менторської підтримки QA trainee. Сучасні інфокомунікаційні технології : зб. тез XV наук.-техн. конф. студентів та молодих вчених ф-ту інформ. технологій (Київ, 9 груд. 2022 р.). Київ : ДУТ, 2022. С. 19–20.
3. Читулян В. О. Методи та засоби організації менторської підтримки процесу тестування в ІТ-проекті. Сучасні комп'ютерні та інформаційні системи і технології : матеріали III Всеукр. наук.-практ. інтернет-конф. (Запоріжжя, 12–19 груд. 2022 р.). Запоріжжя : ТДАТУ, 2022. С. 45.
4. Гангало І. М., Читулян В. О. Особливості оптимізації виробничих процесів з використанням алгоритмів машинного навчання. II Всеукраїнська науково-технічна конференція «Технологічні горизонти: дослідження та застосування інформаційних технологій для технологічного прогресу України і світу» : зб. тез (Київ, 18 листоп. 2024 р.). Київ : ДУІКТ, 2024. С. 109–111.
5. Гангало І. М., Читулян В. О. Стратегії використання хмарних технологій і III для масштабування стартапів. Сучасні досягнення компанії Hewlett Packard Enterprise в галузі ІТ та нові можливості їх вивчення і застосування : зб. тез VI Міжнар. наук.-практ. конф. (Київ, 12 груд. 2024 р.). Київ : ДУІКТ, 2024. С. 46–47.
6. Гангало І. М., Читулян В. О. Хмарні обчислення та їх значення для малого та середнього бізнесу. Сучасні аспекти діджиталізації та інформатизації в програмній та комп'ютерній інженерії : зб. тез II Міжнар. наук.-практ. конф. (Київ, 19–21 груд. 2024 р.). Київ : ДУІКТ, 2024. С. 209–210.

ЗМІСТ

ВСТУП.....	17
РОЗДІЛ 1. АНАЛІЗ СУЧАСНОГО СТАНУ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ ДЛЯ КОРПОРАТИВНИХ СТРУКТУР.....	25
1.1. Корпоративні обчислювальні системи як об'єкт автоматизованого проєктування.....	25
1.2. Сурогатне моделювання та прогнозування продуктивності конфігурованих систем.....	31
1.3. Функціональна стійкість адаптивних корпоративних інформаційних систем	36
1.4. Відмовостійкість і безпека функціонування розподілених корпоративних обчислювальних систем	41
1.5. Аналіз прогалин і постановка задач дослідження	45
Висновки до розділу 1	48
РОЗДІЛ 2. МЕТОД АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ НА ОСНОВІ СУРОГАТНОЇ МОДЕЛІ ГАУСІВСЬКОЇ ПРОЦЕСОВОЇ РЕГРЕСІЇ	50
2.1. Постановка задачі сурогатного моделювання продуктивності обчислювальної системи	50
2.2. Формалізація простору «конфігурація – навантаження»	52
2.3. Гаусівська процесова регресія як апарат сурогатної моделі	56
2.4. Конструювання ознак на основі фізичної моделі системи	60
2.5. Калібрування довірчих інтервалів та кількісна оцінка невизначеності	66
2.6. Критерій активного навчання за предиктивною невизначеністю ..	69

2.7. Метод автоматизованого конфігурування обчислювальної системи на основі сурогатної моделі	73
Висновки до розділу 2	79
РОЗДІЛ 3. ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ЗАБЕЗПЕЧЕННЯ ФУНКЦІОНАЛЬНОЇ СТІЙКОСТІ АДАПТИВНОЇ КОРПОРАТИВНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ	81
3.1. Постановка задачі забезпечення функціональної стійкості	81
3.2. Математична модель функціональної стійкості	88
3.3. Статистичний моніторинг стану системи та виявлення аномалій..	94
3.4. Прогнозування поведінки користувачів методами машинного навчання	98
3.5. Оптимізація інформаційних потоків і ресурсів	102
3.6. Оцінювання ризиків, витрат і рівня інформаційної безпеки	105
3.7. Структура інформаційної технології забезпечення функціональної стійкості.....	108
Висновки до розділу 3	111
РОЗДІЛ 4. МЕТОД ТА ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ СТРУКТУРНО-ПАРАМЕТРИЧНОГО ПРОЄКТУВАННЯ ВІДМОВОСТІЙКИХ І ЗАХИЩЕНИХ РОЗПОДІЛЕНИХ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ.....	113
4.1. Постановка задачі проєктування відмовостійких і захищених розподілених систем	113
4.2. Багатокритеріальна оптимізація розподілу ресурсів безпеки	119
4.3. Модель ймовірності успішної атаки	124
4.4. Моделі живучості, адаптивності та плавної деградації	127
4.5. Модель оцінювання ризиків	131
4.6. Механізми забезпечення захищеності	134

4.7. Метод та інформаційна технологія структурно-параметричного проєктування відмовостійких і захищених систем	138
Висновки до розділу 4	146
РОЗДІЛ 5. ЕКСПЕРИМЕНТАЛЬНА ВЕРИФІКАЦІЯ РЕЗУЛЬТАТІВ	148
5.1. Програма й методика експериментальної верифікації	148
5.2. Верифікація методу сурогатного моделювання продуктивності..	151
5.3. Верифікація інформаційної технології функціональної стійкості	160
5.4. Верифікація методу проєктування відмовостійких і захищених систем	167
5.5. Зведений аналіз результатів верифікації	176
Висновки до розділу 5	178
ВИСНОВКИ.....	180
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	183
Додаток А. Акти впровадження.....	202
Додаток Б. Програмне забезпечення експериментальної верифікації тріади...	208
Додаток В. Специфікація синтетичних наборів даних і калібрувальних моделей	212
Додаток Г. Панелі підтримки рішень інформаційних технологій	216

ВСТУП

Актуальність теми. Сучасний розвиток корпоративних структур – банківських і фінансових установ, телекомунікаційних операторів, електронної комерції, логістичних та освітніх платформ – супроводжується стрімким ускладненням обчислювальних систем та ІТ-інфраструктур, що обслуговують їхні бізнес-процеси. Такі системи розгортаються у хмарних і гібридних середовищах, поєднують десятки сервісів, працюють за змінного навантаження та мають задовольняти жорсткі вимоги до продуктивності, надійності й безпеки функціонування, тож їх проєктування перетворилося з одноразового інженерного акту на неперервний процес, у якому конфігурацію ресурсів, механізми забезпечення стійкості та архітектуру захисту доводиться обґрунтовувати кількісно ще до впровадження.

Автоматизоване проєктування корпоративної обчислювальної системи охоплює щонайменше три взаємопов'язані задачі життєвого циклу. На етапі конфігурування потрібно передбачити продуктивність системи за варіативних конфігурацій і навантажень, не вдаючись до дорогого натурного перебору всіх варіантів; на етапі експлуатації – підтримувати функціональну стійкість, тобто здатність зберігати критичну функціональність за змінних навантажень, змін поведінки користувачів і зовнішніх збурень; на етапі структурного проєктування – обирати архітектуру, стійку до втрати значної частки компонентів і до загроз безпеці функціонування. Кожну з цих задач ускладнюють нелінійність поверхні відгуку продуктивності, висока вартість натурних вимірювань, динамічність експлуатаційного середовища та зростання інтенсивності цілеспрямованих атак на розподілені обчислювальні ресурси.

Аналіз сучасних наукових досліджень показав, що наявні підходи розв'язують ці задачі здебільшого відокремлено та частково. Прогнозування продуктивності часто спирається на адитивні моделі впливу окремих параметрів, які не відтворюють нелінійних взаємодій у просторі «конфігурація – навантаження» й не надають відкаліброваної оцінки невизначеності, тоді як

пряме профілювання лишається витратним. Забезпечення функціональної стійкості реалізується переважно реактивно, окремими засобами моніторингу, без інтеграції їх у єдиний контур адаптації з урахуванням ризиків і витрат. Проєктування відмовостійких і захищених розподілених систем зазвичай виконується якісно, без багатокритеріального обґрунтування розподілу обмежених ресурсів безпеки та без кількісних показників живучості й плавної деградації, які можна було б закласти вже на етапі проєктування.

Загальним недоліком наявних засобів автоматизованого проєктування є те, що продуктивність, функціональну стійкість, надійність і безпеку обчислювальної системи оцінюють окремими моделями на різних етапах життєвого циклу. Причина полягає в тому, що прогнозування продуктивності, теорія стійкості та забезпечення надійності й безпеки сформувалися як самостійні напрями з власними моделями й показниками: адитивні моделі обирали за простоту та інтерпретованість, перевірку рішень покладали на дорогі натурні випробування, а механізми резервування й захисту додавали вже після вибору архітектури. Звідси виникають пізні виявлення архітектурних помилок, коли їх виправлення потребує повернення до раніше ухвалених рішень, надмірне або недостатнє резервування ресурсів і неможливість кількісно обґрунтувати проєктне рішення до розгортання системи. Таким чином, виникає наукова суперечність між зростанням вимог до продуктивності, безперервності функціонування та безпеки обчислювальних систем корпоративних структур і можливостями наявних засобів автоматизованого проєктування, які оцінюють ці властивості відокремлено, спираються на спрощені моделі та значні обсяги натурних вимірювань.

Отже, розроблення методів та інформаційних технологій автоматизованого проєктування обчислювальних систем для корпоративних структур, спрямованих на сурогатне прогнозування продуктивності з відкаліброваною оцінкою невизначеності, інтегроване забезпечення функціональної стійкості в режимі реального часу та багатокритеріальне

проектування відмовостійкої й захищеної архітектури, є актуальним науково-прикладним завданням.

Дисертаційна робота виконана в межах науково-дослідної роботи «Підвищення ефективності процесу управління 3D принтером з використанням методів машинного навчання» (державний реєстраційний номер РК 0124U001849, термін виконання 2024-2026, ДУІКТ, м. Київ) та госпдоговірної науково-дослідної роботи «Комплексна інформаційно-аналітична система дослідження освітніх траєкторій і професійної реалізації випускників».

Мета і завдання дослідження. Метою роботи є підвищення ефективності автоматизованого проектування обчислювальних систем для корпоративних структур шляхом розроблення сурогатної моделі прогнозування продуктивності, інформаційної технології забезпечення функціональної стійкості та методу й інформаційної технології структурно-параметричного проектування відмовостійких і захищених розподілених систем.

Для досягнення мети сформульовано й розв'язано такі **задачі**:

1. Проаналізовано сучасний стан автоматизованого проектування обчислювальних систем для корпоративних структур, підходи до прогнозування продуктивності, забезпечення функціональної стійкості та проектування відмовостійких і захищених розподілених систем; виявлено прогалини наявних підходів.

2. Розроблено метод автоматизованого проектування обчислювальних систем на основі сурогатної моделі гаусівської процесової регресії, що апроксимує нелінійну поверхню відгуку у просторі «конфігурація – навантаження» та надає відкалібровані довірчі інтервали.

3. Удосконалено інформаційну технологію забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи на основі інтеграції статистичного моніторингу аномалій, прогнозування поведінки методами машинного навчання, оптимізації інформаційних потоків та оцінювання ризиків.

4. Розвинено метод та інформаційну технологію структурно-параметричного проектування відмовостійких і захищених розподілених

обчислювальних систем на основі багатокритеріальної оптимізації розподілу ресурсів безпеки з урахуванням функцій живучості, адаптивності та плавної деградації.

5. Виконано експериментальну верифікацію запропонованих методів та інформаційних технологій засобами комп'ютерного моделювання й перевірено достовірність отриманих результатів.

Об'єктом дослідження є процеси автоматизованого проєктування та забезпечення функціонування обчислювальних систем корпоративних структур.

Предметом дослідження є методи, моделі та інформаційні технології автоматизованого проєктування обчислювальних систем корпоративних структур із забезпеченням продуктивності, функціональної стійкості, відмовостійкості та безпеки функціонування.

Методи дослідження. У дослідженні використано методи теорії ймовірностей, математичної статистики та машинного навчання, методи теорії масового обслуговування та конструювання ознак – для формалізації простору «конфігурація – навантаження»; методи статистичного контролю процесів і дисперсійного аналізу – для моніторингу стану системи та виявлення аномалій; методи математичного програмування й теорії графів – для оптимізації інформаційних потоків і розподілу ресурсів; методи багатокритеріальної оптимізації та теорії надійності – для проєктування відмовостійких і захищених розподілених систем, оцінювання живучості; методи оцінювання ризиків і аналізу «витрати – вигоди» – для економічного обґрунтування рішень із забезпечення безпеки; методи математичного й комп'ютерного моделювання та обчислювального експерименту – для верифікації запропонованих методів і технології та перевірки достовірності результатів.

У дисертації надані наступні нові *наукові результати*:

1. Вперше розроблено метод автоматизованого проєктування обчислювальних систем на основі сурогатної моделі гаусівської процесової регресії, яка, на відміну від адитивних моделей впливу, апроксимує нелінійну поверхню відгуку у просторі «конфігурація – навантаження» та забезпечує

оцінювання невизначеності прогнозу за допомогою відкаліброваних довірчих інтервалів, що дозволяє скоротити обсяг натурних вимірювань та обґрунтовано визначати раціональну конфігурацію системи за змінних умов навантаження.

2. Удосконалено інформаційну технологію забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи, яка, на відміну від відомих реактивних рішень, інтегрує статистичний моніторинг аномалій, прогнозування поведінки методами машинного навчання, оптимізацію інформаційних потоків та оцінювання ризиків у єдиний контур керування, що забезпечує адаптацію системи в режимі реального часу та підтримання критичної функціональності в умовах збурень.

3. Набули подальшого розвитку метод та інформаційна технологія структурно-параметричного проєктування відмовостійких і захищених розподілених обчислювальних систем корпоративних структур, які відрізняються від існуючих використанням результатів методу автоматизованого проєктування на основі сурогатної моделі гаусівської процесової регресії для обґрунтування параметрів конфігурації та результатів інформаційної технології забезпечення функціональної стійкості для врахування вимог до стійкості системи, у поєднанні з багатокритеріальною оптимізацією розподілу ресурсів безпеки з урахуванням живучості, адаптивності та плавної деградації, що дозволяє обирати відмовостійку й захищену архітектуру системи.

Практичне значення одержаних результатів полягає в наступному:

1. Розроблений метод автоматизованого проєктування на основі сурогатної моделі гаусівської процесової регресії реалізовано у вигляді програмного засобу, який забезпечує апроксимацію поверхні відгуку продуктивності з коефіцієнтом детермінації 0,970 проти 0,381 в адитивної моделі та відкалібровані довірчі інтервали з покриттям 1,00 за очікуваної похибки калібрування 0,055. Метод дозволяє досягти заданого порогу точності за 75 натурних вимірювань замість 720, тобто скоротити їх обсяг на 89,6%, і придатний для вибору мінімально достатньої конфігурації ресурсів корпоративних систем у хмарних і гібридних середовищах.

2. Удосконалену інформаційну технологію забезпечення функціональної стійкості реалізовано у вигляді програмних модулів і панелі моніторингу. Її застосування підвищує індекс функціональної стійкості з 0,917 до 0,996 і значення цільової функції стійкості з 0,481 до 0,681, повністю усуває недообслуговування інформаційних потоків під час відмов (з 6,0% до 0,0%), забезпечує прогнозування навантаження із середньою абсолютною відсотковою похибкою 5,2% і знижує втрати від часу реагування на 53,5%, що дозволяє службі експлуатації підтримувати безперервність функціонування корпоративної інформаційної системи.

3. Розвинені метод та інформаційну технологію структурно-параметричного проектування реалізовано у вигляді програмних модулів і панелі вибору архітектури. Їх застосування дозволяє на етапі проектування підвищити допустиму частку втрати компонентів із 40% до 70% за умови плавної деградації, знизити інтегральний ризик системи з 1,769 до 0,722, тобто на 59%, підвищити індекс безпеки відмовостійкої основи з 0,103 до 0,736 та обрати збалансовану архітектуру за найвищим композитним критерієм 0,896.

Результати дисертаційної роботи впроваджено в Інституті програмних систем НАН України для обґрунтування проєктних рішень щодо конфігурування, функціональної стійкості та захищеності обчислювальної інфраструктури інформаційно-аналітичних систем установи, а також у навчальний процес Державного університету інформаційно-комунікаційних технологій у дисциплінах «Технології проектування комп'ютерних систем», «Архітектура інформаційних систем» і «Паралельні та розподілені обчислення», що підтверджено актами впровадження та в ТОВ «Байт Орчард Консалтінг», яка надала статистичне підтвердження отриманих результатів.

Особистий внесок здобувача. Усі результати, подані до захисту, отримано автором особисто. У наукових публікаціях, підготовлених у співавторстві, здобувачеві належать: сурогатна модель прогнозування продуктивності корпоративних веб-систем на основі гаусівської процесової регресії та конструювання ознак для поверхні відгуку продуктивності [162]; складові

адаптивної корпоративної інформаційної системи з прогнозуванням поведінки методами машинного навчання як основа забезпечення її функціональної стійкості [163]; математичний апарат оцінювання захищеності розподілених систем – цільова функція безпеки, функції живучості, адаптивності та плавної деградації [164]; UX/UI-рішення захисту персональних даних користувачів у веб- і мобільних застосунках [165]; узагальнення сучасного стану й перспектив автоматизованого проєктування обчислювальних систем для корпоративних структур [166]; модель залишкового ризику, вартісної оптимізації захисту та скорочення часу реагування як складова функціональної стійкості корпоративної платформи [167]; концептуальна модель кіберфізичного моніторингу об'єктів критичної інфраструктури та апарат оцінювання її живучості й захищеності [168].

Апробація результатів дисертації. Ключові положення та практичні результати дисертаційної роботи пройшли апробацію шляхом представлення на всеукраїнських та міжнародних науково-практичних конференціях:

I Міжнародна науково-практична конференція студентів та молодих вчених «Математика та математичне моделювання у сучасному технічному університеті» (Mathematics and Mathematical Simulation in a Modern Technical University, м. Луцьк, 2022);

XV науково-технічна конференція студентів та молодих вчених факультету інформаційних технологій «Сучасні інфокомунікаційні технології» (м. Київ, 2022);

III Всеукраїнська науково-практична інтернет-конференція «Сучасні комп'ютерні та інформаційні системи і технології» (м. Запоріжжя, 2022);

II Всеукраїнська науково-технічна конференція «Технологічні горизонти: дослідження та застосування інформаційних технологій для технологічного прогресу України і світу» (м. Київ, 2024);

VI Міжнародна науково-практична конференція «Сучасні досягнення компанії Hewlett Packard Enterprise в галузі ІТ та нові можливості їх вивчення і застосування» (м. Київ, 2024);

II Міжнародна науково-практична конференція «Сучасні аспекти діджиталізації та інформатизації в програмній та комп'ютерній інженерії» (м. Київ, 2024).

Структура та обсяг дисертації. Дисертаційна робота складається з анотації, змісту, вступу, п'яти розділів, загальних висновків, списку використаних джерел та додатків. Робота містить 42 рисунки, 42 таблиці та 18 сторінок додатків. Список використаних джерел налічує 168 найменувань. Загальний обсяг дисертації становить 220 сторінок, з них 166 сторінок основного тексту.

РОЗДІЛ 1. АНАЛІЗ СУЧАСНОГО СТАНУ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ ДЛЯ КОРПОРАТИВНИХ СТРУКТУР

1.1. Корпоративні обчислювальні системи як об'єкт автоматизованого проєктування

Корпоративні обчислювальні системи є критичною складовою сучасних організацій: вони забезпечують безперервність бізнес-процесів, оброблення та зберігання даних, а також узгоджену взаємодію між територіально й функціонально розподіленими підрозділами. Прискорення цифрової трансформації підприємств перетворило такі системи з допоміжного технічного засобу на основу операційної діяльності, від якості проєктування якої безпосередньо залежать конкурентоспроможність та керованість організації [131]. Під корпоративною обчислювальною системою у цій роботі розуміють програмно-апаратний комплекс, що об'єднує обчислювальні вузли, сховища даних, канали зв'язку та системне і прикладне програмне забезпечення й призначений для обслуговування розподіленого потоку запитів користувачів в інтересах бізнес-процесів організації. Проєктування такої системи полягає у виборі її архітектури та числових параметрів, які найкраще відповідають вимогам конкретної корпоративної структури за наявних ресурсних, вартісних і часових обмежень.

Архітектурні підходи до побудови корпоративних обчислювальних систем зазнали суттєвої еволюції: від монолітних застосунків із єдиною базою коду до сервіс-орієнтованих, а згодом мікросервісних архітектур, у яких система складається з множини слабо зв'язаних, незалежно розгортуваних сервісів [105]. Подальший розвиток привів до безсерверної моделі виконання, за якої проєктувальник оперує функціями та подіями, а керування ресурсами делеговано платформі, і до хмаро-орієнтованих та крайових архітектур, що

розподіляють оброблення між центрами обробки даних і периферійними вузлами в єдиному континуумі «край – хмара» [97, 118, 152]. Розгортання й узгоджене керування такими системами дедалі частіше покладається на засоби контейнерної оркестрації, що автоматизують планування ресурсів між вузлами кластера [29]. Кожен архітектурний підхід має власний профіль масштабованості, складності супроводу та придатності для корпоративного застосування, які й визначають доцільність його вибору на етапі проектування. Порівняльну характеристику основних архітектурних підходів наведено в таблиці 1.1.

Таблиця 1.1

Еволюція архітектурних підходів до побудови корпоративних
обчислювальних систем

Архітектурний підхід	Спосіб розгортання	Масштабованість	Складність супроводу	Придатність для корпоративних систем
Монолітна	Єдиний застосунок на спільному сервері	Обмежена, вертикальна	Низька на старті, зростає з обсягом	Помірна, для невеликих систем
Сервіс-орієнтована	Набір сервісів зі спільною шиною	Середня, за сервісами	Середня, залежить від шини	Помірна
Мікросервісна	Незалежно розгортуються і контейнери	Висока, горизонтальна	Висока, потребує оркестрації	Висока
Безсерверна	Функції та події на платформі	Автоматична, за попитом	Перенесена на платформу	Висока для змінних навантажень
Хмаро-орієнтована та крайова	Континуум «край – хмара»	Дуже висока, розподілена	Висока, розподілене керування	Висока для географічно рознесених структур

Спільною рисою сучасних архітектур є розподіленість, яка має двоїсту природу. З одного боку, рознесення обчислень і даних між багатьма вузлами підвищує доступність та масштабованість системи, оскільки втрата окремого компонента не обов'язково спричиняє зупинку обслуговування, а самі механізми відмовостійкості реалізуються, зокрема, засобами мікросервісної архітектури та узгодження стану реплік [68, 161]. З іншого боку, розподіленість розширює поверхню атаки й ускладнює забезпечення захищеності, бо кожен вузол і кожен канал зв'язку стають потенційною ціллю, а ізоляція ресурсів між орендарями та мережевими зрізами сама потребує окремих механізмів захисту [38, 47]. Динамічний характер навантаження корпоративних систем спонукає до автоматичного масштабування ресурсів засобами машинного навчання, що прогнозує потребу в потужностях наперед, а розподіл оброблення до периферії висуває додаткові вимоги до енергоефективності та обмежених ресурсів крайових вузлів [81, 102, 118]. Отже, вже на етапі вибору архітектури проєктувальник має враховувати одночасно продуктивність, відмовостійкість і захищеність, які в розподіленому середовищі виявляються тісно пов'язаними.

Вибір архітектури та параметрів корпоративної обчислювальної системи визначається сукупністю різнорідних вхідних даних. Бізнес-параметри організації – очікувана кількість користувачів, вимоги до часу безвідмовної роботи, допустимий бюджет – задають зовнішні цілі та обмеження проєкту. Конфігураційні параметри – склад обчислювальних ресурсів, топологія зв'язків, кратність резервування – є керованими змінними, значення яких обирає проєктувальник. Параметри навантаження – інтенсивність потоку запитів, його структура та сезонні коливання – відображають умови експлуатації й безпосередньо не керуються. Атрибути якості – продуктивність, доступність, захищеність, вартість володіння – є цільовими показниками, за якими оцінюють прийнятність рішення. Основні групи параметрів, що описують корпоративну обчислювальну систему як об'єкт проєктування, узагальнено в таблиці 1.2.

Групи параметрів корпоративної обчислювальної системи як об'єкта
проектування

Група параметрів	Приклади	Роль у проектуванні
Бізнес-параметри	Кількість користувачів, вимоги до доступності, бюджет	Зовнішні цілі та обмеження проекту
Конфігураційні параметри	Обсяг ресурсів, топологія, кратність резервування	Керовані змінні вибору рішення
Параметри навантаження	Інтенсивність запитів, структура потоку, сезонність	Зовнішні умови експлуатації
Атрибути якості	Продуктивність, доступність, захищеність, вартість	Цільові показники прийнятності

Традиційний процес проектування корпоративної обчислювальної системи побудовано як послідовність відносно самостійних фаз: збір і аналіз бізнес-вимог, проектування архітектури, розробка та інтеграція компонентів, тестування навантаження й верифікація прийнятих рішень, розгортання та експлуатація. Кожна з цих фаз виконується переважно вручну й залежить від кваліфікації окремих спеціалістів, а перевірка того, чи задовольняє обрана архітектура вимоги за продуктивністю та захищеністю, відкладається на пізні стадії, коли систему вже значною мірою реалізовано.

Унаслідок такої організації помилки архітектурного рівня виявляються тоді, коли їх виправлення потребує дорогого повернення до раніше ухвалених рішень і повторного проходження частини циклу. Узагальнену схему традиційного життєвого циклу проектування з відкладеною верифікацією наведено на рисунку 1.1.

Обмеженість ручного й послідовного проектування спонукала до розвитку технологій автоматизації окремих етапів життєвого циклу. Автоматизоване конфігурування інфраструктури як коду дозволяє описувати цільовий стан обчислювального середовища декларативно й відтворювано розгортати його засобами штучного інтелекту [125].



Рис. 1.1. Традиційний життєвий цикл проектування корпоративної обчислювальної системи з відкладеною верифікацією

Практики експлуатації моделей машинного навчання оформилися в окремий інженерний напрям, що забезпечує відтворюване навчання, розгортання та супровід моделей у промисловому середовищі [89]. Інтеграція засобів безпеки безпосередньо у конвеєр розробки й розгортання перетворює захищеність із фінальної перевірки на неперервну властивість життєвого циклу [111], а застосування машинного інтелекту до задач експлуатації дозволяє автоматично виявляти й усувати відхилення у роботі вже розгорнутих систем [160]. Спільним для цих напрямів є те, що кожен з них автоматизує лише окремий фрагмент життєвого циклу – розгортання, супровід моделей, вбудовування безпеки чи експлуатацію, – тоді як цілісного автоматизованого переходу від бізнес-вимог до верифікованої, відмовостійкої та захищеної обчислювальної інфраструктури вони не утворюють.

Захищеність також не можна відкладати на фазу експлуатації: ізоляцію ресурсів у мережесхемних зрізах, захист крайових і промислових вузлів та завчасний перехід на постквантові криптографічні засоби слід враховувати вже під час вибору архітектури, оскільки їх закладення на пізніх стадіях значно дорожче й менш дієве [13, 54, 108, 134].

Виходячи з наведеного, автоматизоване проєктування корпоративної обчислювальної системи як задача передбачає розв'язання кількох взаємопов'язаних підзадач:

1. Формалізацію відображення бізнес-параметрів і умов експлуатації у конфігураційні параметри та архітектуру системи, придатну до автоматизованого вибору.

2. Прогнозування продуктивності обраної конфігурації до її натурального розгортання за мінімальних витрат на вимірювання.

3. Верифікацію прийнятих рішень за формальними критеріями якості й підтримання функціональної стійкості системи в умовах змінних навантажень.

4. Проєктування відмовостійкої та захищеної структури, що зберігає критичну функціональність за втрати компонентів і за дії загроз безпеці функціонування.

Розв'язання цих підзадач ускладнюється низкою об'єктивних обмежень: різноманітністю вхідних параметрів, що поєднують числові, категоріальні та бінарні величини; високою вартістю натурних навантажувальних вимірювань, кожне з яких потребує значного машинного часу; неповнотою знань про поверхню відгуку системи, що зумовлює невизначеність прогнозів; обмеженістю ресурсів, які доводиться розподіляти між продуктивністю, відмовостійкістю та захищеністю. Саме ці обмеження визначають вимоги до методів і технологій, здатних автоматизувати проєктування корпоративних обчислювальних систем.

Отже, корпоративна обчислювальна система є складним розподіленим об'єктом проєктування, вибір архітектури та параметрів якого визначається різноманітними бізнес-параметрами, умовами навантаження й атрибутами якості за обмежених ресурсів. Традиційне ручне проєктування з відкладеною

верифікацією не забезпечує ані завчасної перевірки продуктивності, ані закладення відмовостійкості й захищеності на етапі вибору архітектури, а наявні засоби автоматизації охоплюють лише окремі фрагменти життєвого циклу, тож постає потреба у формальних методах прогнозування продуктивності, забезпечення функціональної стійкості та проєктування відмовостійких і захищених комп'ютерних систем.

1.2. Сурогатне моделювання та прогнозування продуктивності конфігурованих систем

Продуктивність корпоративної обчислювальної системи під навантаженням є одним із визначальних атрибутів якості, за яким оцінюють прийнятність архітектурного рішення. Її залежність від конфігураційних параметрів середовища розгортання – обсягу пам'яті, кількості обчислювальних ядер, розміру пулів з'єднань і кількості паралельних робочих процесів – має виражено нелінійний характер, а комбінації параметрів породжують ефекти насичення ресурсів і черговості запитів, які неможливо описати незалежним розглядом окремих чинників. Тому прогнозування продуктивності конфігурованих систем оформилося в самостійний напрям досліджень, у якому послідовно розвивалися три групи підходів: аналітичні моделі впливу параметрів, методи машинного навчання та сурогатне моделювання з квантифікацією невизначеності. Ранні роботи цього напрямку зосереджувалися саме на прогнозуванні показників продуктивності вже наявних систем засобами статистичного машинного навчання, а не на виборі архітектури як такої [53].

Історично першим формалізованим інструментом стали моделі впливу продуктивності (performance-influence models), у яких цільова метрика подається як сума доданків, що відповідають окремим параметрам конфігурації та обмеженому набору їхніх попарних взаємодій [142]. Такий адитивний опис є прозорим та інтерпретованим, проте за своєю структурою не відтворює взаємодій вищого порядку й насичувальних ефектів, характерних для реальних

обчислювальних систем, а також не супроводжується оцінкою невизначеності прогнозу. Спроби подолати ці обмеження привели до розширення адитивних моделей довірчими інтервалами [41, 42] та до застосування глибоких нейромереж, що враховують ієрархічні взаємодії параметрів [34, 61]. Систематичні огляди навчання продуктивності конфігурованих систем підтверджують стійку тенденцію переходу від адитивних до нелінійних моделей, здатних апроксимувати складну поверхню відгуку [59, 60]. Водночас емпіричні дослідження показують високу чутливість продуктивності до вхідних даних і структури простору конфігурацій, що ускладнює побудову точних моделей за обмеженої вибірки [71, 94]. Порівняння основних класів методів прогнозування продуктивності конфігурованих систем наведено в таблиці 1.3.

Таблиця 1.3

Порівняння класів методів прогнозування продуктивності
конфігурованих систем

Клас методу	Нелінійні взаємодії	Оцінка невизначеності	Вартість даних	Характерні представники
Адитивні моделі впливу	Обмежено, попарно	Здебільшого відсутня	Помірна	Моделі впливу параметрів
Глибокі нейромережі	Повно, ієрархічно	Обмежена	Висока	Ієрархічні мережі продуктивності
Сурогатні моделі гаусівських процесів	Повно, через ядро	Відкалібрована	Низька за активного навчання	Гаусівська процесова регресія
Сурогатно-керована оптимізація	Повно	Через модель невизначеності	Низька	Баєсова оптимізація

Ключем до скорочення вартості прогнозування є сурогатне (замінне) моделювання, за якого дорогі натурні вимірювання замінюються статистичною моделлю, що апроксимує поведінку системи в усьому просторі конфігурацій за відносно невеликою навчальною вибіркою [66, 98]. Сучасні огляди сурогатного моделювання наголошують на двох його передумовах: раціональному плануванні комп'ютерного експерименту та коректному оцінюванні якості апроксимації [103]. Для рівномірного заповнення багатовимірного простору параметрів широко застосовують схему латинського гіперкубічного вибору, яка,

на відміну від повної сітки, гарантує представленість кожного рівня кожного параметра за істотно меншої кількості точок [106]. Подальший розвиток планування вибірки спрямований на адаптивність та оптимізацію заповнення простору, зокрема засобами еволюційних алгоритмів [25, 70]. Багаторівневе (multi-fidelity) сурогатне моделювання додатково знижує витрати, поєднуючи нечисленні точні вимірювання з численними наближеними [36].

Серед апаратів сурогатного моделювання особливе місце посідає гаусівська процесова регресія, що бере початок від методів просторової інтерполяції (крігінгу) і сформувалася в самостійний імовірнісний підхід машинного навчання [145, 129]. Її перевагою є те, що модель повертає не лише точковий прогноз, а й повний апостеріорний розподіл, з якого безпосередньо впливає оцінка невизначеності у кожній точці простору. Гнучкість гаусівських процесів визначається вибором коваріаційної функції (ядра), теорію та класи яких досліджено у низці робіт [56, 123]. Сучасні огляди підтверджують зрілість цього апарату та його придатність для широкого кола інженерних задач [100], а окремим перспективним напрямом є фізично-керовані гаусівські процеси, у яких структуру ядра узгоджують із відомими закономірностями предметної області [37]. Для задач із багатовимірним простором параметрів розвинуто спеціалізовані методи високовимірної гаусівської моделювання та ефективні обчислювальні реалізації [22, 28].

Принциповою для автоматизованого проєктування властивістю сурогатної моделі є не лише точність точкового прогнозу, а й відкаліброваність оцінки невизначеності, тобто статистична узгодженість довірчих інтервалів із фактичною часткою значень, що в них потрапляють. Огляди оцінювання невизначеності в машинному навчанні наголошують, що некалібрована модель здатна давати систематично завужені або завищені інтервали, знецінюючи їх практичне застосування [55, 143]. Для регресійних задач розроблено спеціальні методи калібрування, що приводять передбачені інтервали у відповідність до емпіричного покриття, а якість калібрування оцінюють діаграмами надійності та інтегральними метриками похибки [90]. Відкаліброваність невизначеності є

передумовою двох застосувань, важливих для проектування: цілеспрямованого поповнення навчальної вибірки та накладання проєктних обмежень не на точковий прогноз, а на його верхню довірчу межу.

Наявність достовірної оцінки невизначеності уможлиблює активне навчання – цілеспрямований вибір наступних точок вимірювання там, де прогноз найменш надійний, замість суцільного покриття простору [95, 150]. Тісно пов'язаний з ним апарат баєсової оптимізації використовує сурогатну модель гаусівського процесу та функцію придбання, що зважає очікуване поліпшення проти невизначеності, для ефективного пошуку в дорогих для обчислення просторах [138, 154]. Сучасні дослідження розвивають функції придбання зі змінною точністю та доступні програмні реалізації баєсової оптимізації, а перенесення знань між середовищами дозволяє повторно використовувати раніше побудовані моделі продуктивності [76, 116, 130, 136]. Узагальнений цикл сурогатного моделювання продуктивності з активним поповненням вибірки наведено на рисунку 1.2.

Поряд із точністю та відкаліброваністю практичне застосування сурогатних моделей у корпоративному середовищі потребує їх інтерпретованості: проєктувальник має розуміти, які саме параметри конфігурації визначають прогнозований рівень продуктивності. Для цього застосовують методи пояснюваного штучного інтелекту, серед яких найпоширенішим є оцінювання внеску ознак на основі значень Шеплі, що дає узгоджену міру важливості кожного параметра [99].

Сучасні огляди пояснюваного штучного інтелекту наголошують, що інтерпретованість підвищує довіру до автоматизованих рекомендацій і є необхідною умовою їх практичного впровадження [12]. Застосування таких методів до сурогатної моделі продуктивності дозволяє не лише прогнозувати відгук системи, а й обґрунтовувати, які ресурсні обмеження є визначальними за конкретного профілю навантаження.



Рис. 1.2. Узагальнений цикл сурогатного моделювання продуктивності конфігурованої системи з активним поповненням вибірки

Отже, сурогатне моделювання на основі гаусівської процесової регресії є найпридатнішим апаратом для прогнозування продуктивності конфігурованих обчислювальних систем, оскільки поєднує нелінійну апроксимацію поверхні відгуку з імовірнісною оцінкою невизначеності. Адитивні моделі впливу, попри інтерпретованість, не відтворюють взаємодій параметрів і не надають довірчих інтервалів [142]. Нелінійні моделі рідко перевіряють на відкаліброваність, а узгоджене застосування гаусівської процесової регресії до автоматизованого конфігурування, коли проєктне обмеження накладається на верхню довірчу межу прогнозу, а обсяг вимірювань скорочується цілеспрямованим поповненням

вибірки, у розглянутій літературі не формалізоване. Тим самим окреслюється потреба у методі, що поєднує точну апроксимацію відгуку, відкалібровану невизначеність та активне скорочення обсягу натурних вимірювань.

1.3. Функціональна стійкість адаптивних корпоративних інформаційних систем

Корпоративна інформаційна система функціонує у середовищі, що безперервно змінюється: коливається інтенсивність звернень користувачів, змінюються типові сценарії їхньої поведінки, зростають вимоги до захищеності даних, а обчислювальні ресурси зазнають часткових відмов і цілеспрямованих кібернетичних впливів. За таких умов класичне поняття надійності, зосереджене на безвідмовності окремих компонентів, виявляється недостатнім, бо не описує здатності системи зберігати корисне функціонування саме тоді, коли частина її ресурсів уже уражена або перевантажена. Тому предметом окремого напрямку досліджень стала ширша системна властивість – функціональна стійкість, тобто здатність системи зберігати працездатність і встановлений рівень якості обслуговування в умовах збурень різної природи за рахунок внутрішньої перебудови ресурсів і режимів роботи [17, 93]. Формування цього напрямку відбило загальний перехід інженерної думки від надійності до стійкості складних систем як інтегральної здатності протистояти збуренням і відновлюватися після них [104].

Функціональну стійкість доцільно розглядати у співвідношенні із суміжними властивостями обчислювальних систем, які нерідко ототожнюють. Надійність характеризує ймовірність безвідмовної роботи компонентів упродовж заданого часу й не передбачає механізмів пристосування до збурень. Відмовостійкість забезпечує збереження працездатності за рахунок заздалегідь закладеного резервування, проте діє пасивно, у межах наперед визначених сценаріїв відмов. Живучість описує здатність зберігати критично важливі функції в умовах масованого ураження, коли втрачено значну частку ресурсів

[69]. Функціональна стійкість посідає проміжне місце: вона свідомо допускає деградацію окремих компонентів, але, на відміну від пасивної відмовостійкості, передбачає активне, кероване в режимі реального часу пристосування системи до збурення шляхом перерозподілу ресурсів, зміни маршрутів інформаційних потоків і коригування політик безпеки. Розмежування цих властивостей наведено в таблиці 1.4. Вітчизняна наукова школа розробила ґрунтовний апарат дослідження функціональної стійкості складних інформаційно-технічних систем, зокрема алгоритми контролю й прогнозування стану, діагностичний моніторинг параметрів працездатності та системний аналіз стійкості об'єктів критичної інфраструктури [2, 3, 4, 18, 19].

Таблиця 1.4

Розмежування функціональної стійкості та суміжних властивостей
обчислювальних систем

Властивість	Характер реагування	Основний механізм	Умови застосування
Надійність	Відсутнє	Безвідмовність компонентів	Штатний режим роботи
Відмовостійкість	Пасивне	Заздалегідь закладене резервування	Наперед визначені сценарії відмов
Живучість	Часткове збереження функцій	Захист критичних сервісів	Масоване ураження ресурсів
Функціональна стійкість	Активне, у реальному часі	Перерозподіл ресурсів і режимів	Збурення різної природи

Кількісне оцінювання функціональної стійкості спирається на міри, розвинуті в межах теорії стійкості складних систем. Класичним є подання втраченої функціональності у вигляді трикутника стійкості, площа якого відображає сукупний вплив збурення від моменту його виникнення до повного відновлення [27]. Узагальнення цього підходу описує стійкість як функцію часу, що враховує глибину провалу функціональності та тривалість відновлення [67]. Сучасні оглядові дослідження систематизують численні метрики стійкості мереж і соціотехнічних систем, наголошуючи на потребі узгодженого набору показників для порівняння різних конфігурацій [128, 146]. Ці міри створюють

формальне підґрунтя для постановки задачі забезпечення функціональної стійкості як оптимізаційної, у якій стійкість максимізують за обмежень на витрати.

Узагальнену структуру замкненого контуру адаптації, що лежить в основі забезпечення функціональної стійкості, наведено на рисунку 1.3.

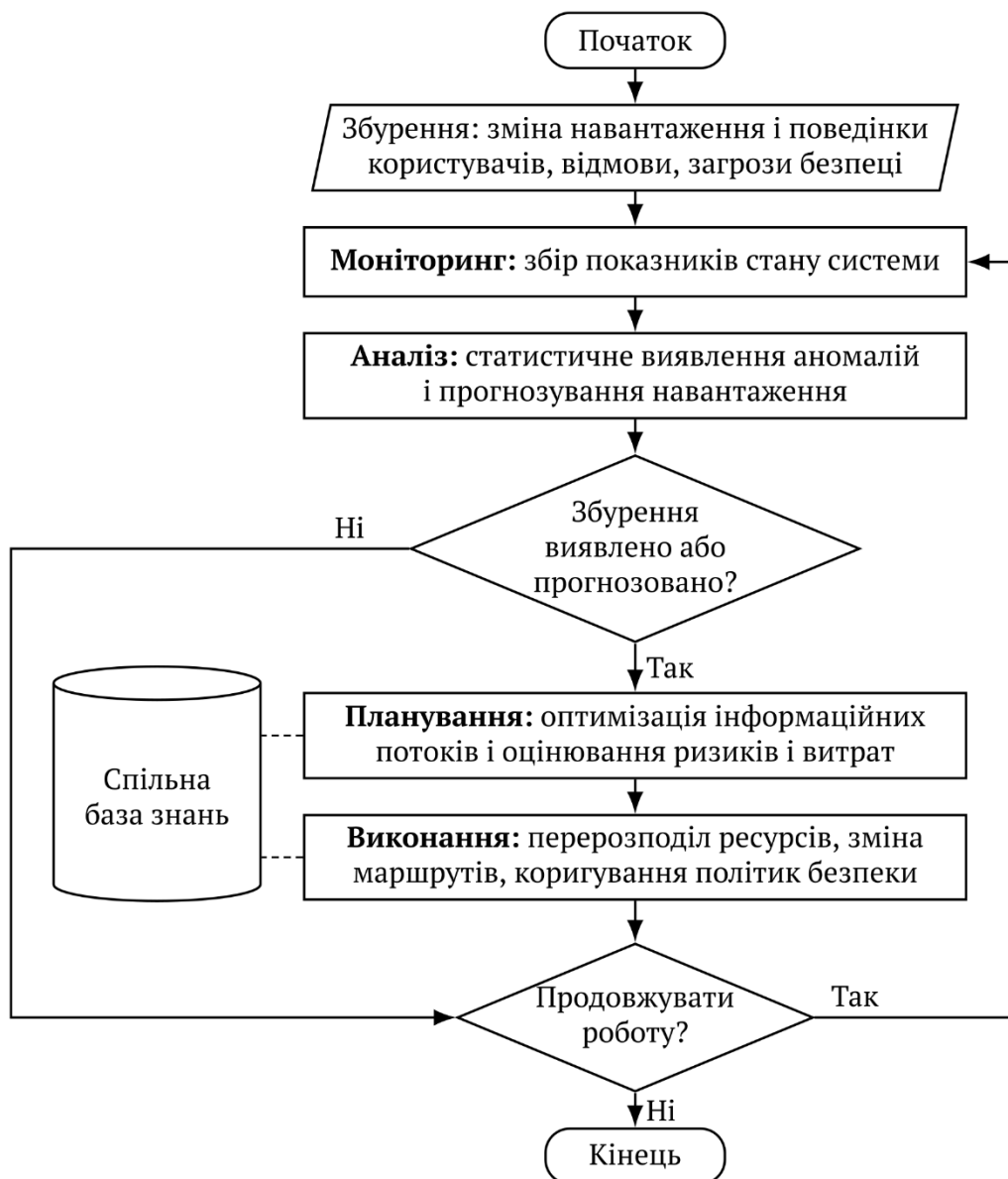


Рис. 1.3. Замкнений контур адаптації для забезпечення функціональної стійкості корпоративної інформаційної системи

Активний характер функціональної стійкості реалізується через контур адаптації, ідея якого сягає концепції автономних обчислень із циклом «моніторинг – аналіз – планування – виконання» та спільною базою знань [82].

Дослідження самоадаптивних систем розвинули цю концепцію в самостійний напрям програмної інженерії, що охоплює архітектури, механізми ухвалення рішень і засоби перевірки адаптивної поведінки [32, 156], а прикладні реалізації довели дієвість нейроадаптивного керування для забезпечення функціональної стійкості виробничих процесів [7].

Першою складовою контуру адаптації є моніторинг стану системи й виявлення аномалій. Класичний апарат статистичного керування процесами – контрольні карти, межа трьох стандартних відхилень і правила серій – дозволяє вчасно фіксувати відхилення показників від штатного режиму [113, 114]. Систематичні огляди методів виявлення аномалій розмежовують статистичні, відстаневі та навчальні підходи й описують їхню придатність для різних типів даних [30, 9]. Розвиток методів машинного навчання поширив виявлення аномалій на потоки телеметрії промислових та мережевих систем, а окремою важливою проблемою стало врахування дрейфу концепту, тобто поступової зміни нормального режиму, що знецінює статичні моделі й потребує їх адаптації [16, 74, 141, 159].

Другою складовою є прогнозування поведінки користувачів і навантаження, яке уможливорює проактивне резервування ресурсів. Для цього застосовують як класичний апарат аналізу часових рядів, так і методи машинного навчання, зокрема градієнтний бустинг та глибокі нейромережі для прогнозування часових рядів [26, 51, 73, 85]. Огляди застосування машинного навчання до прогнозування показників навантаження підтверджують його зрілість і придатність для короткострокових горизонтів [112]. Наявність прогнозу з оцінкою невизначеності дозволяє системі готуватися до очікуваних сплесків попиту заздалегідь, а не реагувати на них постфактум.

Третьою складовою є оптимізація інформаційних потоків і розподілу ресурсів, яка забезпечує обслуговування попиту за часткових відмов. Формальним апаратом тут слугує теорія потоків у мережах: теорема про максимальний потік та алгоритми його обчислення дозволяють перемаршрутизувати навантаження в межах сумарної пропускної здатності

справних вузлів, а задачі розподілу ресурсів природно формалізуються засобами лінійного програмування [49, 10, 21]. Четвертою складовою є оцінювання ризиків, витрат і рівня інформаційної безпеки, що надає адаптації економічно раціонального характеру. Кількісні підходи до оцінювання інформаційного ризику, економіка інвестицій у безпеку з визначенням оптимального бюджету, модель віддачі від витрат на безпеку та чинні стандарти управління ризиками дозволяють обґрунтувати раціональний, а не максимальний рівень захисту; вітчизняні дослідження розвивають інтелектуальні методи оцінювання ризиків інформаційної безпеки корпоративних систем [5, 50, 62, 75, 144]. Складові забезпечення функціональної стійкості та відповідні їм методи узагальнено в таблиці 1.5.

Таблиця 1.5

Складові забезпечення функціональної стійкості та відповідні методи

Складова	Задача складової	Методи та апарат
Моніторинг і виявлення аномалій	Вчасне виявлення відхилень від штатного режиму	Контрольні карти, статистичне та навчальне виявлення аномалій
Прогнозування навантаження	Проактивне резервування за очікуваного попиту	Аналіз часових рядів, градієнтний бустинг, нейромережі
Оптимізація інформаційних потоків	Обслуговування попиту за часткових відмов	Максимальний потік, лінійне програмування
Оцінювання ризиків і витрат	Раціональний рівень захисту за обмеженого бюджету	Кількісний ризик, економіка безпеки, стандарти

Отже, кожен зі складових забезпечення функціональної стійкості достатньо розвинено окремо, проте вони застосовуються переважно ізольовано й у фазі експлуатації: моніторинг фіксує збурення, але не пов'язаний із проактивним прогнозуванням, оптимізація потоків усуває наслідки відмов, але не узгоджена з економічним оцінюванням ризику, а більшість підходів охоплює лише продуктивнісний аспект стійкості без безпекового. Тим самим окреслюється потреба в інформаційній технології, яка інтегрує статистичний

моніторинг аномалій, прогнозування навантаження, оптимізацію інформаційних потоків та оцінювання ризиків у єдиний замкнений контур адаптації в режимі реального часу й узгоджено забезпечує продуктивнісну та безпекову складові функціональної стійкості за раціональних витрат.

1.4. Відмовостійкість і безпека функціонування розподілених корпоративних обчислювальних систем

Корпоративні структури дедалі частіше будують свої обчислювальні системи як розподілені: територіально рознесені вузли, багаторівневе оброблення даних і резервні канали зв'язку підвищують доступність сервісів та масштабованість інфраструктури [148]. Розподіленість має двоїсту природу. З одного боку, вона створює передумови для відмовостійкості, оскільки втрата окремого вузла не обов'язково зупиняє систему; з іншого боку, вона розширює поверхню атаки й ускладнює забезпечення захищеності, бо кожен вузол і кожен канал стають потенційною ціллю [88]. Історично надійність та відмовостійкість, з одного боку, і кібербезпека, з іншого, розвивалися як відносно самостійні інженерні напрями з власними моделями й показниками. Для розподілених обчислювальних систем корпоративних структур такий поділ виявляється штучним, бо і фізична втрата компонентів, і кіберзагрози ведуть до однакового наслідку – зниження здатності системи виконувати критичні функції, що обґрунтовує їх спільний розгляд.

Апарат відмовостійкості розподілених систем спирається на класичні засади надійності та резервування. Теорія відмовостійких обчислювальних систем описує механізми виявлення й маскування відмов, а моделі надійності й готовності – ймовірнісну поведінку систем зі структурним резервуванням, зокрема за схемою «k-з-n», за якої клас зберігає працездатність, доки справні щонайменше k із n його екземплярів [88, 91, 151]. Сучасні дослідження розвивають аналіз надійності й оптимального резервування таких систем засобами марковських ланцюгів і моделювання, а для узгодження стану реплік у

розподіленому середовищі застосовують візантійсько-стійкі алгоритми консенсусу [63, 124, 161]. Прикладні роботи демонструють підвищення відмовостійкості мікросервісних архітектур та оцінювання залишкового ресурсу вузлів розподілених систем критичної інфраструктури [68, 158].

Ключовим для проєктування є поняття живучості – здатності системи зберігати надання критичних послуг за наявності відмов та атак. Класичні означення живучості мережесистем доповнено сучасним апаратом кількісного оцінювання стійкості мереж та плавної деградації, за якої замість бінарного переходу «працює – не працює» система поступово знижує функціональність зі збереженням критичних сервісів [46, 84, 128, 139]. Оглядів дослідження систематизують метрики стійкості складних систем і розмежовують поняття стійкості та живучості [146]. Плавна деградація дозволяє кількісно оцінювати збереження критичної функціональності за наростання втрат ще на етапі проєктування, що робить її придатним проєктним критерієм.

Оцінювання захищеності розподіленої системи потребує формального опису загроз. Для цього застосовують дерева й графи атак, що моделюють можливі шляхи компрометації, а також засоби їх автоматичної побудови, та системи кількісного оцінювання вразливостей за стандартизованими шкалами [48, 86, 87, 140]. Оцінювання ймовірності успішної атаки й ризику доповнюють нечітко-логічними моделями стану системи під час атак типу «відмова в обслуговуванні» та настановами з проведення оцінювання ризиків інформаційної безпеки [8, 92, 122]. Такий апарат дозволяє пов'язати рівень захищеності з рівнем загрози й урахувати періоди її загострення.

Забезпечення захищеності розподіленої системи будується як багаторівневий (ешелонований) захист, у якому кожен рівень протидіє певному класу загроз. Зовнішній рівень утворюють периметровий захист і сегментація мережі; наступний – архітектура нульової довіри, що мінімізує неявну довіру між компонентами й перевіряє кожен доступ незалежно від його джерела [117, 133]. Окремий рівень становить постквантова криптографія, покликана забезпечити довгострокову стійкість до майбутніх квантових обчислень, які

загрожують наявним криптографічним механізмам; її розвиток закріплено сучасними оглядами переходу та стандартами постквантових механізмів інкапсуляції ключів і цифрового підпису [20, 35, 108, 119–121]. Внутрішній рівень формує адаптивне виявлення загроз засобами машинного навчання, здатне розпізнавати атаки в режимі реального часу та пристосовуватися до їх еволюції [15, 135]. Узагальнену структуру багаторівневого захисту на відмовостійкій основі наведено на рисунку 1.4.

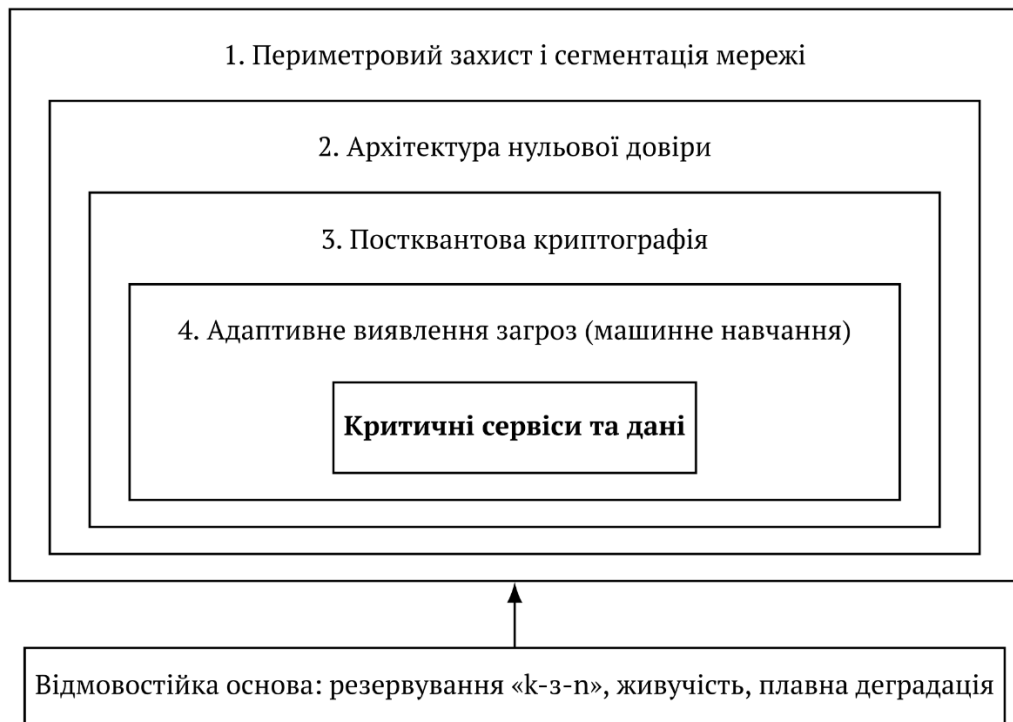


Рис. 1.4. Багаторівневий (ешелонований) захист розподіленої обчислювальної системи на відмовостійкій основі

Крім базових рівнів, сучасні дослідження розвивають активні та розподілені механізми захисту: рухомий захист, що динамічно змінює конфігурацію системи задля ускладнення атаки, техніки кіберобману та пастки, захист ланцюга постачання програмного забезпечення, а також захищене федеративне навчання у крайових середовищах [77, 96, 107, 147]. Окрему увагу приділяють стійкості самих моделей машинного навчання до змагальних атак та обміну даними кіберрозвідки загроз; вітчизняна наукова школа розробила комплексні підходи до побудови систем кібербезпеки, кореляції подій та керування інцидентами на об'єктах критичної інфраструктури [11, 57, 58, 155,

157]. Основні механізми забезпечення захищеності розподілених систем узагальнено в таблиці 1.6.

Таблиця 1.6

Механізми забезпечення захищеності розподілених обчислювальних систем

Механізм захисту	Призначення
Архітектура нульової довіри	Мінімізація неявної довіри, перевірка кожного доступу
Постквантова криптографія	Довгострокова стійкість до квантових обчислень
Адаптивне виявлення загроз	Розпізнавання атак у реальному часі й пристосування
Рухомий захист і кіберобман	Ускладнення атаки динамічною зміною конфігурації
Захист ланцюга постачання й моделей	Стійкість до компрометації залежностей і змагальних атак

Узгоджене впровадження перелічених механізмів потребує обґрунтованого розподілу обмежених ресурсів безпеки між рівнями та компонентами системи, оскільки надмірний захист одних компонентів за незахищеності інших не підвищує загальної живучості. Задачу такого розподілу доцільно формалізувати як багатокритеріальну, у якій одночасно враховують живучість, захищеність, ризик і плавність деградації, а вибір архітектури постає як оптимізаційне, а не суто експертне рішення [45, 109]. Ефективними інструментами пошуку компромісних розв'язків тут є методи багатокритеріальної оптимізації, зокрема еволюційні алгоритми побудови множини Парето [101]. Перевірку отриманих проектних рішень за сценаріями втрат і загроз доцільно виконувати на цифровому двійнику системи, що відтворює її поведінку без ризику для реальної інфраструктури [44, 52, 79]. Функції відмовостійкості й живучості, що входять у такий критерій проектування, узагальнено в таблиці 1.7.

Таблиця 1.7

Функції відмовостійкості й живучості як критерії проектного вибору

Функція або показник	Зміст
Живучість	Імовірність збереження критичної функціональності за втрат і атак
Резервування k-з-n	Збереження класу компонентів за втрати частини екземплярів
Плавна деградація	Поступове зниження функціональності зі збереженням критичних сервісів
Інтегральний ризик	Агрегована оцінка ймовірних втрат з урахуванням рівня загрози

Отже, підходи до забезпечення відмовостійкості та захищеності розподілених систем розвиваються переважно відокремлено й у фазі експлуатації: механізми резервування узгоджують за показниками надійності, засоби захисту додають до вже розгорнутої архітектури, а спільний показник живучості, що поєднував би обидва аспекти, як критерій проєктного рішення не застосовують. Тим самим окреслюється потреба у методі та інформаційній технології структурно-параметричного проєктування, що на етапі проєктування обирають відмовостійку й захищену архітектуру на основі багатокритеріальної оптимізації розподілу ресурсів безпеки з урахуванням функцій живучості, адаптивності та плавної деградації.

1.5. Аналіз прогалин і постановка задач дослідження

Проведений у підрозділах 1.2–1.4 аналіз сучасного стану досліджень охопив три взаємопов'язані напрями автоматизованого проєктування корпоративних обчислювальних систем: прогнозування продуктивності конфігурованих систем, забезпечення функціональної стійкості адаптивних систем та відмовостійкості і безпеку функціонування розподілених систем. У кожному з напрямів досягнуто помітного прогресу, проте виявлено характерну прогалину, що обмежує застосовність наявних підходів до задачі цілісного проєктування. Зведений аналіз виявлених прогалин наведено в таблиці 1.8.

Таблиця 1.8

Зведений аналіз прогалин за трьома напрямми дослідження

Напря́м	Сучасний стан	Виявлена прогалина
Прогнозування продуктивності	Адитивні моделі впливу та нелінійні моделі машинного навчання	Адитивні моделі не відтворюють нелінійних взаємодій і не надають відкаліброваної невизначеності; нелінійні рідко калібрують і не поєднують з активним скороченням вимірювань
Функціональна стійкість	Розвинені моніторинг, прогнозування, оптимізація потоків, оцінювання ризиків	Складові застосовуються ізольовано й у фазі експлуатації, не інтегровані в єдиний контур адаптації, що охоплював би продуктивнісний і безпековий аспекти
Безпека та відмовостійкість	Багаторівневий захист, постквантова криптографія, нульова довіра, адаптивне виявлення загроз	Захищеність забезпечують постфактум і окремо від відмовостійкості; спільний показник живучості не застосовують як критерій проєктного вибору

Порівняння виявлених прогалин показує, що вони мають спільну природу. Загальним недоліком наявних досліджень є відокремлене розв'язання окремих задач: точні моделі продуктивності не поєднуються з оцінкою невизначеності та активним навчанням, складові функціональної стійкості діють ізольовано й лише у фазі експлуатації, а засоби захисту додають до вже спроектованої системи окремо від механізмів відмовостійкості. Причиною такого стану є те, що прогнозування продуктивності, теорія стійкості та забезпечення надійності й безпеки сформувалися як самостійні напрями з власними моделями, показниками й етапами застосування, тоді як узгоджувальна ланка між ними на рівні проєктного рішення не розроблялася. Звідси виникає ситуація, за якої цілісного автоматизованого переходу від бізнес-вимог до верифікованої, відмовостійкої та захищеної обчислювальної інфраструктури наявні дослідження не утворюють, а ефект узгодженого поєднання підходів залишається нереалізованим. Сукупна прогалина зумовлює потребу в розробленні узгодженого апарату автоматизованого проєктування корпоративних обчислювальних систем, що поєднає прогнозовану

продуктивність, функціональну стійкість, відмовостійкість і безпеку функціонування в межах єдиного підходу.

Такий апарат доцільно будувати не як удосконалення окремого напрямку, а як триєдиний підхід, у якому три взаємодоповнювані складові узгоджено покривають життєвий цикл корпоративної обчислювальної системи. Перша складова діє на етапі проектування за критерієм продуктивності: сурогатна модель дозволяє передбачити відгук системи й обрати конфігурацію ресурсів за мінімуму натурних вимірювань. Друга складова діє на етапі експлуатації: інтегрована технологія забезпечує функціональну стійкість під змінним навантаженням і збуреннями, поєднуючи моніторинг, прогнозування, оптимізацію потоків та ризик-економічне керування. Третя складова діє на етапі структурного проектування й спирається на дві попередні: метод та інформаційна технологія, використовуючи прогноз продуктивності та оцінку функціональної стійкості, обирають відмовостійку й захищену архітектуру, стійку до втрати компонентів і до загроз безпеці функціонування. Взаємозв'язок трьох складових у межах життєвого циклу системи наведено на рисунку 1.5.

Запропонований триєдиний підхід має і теоретичне, і практичне значення. З теоретичного погляду він об'єднує в межах єдиного методологічного апарату задачі, що раніше розглядалися відокремлено, і встановлює між ними змістовні зв'язки: відкалібрована невизначеність сурогатної моделі уможливорює активне скорочення вимірювань, статистичний моніторинг у складі технології стійкості виконує функцію виявлення аномального режиму, а спільний показник живучості пов'язує відмовостійкість із безпекою функціонування.

З практичного погляду підхід дозволяє обґрунтовано, ще до розгортання, обирати конфігурацію та архітектуру корпоративної обчислювальної системи з наперед заданими показниками продуктивності, стійкості й захищеності в межах доступних ресурсів, скорочуючи обсяг дорогих натурних випробувань.



Рис. 1.5. Інтегрований підхід до автоматизованого проєктування корпоративної обчислювальної системи впродовж її життєвого циклу

Висновки до розділу 1

У розділі проаналізовано сучасний стан автоматизованого проєктування обчислювальних систем для корпоративних структур. Корпоративну обчислювальну систему розглянуто як складний розподілений об'єкт проєктування, для якого традиційне ручне проєктування з відкладеною верифікацією не забезпечує ані завчасної перевірки продуктивності, ані

закладення відмовостійкості й захищеності на етапі вибору архітектури. У кожному з трьох напрямів виявлено характерну прогалину: адитивні моделі продуктивності не відтворюють нелінійних взаємодій і не надають відкаліброваної невизначеності, складові функціональної стійкості застосовуються ізольовано й у фазі експлуатації, а захищеність забезпечують постфактум і окремо від відмовостійкості.

Спільною причиною прогалин є відокремлений розвиток прогнозування продуктивності, теорії стійкості та забезпечення надійності й безпеки, тож обґрунтовано потребу в узгодженому апараті автоматизованого проєктування корпоративних обчислювальних систем, що поєднує прогнозовану продуктивність, функціональну стійкість, відмовостійкість і безпеку функціонування, і запропоновано триєдиний підхід, який покриває життєвий цикл системи від проєктування за продуктивністю до проєктування відмовостійкої та захищеної архітектури.

РОЗДІЛ 2. МЕТОД АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ НА ОСНОВІ СУРОГАТНОЇ МОДЕЛІ ГАУСІВСЬКОЇ ПРОЦЕСОВОЇ РЕГРЕСІЇ

2.1. Постановка задачі сурогатного моделювання продуктивності обчислювальної системи

Продуктивність корпоративної обчислювальної системи під навантаженням визначається сукупністю конфігураційних параметрів середовища розгортання: обсягом оперативної пам'яті, кількістю ядер процесора, розміром пулу з'єднань до бази даних та кількістю паралельних робочих процесів. Кожен із цих параметрів нелінійно впливає на цільові показники якості обслуговування, а їхні комбінації породжують ефекти насичення ресурсів і черговості запитів, які неможливо описати незалежним розглядом окремих параметрів [53, 142]. Пряме визначення оптимальної конфігурації повним перебором простору параметрів є практично нездійсненним: навіть для шести параметрів з кількома дискретними рівнями кожного повна сітка налічує сотні унікальних конфігурацій, а кожне натурне навантажувальне вимірювання триває десятки хвилин, унаслідок чого сумарна вартість повного перебору сягає сотень годин машинного часу [162].

Ефективним способом подолання цієї обмеженості є сурогатне (замінне) моделювання: за відносно невеликою кількістю натурних вимірювань будується статистична модель, що апроксимує поведінку системи в усьому просторі конфігурацій і навантажень та замінює собою дорогі експерименти [1, 53]. Позначимо через x вектор конфігураційних параметрів системи, а через l – вектор характеристик навантаження. Задача сурогатного моделювання полягає у побудові відображення, яке для кожної пари «конфігурація – навантаження» повертає не лише точкову оцінку цільової метрики продуктивності, а й кількісну оцінку невизначеності цієї оцінки:

$$f: (x, l) \mapsto (\mu, \sigma), \mu = E[y | x, l], \quad \sigma^2 = \text{Var}[y | x, l], \quad (2.1)$$

де y – спостережуване значення цільової метрики продуктивності, μ – прогнозоване (середнє) значення, σ – середньоквадратичне відхилення прогнозу, що характеризує впевненість моделі. Як цільову метрику обрано 95-й перцентиль затримки відповіді (p95-затримку), оскільки саме перцентильні показники, на відміну від середнього значення, адекватно відображають досвід користувача у хвостовій частині розподілу затримок і є типовим предметом угод про рівень обслуговування.

Принципова відмінність задачі (2.1) від класичної регресії полягає у вимозі до відкаліброваності оцінки невизначеності. Модель повинна не лише мінімізувати похибку точкового прогнозу, а й забезпечувати статистичну узгодженість довірчих інтервалів: частка реальних значень, що потрапляють у передбачений інтервал довіри рівня α , має дорівнювати α . Ця властивість є критичною для цілеспрямованого поповнення навчальної вибірки та для накладання проєктних обмежень на верхню довірчу межу прогнозу. Наявні підходи до прогнозування продуктивності конфігурованих систем здебільшого зосереджені виключно на точності точкового прогнозу [53, 142] або спираються на адитивні моделі впливу, які за своєю структурою не відтворюють нелінійну взаємодію параметрів і не надають оцінки невизначеності [142].

Формально мету методу можна подати як пошук мінімальної кількості натурних вимірювань, достатньої для досягнення заданих порогів якості. Нехай $\hat{f}_N$ – сурогатна модель, побудована за навчальною вибіркою обсягом N , $R^2(\hat{f}_N)$ – коефіцієнт детермінації на незалежній тестовій вибірці, а $C_\alpha(\widehat{f}_N)$ – фактичне покриття довірчого інтервалу рівня α . Тоді цільова задача формулюється так:

$$N^* = \min\{N \mid R^2(\widehat{f}_N) \geq R_{min}^2 \wedge |C_\alpha(\widehat{f}_N) - \alpha| \leq \varepsilon\}, \quad (2.2)$$

де R_{min}^2 – мінімально прийнятний рівень точності, ε – допустиме відхилення фактичного покриття від заявленого. У дослідженні прийнято $R_{min}^2 = 0,90$ та контроль покриття для рівня $\alpha = 0,95$. Величина N^* безпосередньо визначає

економію ресурсів порівняно з повним перебором і є основним практичним показником ефективності сурогатного підходу.

Для формування навчальної вибірки застосовано планування комп'ютерного експерименту за схемою латинського гіперкубічного вибору (Latin Hypercube Sampling), яка забезпечує рівномірне заповнення багатовимірного простору параметрів і, на відміну від рівномірної сітки, гарантує, що кожен рівень кожного параметра трапляється у проєкції рівно один раз [106]. Такий план є значно ефективнішим за повний перебір для задач апроксимації відгуку і дозволяє суттєво скоротити кількість необхідних вимірювань за збереження репрезентативності вибірки.

Отже, простір вхідних змінних сурогатної моделі формалізовано з розмежуванням керованих параметрів конфігурації та зовнішніх параметрів навантаження, а мультиплікативна модель спостережень разом із логарифмічним перетворенням цільової змінної і стандартизацією ознак створює формальне підґрунтя для вибору апарату апроксимації поверхні відгуку.

2.2. Формалізація простору «конфігурація – навантаження»

Побудова сурогатної моделі (2.1) вимагає попередньої формалізації простору вхідних змінних, у якому визначено поверхню відгуку продуктивності. Цей простір складається з двох принципово різних за природою груп параметрів: керованих параметрів конфігурації системи, значення яких обирає проєктувальник на етапі розгортання, та параметрів навантаження, які визначаються зовнішніми умовами експлуатації й не підлягають безпосередньому керуванню. Розмежування цих двох груп є методологічно суттєвим, оскільки мета автоматизованого проєктування полягає в оптимізації саме керованих параметрів конфігурації за очікуваного або найгіршого профілю навантаження.

Керовану конфігурацію системи подамо вектором

$$x = (x_1, x_2, x_3, x_4)^T = (\text{RAM}, \text{CPU}, \text{pool}, \text{workers})^T, \quad (2.3)$$

де x_1 – обсяг оперативної пам'яті, виділеної системі (ГБ); x_2 – кількість доступних ядер процесора; x_3 – розмір пулу з'єднань до бази даних; x_4 – кількість паралельних робочих процесів застосунку. Ці чотири параметри охоплюють основні ресурсні обмеження, що визначають пропускну здатність корпоративного веб-стеку, і безпосередньо задаються засобами оркестрації контейнерів на етапі розгортання [162]. Кожен із параметрів набуває значень у скінченному діапазоні, обумовленому апаратними можливостями цільового середовища, причому параметри x_2 , x_3 , x_4 є цілочисловими за своєю фізичною природою.

Умови експлуатації подамо вектором характеристик навантаження

$$l = (l_1, l_2)^T = (VUs, w)^T, \quad (2.4)$$

де l_1 – кількість віртуальних користувачів, що одночасно взаємодіють із системою (інтенсивність надходження запитів); $l_2 = w$ – частка операцій запису в загальному потоці запитів. Параметр w уведено окремо, оскільки операції запису у реляційній базі даних є істотно повільнішими за операції читання внаслідок накладних витрат на журналювання та підтримання узгодженості, тому співвідношення читання і запису суттєво впливає на затримку відповіді навіть за незмінної загальної інтенсивності навантаження [162].

Об'єднання (2.3) і (2.4) утворює повний вектор вхідних змінних сурогатної моделі

$$z = (x^T, l^T)^T \in \mathcal{D} \subset R^6, \quad (2.5)$$

де $\mathcal{D}$ – допустима область простору параметрів, задана прямим добутком діапазонів окремих змінних із урахуванням цілочислових обмежень. Розмірність простору дорівнює шести; саме у цьому шестивимірному просторі й визначено нелінійну поверхню відгуку продуктивності, яку належить апроксимувати. Діапазони та характер значень параметрів, прийняті у дослідженні, наведено в таблиці 2.1.

Повна сітка значень за фіксованих дискретних рівнів параметрів налічує $N_{\text{grid}} = 720$ конфігурацій, кожна з яких потребує окремого натурального навантажувального вимірювання. Саме ця величина є базою для оцінювання

економії, яку забезпечує сурогатний підхід згідно з формулою (2.2). Кардинальність повної сітки обчислюється як добуток кількості рівнів за кожним параметром і зростає мультиплікативно з розмірністю простору, що робить повний перебір непридатним уже за помірної кількості параметрів [142, 106].

Таблиця 2.1

Параметри простору «конфігурація – навантаження»

Параметр	Позначення	Група	Діапазон значень	Характер
Оперативна пам'ять, ГБ	RAM	конфігурація	від 2 до 16	дискретний
Ядра процесора	CPU	конфігурація	від 1 до 8	цілочисловий
Розмір пулу з'єднань	pool	конфігурація	від 10 до 200	цілочисловий
Робочі процеси	workers	конфігурація	від 2 до 16	цілочисловий
Віртуальні користувачі	VUs	навантаження	від 10 до 400	цілочисловий
Частка операцій запису	w	навантаження	від 0 до 0,6	неперервний

Спостережуване значення цільової метрики не збігається з істинним значенням поверхні відгуку через варіативність натурних вимірювань: повторні навантажувальні тести однієї й тієї самої конфігурації дають дещо різні результати внаслідок фонові активності операційної системи, коливань планувальника та мережних затримок. Тому модель спостережень подамо у мультиплікативній формі

$$y = g(\mathbf{z}) \cdot \varepsilon, \ln \varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2), \quad (2.6)$$

де $g(\mathbf{z})$ – детермінована поверхня відгуку р95-затримки, ε – мультиплікативний шум вимірювань, розподілений за логнормальним законом. Мультиплікативний характер шуму відповідає фізичній природі затримок: абсолютна варіативність вимірювань зростає пропорційно до рівня затримки, а самі затримки є додатними величинами з асиметричним, витягнутим у бік великих значень розподілом. Логнормальна модель шуму природно узгоджується з обома цими властивостями.

Асиметрія та широкий динамічний діапазон розподілу затримок, що охоплює кілька порядків – від десятків мілісекунд у ненавантаженому режимі до кількох секунд поблизу насичення ресурсів, – ускладнюють безпосереднє моделювання величини y . Для нормалізації розподілу та стабілізації дисперсії цільову змінну піддають логарифмічному перетворенню

$$\tilde{y} = \ln y, \quad (2.7)$$

після чого сурогатну модель будують у просторі логарифмованої затримки, а при формуванні прогнозу застосовують зворотнє перетворення $\hat{y} = \exp(\hat{\tilde{y}})$. Логарифмічне перетворення переводить мультиплікативний шум (2.6) в адитивний, наближає розподіл цільової змінної до нормального й забезпечує однорідність дисперсії за всім діапазоном навантажень, що є необхідною передумовою коректного застосування апарату гаусівської процесової регресії, розглянутого у підрозділі 2.3.

Компоненти вектора z мають різну фізичну природу та різні масштаби: кількість ядер вимірюється одиницями, обсяг пам'яті – одиницями та десятками, кількість користувачів – сотнями. Безпосереднє використання таких різномасштабних змінних спотворює обчислення відстаней у просторі ознак, від яких залежить робота ядрових методів. Тому перед навчанням кожен ознаку піддають стандартизації до нульового середнього та одиничної дисперсії

$$z_j^{\text{ст}} = \frac{z_j - \bar{z}_j}{s_j}, \quad j = 1, \dots, d, \quad (2.8)$$

де $\bar{z}_j$ та s_j – відповідно вибіркове середнє та стандартне відхилення j -ї ознаки, обчислені виключно на навчальній вибірці для запобігання просоченню інформації з тестових даних, d – загальна кількість ознак. Параметри стандартизації зберігаються та застосовуються без змін до валідаційної й тестової вибірок, а також до всіх нових точок прогнозування.

Слід підкреслити, що безпосередніх шести первинних параметрів (2.5) недостатньо для адекватного опису поверхні відгуку: визначальним для продуктивності є не абсолютне значення навантаження чи ресурсів окремо, а їхнє співвідношення, тобто ступінь завантаженості системи. Тому простір ознак

розширюють конструйованими змінними, що відображають фізичну семантику взаємодії навантаження й потужності системи; побудову цих ознак на основі теорії черг детально розглянуто у підрозділі 2.4. У формалізованому вигляді простір ознак моделі $\phi(z)$ є образом первинного простору (2.5) під дією відображення конструювання ознак, а сурогатна модель (2.1) фактично апроксимує відгук у розширеному просторі $\phi(z)$.

Отже, формалізовано простір вхідних змінних сурогатної моделі: розмежовано керовані параметри конфігурації (2.3) і зовнішні параметри навантаження (2.4), задано допустиму область (2.5) з діапазонами за таблицею 2.1, побудовано мультиплікативну модель спостережень із логнормальним шумом (2.6) та обґрунтовано логарифмічне перетворення цільової змінної (2.7) і стандартизацію ознак (2.8). Таким чином створено формальне підґрунтя для вибору апарату апроксимації поверхні відгуку.

2.3. Гаусівська процесова регресія як апарат сурогатної моделі

Вибір апарату апроксимації поверхні відгуку зумовлений двоєдиною вимогою задачі (2.1): модель має не лише відтворювати нелінійну залежність р95-затримки від параметрів конфігурації й навантаження, а й супроводжувати кожен прогноз відкаліброваною оцінкою невизначеності. Цій вимозі найповніше відповідає гаусівська процесова регресія (Gaussian Process Regression, GPR) – непараметричний байєсівський метод, що задає ймовірнісний розподіл безпосередньо над простором функцій і природно поєднує точковий прогноз із його дисперсією [129]. На відміну від параметричних моделей, які фіксують наперед задану функціональну форму залежності, гаусівський процес допускає довільну гладку поверхню відгуку, форму якої визначають самі дані через структуру коваріаційної функції.

Гаусівський процес означають як сукупність випадкових величин, будь-яка скінченна підмножина яких має спільний багатовимірний нормальний

розподіл. Такий процес повністю задається функцією середнього $m(z)$ та коваріаційною функцією (ядром) $k(z, z')$:

$$f(z) \sim \mathcal{GP}(m(z), k(z, z')), \quad (2.9)$$

де $m(z) = E[f(z)]$, а ядро $k(z, z') = E[(f(z) - m(z))(f(z') - m(z'))]$ визначає коваріацію значень функції у двох точках простору ознак. Без обмеження загальності функцію середнього приймають нульовою після центрування цільової змінної, тож уся структура моделі зосереджується у ядрі. Коваріаційна функція відображає базове припущення методу: близькі у просторі ознак конфігурації дають близькі значення продуктивності, а міра цієї близькості керується гіперпараметрами ядра.

Нехай навчальна вибірка складається з n спостережень $\{(\mathbf{z}_i, \tilde{y}_i)\}_{i=1}^n$, де $\tilde{y}_i$ – логарифмована згідно з (2.7) р95-затримка. Позначимо через Z матрицю входів навчальної вибірки, через $\tilde{\mathbf{y}}$ – вектор відповідних цільових значень. Відповідно до моделі спостережень цільові значення вважаються зашумленими реалізаціями латентної функції:

$$\tilde{y}_i = f(\mathbf{z}_i) + \eta_i, \quad \eta_i \sim \mathcal{N}(0, \sigma_n^2), \quad (2.10)$$

де σ_n^2 – дисперсія шуму вимірювань, що після логарифмічного перетворення (2.7) з мультиплікативного стає адитивним і нормальним. Позначимо коваріаційну матрицю навчальних точок як $\mathbf{K} \in \mathbb{R}^{n \times n}$ з елементами $K_{ij} = k(\mathbf{z}_i, \mathbf{z}_j)$, вектор коваріацій між навчальними точками й новою точкою прогнозування $\mathbf{z}_*$ як $\mathbf{k}_*$ з компонентами $k_i = k(\mathbf{z}_i, \mathbf{z}_*)$, а власну коваріацію нової точки як $k_{**} = k(\mathbf{z}_*, \mathbf{z}_*)$. Тоді спільний розподіл спостережуваних значень і значення функції у новій точці є нормальним, а застосування правила умовного гаусівського розподілу дає апостеріорний прогнозний розподіл, який також є нормальним із такими середнім і дисперсією::

$$\mu(\mathbf{z}_*) = \mathbf{k}_*^\top (\mathbf{K} + \sigma_n^{2I})^{-1} \tilde{\mathbf{y}}; \quad (2.11)$$

$$\sigma^2(\mathbf{z}_*) = k_{**} - \mathbf{k}_*^\top (\mathbf{K} + \sigma_n^{2I})^{-1} \mathbf{k}_*. \quad (2.12)$$

Формули (2.11) і (2.12) становлять ядро методу. Прогнозоване середнє (2.11) є лінійною комбінацією навчальних спостережень з вагами, визначеними

коваріаційною структурою, і після зворотного експоненційного перетворення дає точкову оцінку μ у постановці (2.1). Прогнозна дисперсія (2.12) дає оцінку невизначеності σ^2 і має принципову властивість: вона зростає в областях простору ознак, віддалених від навчальних точок, і спадає поблизу них.

Доданок σ_n^{2I} у формулах (2.11), (2.12) виконує подвійну роль. З погляду статистики він моделює шум вимірювань (2.10), а з погляду обчислень – забезпечує регуляризацію коваріаційної матриці, гарантуючи її додатну визначеність і чисельну стійкість обернення. У практиці високопродуктивних реалізацій GPR цей доданок відомий як параметр згладжування шуму (nugget), і його коректний добір є критичним для стабільної роботи моделі; занадто мале значення призводить до чисельної нестійкості, зavelike – до надмірного згладжування поверхні відгуку [28]. У дослідженні значення σ_n^2 визначається автоматично разом з іншими гіперпараметрами.

Гіперпараметри ядра та дисперсію шуму визначають шляхом максимізації логарифма маргінальної правдоподібності навчальних даних:

$$\ln p(\tilde{y} | Z) = -\frac{1}{2} \tilde{y}^\top (K + \sigma_n^{2I})^{-1} \tilde{y} - \frac{1}{2} \ln |K + \sigma_n^{2I}| - \frac{n}{2} \ln 2\pi. \quad (2.13)$$

Вираз (2.13) втілює принцип бритви Оккама у байєсівському формулюванні: перший доданок карає невідповідність моделі даним, другий – надмірну складність (об'єм простору функцій, що охоплюється ядром), а їхній баланс автоматично запобігає перенавчанню. Оптимізацію (2.13) виконують градієнтними методами з кількома перезапусками з різних початкових наближень для уникнення локальних максимумів.

Властивості поверхні відгуку, яку здатна відтворити модель, повністю визначаються вибором ядра. У дослідженні порівняно три коваріаційні функції, що відрізняються припущеннями про гладкість відгуку. Радіально-базисне ядро (RBF) визначається як

$$k_{\text{RBF}}(r) = \sigma_f^2 \exp\left(-\frac{r^2}{2l^2}\right), \quad r = |z - z'|, \quad (2.14)$$

де σ_f^2 – дисперсія сигналу, l – масштаб довжини (характерна відстань кореляції). Ядро RBF припускає нескінченну диференційовність функції відгуку, що для реальних систем часто є надмірно сильним припущенням і призводить до занадто гладких прогнозів. Ядро Матерн з параметром гладкості ν узагальнює цей клас, дозволяючи керувати регулярністю функції; для практично важливого значення $\nu = 5/2$ воно набуває вигляду

$$k_{\text{Mat}}^{\nu=5/2}(r) = \sigma_f^2 \left(1 + \frac{\sqrt{5}r}{l} + \frac{5r^2}{3l^2} \right) \exp\left(-\frac{\sqrt{5}r}{l}\right). \quad (2.15)$$

Значенню $\nu = 5/2$ відповідає двічі диференційовна функція відгуку [145], що фізично узгоджується з характером зміни продуктивності веб-системи: залежність затримки від параметрів є плавною, проте не нескінченно гладкою через ефекти насичення ресурсів і черговості запитів. Раціональна квадратична функція (RQ) є нескінченною сумішшю RBF-ядер з різними масштабами довжини і теоретично гнучкіша:

$$k_{\text{RQ}}(r) = \sigma_f^2 \left(1 + \frac{r^2}{2\alpha l^2} \right)^{-\alpha}, \quad (2.16)$$

де $\alpha > 0$ – параметр, що керує відносним внеском масштабів; при $\alpha \rightarrow \infty$ ядро RQ переходить у RBF (2.14). Попри більшу гнучкість, ядро RQ схильне до надмірної апроксимації шуму за обмеженого обсягу навчальних даних.

Порівняльні характеристики трьох коваріаційних функцій за припущеннями про гладкість відгуку узагальнено в таблиці 2.2.

Таблиця 2.2

Порівняння коваріаційних функцій (ядер) гаусівської процесової регресії

Ядро	Коваріаційна функція	Гладкість відгуку	Особливості застосування
RBF	$\sigma_f^2 \cdot \exp(-r^2/2\ell^2)$	нескінченно диференційовна	надмірно гладкі прогнози; проміжна точність
Матерн 5/2	$\sigma_f^2(1+\sqrt{5} \cdot r/\ell + 5r^2/3\ell^2) \cdot \exp(-\sqrt{5} \cdot r/\ell)$	двічі диференційовна	фізично обґрунтована; обрана як основна
Раціональна квадратична	$\sigma_f^2(1+r^2/2\alpha\ell^2)^{-\alpha}$	суміш масштабів довжини	гнучка; схильна до апроксимації шуму

Важливим для стійкості методу є вибір параметризації масштабу довжини. Ізотропне ядро з єдиним спільним масштабом l для всіх ознак містить мінімальну кількість гіперпараметрів і забезпечує стабільну оптимізацію (2.13) навіть за малого обсягу навчальної вибірки. Альтернативна анізотропна параметризація з окремим масштабом для кожної ознаки (автоматичне визначення релевантності) теоретично точніша, проте за десятків гіперпараметрів і десятків навчальних точок призводить до вироджених розв'язків, коли масштаби окремих вимірів впираються у межі допустимого діапазону, а прогноз стає нестійким. З огляду на невеликий обсяг навчальної вибірки, притаманний задачі скорочення натурних вимірювань, у дослідженні застосовано ізотропну параметризацію з обмеженими діапазонами гіперпараметрів, що забезпечує відтворюваність і стійкість результатів.

За сукупністю гнучкості, стійкості на малих вибірках і фізичної обґрунтованості перевагу віддано ядру Матерн $5/2$, вибір якого спирається на змістовну відповідність параметра гладкості $5/2$ характеру поверхні відгуку продуктивності: залежність затримки від параметрів є плавною, проте не нескінченно гладкою через ефекти насичення ресурсів. Придатність апарату гаусівської процесової регресії для прогнозування у прикладних інженерних задачах з обмеженими вибірками підтверджується і незалежними вітчизняними дослідженнями суміжних предметних областей [6].

Отже, гаусівська процесова регресія надає сурогатній моделі апостеріорний прогнозний розподіл, у якому точкова оцінка продуктивності супроводжується дисперсією, а гіперпараметри визначаються максимізацією маргінальної правдоподібності; серед розглянутих коваріаційних функцій основною обрано ядро Матерн $5/2$.

2.4. Конструювання ознак на основі фізичної моделі системи

Якість апроксимації поверхні відгуку сурогатною моделлю визначається не лише вибором ядра, а й тим, у якому просторі ознак ця поверхня описується.

Як зазначено у підрозділі 2.2, шести первинних параметрів (2.5) недостатньо для адекватного опису продуктивності, і причина цього має фундаментальний, а не технічний характер. Затримка відповіді залежить не від абсолютних значень ресурсів чи навантаження, узятих окремо, а від їхнього співвідношення. Система з потужною апаратною конфігурацією, завантажена сотнями користувачів, і слабка система за десятків користувачів можуть демонструвати однакову затримку, якщо співвідношення навантаження до потужності в обох випадках збігається. Модель, яка оперує лише первинними параметрами, змушена відновлювати це співвідношення опосередковано через складну взаємодію ознак, тоді як безпосереднє введення відповідних конструйованих ознак робить структуру залежності явною і різко спрощує задачу апроксимації. Саме тому конструювання ознак на основі фізичної моделі системи є ключовим елементом запропонованого методу.

Фізичним підґрунтям для побудови ознак слугує теорія масового обслуговування, а конкретно – модель багатоканальної черги з очікуванням, у якій потік запитів обслуговується кількома паралельними обробниками [83]. Корпоративний веб-стек природно відображається на таку модель: запити користувачів утворюють вхідний потік, робочі процеси й ядра процесора виконують роль каналів обслуговування, а обмежений пул з'єднань до бази даних задає верхню межу одночасно оброблюваних запитів. Центральним поняттям цієї моделі є коефіцієнт завантаженості системи – відношення інтенсивності надходження запитів до сумарної інтенсивності їх обслуговування:

$$\rho = \frac{\lambda}{c\mu}, \quad (2.17)$$

де λ – інтенсивність вхідного потоку запитів, пропорційна кількості віртуальних користувачів; μ – інтенсивність обслуговування одним каналом; c – ефективна кількість паралельних каналів обслуговування.

Значення ρ лежить у межах від нуля до одиниці для стійкого режиму роботи; що ближче воно до одиниці, то щільніше система завантажена.

Принципова властивість черги полягає в тому, що середній час перебування запиту в системі зростає непропорційно до ρ : поки завантаженість помірна, затримка змінюється повільно, проте з наближенням ρ до одиниці вона зростає лавиноподібно, прямуючи до нескінченності. Цей ефект породжує характерну точку перелому (коліно) на поверхні відгуку, за якою система переходить у режим насичення. Саме нездатність адитивних моделей відтворити цю нелінійну залежність, зосереджену у відношенні (2.17), а не в окремих параметрах, є теоретичною причиною їхньої принципової непридатності для задачі.

Виходячи з цієї фізичної картини, до простору ознак уведено чотири конструйовані змінні, що відображають ключові аспекти взаємодії навантаження й ресурсів; окремо серед вхідних ознак збережено первинну ознаку частки операцій запису w (2.4). Першою є оцінка потужності системи, яка агрегує ресурси паралельного оброблення. Кількість паралельних обробників визначається насамперед добутком числа ядер процесора та робочих процесів, тоді як розмір пулу з'єднань впливає на потужність із насиченням – збільшення пулу понад певну межу вже не додає пропускну здатності, оскільки вузьким місцем стає процесор. Тому внесок пулу враховують логарифмічно:

$$\text{Capacity} = x_2 x_4 \log_2(x_3), \quad (2.18)$$

де x_2 – кількість ядер, x_4 – кількість робочих процесів, x_3 – розмір пулу з'єднань. Величина (2.18) є конструктивним аналогом сумарної інтенсивності обслуговування c_m з формули (2.17) і характеризує здатність системи опрацьовувати паралельне навантаження.

Другою і найважливішою конструйованою ознакою є інтенсивність навантаження – відношення кількості користувачів до потужності системи:

$$\text{LoadIntensity} = \frac{l_1}{\text{Capacity}} = \frac{VUs}{x_2 x_4 \log_2(x_3)}. \quad (2.19)$$

Ця ознака є безпосереднім конструктивним аналогом коефіцієнта завантаженості ρ (2.17). Вона в одній величині поєднує інформацію про навантаження й про всі ресурси системи, а тому несе основну частину інформації про режим роботи. Введення саме цієї ознаки дозволяє сурогатній моделі

відтворити коліно насичення природним чином, оскільки поверхня відгуку у координатах інтенсивності навантаження має просту монотонну форму з різким зростанням поблизу критичного значення, тоді як у координатах первинних параметрів та сама поверхня виглядає складною й багатовимірною.

Третя ознака характеризує конкуренцію за з'єднання до бази даних. Коли кількість одночасних запитів перевищує розмір пулу, частина запитів змушена очікувати на звільнення з'єднання, що додає затримку, не пов'язану безпосередньо з обчислювальними ресурсами:

$$\text{PoolContention} = \frac{l_1}{x_3} = \frac{\text{VUs}}{\text{pool}}. \quad (2.20)$$

Четверта ознака відображає забезпеченість пам'яттю в розрахунку на один робочий процес. Недостатній обсяг пам'яті на процес спричиняє витіснення даних із кешу та зростання частки повільних звернень до диску, що підвищує ефективний час обслуговування:

$$\text{RAMperWorker} = \frac{x_1}{x_4} = \frac{\text{RAM}}{\text{workers}}. \quad (2.21)$$

Частку операцій запису w , уведено ще у складі вектора навантаження (2.4), збережено серед вхідних ознак з огляду на істотний і безпосередній вплив на час обслуговування: операції запису у реляційній базі даних приблизно у півтора раза повільніші за операції читання [162]. Отже, розширений простір ознак сурогатної моделі утворюється відображенням первинного простору у сукупність первинних і конструйованих змінних:

$\phi(z) = (x, l, \text{Capacity}, \text{LoadIntensity}, \text{PoolContention}, \text{RAMperWorker})$, (2.22)
після чого гаусівська процесова регресія (2.11), (2.12) будується вже у просторі $\phi(z)$, а не у первинному просторі (2.5). Слід підкреслити, що конструйовані ознаки не додають нової інформації понад ту, що міститься у первинних параметрах, – усі вони є їхніми детермінованими функціями. Їхня роль полягає в іншому: вони переводять залежність, приховану у складній взаємодії первинних параметрів, у явну форму, придатну для ефективної апроксимації гладким ядром,

що узгоджується із загальним принципом, згідно з яким вдале подання даних часто важливіше за вибір самого алгоритму навчання.

Провідну роль ознаки інтенсивності навантаження доцільно перевіряти кількісно засобами аналізу інтерпретованості моделей. Для оцінювання внеску кожної ознаки у прогноз застосовано метод адитивних пояснень Шеплі (SHapley Additive exPlanations, SHAP), який на основі кооперативної теорії ігор розподіляє прогнозоване значення між ознаками пропорційно до їхнього середнього граничного внеску за всіма можливими комбінаціями ознак [99]. На відміну від простих показників важливості, метод SHAP має міцне теоретичне підґрунтя і задовольняє властивості локальної точності та узгодженості, тож придатний для обґрунтованого ранжування ознак. Відповідно до наведеного теоретичного положення від сурогатної моделі очікується домінування ознаки інтенсивності навантаження над первинними параметрами, узятими окремо, оскільки поведінку системи описує саме співвідношення навантаження й потужності.

Таблиця 2.3

Конструйовані ознаки та їхній фізичний зміст

Ознака	Формула	Фізичний зміст	Аналог у теорії черг
Потужність (Capacity)	$CPU \cdot workers \cdot \log_2(pool)$	ресурс паралельного оброблення	інтенсивність обслуговування $c \cdot \mu$
Інтенсивність навантаження	$VUs/Capacity$	ступінь завантаженості системи	коефіцієнт завантаженості ρ
Конкуренція за пул	$VUs/pool$	очікування на з'єднання до БД	довжина черги за ресурс
Пам'ять на процес	$RAM/workers$	забезпеченість пам'яттю	вплив на час обслуговування $1/\mu$
Частка запису (w)	w	частка повільніших операцій	диференціація часу обслуговування

Схему конструювання ознак від первинних параметрів до простору ознак сурогатної моделі наведено на рисунку 2.1.

Практична цінність такого положення виходить за межі підвищення точності моделі. Знання про домінування інтенсивності навантаження дає проєктувальникові чіткий орієнтир: для утримання затримки в допустимих

межах слід контролювати саме відношення (2.19), а не нарощувати окремі ресурси наосліп. Крім того, кількісна оцінка впливу конкуренції за з'єднання (2.20) обґрунтовує рішення щодо обмеження інтенсивності запитів і розміру пулу, а оцінка забезпеченості пам'яттю (2.21) – рішення щодо розподілу пам'яті між процесами. Таким чином, конструйовані ознаки слугують не лише входами моделі, а й інтерпретованими показниками стану системи, придатними для підтримки проєктних рішень.

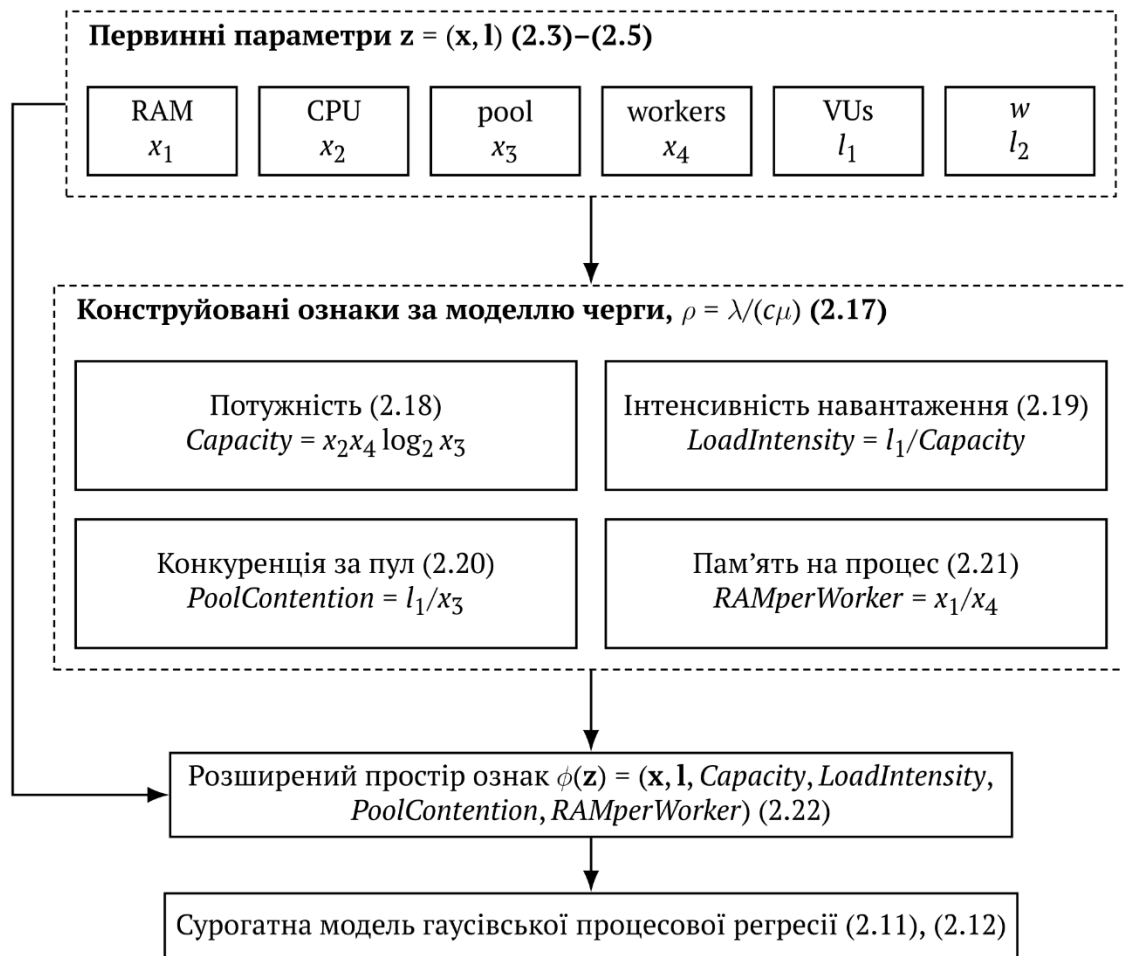


Рис. 2.1. Схема конструювання ознак на основі фізичної моделі системи

Отже, перехід від первинних параметрів до розширеного простору ознак обґрунтовано фізичною моделлю черги, у якій центральною конструйованою ознакою є інтенсивність навантаження як конструктивний аналог коефіцієнта завантаженості, тож конструювання ознак створює передумови для точної та інтерпретованої апроксимації й пояснює непридатність адитивних моделей.

2.5. Калібрування довірчих інтервалів та кількісна оцінка невизначеності

Друга складова вимоги до сурогатної моделі, сформульованої у задачі (2.1), стосується якості оцінки невизначеності. Здатність гаусівської процесової регресії надавати разом із прогнозом його дисперсію (2.12) сама по собі ще не гарантує, що ця дисперсія є достовірною мірою впевненості. Модель може бути надмірно самовпевненою, занижуючи невизначеність, або, навпаки, надмірно обережною, завищуючи її. В обох випадках супровідний довірчий інтервал уводить користувача в оману: у першому випадку реальні значення систематично виходитимуть за межі передбачених інтервалів частіше, ніж заявлено, у другому – інтервали будуть не виправдано широкими й малокорисними для прийняття рішень. Тому окремою і принциповою вимогою до методу є відкаліброваність довірчих інтервалів, тобто статистична узгодженість заявленого та фактичного рівнів довіри. Ця вимога є ключовою відмінністю задачі від класичної регресії, у якій достатньо мінімізувати похибку точкового прогнозу.

Формально відкаліброваність означають так. Для кожного рівня довіри $\alpha \in (0,1)$ модель на основі прогнозного розподілу будує довірчий інтервал, що має містити реальне значення з імовірністю α . Оскільки апостеріорний прогноз GPR у просторі логарифмованої затримки є нормальним із середнім $\mu(z_*)$ і дисперсією $\sigma^2(z_*)$, довірчий інтервал рівня α подається у симетричній формі відносно середнього:

$$I_\alpha(z_*) = [\mu(z_*) - z_\gamma \sigma(z_*), \mu(z_*) + z_\gamma \sigma(z_*)], \quad \gamma = \frac{1 + \alpha}{2}, \quad (2.23)$$

де z_γ – квантиль стандартного нормального розподілу рівня γ . Для практично важливого рівня $\alpha = 0,95$ множник z_γ становить приблизно 1,96. Межі інтервалу (2.23) обчислюються у логарифмованому просторі, після чого до них застосовують зворотнє експоненційне перетворення, отримуючи асиметричний

інтервал у вихідній шкалі затримки, що природно узгоджується з асиметрією її розподілу.

Модель вважається відкаліброваною, якщо фактична частка реальних значень, що потрапляють у передбачений інтервал (2.23), дорівнює заявленому рівню α для всіх його значень. Кількісною мірою цієї частки є емпіричне покриття, обчислене на незалежній тестовій вибірці обсягом m :

$$C(\alpha) = \frac{1}{m} \sum_{i=1}^m \mathbb{1}\{\tilde{y}_i \in I_{\alpha}(z_i)\}, \quad (2.24)$$

де $\mathbb{1} \cdot$ – індикаторна функція, що дорівнює одиниці, якщо реальне значення потрапило в інтервал, і нулю в іншому разі. Для ідеально відкаліброваної моделі емпіричне покриття збігається із заявленим рівнем, тобто $C(\alpha) = \alpha$ для всіх α . Окремим важливим показником є покриття для рівня 0,95, позначене $\text{Coverage}_{95} = C(0,95)$, яке безпосередньо характеризує надійність найуживанішого 95-відсоткового довірчого інтервалу.

Наочним інструментом перевірки відкаліброваності є діаграма надійності (reliability diagram). Її будують так: для послідовності рівнів довіри α у діапазоні від значень, близьких до нуля, до значень, близьких до одиниці, обчислюють фактичне покриття (2.24) і відкладають його проти заявленого рівня. Точки ідеально відкаліброваної моделі лягають на діагональ $C(\alpha) = \alpha$. Відхилення кривої вгору від діагоналі свідчить про надмірну обережність моделі (інтервали ширші, ніж потрібно), відхилення вниз – про надмірну самовпевненість (інтервали завузькі). Діаграма надійності дає повну картину калібрування за всіма рівнями довіри одночасно, а не лише для одного вибраного значення.

Для згортання діаграми надійності в єдиний числовий показник застосовують очікувану похибку калібрування (Expected Calibration Error, ECE) – усереднене за рівнями довіри абсолютне відхилення фактичного покриття від заявленого:

$$\text{ECE} = \frac{1}{L} \sum_{k=1}^L |C(\alpha_k) - \alpha_k|, \quad (2.25)$$

де $\alpha_{k=1}^L$ – сітка рівнів довіри, рівномірно розподілених у робочому діапазоні. Що менше значення ESE, то ближча модель до ідеального калібрування; нульове значення відповідає повній узгодженості заявленого та фактичного покриття. Метрика (2.25) є зручним скалярним критерієм для порівняння різних моделей і ядер за якістю оцінки невизначеності, доповнюючи традиційні метрики точності точкового прогнозу.

Ключову роль у забезпеченні відкаліброваності відіграє параметр згладжування шуму σ_n^2 , уведений у моделі спостережень (2.10) і присутній у формулах прогнозу (2.11), (2.12). Цей параметр безпосередньо визначає ширину прогнозних інтервалів: занижене значення σ_n^2 звужує інтервали й робить модель надмірно самовпевненою, що виявляється у покритті нижчому за заявлене; завищене значення надмірно розширює інтервали, знижуючи їхню інформативність. Оскільки σ_n^2 визначається автоматично разом з іншими гіперпараметрами шляхом максимізації маргінальної правдоподібності (2.13), калібрування досягається природним чином у процесі навчання, без окремого етапу підстроювання. Водночас коректний добір діапазону допустимих значень цього параметра є критичним для чисельної стійкості моделі: у високопродуктивних реалізаціях гаусівської процесової регресії саме параметр згладжування шуму забезпечує додатну визначеність коваріаційної матриці й запобігає її виродженню [28]. У випадках, коли автоматичне налаштування не забезпечує достатньої відкаліброваності, застосовують додаткові процедури апостеріорного калібрування регресійних моделей, які коригують ширину інтервалів за окремою калібрувальною вибіркою [90]; у запропонованому методі потреби в такому доопрацюванні не виникло.

Сукупність метрик для оцінювання точності та відкаліброваності сурогатної моделі узагальнено в таблиці 2.4.

Отже, відкаліброваність довірчих інтервалів є самостійною вимогою до сурогатної моделі, яку перевіряють емпіричним покриттям, діаграмою надійності та очікуваною похибкою калібрування, а визначальну роль у її

забезпеченні відіграє параметр згладжування шуму; саме ця вимога вирізняє запропонований підхід серед адитивних альтернатив.

Таблиця 2.4

Метрики оцінювання сурогатної моделі

Метрика	Позначення	Що характеризує	Напрямок кращого значення
Середньоквадратична похибка	RMSE	точність, посилений штраф великих похибок	менше
Середня абсолютна похибка	MAE	середня точність прогнозу	менше
Коефіцієнт детермінації	R^2	частка поясненої дисперсії відгуку	більше, до 1
Покриття довірчого інтервалу	Coverage ₉₅	відкаліброваність 95% інтервалу	ближче до 0,95
Очікувана похибка калібрування	ECE	відкаліброваність за всіма рівнями довіри	менше, до 0

2.6. Критерій активного навчання за предиктивною невизначеністю

Скорочення обсягу натурних вимірювань, задане як основна мета методу у формулі (2.2), досягається не лише за рахунок якісної апроксимації, а й завдяки раціональному вибору того, які саме конфігурації підлягають вимірюванню. Початкове формування навчальної вибірки за схемою латинського гіперкубічного вибору (підрозділ 2.2) забезпечує рівномірне заповнення простору параметрів, проте воно є пасивним: точки обираються наперед, без урахування поведінки самої поверхні відгуку. Тим часом різні області простору конфігурацій потребують різної щільності вимірювань. Там, де поверхня відгуку є плавною і добре передбачуваною, достатньо небагатьох точок, тоді як поблизу коліна насичення чи в областях складної взаємодії параметрів модель потребує більшої кількості спостережень. Активне навчання дозволяє врахувати цю нерівномірність і спрямувати обмежений бюджет вимірювань туди, де кожне нове спостереження приносить найбільший приріст якості моделі.

Ключовою передумовою активного навчання є наявність достовірної оцінки невизначеності, яку гаусівська процесова регресія надає безпосередньо у

вигляді прогнозової дисперсії (2.12). Як зазначено у підрозділі 2.3, ця дисперсія має принципову властивість: вона зростає в областях простору ознак, віддалених від навчальних точок, і спадає поблизу них. Отже, дисперсія прогнозу є природним показником того, наскільки погано досліджено ту чи іншу область простору конфігурацій, звідки безпосередньо впливає стратегія вибору наступного вимірювання: вимірювати слід там, де модель найменш упевнена, тобто де прогнозна дисперсія максимальна. Такий критерій відомий як вибірка за максимальною дисперсією (Maximum Variance Sampling) і формалізується як задача пошуку точки, що максимізує невизначеність прогнозу серед множини кандидатів:

$$z_{nx} = \arg \max_{z \in \mathcal{C}} \sigma^2(z), \quad (2.26)$$

де $\mathcal{C} \subset \mathcal{D}$ – множина конфігурацій-кандидатів у допустимій області, а $\sigma^2(z)$ – прогнозна дисперсія (2.12), обчислена поточною моделлю. Обрана згідно з (2.26) точка є тією конфігурацією, вимірювання якої максимально зменшить глобальну невизначеність моделі, оскільки саме в ній модель має найменше інформації.

Критерій (2.26) реалізує чисту стратегію дослідження простору конфігурацій. Він є граничним випадком ширшого класу функцій вибору, застосовуваних у байєсівській оптимізації, де рішення про наступну точку приймають, балансуючи між дослідженням недосліджених областей та використанням уже виявлених перспективних зон [138]. Для задачі побудови сурогатної моделі, метою якої є рівномірно якісна апроксимація всієї поверхні відгуку, а не пошук окремого оптимуму, доцільною є саме стратегія дослідження, що мінімізує загальну невизначеність. Застосування прогнозової дисперсії гаусівської процесової регресії як критерію вибору конфігурацій для вимірювання узгоджується з підходами, апробованими у суміжних дослідженнях моделювання продуктивності конфігурованих систем [116].

Практично активне навчання реалізують у формі ітеративного циклу поповнення навчальної вибірки, який складається з таких кроків.

1. Побудова початкової моделі за невеликою навчальною вибіркою, сформованою за схемою латинського гіперкубічного вибору.

2. Обчислення прогнозової дисперсії (2.12) для всіх конфігурацій-кандидатів із множини $\mathcal{C}$.
3. Вибір конфігурації з максимальною дисперсією (2.26).
4. Проведення натурного вимірювання для обраної конфігурації та додавання його результату до навчальної вибірки.
5. Перенавчання моделі на розширеній вибірці з переоцінюванням гіперпараметрів.

Послідовність операцій циклу активного навчання за критерієм максимальної дисперсії наведено на рисунку 2.2.

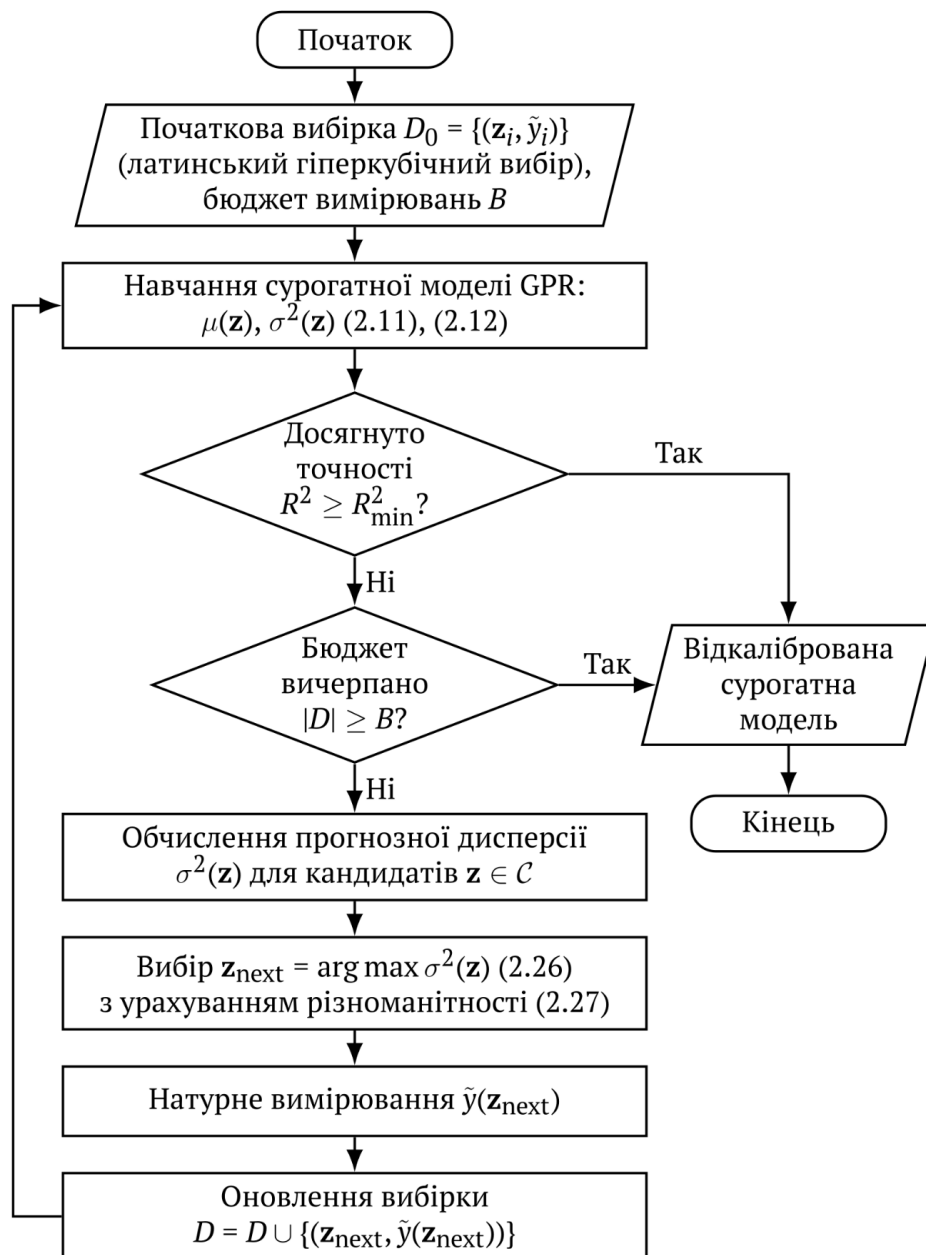


Рис. 2.2. Цикл активного навчання за критерієм максимальної дисперсії

На практиці замість вибору однієї точки за ітерацію часто відбирають групу з кількох конфігурацій з найвищою дисперсією, що дозволяє проводити вимірювання паралельно і скорочує кількість дорогих циклів перенавчання. Наївний відбір групи за критерієм (2.26) має, однак, ваду: конфігурації з найвищою дисперсією схильні концентруватися в одній і тій самій недослідженій області, тому їхнє одночасне вимірювання надає надлишкову інформацію.

Для уникнення цього ефекту застосовують відбір з урахуванням просторової різноманітності, коли до групи включають точки, рознесені у просторі ознак:

$$\mathcal{B} = \{z_{(1)}, \dots, z_{(k)}\}, \quad |z_{(i)} - z_{(j)}| \geq \delta \quad \text{для всіх } i \neq j, \quad (2.27)$$

де $\mathcal{B}$ – група з k конфігурацій для наступного пакету вимірювань, δ – мінімальна допустима відстань між ними. Такий підхід забезпечує, що кожне вимірювання пакета збагачує модель інформацією про власну область простору, а не дублює сусідів.

Наочним відображенням роботи критерію активного навчання є карта невизначеності – розподіл прогнозованої дисперсії у площині двох найінформативніших параметрів. На такій карті області з високою дисперсією відповідають недослідженим або нетиповим комбінаціям параметрів, а рекомендовані до вимірювання конфігурації розташовуються в зонах максимальної невизначеності. З властивостей прогнозованої дисперсії (2.12) випливає, що найвища невизначеність очікується в областях крайніх і рідкісних комбінацій параметрів, де щільність навчальних даних є найменшою, тоді як у добре заповнених центральних областях простору модель упевнена у своїх прогнозах.

Сукупний ефект початкового рівномірного планування вибірки та активного поповнення виявляється у залежності якості моделі від обсягу навчальних даних. Ця залежність, побудована як крива коефіцієнта детермінації від кількості навчальних точок, дозволяє визначити мінімальну кількість вимірювань N^* , достатню для досягнення цільового рівня точності згідно з

задачею (2.2). Економію ресурсів, яку забезпечує сурогатний підхід порівняно з повним перебором сітки, оцінюють показником відносного скорочення обсягу вимірювань:

$$E = \left(1 - \frac{N^*}{N_{\text{grid}}}\right) \times 100\%, \quad (2.28)$$

де N_{grid} – кількість конфігурацій повного перебору, визначена у підрозділі 2.2.

Слід підкреслити методологічний зв'язок між відкаліброваністю невизначеності, обґрунтованою у підрозділі 2.5, і коректністю критерію активного навчання. Стратегія (2.26) спирається на припущення, що прогнозна дисперсія достовірно відображає брак інформації в тій чи іншій області. Якби оцінка невизначеності була невідкаліброваною, критерій спрямовував би вимірювання у неправильні області, і активне навчання втратило б ефективність. Таким чином, висока відкаліброваність моделі є не лише самостійною метою, а й необхідною умовою працездатності механізму скорочення обсягу вимірювань.

Отже, критерій активного навчання за максимальною прогноною дисперсією разом із його пакетним узагальненням спрямовує обмежений бюджет вимірювань в області найбільшої невизначеності, а відкаліброваність невизначеності є передумовою коректності критерію, тож сукупна дія планування вибірки та активного поповнення забезпечує скорочення обсягу натурних вимірювань порівняно з повним перебором.

2.7. Метод автоматизованого конфігурування обчислювальної системи на основі сурогатної моделі

Розглянуті у попередніх підрозділах складові – формалізація простору «конфігурація – навантаження» (підрозділ 2.2), апарат гаусівської процесової регресії (підрозділ 2.3), конструювання ознак на основі фізичної моделі системи (підрозділ 2.4), калібрування довірчих інтервалів (підрозділ 2.5) та критерій активного навчання (підрозділ 2.6) – самі по собі є необхідними, але не достатніми елементами розв'язання задачі автоматизованого проєктування. Їхня наукова цінність повністю розкривається лише в інтеграції, коли вони

утворюють замкнений технологічний цикл, що перетворює обмежений набір натурних вимірювань на діючий інструмент прийняття проєктних рішень. Метою цього підрозділу є узагальнення окремих складових у цілісний метод автоматизованого конфігурування обчислювальної системи, який реалізує обидві вимоги постановки (2.1) – високу точність апроксимації нелінійної поверхні відгуку та відкаліброване оцінювання невизначеності – і забезпечує скорочення обсягу натурних вимірювань згідно з (2.28).

Метод у цілому є композицією послідовно застосовуваних перетворень, яку у стислій формі можна записати як відображення від бюджету вимірювань до відкаліброваної сурогатної моделі та похідних від неї проєктних рішень:

$$\mathcal{M}: (\mathcal{D}, B) \mapsto \left(\hat{f}: (x, l) \mapsto (\mu, \sigma) \right) \mapsto x^{ot}, \quad (2.29)$$

де $\mathcal{D}$ – допустима область простору параметрів (2.5), B – відведений бюджет натурних вимірювань, $\hat{f}$ – побудована сурогатна модель, а x^{ot} – рекомендована конфігурація системи. Розгортання цієї композиції у послідовність технологічних етапів утворює основу методу, узагальнену схему якого наведено на рисунку 2.3.

Перший етап методу полягає у формалізації простору вхідних змінних відповідно до підрозділу 2.2: визначенні керованих параметрів конфігурації, зовнішніх параметрів навантаження, їхніх діапазонів і допустимої області. На цьому етапі також фіксують цільову метрику продуктивності (p95-затримку) та порогові значення точності й відкаліброваності, що входять до задачі (2.2). Коректна формалізація простору є визначальною, оскільки саме вона задає межі, у яких сурогатна модель зберігає достовірність, і масштаб економії, обчислюваний відносно повного перебору сітки.

Другий етап – планування початкової навчальної вибірки за схемою латинського гіперкубічного вибору, що забезпечує рівномірне заповнення багатовимірного простору параметрів мінімальною кількістю точок [106]. На відміну від повного перебору сітки, який зростає мультиплікативно з розмірністю, латинський гіперкубічний план дозволяє почати побудову моделі з невеликої, але репрезентативної вибірки, що є першим джерелом скорочення

обсягу вимірювань. Обрані на цьому етапі конфігурації підлягають натурному навантажувальному тестуванню, у результаті якого отримують значення p95-затримки для кожної точки початкової вибірки.

Третій етап охоплює підготовку даних до навчання: конструювання ознак на основі фізичної моделі черги (підрозділ 2.4), логарифмічне перетворення цільової змінної (2.7) для нормалізації розподілу затримок та стандартизацію ознак (2.8) для узгодження їхніх масштабів. Саме на цьому етапі первинний шестивимірний простір параметрів перетворюється на розширений простір ознак (2.22), у якому нелінійна поверхня відгуку набуває форми, придатної для ефективної апроксимації гладким ядром. Значущість цього етапу важко переоцінити: як показано у підрозділі 2.4, без конструйованих ознак, і насамперед без ознаки інтенсивності навантаження, залежність продуктивності від параметрів залишається прихованою у складній взаємодії первинних змінних, і жодне ядро не здатне відтворити її з тією самою точністю.

Четвертий етап – власне навчання сурогатної моделі методом гаусівської процесової регресії (підрозділ 2.3). Він включає вибір коваріаційної функції з обґрунтуванням на користь ядра Матерн $5/2$, автоматичну оптимізацію гіперпараметрів ядра та параметра згладжування шуму шляхом максимізації маргінальної правдоподібності (2.13), а також обчислення апостеріорного прогнозного розподілу (2.11), (2.12) для будь-якої точки простору. Результатом етапу є модель, що для кожної пари «конфігурація – навантаження» повертає прогнозоване значення p95-затримки разом з оцінкою невизначеності цього прогнозу.

П'ятий етап – перевірка відкаліброваності побудованої моделі засобами діаграми надійності та метрики очікуваної похибки калібрування (підрозділ 2.5). Цей етап є контрольною точкою методу: лише за умови підтвердженої відкаліброваності оцінка невизначеності може використовуватися для подальших рішень. Якщо калібрування виявляється незадовільним, застосовують коригування діапазону параметра згладжування шуму або процедури апостеріорного калібрування.

Шостий етап реалізує механізм активного навчання (підрозділ 2.6). За побудованою моделлю оцінюють досягнутий рівень точності, і якщо він не відповідає цільовому порогу, за критерієм максимальної дисперсії (2.26) обирають наступні конфігурації для вимірювання, проводять для них натурні тести, поповнюють навчальну вибірку й повертаються до етапів підготовки даних і навчання. Цей цикл повторюється доти, доки не буде досягнуто цільової точності або вичерпано бюджет вимірювань. Ітеративне поповнення вибірки саме в областях найбільшої невизначеності є другим і головним джерелом скорочення обсягу натурних вимірювань, оскільки дозволяє досягти цільової точності за мінімальної кількості спостережень N^* згідно з (2.2), (2.28).

Сьомий, завершальний етап полягає у використанні готової відкаліброваної сурогатної моделі для розв'язання прикладних задач проектування. Основною з них є оптимізація конфігурації – пошук такої комбінації керованих параметрів, яка забезпечує задану цільову продуктивність за мінімальних витрат ресурсів. Ця задача формалізується як умовна оптимізація:

$$\mathbf{x}^{\text{ot}} = \arg \min_{\mathbf{x} \in D} \text{Cost}(\mathbf{x}) \text{ за умови } \mu(\mathbf{x}, l) + z_\gamma \sigma(\mathbf{x}, l) \leq T, \quad (2.30)$$

де $\text{Cost}(\mathbf{x})$ – функція вартості ресурсів конфігурації, T – цільове (гранично допустиме) значення р95-затримки, l – очікуваний або найгірший профіль навантаження, а D – область допустимих конфігурацій. Принципово важливим у формулюванні (2.30) є те, що обмеження накладається не на точковий прогноз μ , а на його верхню довірчу межу $\mu + z_\gamma \sigma$. Отже, метод обирає конфігурацію, яка задовольняє цільову затримку з урахуванням невизначеності прогнозу, тобто є робастною до похибки моделі. Такий підхід безпосередньо використовує відкалібровану оцінку невизначеності як інструмент прийняття рішень: у погано досліджених областях простору, де σ велика, метод діє обережно, вимагаючи більшого запасу за затримкою, а в добре вивчених областях – наближає обрану конфігурацію до теоретичного оптимуму.

Оскільки сурогатна модель дозволяє обчислювати прогноз для довільної конфігурації без натурального вимірювання практично миттєво, задачу (2.30) розв'язують суцільним перебором або чисельною оптимізацією у просторі

конфігурацій, що є обчислювально дешевим порівняно з натурними випробуваннями. Побудована модель також дає змогу оцінювати чутливість продуктивності до окремих параметрів, обґрунтовувати рішення щодо обмеження інтенсивності запитів і розміру пулу з'єднань.

Запропонований метод вигідно вирізняється на тлі наявних підходів до налаштування конфігурованих систем. Адитивні моделі впливу забезпечують інтерпретованість, проте принципово не відтворюють нелінійну взаємодію параметрів і у базовій формі не надають оцінки невизначеності [142, 42]. Сучасні методи налаштування, що спираються на причинно-наслідковий аналіз конфігурацій, підвищують ефективність пошуку оптимуму, проте зосереджені саме на оптимізації, а не на побудові універсальної відкаліброваної моделі всієї поверхні відгуку [33]. Порівняння запропонованого методу з наявними підходами до моделювання продуктивності за ключовими властивостями наведено в таблиці 2.5. Похідні застосування показано на рисунку 2.3 як напрями використання готової сурогатної моделі.

Таблиця 2.5

Порівняння підходів до моделювання продуктивності конфігурованих систем

Властивість	Адитивні моделі впливу	Дерева рішень, бустинг	GPR-сурогат (метод)
Нелінійна поверхня відгуку	ні	так	так
Оцінка невизначеності (довірчі інтервали)	лише в розширеннях	ні	так
Відкаліброваність довірчих інтервалів	не перевіряється	не застосовна	так, перевірено
Активне навчання за невизначеністю	ні	обмежено	так
Робастна оптимізація конфігурації	ні	ні	так
Інтерпретованість	висока	середня (SHAP)	висока (SHAP + фізичні ознаки)

Запропонований метод, натомість, поєднує в одному технологічному циклі чотири властивості, які раніше розглядалися окремо: апроксимацію нелінійної поверхні відгуку у повному просторі «конфігурація – навантаження», відкалібровану кількісну оцінку невизначеності, активне скорочення обсягу

натурних вимірювань і робастну оптимізацію конфігурації з урахуванням невизначеності.

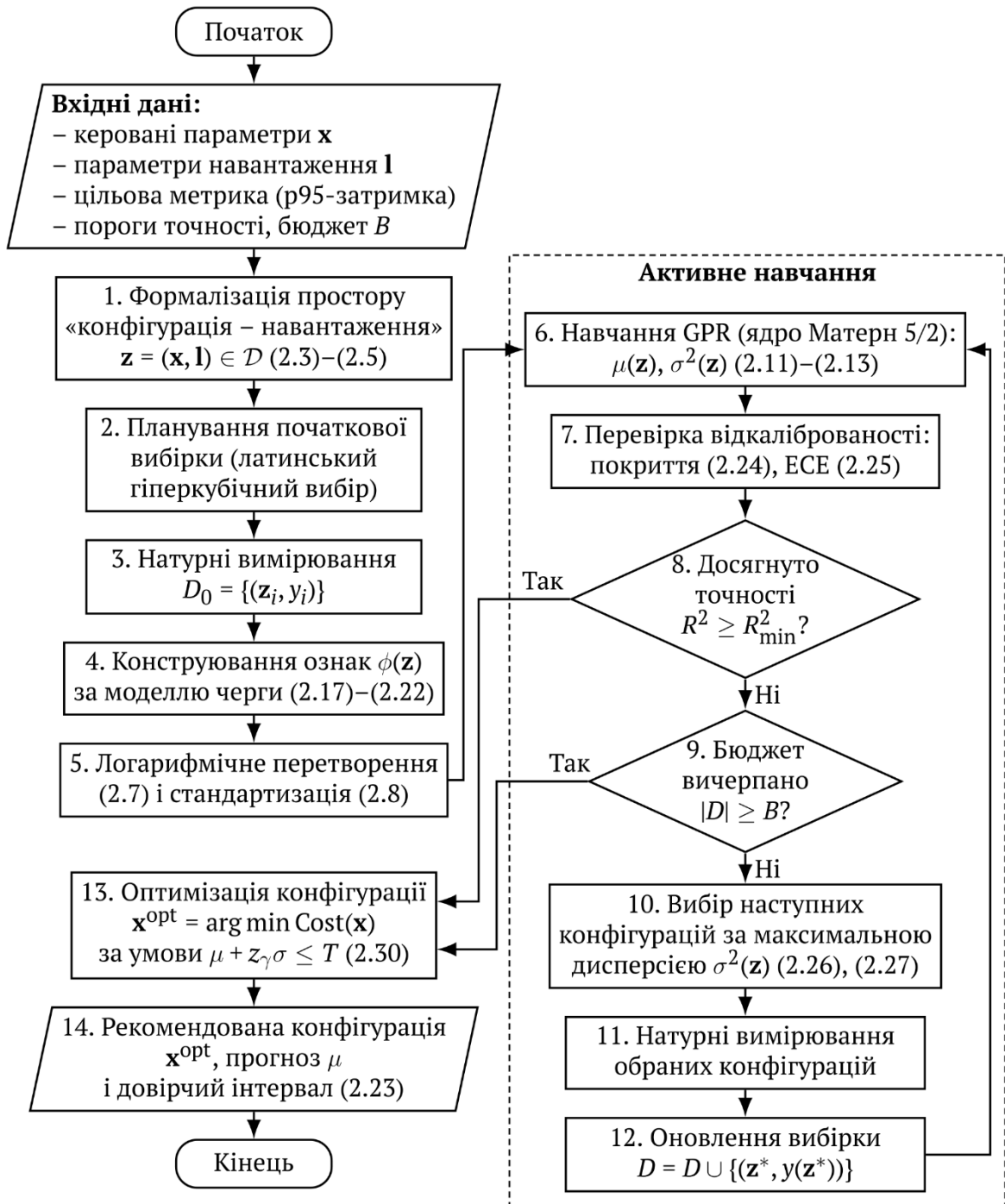


Рис. 2.3. Узагальнена схема методу автоматизованого конфігурування обчислювальної системи на основі сурогатної моделі гаусівської процесової регресії

Перспективним напрямом розвитку методу є перенесення побудованої сурогатної моделі між різними апаратними та програмними середовищами

засобами трансферного навчання, що дозволило б повторно використовувати накопичені вимірювання й додатково скоротити витрати на калібрування моделі для нових середовищ [76, 80].

Висновки до розділу 2

У розділі викладено теоретичні засади методу автоматизованого проектування (конфігурування) обчислювальних систем на основі сурогатної моделі гаусівської процесової регресії, який замінює ресурсомісткий повний перебір простору конфігурацій відкаліброваною статистичною моделлю продуктивності. Основні результати розділу полягають у такому:

1. Задачу сурогатного моделювання продуктивності сформульовано як побудову відображення пари «конфігурація – навантаження» у прогноз цільової метрики разом з оцінкою його невизначеності, а мету методу – як пошук мінімальної кількості натурних вимірювань, достатньої для заданих порогів точності та відкаліброваності.

2. Формалізовано простір вхідних змінних із розмежуванням керованих параметрів конфігурації та зовнішніх параметрів навантаження й обґрунтовано мультиплікативну модель спостережень, логарифмічне перетворення цільової змінної та стандартизацію ознак як передумови коректного застосування гаусівської процесової регресії.

3. Обґрунтовано вибір гаусівської процесової регресії як апарату, що поєднує точковий прогноз із його дисперсією; основним обрано ядро Матерн $5/2$ з огляду на фізичну відповідність параметра гладкості характеру поверхні відгуку, а для стійкості на малих вибірках застосовано ізотропну параметризацію.

4. Розроблено конструювання ознак на основі моделі черги масового обслуговування, центральною ознакою якого є інтенсивність навантаження як конструктивний аналог коефіцієнта завантаженості; показано, що визначальним

для продуктивності є співвідношення навантаження й потужності, чим пояснюється принципова непридатність адитивних моделей впливу.

5. Відкаліброваність довірчих інтервалів сформульовано як самостійну вимогу до сурогатної моделі з інструментами перевірки – емпіричним покриттям, діаграмою надійності та очікуваною похибкою калібрування, а визначальну роль у її забезпеченні відведено параметру згладжування шуму.

6. Запропоновано критерій активного навчання за максимальною прогнозною дисперсією з пакетним узагальненням, що враховує просторову різноманітність, і показано, що його коректність ґрунтується на відкаліброваності невизначеності.

7. Складові узагальнено в цілісний метод із семи технологічних етапів, завершений робастною оптимізацією конфігурації, у якій обмеження за цільовою затримкою накладається на верхню довірчу межу прогнозу, а не на точкову оцінку.

Наукова новизна результату полягає в поєднанні в одному технологічному циклі апроксимації нелінійної поверхні відгуку у просторі «конфігурація – навантаження», відкаліброваної оцінки невизначеності, активного скорочення обсягу натурних вимірювань і робастної оптимізації конфігурації, завдяки чому вибір конфігурації обчислювальної системи переходить від натурних випробувань до обчислювально дешевого аналізу за сурогатною моделлю.

РОЗДІЛ 3. ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ЗАБЕЗПЕЧЕННЯ ФУНКЦІОНАЛЬНОЇ СТІЙКОСТІ АДАПТИВНОЇ КОРПОРАТИВНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

3.1. Постановка задачі забезпечення функціональної стійкості

Корпоративна інформаційна система функціонує в середовищі, де змінюються інтенсивність звернень і сценарії поведінки користувачів, а обчислювальні ресурси зазнають часткових відмов і цілеспрямованих деструктивних впливів. Навіть за високої безвідмовності окремих вузлів система в цілому може втрачати працездатність унаслідок непередбачених сплесків навантаження, поступової зміни поведінки користувачів чи узгодженої атаки, тому предметом розділу є функціональна стійкість – здатність інформаційної системи зберігати працездатність і встановлений рівень якості обслуговування в умовах збурень різної природи за рахунок внутрішньої перебудови ресурсів і режимів роботи [17, 93].

На відміну від надійності, відмовостійкості та живучості, функціональна стійкість свідомо допускає деградацію окремих компонентів і передбачає активне, кероване в режимі реального часу пристосування системи до збурення шляхом перерозподілу ресурсів, зміни маршрутів інформаційних потоків та коригування політик безпеки [69, 104]. Саме така активна адаптивність відрізняє функціонально стійку систему від просто надійної і є центральною ідеєю запропонованої інформаційної технології.

Об'єктом розгляду є адаптивна корпоративна інформаційна система, під якою розуміють програмно-апаратний комплекс, що обслуговує розподілений потік запитів користувачів і здатний змінювати конфігурацію обчислювальних ресурсів, маршрути інформаційних потоків та політики безпеки без припинення надання послуг. Типова корпоративна платформа охоплює контур користувацьких сервісів, контур інтеграції із зовнішніми системами та контур

нагляду й безпеки, кожен з яких робить внесок у сукупну функціональність і водночас є потенційним місцем виникнення збурень. Адаптивність такої системи означає наявність зворотного зв'язку, що на основі поточних вимірювань стану ухвалює рішення про перебудову й реалізує їх у реальному часі, не перериваючи обслуговування користувачів. Стан такої системи у кожен момент часу t зручно характеризувати інтегральним показником якості функціонування, який відображає частку корисної функціональності, що фактично надається користувачам, відносно номінального рівня:

$$Q(t) = \frac{\Phi(t)}{\Phi_{\text{ном}}}, Q(t) \in [0,1], \quad (3.1)$$

де $\Phi(t)$ – фактично забезпечувана функціональність, тобто обсяг своєчасно й коректно обслугованих запитів із дотриманням політик безпеки, $\Phi_{\text{ном}}$ – її номінальне значення за відсутності збурень. Значення $Q(t) = 1$ відповідає повноцінному функціонуванню, $Q(t) = 0$ – повній втраті працездатності, а проміжні значення – частковій деградації. Показник (3.1) є безрозмірним і зводить різноманітні прояви збурень – зростання затримки відповіді, відмову в обслуговуванні, зниження рівня захищеності – до єдиної шкали, придатної для кількісного оцінювання стійкості й порівняння різних конфігурацій системи.

За своєю природою якість функціонування корпоративної системи визначається двома взаємопов'язаними, але відмінними аспектами: продуктивнісним, тобто здатністю своєчасно обслуговувати наявний потік запитів наявними ресурсами, і безпековим, тобто здатністю зберігати захищеність і цілісність обслуговування під впливом загроз. Тому показник якості доцільно подати у вигляді добутку двох складових:

$$Q(t) = Q_{\text{perf}}(t) \cdot Q_{\text{sec}}(t), \quad (3.2)$$

де $Q_{\text{perf}}(t)$ – продуктивнісна складова, що спадає за перевищення попиту доступної потужності оброблення, а $Q_{\text{sec}}(t)$ – безпекова складова, що спадає за зниження рівня захищеності внаслідок реалізації загроз. Мультиплікативна форма (3.2) відображає ту обставину, що втрата функціональності настає як за вичерпання продуктивності, так і за порушення безпеки: достатньо істотного

спаду будь-якої зі складових, щоб система в цілому втратила корисне функціонування. Такий поділ дозволяє надалі розмежувати механізми протидії продуктивнісним і безпековим збуренням, зберігаючи єдину інтегральну оцінку стану системи.

Збурення, що загрожують функціональній стійкості корпоративної інформаційної системи, доцільно розмежувати за природою й характером впливу на показник якості. Змінне навантаження та зміни поведінки користувачів діють на систему з боку входу, підвищуючи інтенсивність потоку запитів і змінюючи його структуру; кіберзагрози діють на безпекову складову, знижуючи довіру до системи й спричиняючи додаткові затримки та відмови; відмови компонентів діють на ресурсний контур, зменшуючи доступну потужність оброблення. Спільним для всіх типів збурень є те, що вони виводять систему з номінального режиму й спричиняють тимчасове або тривале зниження показника $Q(t)$, проте механізми протидії кожному типу різні. Класифікацію збурень, характер їхнього первинного впливу та відповідну складову технології, що протидіє кожному типу, наведено в таблиці 3.1.

Таблиця 3.1

Класифікація збурень функціональної стійкості корпоративної інформаційної системи

Тип збурення	Джерело	Первинний вплив на систему	Протидійна складова технології
Змінне навантаження	зовнішнє середовище	зростання інтенсивності потоку запитів	прогнозування та проактивне резервування
Зміна поведінки користувачів	користувачі	відхилення профілю активності від типового	статистичний моніторинг, машинне навчання
Кіберзагрози	зловмисник	зниження захищеності, відмови в обслуговуванні	оцінювання ризиків, активний контроль загроз
Відмова компонента	внутрішні збої	зменшення доступної потужності оброблення	оптимізація потоків, перемаршрутизація

Функціональну стійкість системи на заданому горизонті спостереження T кількісно оцінюють інтегральним показником – усередненим у часі значенням якості функціонування, яке далі позначаємо як індекс функціональної стійкості:

$$FSI = \frac{1}{T} \int_0^T Q(t) dt \quad (3.3)$$

Індекс (3.3) набуває значень у діапазоні від нуля до одиниці й тим ближчий до одиниці, чим повніше система зберігає корисне функціонування протягом усього горизонту, включно з періодами збурень. На відміну від миттєвого показника $Q(t)$, індекс FSI характеризує систему інтегрально, згортаючи всю траєкторію якості в єдине число, і тому придатний для порівняння різних конфігурацій та технологій забезпечення стійкості за єдиним критерієм. Саме цей показник обрано за основну кількісну міру ефективності запропонованої технології, а його приріст порівняно з базовою конфігурацією – за головний практичний результат.

Окремий інтерес становить поведінка системи безпосередньо під час збурення, коли показник якості відхиляється від номінального рівня, а потім поступово відновлюється. Величину втраченої функціональності за час дії окремого збурення, що почалося у момент t_0 і триває впродовж інтервалу τ , оцінюють площею провалу показника якості нижче номінального рівня:

$$\Theta = \int_{t_0}^{t_0+\tau} (Q_{\text{ном}} - Q(t)) dt. \quad (3.4)$$

Геометрично величина (3.4) відповідає площі так званого трикутника стійкості – області між номінальним рівнем і фактичною траєкторією $Q(t)$ протягом збурення та наступного відновлення [67, 27]. Ця міра акумулює в собі дві істотні характеристики збурення одночасно: глибину провалу, тобто наскільки сильно впала якість у найгіршій точці, і його тривалість, тобто як довго система перебувала у зниженому режимі. Дві системи з однаковою глибиною провалу, але різним часом відновлення матимуть різну площу деградації, і стійкішою вважається та, що повертається до норми швидше. Тривалість відновлення, тобто час T_{rec} від початку збурення до повернення показника якості

у припустимий окіл номінального рівня, є самостійною характеристикою стійкості й безпосередньо визначає економічні наслідки збурення, оскільки триваліший простій у корпоративній системі означає більші фінансові втрати та репутаційну шкоду.

Виходячи з наведених означень, вимогу функціональної стійкості формалізують як сукупність умов, що обмежують допустиму деградацію системи під збуреннями. Ця вимога полягає в утриманні показника якості не нижче припустимого порогу протягом усього горизонту спостереження та в обмеженні часу відновлення після кожного збурення:

$$Q(t) \geq Q_{min} \text{ для всіх } t \in [0, T], T_{rec} \leq T_{max}, \quad (3.5)_{min}$$

де Q_{min} – мінімально припустимий рівень якості функціонування, наприклад визначена в угоді про рівень обслуговування частка запитів, оброблених вчасно й захищено, а T_{max} – гранично допустимий час відновлення після збурення. Умови (3.5) відображають дві взаємодоповняльні вимоги: система не повинна занадто глибоко втрачати функціональність під час збурення й не повинна занадто довго перебувати у зниженому режимі. Порушення хоча б однієї з цих умов свідчить про недостатню функціональну стійкість і є підставою для впровадження відповідних інформаційних технологій, здатних активно протидіяти збуренням. Пороги Q_{min} і T_{max} визначаються вимогами конкретної корпоративної платформи й закріплюються в угодах про рівень обслуговування, тому запропонована технологія має забезпечувати їх виконання за будь-яких очікуваних профілів збурень.

Забезпечення функціональної стійкості неможливо звести до окремого технічного заходу: воно вимагає узгодженої дії кількох механізмів, що безперервно спостерігають за станом системи, розпізнають збурення, прогнозують їхній розвиток і своєчасно перебудовують ресурси та політики безпеки. Такий замкнений контур керування узгоджується із загальною парадигмою автономних і самоадаптивних обчислювальних систем, у якій виокремлюють послідовні етапи моніторингу стану, аналізу отриманих даних, планування коригувальних дій та їх виконання з опорою на спільну базу знань

про систему [82, 32]. Стосовно корпоративної інформаційної системи цей контур конкретизується у взаємоузгоджену послідовність складових: статистичний моніторинг стану й виявлення аномалій, що виявляє факт і момент збурення; прогнозування поведінки користувачів методами машинного навчання, що дозволяє передбачити зміну навантаження й діяти проактивно; оптимізацію інформаційних потоків і ресурсів, що перерозподіляє обмежену потужність на користь критичних сервісів; та оцінювання ризиків і рівня інформаційної безпеки, що керує захисними заходами й розподілом бюджету. Узагальнену блок-схему цього контуру з позначенням характеру зв'язків між складовими наведено на рисунку 3.1.

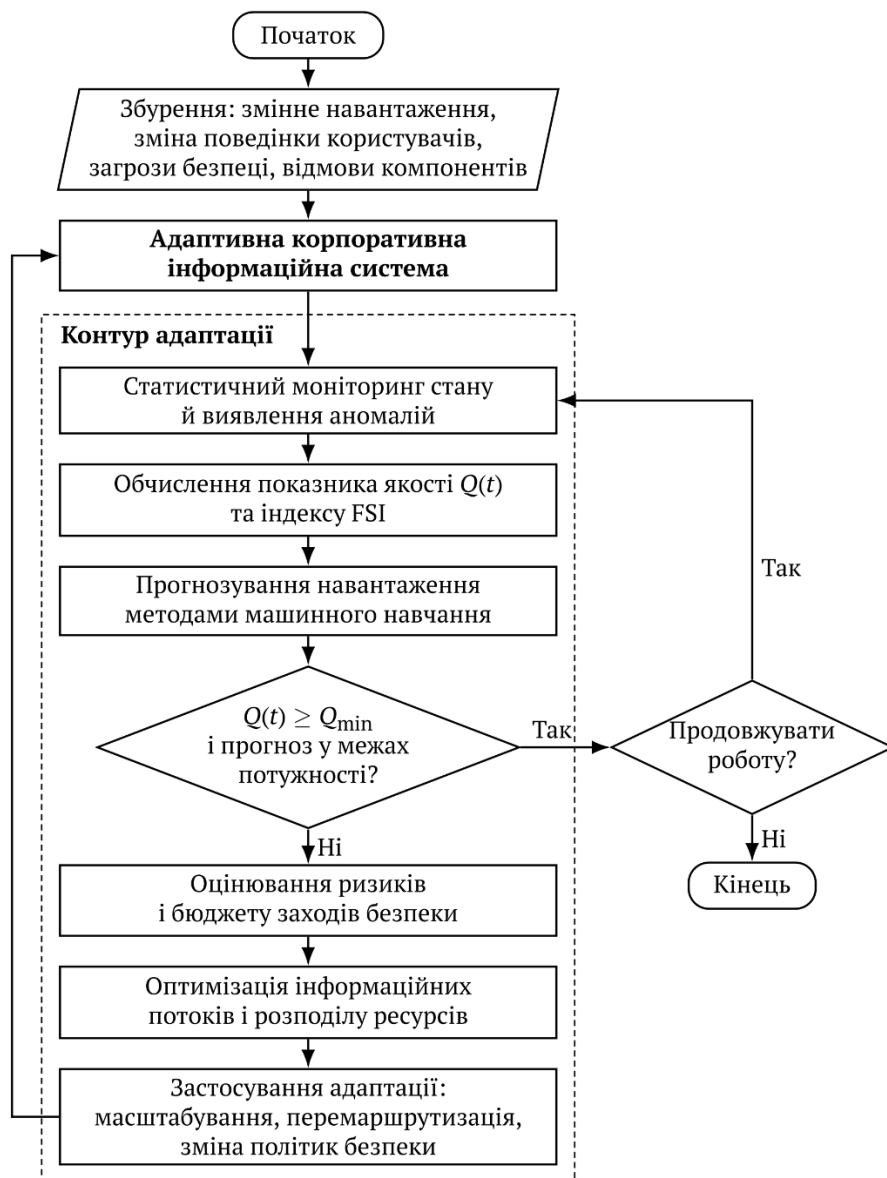


Рис. 3.1. Блок-схема контуру забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи

Наявність окресленого контуру складових дозволяє сформулювати задачу цього розділу як задачу синтезу інформаційної технології забезпечення функціональної стійкості. Нехай M – шукана інформаційна технологія, що об'єднує перелічені складові та їхні параметри в єдиний контур адаптації, $FSI(M)$ – досягнутий за цією технологією індекс функціональної стійкості (3.3), а $C(M)$ – сукупні витрати на її впровадження та експлуатацію. Тоді задача полягає у доборі такої технології, яка максимізує функціональну стійкість системи за обмеження на допустимі витрати:

$$M^* = \arg \max_M FSI(M) \text{ за умови } C(M) \leq C_{\text{доп}}, \quad (3.6)$$

де $C_{\text{доп}}$ – гранично допустимий бюджет. Постановка (3.6) підкреслює принципову особливість задачі: функціональну стійкість неможливо нарощувати необмежено, оскільки кожна складова технології споживає обчислювальні й фінансові ресурси, а віддача від додаткових витрат спадає після досягнення певного рівня захищеності. Тому шукається не максимально можлива, а економічно раціональна стійкість, що задовольняє вимоги (3.5) за прийнятних витрат. Кількісне розв'язання компромісу «стійкість – витрати», зокрема визначення критичного рівня витрат на безпеку та оптимального розподілу бюджету між захисними заходами й резервуванням, спирається на економічну модель [167].

Сформульована задача (3.6) конкретизує другий науковий результат дисертації – удосконалену інформаційну технологію забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи, та розвинену безпековою й економічною складовою. На відміну від наявних підходів, що розглядають окремі аспекти – підвищення захищеності або оптимізацію витрат – ізольовано, запропонована технологія інтегрує статистичний моніторинг аномалій, прогнозування поведінки методами машинного навчання, оптимізацію інформаційних потоків та оцінювання ризиків у єдиний контур адаптації в режимі реального часу, що й забезпечує одночасне досягнення продуктивнісної та безпекової складових якості

функціонування (3.2). Сукупність показників, за якими надалі оцінюють досягнення функціональної стійкості, узагальнено в таблиці 3.2.

Таблиця 3.2

Показники оцінювання функціональної стійкості

Показник	Позначення	Що характеризує	Напрямок кращого значення
Якість функціонування	$Q(t)$	миттєва частка збереженої функціональності	ближче до 1
Індекс функціональної стійкості	FSI	усереднена в часі якість функціонування	більше, до 1
Площа деградації	Θ	втрачена функціональність за час збурення	менше, до 0
Час відновлення	T_{rec}	тривалість повернення до номінального режиму	менше
Сукупні витрати	C	вартість упровадження та експлуатації технології	у межах бюджету

Отже, задачу забезпечення функціональної стійкості формалізовано через інтегральний показник якості функціонування з продуктивнісною та безпековою складовими, на основі якого побудовано взаємодоповняльні міри стійкості – індекс функціональної стійкості, площу деградації та час відновлення, а саму задачу подано як синтез економічно раціональної інформаційної технології, що максимізує стійкість за обмеження на витрати.

3.2. Математична модель функціональної стійкості

Постановка (3.6) вимагає кількісного критерію, який згорнув би різномірні аспекти функціональної стійкості – продуктивність, захищеність, ризик і здатність до відновлення – в єдину скалярну величину, придатну для оптимізації на етапі проєктування та керування. Такий критерій побудовано на основі математичної моделі функціональної стійкості адаптивної корпоративної системи. Модель оперує чотирма інтегральними змінними стану системи: функціональністю T , що відображає повноту й своєчасність обслуговування запитів; рівнем безпеки S , що характеризує захищеність системи від реалізації

загроз; рівнем ризику R , що узагальнює очікувані втрати від можливих загроз; та адаптивністю A , що описує здатність системи відновлювати функціонування після збурень. Ці чотири змінні охоплюють обидві складові якості функціонування, введені у співвідношенні (3.2): функціональність і адаптивність стосуються переважно продуктивнісної складової, а безпека й ризик – безпекової.

Цільову функцію функціональної стійкості подають як зважену згортку чотирьох змінних, у якій корисні властивості – функціональність, безпека та адаптивність – входять зі знаком плюс, а ризик як небажаний чинник – зі знаком мінус:

$$\max F(T, S, R, A) = \alpha_1 T + \alpha_2 S - \alpha_3 R + \alpha_4 A, \quad (3.7)$$

де $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ – вагові коефіцієнти, що відображають відносну важливість кожної складової для конкретної корпоративної платформи. Вагові коефіцієнти нормують так, щоб їхня сума дорівнювала одиниці, а самі вони були невід’ємними:

$$\sum_{i=1}^4 \alpha_i = 1, \alpha_i \geq 0. \quad (3.8)$$

Нормування (3.8) надає цільовій функції зміст зваженого середнього й забезпечує зіставність значень F для систем різного масштабу. Конкретні значення ваг визначаються пріоритетами організації: у дослідженні прийнято розподіл, за якого провідну роль відведено функціональності, а безпеці й ризику надано однакову помірну вагу за меншого внеску адаптивності, що відповідає типовим вимогам корпоративних освітніх платформ до безперервності обслуговування. Максимізація функції (3.7) означає пошук такого режиму роботи й розподілу ресурсів, за якого система одночасно надає якнайповнішу функціональність, підтримує достатній рівень безпеки та адаптивності й утримує ризик на низькому рівні.

Функціональність системи визначають як нормовану міру повноти обслуговування критично важливих функцій із урахуванням їхньої відносної значущості. Якщо система надає k критичних функцій, кожна з яких

характеризується часткою успішного виконання φ_j та вагою важливості w_j , то функціональність обчислюють як зважене середнє:

$$T = \frac{\sum_{j=1}^k w_j \varphi_j}{\sum_{j=1}^k w_j}, T \in [0,1]. \quad (3.9)$$

Показник (3.9) набуває значення одиниці, коли всі критичні функції виконуються повністю, і спадає в міру того, як окремі функції деградують або стають недоступними; вага w_j гарантує, що втрата важливіших функцій знижує T сильніше, ніж втрата другорядних. Така побудова узгоджується з декомпозицією якості (3.2): функціональність T є інтегральним відображенням продуктивнісної складової Q_{perf} на множині критичних сервісів системи.

Рівень ризику узагальнює очікувані втрати від можливих загроз і обчислюється як сума добутків імовірності реалізації кожної загрози на її вплив, обмежена зверху допустимим значенням:

$$R = \sum_{i=1}^m P_i I_i \leq R_{\text{max}}, \quad (3.10)$$

де P_i – імовірність реалізації i -ї загрози, I_i – її вплив у вигляді очікуваних втрат, m – кількість розглянутих загроз, R_{max} гранично допустимий сукупний ризик. Співвідношення (3.10) є водночас визначенням ризику й обмеженням на нього: сукупний ризик не повинен перевищувати межу R_{max} а якої безпека системи вважається прийнятною. Деталізацію ймовірності реалізації загрози через імовірність загрози, вразливість і рівень контролю, а також динаміку залишкового ризику залежно від витрат на безпеку розкрито у підрозділі 3.6.

Адаптивність характеризує здатність системи відновлювати функціонування після збурень і природно виражається через уведену в підрозділі 3.1 площу деградації. Що менша втрачена під час збурення функціональність, то вища адаптивність, тому адаптивність визначають як доповнення нормованої площі деградації до одиниці:

$$A = 1 - \frac{\Theta}{Q_{\text{ном}} \tau}, A \in [0,1], \quad (3.11)$$

де Θ – площа деградації (3.4), τ – тривалість збурення, $Q_{\text{ном}}$ – максимально можлива втрата функціональності за повного простою протягом усього

збурення. Значення $A = 1$ відповідає системі, що зовсім не втрачає функціональності під збуренням або відновлюється миттєво, а $A = 0$ – системі, що повністю простоє протягом усього збурення. Означення (3.11) безпосередньо пов’язує адаптивність із часовою поведінкою показника якості й робить її вимірюваною величиною. Змінні моделі, їхні позначення та фізичний зміст узагальнено в таблиці 3.3.

Таблиця 3.3

Змінні математичної моделі функціональної стійкості

Змінна	Позначення	Фізичний зміст	Діапазон
Функціональність	T	повнота й своєчасність обслуговування критичних функцій	від 0 до 1
Рівень безпеки	S	захищеність системи від реалізації загроз	від 0 до 1
Рівень ризику	R	очікувані сукупні втрати від можливих загроз	від 0 до R_{max}
Адаптивність	A	здатність відновлювати функціонування після збурень	від 0 до 1
Вагові коефіцієнти	$\alpha_1 \dots \alpha_4$	відносна важливість складових цільової функції	сума дорівнює 1

Максимізація цільової функції (3.7) відбувається не безумовно, а за системи обмежень, що задають допустимі межі роботи системи й відображають реальні ресурсні та безпекові вимоги. Перше обмеження вже наведено у складі визначення ризику (3.10) і вимагає, щоб сукупний ризик не перевищував межу R_{max} руге й третє обмеження встановлюють мінімально припустимі рівні безпеки та адаптивності, нижче яких система вважається функціонально нестійкою незалежно від значення інших складових:

$$S \geq S_{min}, A \geq A_{min}, \quad (3.12)$$

де S_{min} – мінімальний рівень безпеки, а A_{min} – мінімальний рівень адаптивності. Ці обмеження запобігають небажаним розв’язкам, за яких висока функціональність досягається ціною неприпустимого зниження захищеності або втрати здатності до відновлення; вони виражають принцип, згідно з яким функціональна стійкість вимагає одночасного, а не компенсаційного задоволення всіх складових.

Четверте обмеження є бюджетним і вимагає, щоб сумарні витрати на забезпечення функціональності, безпеки та управління ризиками не перевищували допустимого бюджету:

$$C_T + C_S + C_R \leq C_{\text{доп}}, \quad (3.13)$$

де C_T – витрати на підтримання функціональності, C_S – витрати на заходи безпеки, C_R – витрати на управління ризиками, $C_{\text{доп}}$ – гранично допустимий бюджет, узгоджений із постановкою (3.6). Обмеження (3.13) робить модель економічно змістовною: воно відображає обмеженість ресурсів організації й унеможливорює нарощування стійкості без урахування вартості, що узгоджується із загальним принципом раціональних інвестицій в інформаційну безпеку [62].

Сукупність обмежень моделі та їхній економічний і безпековий зміст наведено в таблиці 3.4.

Таблиця 3.4

Обмеження математичної моделі функціональної стійкості

Обмеження	Формула	Зміст	Наслідок порушення
Обмеження ризику	$R \leq R_{\text{max}}$	сукупний ризик у допустимих межах	неприйнятна ймовірність втрат
Мінімальна безпека	$S \geq S_{\text{min}}$	достатній рівень захищеності	вразливість до загроз безпеці
Мінімальна адаптивність	$A \geq A_{\text{min}}$	здатність відновлюватися	тривалі простої після збурень
Бюджетне обмеження	$C_T + C_S + C_R \leq C_{\text{доп}}$	витрати в межах бюджету	економічна недоцільність

Математична модель (3.7)–(3.13) є задачею умовної оптимізації, у якій керованими є параметри розподілу обчислювальних ресурсів та бюджету безпеки, а цільова функція й обмеження обчислюються за поточними вимірюваннями стану системи. Слід розмежувати два рівні застосування моделі. Цільова функція $F(T, S, R, A)$ є проєктною, агрегованою мірою: вона згортає стан системи в єдине число й слугує критерієм вибору конфігурації та розподілу ресурсів.

Уведені в підрозділі 3.1 показник якості $Q(t)$ та індекс функціональної стійкості FSI є, натомість, часовими мірами: вони описують фактичну траєкторію системи в режимі реального часу під дією збурень.

Послідовність операцій побудови рішення за математичною моделлю наведено на рисунку 3.2.

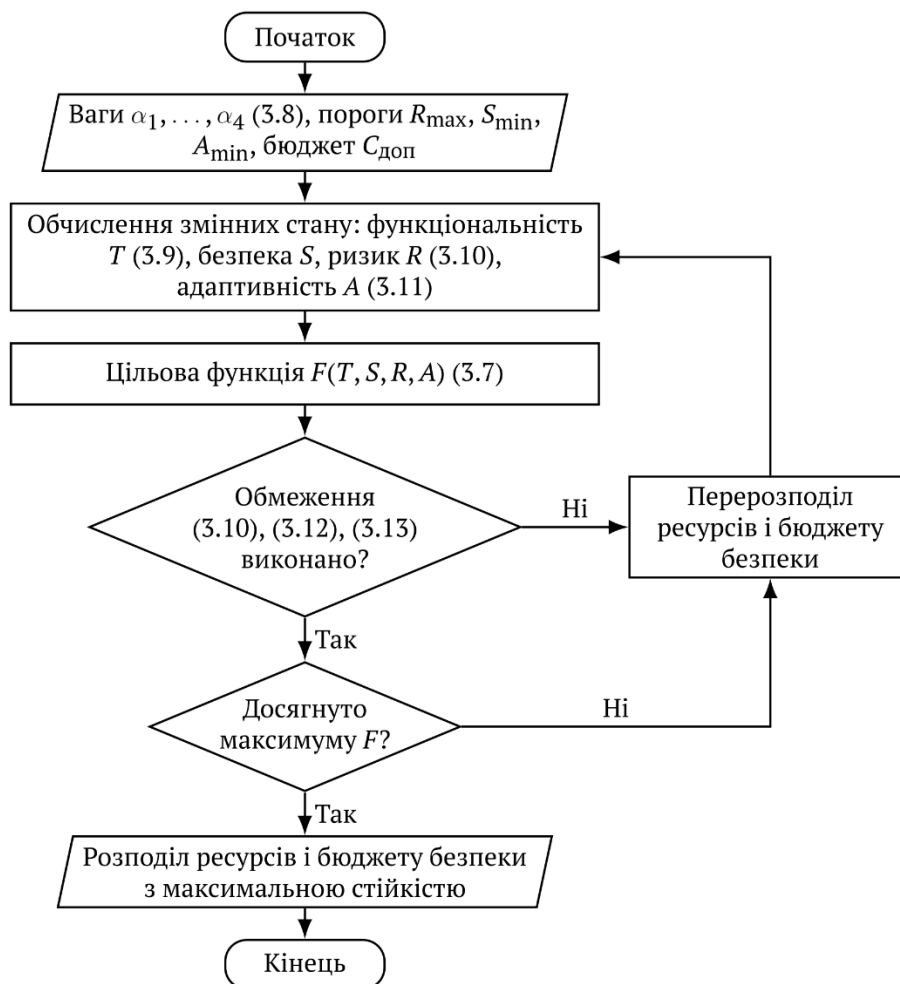


Рис. 3.2. Блок-схема побудови рішення за математичною моделлю функціональної стійкості

Таким чином, математична модель функціональної стійкості є задачею умовної оптимізації з цільовою функцією у вигляді нормованої зваженої згортки функціональності, безпеки, ризику й адаптивності та з обмеженнями на ризик, мінімальні рівні безпеки й адаптивності та бюджет. Означення адаптивності через площу деградації пов'язує модель із часовими мірами якості, тож максимізація цільової функції на етапі проєктування спрямована на підвищення індексу функціональної стійкості під час експлуатації.

3.3. Статистичний моніторинг стану системи та виявлення аномалій

Першою складовою інформаційної технології забезпечення функціональної стійкості є статистичний моніторинг стану системи, задача якого – своєчасно виявити факт і момент збурення, тобто відхилення поведінки системи від типової. Без такого виявлення решта складових технологій не можуть спрацювати вчасно, тому саме моніторинг замикає зворотний зв'язок контуру адаптації, поданого на рисунку 3.1. Основою моніторингу є апарат статистичного контролю процесів, у якому за спостережуваним показником будують контрольні межі, а вихід показника за ці межі трактують як сигнал про аномалію [114]. У корпоративній інформаційній системі таким показником слугує один із індикаторів якості обслуговування – наприклад, латентність відповіді або інтенсивність помилок, – що чутливо реагує на всі типи збурень: сплеск навантаження, кіберінцидент чи відмову компонента.

Пряме застосування класичних контрольних меж до показника корпоративної системи ускладнене тим, що його типова поведінка не є стаціонарною: інтенсивність звернень і похідна від неї латентність мають виражену добову й тижневу сезонність, тому фіксовані межі давали б хибні спрацювання під час звичайних денних піків. Щоб усунути цей вплив, показник попередньо позбавляють сезонної складової. Для цього за передісторією будують сезонний профіль – характерне значення показника для кожного слоту доби, обчислене як медіана спостережень цього слоту за всі доби, що є робастним до поодиноких збурень:

$$\mu_s = \text{med}\{X_t : s(t) = s\}, s = 0, \dots, L - 1, \quad (3.14)$$

де X_t – значення показника у момент t , $s(t)$ – номер слоту доби, до якого належить момент t , L – кількість слотів у добі. Відхилення поточного значення від сезонного профілю утворює залишок, що вже не містить регулярної добової складової й придатний для контролю:

$$r_t = X_t - \mu_{s(t)}. \quad (3.15)$$

Масштаб природної мінливості залишку оцінюють робастно, через медіану абсолютних відхилень, окремо для кожного слоту доби, оскільки в пікові години розкид показника більший, ніж у нічні:

$$\sigma_s = 1,4826 \cdot \text{med}\{|r_t - \text{med}(r)| : s(t) = s\}. \quad (3.16)$$

Множник у формулі (3.16) переводить медіану абсолютних відхилень у сумісну зі стандартним відхиленням шкалу за припущення про приблизно нормальний розподіл залишку, а робастність оцінки запобігає завищенню меж під впливом самих аномалій.

За обчисленими сезонним профілем і масштабом залишку будують контрольні межі та формують сигнал аномалії. Кандидатом на аномалію вважають відлік, залишок якого виходить за межу трьох стандартних відхилень, що відповідає класичному правилу контрольних карт і узгоджується з підходом до виявлення аномалій за правилом $\mu \pm 3\sigma$. Проте поодинокі перевищення меж може бути наслідком випадкового шуму, тому сигнал підтверджують правилом серій статистичного контролю процесів: аномалію фіксують лише тоді, коли межу перевищено щонайменше у двох послідовних відліках:

$$|r_t| > 3\sigma_s \text{ і } |r_{t-1}| > 3\sigma_s. \quad (3.17)$$

Правило серій (3.17) істотно зменшує частку хибних спрацювань порівняно з простим порогом, майже не втрачаючи чутливості до реальних збурень, оскільки справжні аномалії, на відміну від шумових викидів, тривають кілька відліків поспіль [114]. Затримка виявлення при цьому залишається малою, бо для підтвердження достатньо двох послідовних перевищень.

Поряд із контролем за межами 3σ застосовують дисперсійний аналіз, який виявляє не поодинокі викиди, а зсув режиму роботи – поступову зміну середнього рівня показника, характерну для повільно розгортуваних збурень. Для цього суміжні часові вікна залишку розглядають як групи спостережень і перевіряють гіпотезу про рівність їхніх середніх за критерієм Фішера, обчислюючи відношення міжгрупової дисперсії до внутрішньогрупової:

$$F = \frac{MSB}{MSW} = \frac{\frac{1}{g-1} \sum_{i=1}^g n_i (\bar{X}_i - \bar{X})^2}{\frac{1}{N-g} \sum_{i=1}^g \sum_{j=1}^{n_i} (X_{ij} - \bar{X}_i)^2}, \quad (3.18)$$

де MSB – міжгрупова дисперсія, MSW – внутрішньогрупова дисперсія, g – кількість груп (вікон), n_i – обсяг i-ї групи, $\bar{X}_i$ – середнє i-ї групи, $\bar{X}$ – загальне середнє, N – загальна кількість спостережень. Велике значення F свідчить про статистично значущу відмінність середніх сусідніх вікон, тобто про зміну режиму, і слугує додатковим сигналом моніторингу, який доповнює контроль за межами 3σ . Послідовність операцій алгоритму моніторингу наведено на рисунку 3.3.

Якість виявлення аномалій оцінюють за трьома взаємопов'язаними метриками, обчисленими шляхом зіставлення сигналів детектора з еталонними мітками збурень. Точність визначає частку справжніх аномалій серед усіх поданих сигналів, повнота – частку виявлених аномалій серед усіх наявних, а їхнє гармонічне середнє F1 дає єдину узагальнену оцінку:

$$\text{Precision} = \frac{TP}{TP + FP}, \text{Recall} = \frac{TP}{TP + FN}, F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (3.19)$$

де TP – кількість правильно виявлених аномальних відліків, FP – кількість хибних спрацювань, FN – кількість пропущених аномалій. Точність і повнота перебувають у природному компромісі: зниження порогу підвищує повноту ціною зростання хибних спрацювань, тому правило серій (3.17) є засобом поліпшити точність за майже незмінної повноти. Зміст метрик і бажаний напрям їхніх значень узагальнено в таблиці 3.5.

Таблиця 3.5

Метрики якості виявлення аномалій

Метрика	Позначення	Що характеризує	Напрямок кращого значення
Точність	Precision	частка справжніх аномалій серед сигналів	більше, до 1
Повнота	Recall	частка виявлених аномалій серед наявних	більше, до 1
Узагальнена метрика	F1	гармонічне середнє точності та повноти	більше, до 1
Затримка виявлення	Tdet	час від початку збурення до сигналу	менше

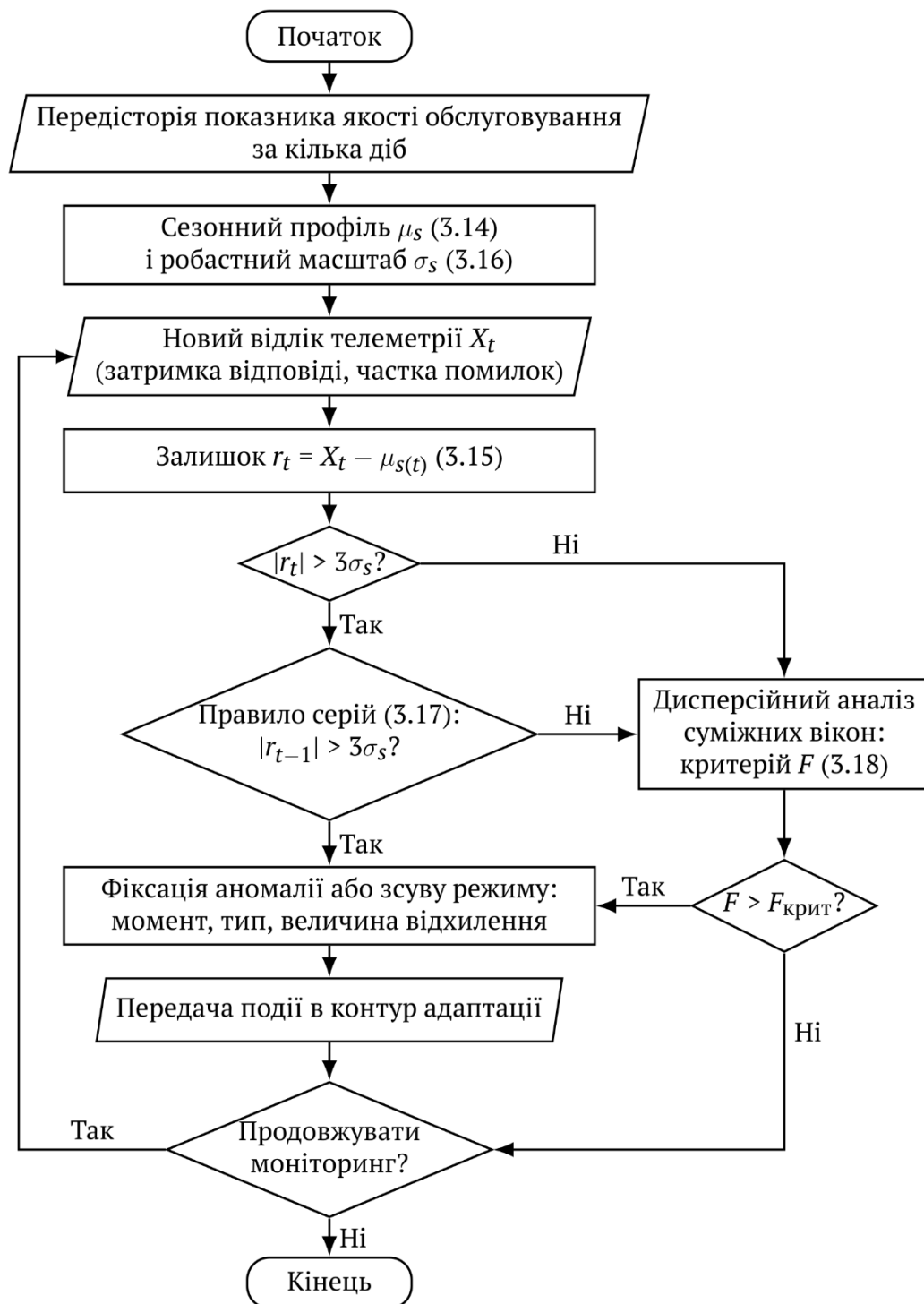


Рис. 3.3 Блок-схема алгоритму статистичного моніторингу стану системи та виявлення аномалій

Отже, поєднання сезонної моделі показника з робастною оцінкою масштабу, контрольної межі з правилом серій і дисперсійного критерію зсуву режиму дає детектор, чутливий до різномірних збурень і стійкий до хибних тривог, який вчасно ініціює адаптацію системи.

3.4. Прогнозування поведінки користувачів методами машинного навчання

Статистичний моніторинг, розглянутий у підрозділі 3.3, виявляє збурення вже після його настання, тому сам по собі забезпечує лише реактивне реагування. Для випереджального, проактивного забезпечення функціональної стійкості потрібно передбачати зміну поведінки користувачів заздалегідь, щоб система встигала перерозподілити ресурси до того, як зростання навантаження спричинить деградацію якості. Цю роль виконує друга складова технології – прогнозування поведінки користувачів методами машинного навчання, яке за передісторією звернень оцінює майбутню інтенсивність навантаження й тим самим переводить керування з реактивного режиму у випереджальний. Для прогнозування поведінки застосовано лінійну регресію, у якій прогнозований показник подають як лінійну комбінацію вхідних чинників:

$$Y = \beta_0 + \sum_{i=1}^p \beta_i X_i + \varepsilon, \quad (3.20)$$

де Y – прогнозований показник, X_i – вхідні чинники, β_i – коефіцієнти моделі, ε – випадкова похибка. Лінійна модель (3.20) є простою й інтерпретованою, проте не відтворює нелінійних ефектів і немонотонного тренду, притаманних реальному профілю навантаження корпоративної системи, тому в дисертації її узагальнено до нелінійної моделі на основі ансамблю дерев рішень.

Задачу прогнозування формують як передбачення значення навантаження на h кроків уперед за вектором ознак, побудованим на момент t :

$$\widehat{d_{t+h}} = f(z_t), \quad (3.21)$$

де $\widehat{d_{t+h}}$ – прогнозоване навантаження, f – навчена модель, z_t – вектор ознак. Вектор ознак поєднує запізнені значення самого навантаження, які несуть інформацію про його поточний рівень і найближчу динаміку, з календарними ознаками, що кодують добову та тижневу сезонність, і з ознакою астрономічного часу, що відображає повільний тренд зростання активності:

$$z_t = (d_t, d_{t-1}, \dots, d_{t-p}, \sin \theta_t, \cos \theta_t, e_t, t/N) \quad (3.22)$$

де d_{t-i} – запізнені значення навантаження, θ_t – фаза доби, $\sin \theta_t$ та $\cos \theta_t$ – гармоніки добового циклу, e_t – ознака вихідного дня, t/N – нормований час. Розкладання доби на пару гармонік забезпечує неперервність кодування часу через межу доби, а ознака нормованого часу дозволяє моделі врахувати тренд, який запізнені значення самі по собі відтворюють неповно.

Як модель прогнозування застосовано градієнтний бустинг над деревами рішень – ансамблевий метод, що будує прогноз послідовним додаванням простих дерев, кожне з яких зменшує похибку попередньої суми, апроксимуючи її антиградієнт [51]. Модель формують ітеративно за правилом

$$f_b(z) = f_{b-1}(z) + v\gamma_b h_b(z), \quad (3.23)$$

де f_b – наближення після b -ї ітерації, h_b – чергове дерево, γ_b – його ваговий коефіцієнт, а v – темп навчання, що стримує внесок кожного дерева й запобігає перенавчанню. Градієнтний бустинг природно відтворює нелінійні залежності й порогові ефекти навантаження, стійкий до різномасштабних ознак і не потребує їх попереднього нормування, чим вигідно відрізняється від лінійної моделі (3.20). Разом із ознакою нормованого часу (3.22) він адекватно апроксимує профіль навантаження з добовою сезонністю та трендом зростання.

Окрім точкового прогнозу, для проактивного резервування важлива оцінка невизначеності, оскільки резерв потужності доцільно закладати з урахуванням можливого відхилення фактичного навантаження від прогнозу. Інтервал прогнозу будують симетрично навколо точкової оцінки за стандартним відхиленням залишків моделі:

$$I_\gamma(\widehat{d_{t+h}}) = [\widehat{d_{t+h}} - z_\gamma \widehat{\sigma}, \widehat{d_{t+h}} + z_\gamma \widehat{\sigma}], \quad (3.24)$$

де $\widehat{\sigma}$ – стандартне відхилення залишків прогнозу на навчальній вибірці, z_γ – квантиль стандартного нормального розподілу рівня γ ; для 95-відсоткового інтервалу множник z_γ становить приблизно 1,96. Верхня межа інтервалу (3.24) слугує орієнтиром для проактивного масштабування потужності: систему готують до навантаження, що з високою ймовірністю не буде перевищене, чим

досягається запас стійкості без надмірного резервування. Побудова прогнозних інтервалів для випереджального керування узгоджується із засадами прогнозування часових рядів [73, 26]. Склад ознак моделі прогнозування наведено в таблиці 3.6.

Таблиця 3.6

Ознаки моделі прогнозування навантаження

Група ознак	Позначення	Що відображає	Призначення
Запізнені значення	dt, dt-1, ..., dt-p	поточний рівень і динаміка навантаження	короткострокова інерція
Добові гармоніки	sin θt , cos θt	фаза доби	добова сезонність
Ознака вихідного	et	робочий чи вихідний день	тижнева сезонність
Нормований час	t/N	положення на горизонті	тренд зростання

Якість прогнозування оцінюють трьома метриками, що характеризують різні аспекти похибки: середньою абсолютною похибкою, середньою абсолютною відсотковою похибкою та коефіцієнтом детермінації:

$$MAE = \frac{1}{n} \sum_{i=1}^n |d_i - \hat{d}_i|, MAPE = \frac{100\%}{n} \sum_{i=1}^n \frac{|d_i - \hat{d}_i|}{d_i}, R^2 = 1 - \frac{\sum_i (d_i - \hat{d}_i)^2}{\sum_i (d_i - \bar{d})^2}, (3.25)$$

де d_i – фактичне навантаження, $\hat{d}_i$ – прогнозоване, $\bar{d}$ – середнє фактичне значення, n – обсяг вибірки. Середня абсолютна похибка виражає типове відхилення в одиницях навантаження, відсоткова похибка робить оцінку зіставною незалежно від масштабу, а коефіцієнт детермінації показує частку поясненої моделлю дисперсії. Метрики обчислюють на штатних, неаномальних відрізках часового ряду, оскільки прогнозувальник моделює звичайну поведінку користувачів, тоді як самі збурення є предметом детектора аномалій, а не прогнозу.

Достатня точність короткострокового прогнозу разом із верхньою межею інтервалу (3.24) дозволяє завчасно масштабувати потужність під очікуване зростання навантаження й тим самим утримувати продуктивнісну складову якості (3.2) під час пікових періодів. Застосування машинного навчання для передбачення поведінки й раннього реагування узгоджується із сучасними

підходами до адаптивного захисту інформаційних систем [112]. Послідовність операцій прогнозування та проактивного резервування наведено на рисунку 3.4.

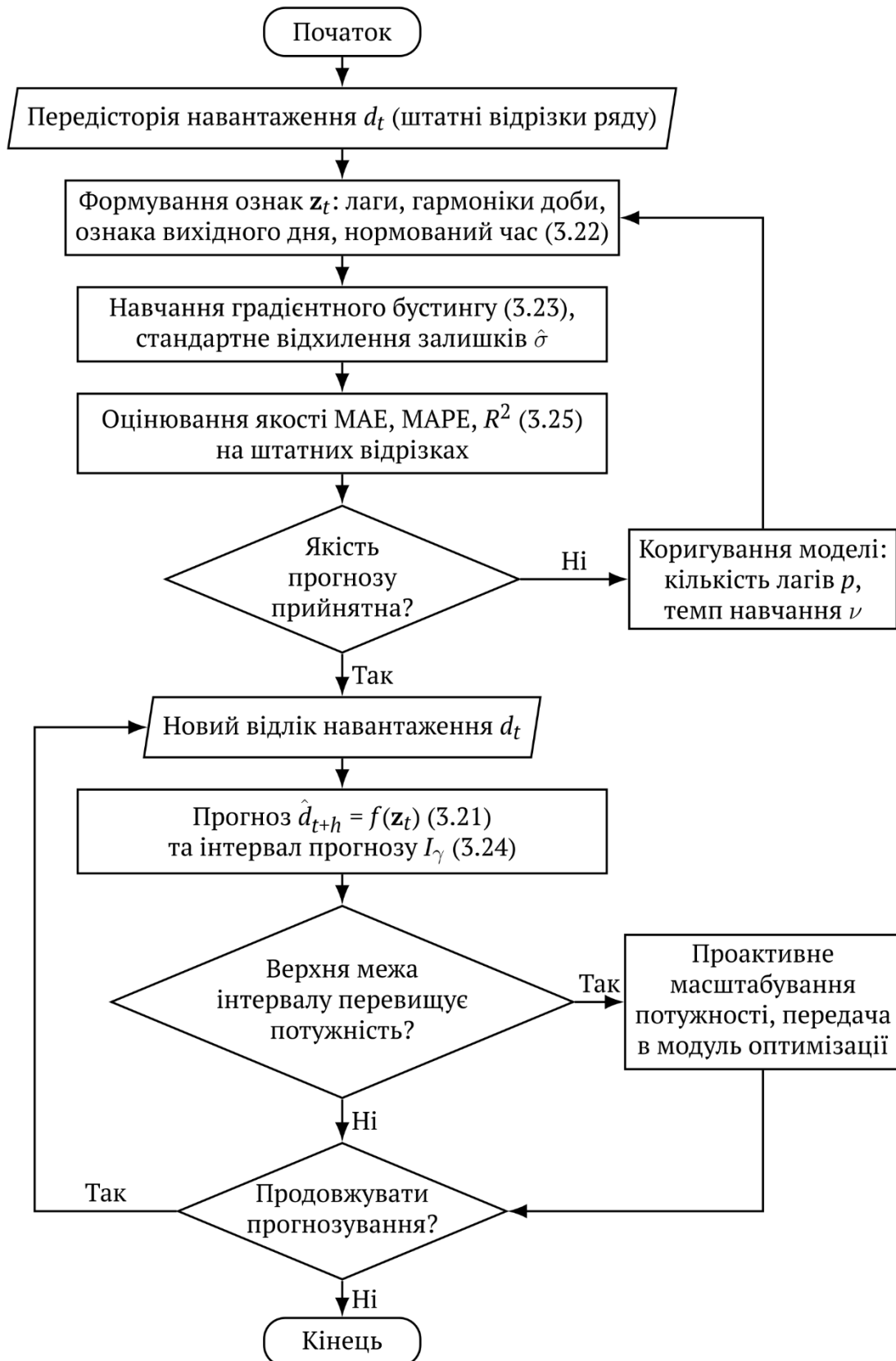


Рис. 3.4. Блок-схема прогнозування навантаження та проактивного резервування ресурсів методами машинного навчання

Отже, прогнозування навантаження градієнтним бустингом разом з інтервалом прогнозу переводить керування ресурсами з реактивного режиму у випереджальний, забезпечуючи запас потужності без надмірного резервування.

3.5. Оптимізація інформаційних потоків і ресурсів

Виявлення збурень моніторингом і передбачення навантаження прогнозуванням лише готують підстави для дії; безпосередньо ж функціональну стійкість підтримує третя складова технології – оптимізація інформаційних потоків і ресурсів, що перерозподіляє обмежену обчислювальну потужність на користь критичних сервісів. Ця складова діє у двох типових ситуаціях: за пікового навантаження вона розподіляє ресурси так, щоб мінімізувати витрати за дотримання вимог до якості, а за відмови компонента – перемаршрутизовує потоки запитів на справні вузли, відновлюючи пропускну здатність системи. Коли розглядають кілька варіантів розподілу за різними критеріями, їх зводять до єдиної оцінки методом зваженої суми, у якому часткові показники підсумовують із вагами їхньої важливості:

$$Z = \sum_{i=1}^n w_i X_i, \quad (3.26)$$

де X_i – значення i -го показника варіанта розподілу, w_i – його вага. Критерій (3.26) дозволяє порівнювати варіанти, що відрізняються за вартістю, затримкою та надійністю, зводячи багатокритеріальний вибір до впорядкування за єдиною величиною [163].

Задачу економного розподілу ресурсів формалізують як задачу лінійного програмування, у якій мінімізують сумарну вартість обслуговування інформаційного потоку за обмежень на доступні ресурси. Цільова функція має вигляд

$$\min Z = \sum_{j=1}^n c_j x_j, \quad (3.27)$$

а обмеження задають доступні обсяги ресурсів та невід'ємність змінних:

$$\sum_{j=1}^n a_{ij}x_j \leq b_i, x_j \geq 0, i = 1, \dots, m, \quad (3.28)$$

де x_j – обсяг j -го ресурсу, задіяного в обслуговуванні, c_j – питома вартість його одиниці, a_{ij} – витрата i -го ресурсу на одиницю j -ї змінної, b_i – доступний обсяг i -го ресурсу. Модель (3.27), (3.28) є класичною задачею лінійного програмування й розв'язується стандартними ефективними методами, що гарантують глобальний оптимум за лінійних обмежень [21]. Її застосування дозволяє мінімізувати вартість або час обслуговування потоку без порушення ресурсних обмежень системи.

Для задачі перемаршрутизації за відмов компонентів систему подають як мережу обслуговування, у якій джерело відповідає вхідному потоку запитів, проміжні вузли – паралельним обробникам (робочим процесам, вузлам оброблення), а сток – завершенню обслуговування; кожне ребро має обмежену пропускну здатність. Мета перемаршрутизації полягає у максимізації сумарного потоку від джерела до стоку, тобто обсягу запитів, які система здатна обслужити за поточних пропускну здатностей

$$F_{max} = \max \sum_{v \in V} f(s, v) \quad (3.29)$$

обмежень пропускну здатності ребер і збереження потоку у проміжних вузлах:

$$0 \leq f(u, v) \leq c(u, v), \sum_{u \in V} f(u, v) = \sum_{w \in V} f(v, w), \quad (3.30)$$

де $f(u, v)$ – потік ребром від вузла u до вузла v , $c(u, v)$ – його пропускну здатність, s – джерело, V – множина вузлів мережі. Задачу (3.29), (3.30) розв'язують алгоритмом пошуку максимального потоку, теоретичну основу якого закладено теоремою про максимальний потік і мінімальний розріз [49, 10]. Практична цінність такого підходу для функціональної стійкості полягає у тому, що за відмови окремого вузла максимальний потік автоматично перерозподіляється на справні вузли, тоді як статична рівномірна маршрутизація втрачає частину запитів, спрямованих на уражений вузол.

Ефект перемаршрутизації кількісно оцінюють часткою необслугованого попиту – відносною різницею між попитом і фактично обслугованим потоком:

$$U = 1 - \frac{\min(d, F_{\max})}{d}, \quad (3.31)$$

де d – поточний попит, $F_{\max}$ – максимальний потік, який система здатна пропустити. Що менше значення U , то повніше задовольняється попит; для функціонально стійкої системи ця величина має залишатися близькою до нуля навіть під час відмов. Послідовність операцій складової показано на рисунку 3.5. Порівняння статичної маршрутизації та оптимізації потоків за поведінкою під відмовою наведено в таблиці 3.7.

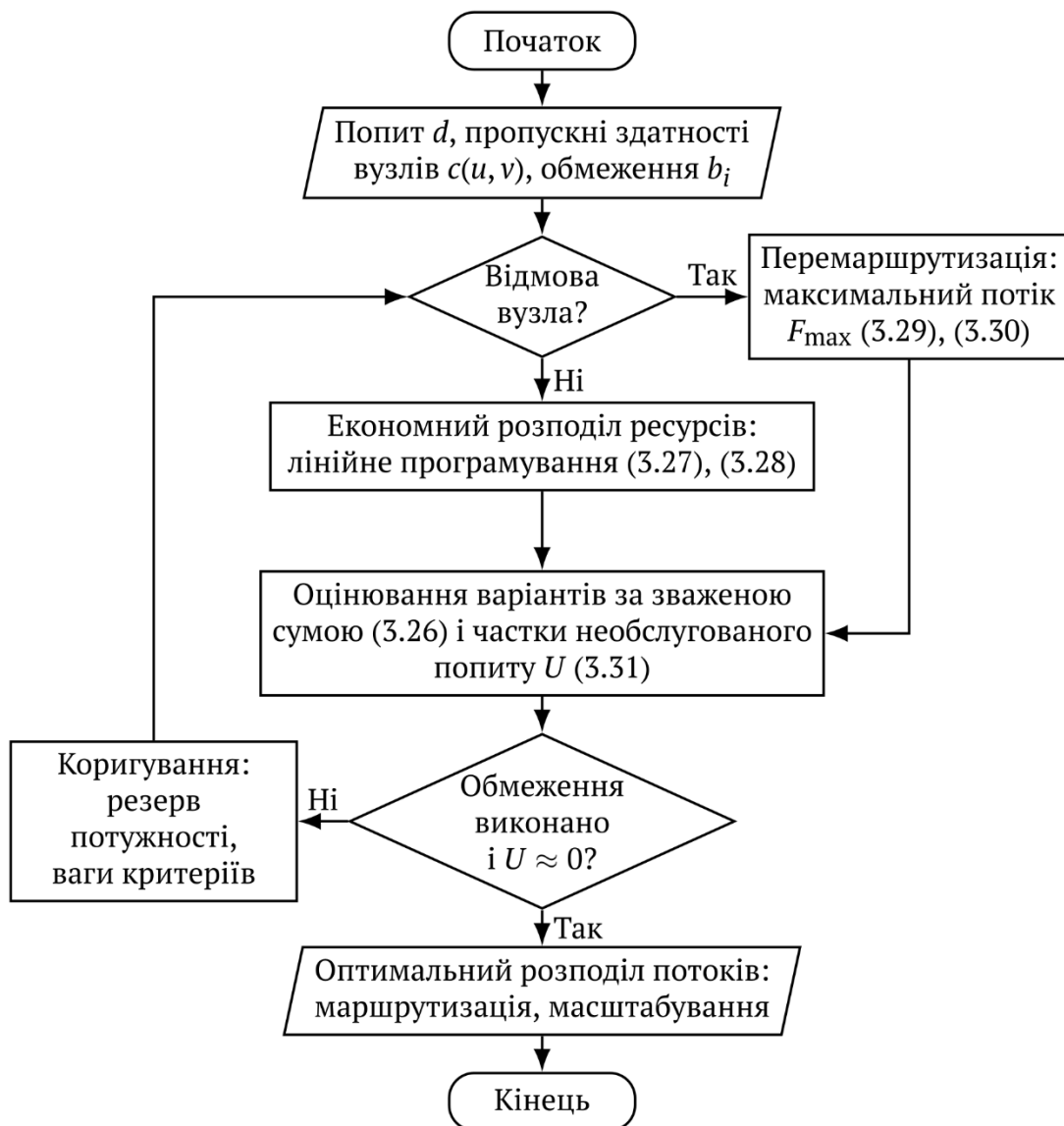


Рис. 3.5. Блок-схема оптимізації інформаційних потоків і ресурсів на основі максимального потоку

З теоретичних співвідношень (3.29)–(3.31) випливає ключова властивість складової оптимізації: доки сумарна пропускна здатність справних вузлів не менша за поточний попит, оптимальна маршрутизація здатна обслужити його повністю, зводячи необслугований попит до нуля, тоді як статична рівномірна маршрутизація втрачає частину запитів, спрямованих на уражений вузол. Саме тому перемаршрутизація на основі максимального потоку безпосередньо підтримує продуктивнісну складову якості (3.2) та адаптивність системи (3.11), скорочуючи площу деградації під час відмов.

Отже, перемаршрутизація інформаційних потоків на основі максимального потоку разом із лінійним програмуванням розподілу ресурсів і зваженим вибором варіантів забезпечує повне обслуговування попиту за відмов у межах сумарної пропускної здатності справних вузлів.

3.6. Оцінювання ризиків, витрат і рівня інформаційної безпеки

Четверта складова технології забезпечує безпекову частину функціональної стійкості, поєднуючи оцінювання кіберризиків із вартісною оптимізацією захисних заходів. Її потреба зумовлена тим, що безпека й ризик входять до цільової функції (3.7) як окремі змінні, а їхнє забезпечення споживає бюджет, обмежений умовою (3.13); тому захист має бути не максимальним, а економічно раціональним. Відправною точкою є оцінка ймовірності успішної реалізації загрози, що враховує ймовірність самої загрози, вразливість системи та рівень контролю з боку захисних заходів:

$$P_{\text{success}} = \frac{P_{\text{threat}} \cdot P_{\text{vuln}}}{P_{\text{control}}}, \quad (3.32)$$

де P_{threat} – ймовірність виникнення загрози, P_{vuln} – ймовірність наявності вразливості, P_{control} – рівень контролю, що відображає ефективність захисних механізмів. Зі співвідношення (3.32) випливає, що посилення контролю знижує ймовірність успішної атаки. Очікувані втрати від успішних атак підсумовують за всіма розглянутими загрозами:

$$R_{\text{attack}} = \sum_{i=1}^m P_{\text{success},i} \cdot I_{\text{loss},i}, \quad (3.33)$$

де $I_{\text{loss},i}$ – потенційні втрати від i -ї загрози, m – кількість загроз. Величина (3.33) конкретизує рівень ризику R математичної моделі (3.10) у кібербезпековому контексті й узгоджується зі стандартними підходами до кількісного оцінювання інформаційних ризиків [50, 75]. Значення ймовірності успішної реалізації загрози за потреби обмежують зверху одиницею, оскільки воно не може перевищувати достовірної події.

Вплив витрат на безпеку на рівень захищеності системи має спадну віддачу: перші вкладення різко знижують ризик, тоді як подальші дають дедалі менший приріст. Цю залежність описують експоненційною моделлю залишкового ризику, за якою ризик спадає з витратами на безпеку:

$$R_{\text{res}} = R_{\text{init}} \cdot \exp(-\eta M_{\text{sec}}), \quad (3.34)$$

де R_{init} – початковий рівень ризику за відсутності захисту, M_{sec} – витрати на заходи безпеки, η – коефіцієнт ефективності захисних заходів. Симетричним відображенням цієї залежності є нелінійна модель рівня стійкості системи з пороговим ефектом, за якою до певного критичного рівня витрат захищеність зростає повільно, а після його подолання – різко:

$$S_{\text{sys}} = \frac{1}{1 + \exp(-\alpha(M_{\text{sec}} - M_{\text{crit}}))}, \quad (3.35)$$

де M_{crit} – критичний рівень витрат, за якого досягається мінімально прийнятна стійкість, α – параметр чутливості. Сигмоїдна форма (3.35) відображає загальновідому властивість інвестицій в інформаційну безпеку: витрати нижче критичного порогу майже не впливають на захищеність, тоді як їх перевищення різко підвищує стійкість, після чого віддача знову спадає, а надмірні вкладення стають неефективними [62, 144]. Оптимальний бюджет безпеки лежить у околі критичного рівня M_{crit} , де перетинаються криві залишкового ризику (3.34) та стійкості (3.35). Загальний бюджет забезпечення стійкості розподіляють між заходами безпеки та резервуванням:

$$M_{\text{sec}} + M_{\text{backup}} = C_{\text{safe}}, \quad (3.36)$$

де M_{backup} – витрати на резервування та відновлення, C_{safe} – сукупний бюджет. Розподіл (3.36) є предметом оптимізації: недостатнє фінансування резервування подовжує простої за відмов, тоді як надмірне – невиправдано зменшує кошти на активний захист.

Час реагування на інцидент безпосередньо визначає масштаб втрат: чим довше система відновлюється, тим більших збитків завдає інцидент. Публікація [167] задає цю залежність співвідношенням, у якому втрати спадають з часом реагування, що суперечить змістовому трактуванню; тому в дисертації модель подано у виправленій, монотонно зростальній формі, за якою втрати зростають із часом реагування й насичуються:

$$R_{\text{loss}}(T_{\text{rec}}) = L_{\text{max}}(1 - \exp(-T_{\text{rec}}/\tau)), \quad (3.37)$$

де L_{max} – гранична величина втрат за тривалого простою, T_{rec} – час реагування, τ – характерний час наростання втрат. За малого часу реагування втрати близькі до нуля, а з його зростанням прямують до граничного рівня L_{max} аме тому скорочення часу реагування є економічно доцільним. Загальну вартість забезпечення стійкості оцінюють як інтегральну суму втрат від ризику та витрат на безпеку у часі, яку мінімізують добором бюджету:

$$\min C_{\text{total}} = \int_0^T (k_1 R_{\text{loss}}(t) + k_2 M_{\text{sec}}(t)) dt, \quad (3.38)$$

де k_1 та k_2 – вагові коефіцієнти пріоритетів зниження втрат і витрат.

Критерій (3.38) формалізує компроміс «безпека – витрати» з постановки (3.6) і разом із моделями (3.34), (3.35) дозволяє визначити раціональний бюджет, за якого досягається максимальне зниження ризику за прийнятних витрат [14]. З моделей залишкового ризику (3.34) та втрат від часу реагування (3.37) випливає, що комплексний захист, який одночасно знижує залишковий ризик і скорочує час реагування, дає менші сукупні втрати, ніж лише резервування чи відсутність активного захисту.

Послідовність операцій складової наведено на рисунку 3.6.

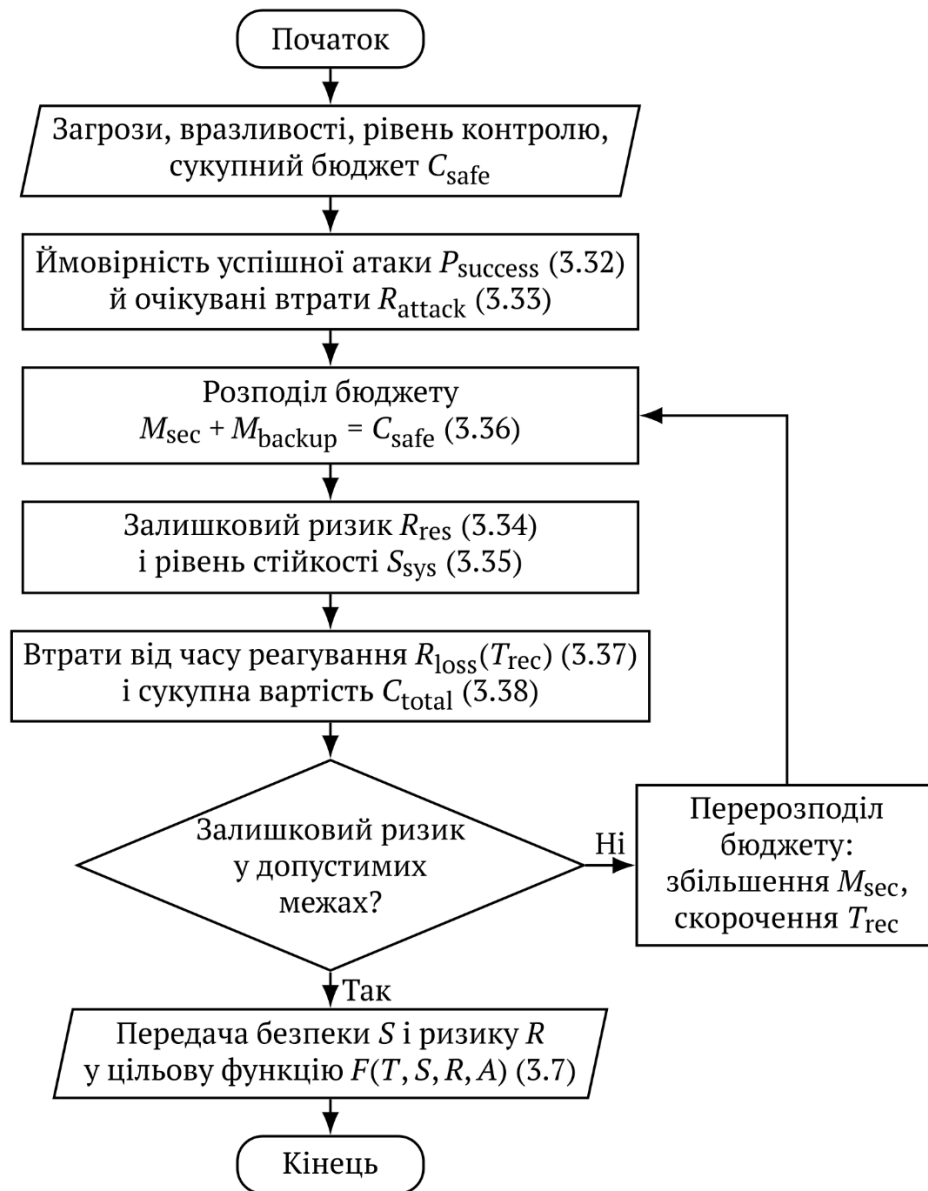


Рис. 3.6. Блок-схема оцінювання ризиків і вартісної оптимізації безпеки

3.7. Структура інформаційної технології забезпечення функціональної стійкості

Розглянуті у попередніх підрозділах складові набувають повної дієвості лише у взаємодії, коли їх об'єднано в єдину інформаційну технологію з чотирьох складових: моніторингу й виявлення аномалій, прогнозування поведінки, оптимізації потоків і ресурсів, оцінювання ризиків і керування бюджетом безпеки. Складові утворюють замкнений контур керування, що узгоджується з парадигмою автономних систем і послідовно виконує етапи моніторингу, аналізу, планування та виконання з опорою на спільну базу знань [32, 82]. У

кожному циклі технологія відображає поточний стан системи та збурення на коригувальну дію, спрямовану на максимізацію функціональної стійкості:

$$\mathcal{M}: (Q(t), v(t)) \mapsto a(t) \quad (3.39)$$

де $Q(t)$ – показник якості функціонування, $v(t)$ – вектор поточних збурень, $a(t)$ – коригувальна дія адаптації, що охоплює масштабування потужності, перемаршрутизацію потоків і зміну політик безпеки. Саме замкненість контуру (3.39) відрізняє запропоновану технологію від набору ізольованих заходів: результат кожної складової є входом для наступної, а результат адаптації повертається на вхід моніторингу наступного циклу.

Взаємодія складових технології має чіткий розподіл впливу на змінні математичної моделі (3.7): кожна складова відповідає переважно за окрему змінну цільової функції, а разом вони забезпечують її одночасну максимізацію. Відповідність складових технології змінним моделі, підрозділам їх викладу та характеру впливу на функціональну стійкість наведено в таблиці 3.7.

Таблиця 3.7

Відповідність складових технології змінним моделі функціональної стійкості

Складова технології	Підрозділ	Змінна моделі	Внесок у функціональну стійкість
Моніторинг і виявлення аномалій	3.3	функціональність Т	своєчасне виявлення збурень
Прогнозування поведінки	3.4	функціональність Т	проактивне резервування ресурсів
Оптимізація потоків і ресурсів	3.5	функціональність Т, адаптивність А	перемаршрутизація, скорочення деградації
Оцінювання ризиків і безпеки	3.6	безпека S, ризик R	раціональний захист і бюджет

Об'єднавчим елементом є спільна база знань, що зберігає математичну модель стійкості (3.7), навчені моделі прогнозування та поточні параметри системи й доступна всім складовим контуру. Узагальнену блок-схему інтегрованої технології із замкненим контуром адаптації та базою знань наведено на рисунку 3.7, а приклад панелі моніторингу, через яку технологія подає результати службі експлуатації, – у додатку Г.

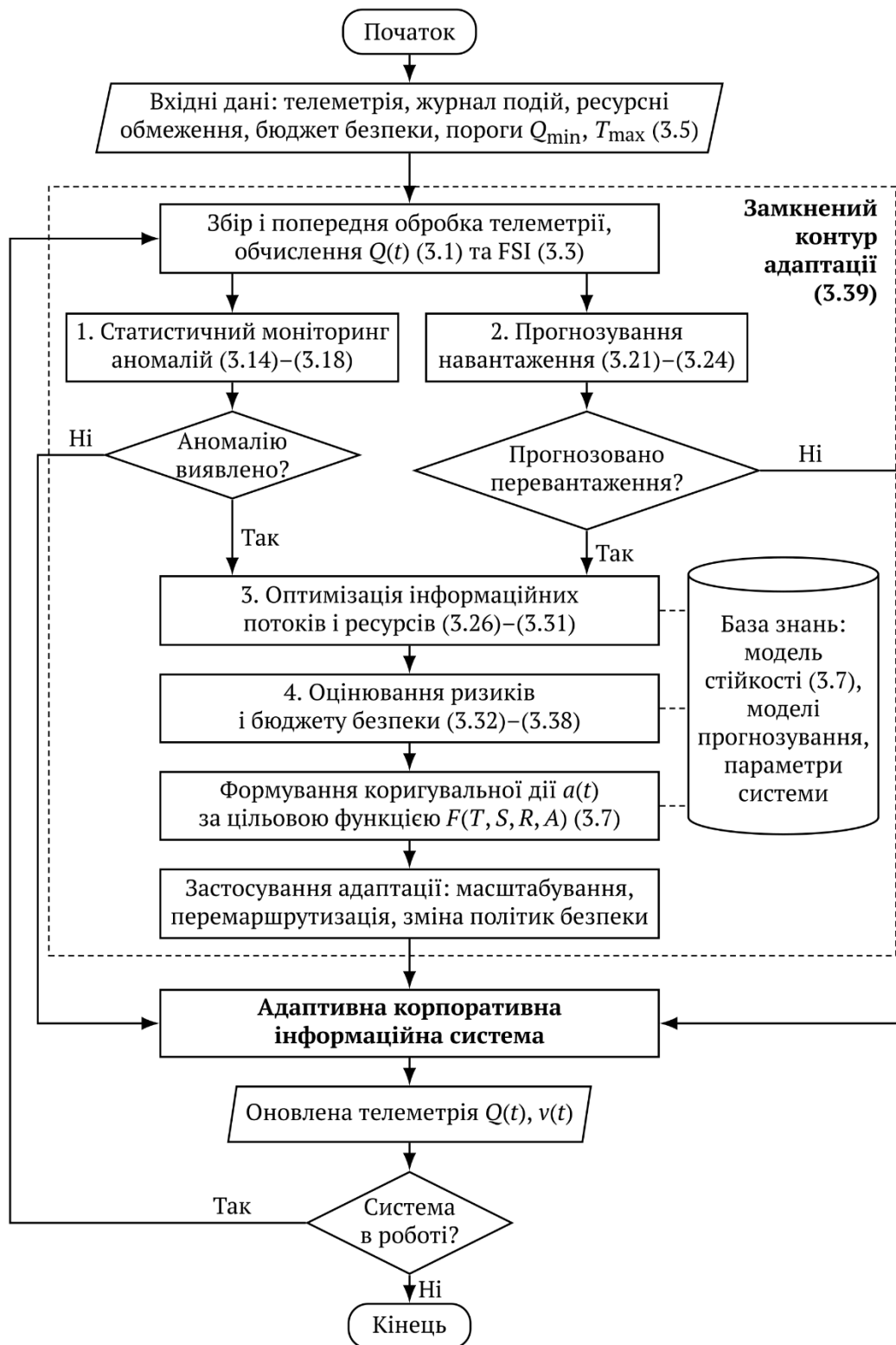


Рис. 3.7. Блок-схема інтегрованої інформаційної технології забезпечення функціональної стійкості

Інтегрований характер технології дає якісно новий ефект порівняно з ізольованим застосуванням окремих складових. Приріст функціональної стійкості, який забезпечує технологія, виражають різницею індексів

функціональної стійкості системи з упровадженою технологією та базової конфігурації без неї, і додатне значення цієї різниці є інтегральним показником корисності всього контуру адаптації. Джерелом приросту є саме взаємодія складових: прогнозування робить дієвим проактивне резервування, моніторинг вчасно запускає перемаршрутизацію, а оцінювання ризиків узгоджує захист із бюджетом, унаслідок чого одночасно зростають функціональність, безпека й адаптивність, а ризик знижується.

Відмінності інтегрованої технології від базової реактивної конфігурації за складовими цільової функції узагальнено в таблиці 3.8.

Таблиця 3.8

Відмінності інтегрованої технології від базової конфігурації

Складова цільової функції	Базова конфігурація	Інтегрована технологія
Функціональність T	статичні ресурси, реактивне реагування	проактивне резервування, перемаршрутизація
Безпека S	слабкий контроль загроз	активний контроль, раціональний бюджет
Ризик R	високий залишковий ризик	знижений залишковий ризик
Адаптивність A	повільне відновлення	швидке відновлення в режимі реального часу

Отже, складові моніторингу, прогнозування, оптимізації потоків та оцінювання ризиків об'єднано в цілісну інформаційну технологію із замкненим контуром адаптації, яка відображає стан системи та збурення на коригувальну дію в режимі реального часу, а приріст індексу функціональної стійкості слугує інтегральною мірою її корисності.

Висновки до розділу 3

У розділі викладено теоретичні засади інформаційної технології забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи. Функціональну стійкість розмежовано із суміжними властивостями надійності, відмовостійкості та живучості, а її визначальною ознакою визнано активну, керовану в режимі реального часу адаптацію до збурень. Різномірні

прояви збурень зведено до єдиного показника якості функціонування з продуктивнісною та безпековою складовими, на основі якого побудовано індекс функціональної стійкості, площу деградації та час відновлення.

Ядром апарату є математична модель функціональної стійкості як задача умовної оптимізації, у якій функціональність, безпека та адаптивність максимізуються, ризик мінімізується, а обмеження на ризик, мінімальні рівні безпеки й адаптивності та бюджет надають пошуку стійкості економічно раціонального характеру. Означення адаптивності через площу деградації пов'язує проєктний критерій із часовою поведінкою системи.

Технологію сформовано з чотирьох складових: статистичного моніторингу із сезонною моделлю, робастною оцінкою масштабу та правилом серій; прогнозування навантаження градієнтним бустингом з інтервалом прогнозу для проактивного резервування; перемаршрутизації інформаційних потоків на основі максимального потоку, що забезпечує повне обслуговування попиту за відмов у межах сумарної пропускної здатності справних вузлів; оцінювання ризиків і вартісної оптимізації безпеки з монотонною залежністю втрат від часу реагування. Складові утворюють замкнений контур моніторингу, аналізу, планування та виконання зі спільною базою знань, у якому результат кожної складової є входом для наступної.

Наукова новизна результату полягає в тому, що, удосконалено інформаційну технологію забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи, яка, на відміну від відомих реактивних рішень, інтегрує статистичний моніторинг аномалій, прогнозування поведінки методами машинного навчання, оптимізацію інформаційних потоків та оцінювання ризиків у єдиний контур керування, що забезпечує адаптацію системи в режимі реального часу та підтримання критичної функціональності в умовах збурень.

РОЗДІЛ 4. МЕТОД ТА ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ СТРУКТУРНО-ПАРАМЕТРИЧНОГО ПРОЄКТУВАННЯ ВІДМОВОСТІЙКИХ І ЗАХИЩЕНИХ РОЗПОДІЛЕНИХ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ

4.1. Постановка задачі проєктування відмовостійких і захищених розподілених систем

Корпоративні структури дедалі частіше будують свої обчислювальні системи як розподілені, що підвищує доступність сервісів і масштабованість інфраструктури, проте розширює поверхню атаки, оскільки кожен вузол і кожен канал зв'язку стають потенційною ціллю [17, 88, 148]. Фізична втрата компонентів і реалізація загроз безпеці ведуть до однакового наслідку – зниження здатності системи виконувати критичні функції, тому відмовостійкість і захищеність доцільно поєднати в межах єдиної задачі проєктування.

Суть задачі полягає в тому, щоб на етапі проєктування, коли архітектура ще не зафіксована, обрати таку структуру й такі параметри розподіленої обчислювальної системи, які забезпечують збереження критичної функціональності одночасно за втрати значної частки компонентів і за дії загроз безпеці функціонування. Математичний апарат цієї задачі – цільову функцію безпеки, моделі ймовірності успішної атаки, живучості, адаптивності та плавної деградації, а також інтегральний ризик із множителем підвищеної загрози – раніше обґрунтовано для кіберфізичної системи екологічного моніторингу в умовах воєнного стану [164, 168]. У цьому розділі апарат узагальнено й перенесено на розподілену корпоративну обчислювальну систему, а множник воєнного часу трактується ширше – як множник підвищеної загрози, що відповідає періодам загострення атак, зокрема кампаній цілеспрямованих тривалих загроз [14].

Розподілену обчислювальну систему подамо орієнтованим графом

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}) \quad (4.1)$$

де $\mathcal{V}$ – множина компонентів системи, а $\mathcal{E}$ – множина зв'язків між ними. Компоненти об'єднуються у скінченну кількість функціональних класів (крайові вузли доступу, транспорт даних, обчислювальні вузли, сховища даних, підсистема безпеки), кожному з яких приписано вагу важливості, що відображає його критичність для роботи системи в цілому. Такий поділ дозволяє описувати різноманітні за призначенням компоненти в єдиній моделі та зважувати їхній внесок у сукупну функціональність.

Проектне рішення описується архітектурою, яку зручно подати як структурно-параметричну пару

$$\mathcal{A} = (\mathcal{S}, p), \mathcal{S} \subset \mathcal{V} \times \mathcal{V}, p \in \mathcal{P} \quad (4.2)$$

де структурна частина $\mathcal{S}$ задає топологію зв'язків, схему розподілу компонентів за рівнями та кратність резервування, а параметрична частина p визначає числові налаштування: розподіл обмежених ресурсів безпеки між компонентами, порогові значення систем виявлення, обсяг резервних потужностей і конфігурацію обчислювальних вузлів. Саме поєднання структурних і параметричних рішень зумовлює назву методу – структурно-параметричне проектування. Основні групи проектних рішень та їх призначення узагальнено в таблиці 4.1.

Таблиця 4.1

Структурні та параметричні проектні рішення

Група рішень	Проектний параметр	Призначення
Структурні	Топологія зв'язків	Усунення єдиних точок відмови
Структурні	Кратність резервування k-з-п	Збереження класу за втрати частини екземплярів
Структурні	Розподіл компонентів за рівнями	Локалізація наслідків втрат і автономність рівнів
Параметричні	Розподіл ресурсів безпеки	Максимізація рівня захищеності за обмеженого бюджету
Параметричні	Порогові значення виявлення	Налаштування чутливості засобів моніторингу
Параметричні	Обсяг резервних потужностей	Компенсація втрат і пікових навантажень
Параметричні	Конфігурація обчислювального вузла	Задана продуктивність за мінімальної вартості

Постановка спирається на два попередні результати дисертації. Параметри конфігурації обчислювальних вузлів кожної архітектури-кандидата обґрунтовуються методом автоматизованого проєктування на основі сурогатної моделі гаусівської процесової регресії: за задачею (2.30) обирають найдешевшу конфігурацію вузла, верхня довірча межа прогнозованої затримки якої не перевищує допустимого значення за навантаження, що припадає на мінімально потрібну кількість справних вузлів. Функціональну стійкість кожного кандидата оцінюють інформаційною технологією забезпечення функціональної стійкості, яка відтворює траєкторію показника якості (3.1) за сценаріями збурень і дає індекс функціональної стійкості (3.3). Обидва результати утворюють цифровий двійник, на якому перевіряють проєктні рішення перед остаточним вибором архітектури.

Зовнішні впливи, яких зазнає система впродовж експлуатації, зручно описати множиною сценаріїв Ω . Кожен сценарій $\omega \in \Omega$ відповідає певній комбінації втрат компонентів і загроз: відмові окремого вузла, компрометації через кібератаку, перевантаженню за сплеску попиту або одночасній дії кількох чинників. Класи збурень і загроз, врахованих у моделі, наведено в таблиці 4.2.

Об'єднання фізичних відмов і загроз безпеці в одному переліку підкреслює спільність їхнього наслідку для функціональності та обґрунтовує потребу спільного проєктування захищеності й відмовостійкості.

Таблиця 4.2

Класи збурень і загроз у розподіленій обчислювальній системі

Клас збурення	Природа	Вплив на систему
Відмова компонента	Апаратний або програмний збій вузла	Втрата працездатності класу за недостатнього резервування
Кіберінцидент	Компрометація вузла чи каналу внаслідок атаки	Порушення конфіденційності, цілісності або доступності
Перевантаження	Сплеск попиту понад проєктну потужність	Деградація продуктивності, відмови в обслуговуванні
Фізична втрата	Пошкодження чи недоступність частини інфраструктури	Одночасна втрата групи компонентів
Комбінований сценарій	Одночасна дія кількох чинників	Накладання втрат, прискорене зниження функціональності

Здатність системи виконувати корисну роботу за конкретного сценарію оцінюється функціональністю

$$\Phi(\mathcal{A}, \omega) = \sum_{i=1}^n w_i \varphi_i(\mathcal{A}, \omega), \sum_{i=1}^n w_i = 1, \quad (4.3)$$

де $\varphi_i(\mathcal{A}, \omega)$ – показник працездатності i -го класу компонентів за архітектури $\mathcal{A}$ й сценарію ω , що набуває одиничного значення, якщо клас зберігає працездатність (зокрема завдяки резервуванню), і нульового – у протилежному разі, а w_i – вага важливості класу. Функціональність нормовано так, що за повної працездатності всіх класів вона дорівнює одиниці, а за втрати частини класів пропорційно зменшується. Критичним вважається рівень функціональності Φ_{crit} , нижче якого система не забезпечує виконання найважливіших операцій.

Оскільки сценарії втрат мають імовірнісну природу, ключовим показником проєктування є живучість – імовірність того, що функціональність не опуститься нижче критичного рівня:

$$S(\mathcal{A}) = P(\Phi(\mathcal{A}, \omega) \geq \Phi_{\text{crit}}), \omega \in \Omega. \quad (4.4)$$

Живучість у цьому означенні узгоджується з класичним трактуванням живучості мережевих систем як здатності зберігати надання критичних послуг за наявності відмов і атак [46, 84], а також із кількісним підходом до стійкості складних систем як функції часу й ступеня ушкодження [69]. Поряд із живучістю проєктне рішення характеризують ще три функціонали: рівень захищеності $Z(\mathcal{A})$, що відображає стійкість до загроз; інтегральний ризик $R(\mathcal{A})$, що агрегує ймовірні втрати; коефіцієнт плавної деградації $D(\mathcal{A})$, що описує плавність зниження функціональності за наростання втрат. Ці функціонали докладно розглянуто в підрозділах 4.2–4.5; тут вони вводяться як складові єдиного критерію проєктування.

Задача проєктування формулюється як багатокритеріальна, оскільки перелічені показники частково суперечать один одному: підвищення живучості й захищеності потребує додаткових ресурсів і, відповідно, витрат. Для зведення показників до єдиної оцінки застосовується композитний критерій

$$G(\mathcal{A}) = \lambda_1 S(\mathcal{A}) + \lambda_2 Z(\mathcal{A}) + \lambda_3 D(\mathcal{A}) - \lambda_4 R(\mathcal{A}), \quad (4.5)$$

де $\lambda_1, \dots, \lambda_4$ – невід'ємні вагові коефіцієнти пріоритетності показників, що задаються особою, яка приймає рішення, з урахуванням значущості кожного показника для конкретної корпоративної структури. Ризик входить у критерій зі знаком мінус, бо його бажано мінімізувати, тоді як живучість, захищеність і плавність деградації – максимізувати. Оптимальне проєктне рішення визначається як

$$\mathcal{A}^* = \arg \max_{\mathcal{A} \in \mathcal{F}} G(\mathcal{A}), \quad (4.6)$$

де $\mathcal{F}$ – множина допустимих архітектур, що задовольняють проєктні обмеження

$$C(\mathcal{A}) \leq C_{\text{доп}}, T(\mathcal{A}) \leq T_{\text{доп}}, S(\mathcal{A}) \geq S_{\min}, Z(\mathcal{A}) \geq Z_{\min}. \quad (4.7)$$

Тут $C(\mathcal{A})$ і $T(\mathcal{A})$ – сукупна вартість і час розгортання архітектури, обмежені допустимими значеннями $C_{\text{доп}}$ та $T_{\text{доп}}$, а $S_{\min}$ і $Z_{\min}$ імально прийнятні рівні живучості й захищеності. Обмеження (4.7) відсікають нереалізовані або недостатньо надійні варіанти й гарантують, що серед допустимих архітектур розглядаються лише ті, що вкладаються в бюджет і строки та забезпечують базові вимоги до стійкості. Проєктування за поданою постановкою має ітеративний характер, що відображено на рисунку 4.1.

Відправною точкою є вихідні вимоги до системи: цільові показники продуктивності й доступності, модель загроз, а також обмеження щодо бюджету й часу впровадження. На їх основі формується простір архітектур, у якому варіюються структурні рішення й параметри. Кожна архітектура-кандидат оцінюється за живучістю, захищеністю, ризиком і плавністю деградації; якщо проєктні обмеження порушено або композитний критерій не досяг максимуму, виконується корекція структури й параметрів, і оцінювання повторюється. Процес завершується вибором архітектури, що задовольняє обмеженням і максимізує критерій.

Запропонована постановка відрізняється від традиційних підходів двома рисами. По-перше, захищеність і відмовостійкість розглядаються спільно, а не як окремі задачі, що розв'язуються послідовно.

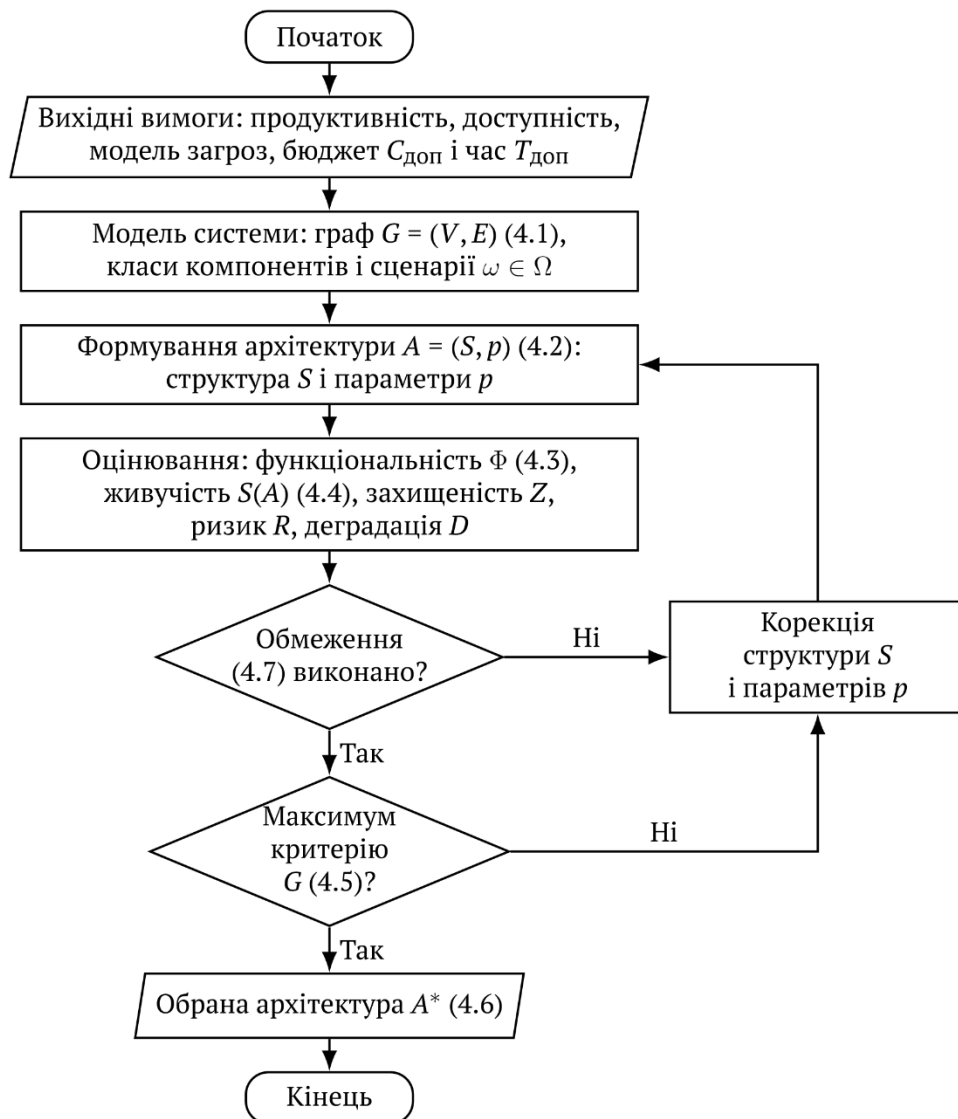


Рис. 4.1. Блок-схема постановки задачі проєктування відмовостійкої й захищеної розподіленої обчислювальної системи

По-друге, рішення ухвалюється на етапі проєктування, а не додається до вже розгорнутої системи, коли можливості зміни структури обмежені. Ранній вибір архітектури з наперед заданими показниками живучості й захищеності узгоджується з дисципліною проєктування живучих систем і щодо оцінювання залишкового ресурсу вузлів розподілених систем критичної інфраструктури [46, 84, 157, 158]. Прийняті припущення полягають у тому, що відмови й компрометації різних компонентів вважаються переважно незалежними подіями, а ваги важливості класів і вагові коефіцієнти критерію визначаються експертно з урахуванням специфіки корпоративної структури.

Отже, задачу проектування відмовостійкої й захищеної розподіленої обчислювальної системи сформульовано як багатокритеріальний вибір структурно-параметричної архітектури, що максимізує композитний критерій живучості, захищеності та плавності деградації за мінімізації ризику й дотримання обмежень щодо вартості, часу та мінімально прийнятних рівнів стійкості. Така постановка об'єднує відмовостійкість і безпеку функціонування в межах єдиної моделі й переносить ухвалення відповідних рішень на етап проектування, що створює основу для подальшої формалізації окремих складових методу.

4.2. Багатокритеріальна оптимізація розподілу ресурсів безпеки

Рівень захищеності розподіленої обчислювальної системи визначається не лише переліком упроваджених механізмів захисту, а й тим, як обмежені ресурси безпеки розподілено між компонентами. Бюджет і час розгортання завжди обмежені, тому застосувати всі засоби до всіх компонентів неможливо, і постає задача оптимального розподілу: обрати такий набір застосувань засобів захисту, який за наявних обмежень максимізує сукупну захищеність системи. Цільову функцію задачі узагальнено для розподіленої корпоративної системи на основі раніше обґрунтованої моделі захищеності [164]. Інтегральний рівень захищеності подається як

$$Z(X) = \sum_{i=1}^n w_i R_i (1 - P_i(X)) E_i, \quad (4.8)$$

де w_i – вага важливості i -го класу компонентів, R_i – його надійність, $P_i(X)$ – ймовірність успішної атаки на цей клас за поточного розподілу засобів, а E_i – коефіцієнт ефективності компонента в умовах підвищеної загрози. Захищеність зростає, коли підвищується надійність критичних компонентів і знижується ймовірність їх успішної компрометації; множник E_i враховує, що частина компонентів чутливіша до зовнішніх збурень.

Розподіл засобів захисту описується булевою матрицею

$$X = [x_{ij}], x_{ij} \in \{0,1\}, i = 1, \dots, n, j = 1, \dots, m, \quad (4.9)$$

де x_{ij} дорівнює одиниці, якщо j -й механізм захисту застосовано до i -го класу компонентів, і нулю в протилежному разі, n – кількість класів компонентів, m – кількість доступних механізмів. Сукупний вплив застосованих до компонента механізмів на його вразливість описується часткою знятої вразливості

$$\pi_i(X) = 1 - \prod_{j=1}^m (1 - e_j x_{ij}), \quad (4.10)$$

де e_j – ефективність j -го механізму. Мультиплікативна форма відображає ефект спадної віддачі: механізми частково перекривають функції один одного, тому навіть повний їх набір лишає залишкову вразливість, а не усуває її повністю. Саме через залишкову вразливість визначається ймовірність успішної атаки $P_i(X)$, модель якої докладно розглянуто в підрозділі 4.3. Складові задачі оптимізації розподілу ресурсів безпеки узагальнено в таблиці 4.3.

Таблиця 4.3

Складові задачі оптимізації розподілу ресурсів безпеки

Складова	Позначення	Зміст
Змінні рішення	x_{ij}	Застосування j -го механізму до i -го компонента (0 або 1)
Цільова функція	Z	Інтегральний рівень захищеності системи
Ефективність засобу	e_j	Частка зниження вразливості j -м механізмом
Вартість засобу	c_j	Нормовані витрати на застосування засобу
Час засобу	τ_j	Нормований час розгортання засобу
Бюджетне обмеження	B	Гранично допустимі сукупні витрати на безпеку
Часове обмеження	T_{max}	Гранично допустимий час розгортання захисту

Оптимізація виконується за наявності ресурсних обмежень. Бюджетне обмеження вимагає, щоб сукупна вартість застосованих засобів не перевищувала виділеного бюджету безпеки:

$$C(X) = \sum_{i=1}^n \sum_{j=1}^m c_j x_{ij} \leq B, \quad (4.11)$$

де c_j – нормована вартість застосування j -го механізму, а B – доступний бюджет. Аналогічно, часове обмеження не дозволяє перевищити граничний строк розгортання захисту:

$$T(X) = \sum_{i=1}^n \sum_{j=1}^m \tau_j x_{ij} \leq T_{max}, \quad (4.12)$$

де τ_j – нормований час упровадження j -го механізму, а T_{max} граничний строк. Обидва обмеження відображають планування портфеля захисних заходів за фіксованої стелі витрат і часу [62]. За таких умов задача набуває вигляду:

$$X^* = \arg \max_X Z(X) \text{ за умов } C(X) \leq B, T(X) \leq T_{max}. \quad (4.13)$$

За своєю структурою це комбінаторна задача цілочислового програмування, споріднена із задачею про рюкзак: кожне застосування засобу дає певний приріст захищеності ціною певних витрат, а сукупний вибір обмежено бюджетом. Точний перебір усіх варіантів має експоненційну складність за кількістю пар «компонент – механізм», тому для систем корпоративного масштабу застосовують наближені методи розв'язання [21].

Водночас вартість захисту є не лише обмеженням, а й самостійним критерієм, який бажано мінімізувати. Тому задача по суті двокритеріальна: одночасно максимізувати захищеність Z і мінімізувати витрати C . Її розв'язком є не єдина точка, а множина Парето-оптимальних розподілів, що утворюють межу компромісу «захищеність – вартість» [39, 45, 109]. Розподіл вважається Парето-оптимальним, якщо неможливо підвищити захищеність, не збільшивши витрат, і неможливо зменшити витрати, не знизивши захищеності. Особа, яка приймає рішення, обирає на цій межі робочу точку, що відповідає прийнятному для корпоративної структури бюджету безпеки.

Практично Парето-межу зручно будувати жадібним алгоритмом за маргіальною ефективністю витрат. Для кожного ще не застосованого засобу обчислюється відношення приросту захищеності до його вартості:

$$\rho_{ij} = \frac{\Delta Z_{ij}}{c_j}, \quad (4.14)$$

де ΔZ_{ij} – приріст цільової функції Z від застосування j -го механізму до i -го компонента. На кожному кроці до розподілу додається засіб із найбільшим значенням ρ_{ij} , що вкладається в залишок бюджету й часу; отримана точка фіксується на Парето-межі, і крок повторюється, доки ресурси не вичерпано. Такий підхід дає розподіл, близький до Парето-оптимального, за прийнятних обчислювальних витрат. Послідовність кроків алгоритму подано на рисунку 4.2.

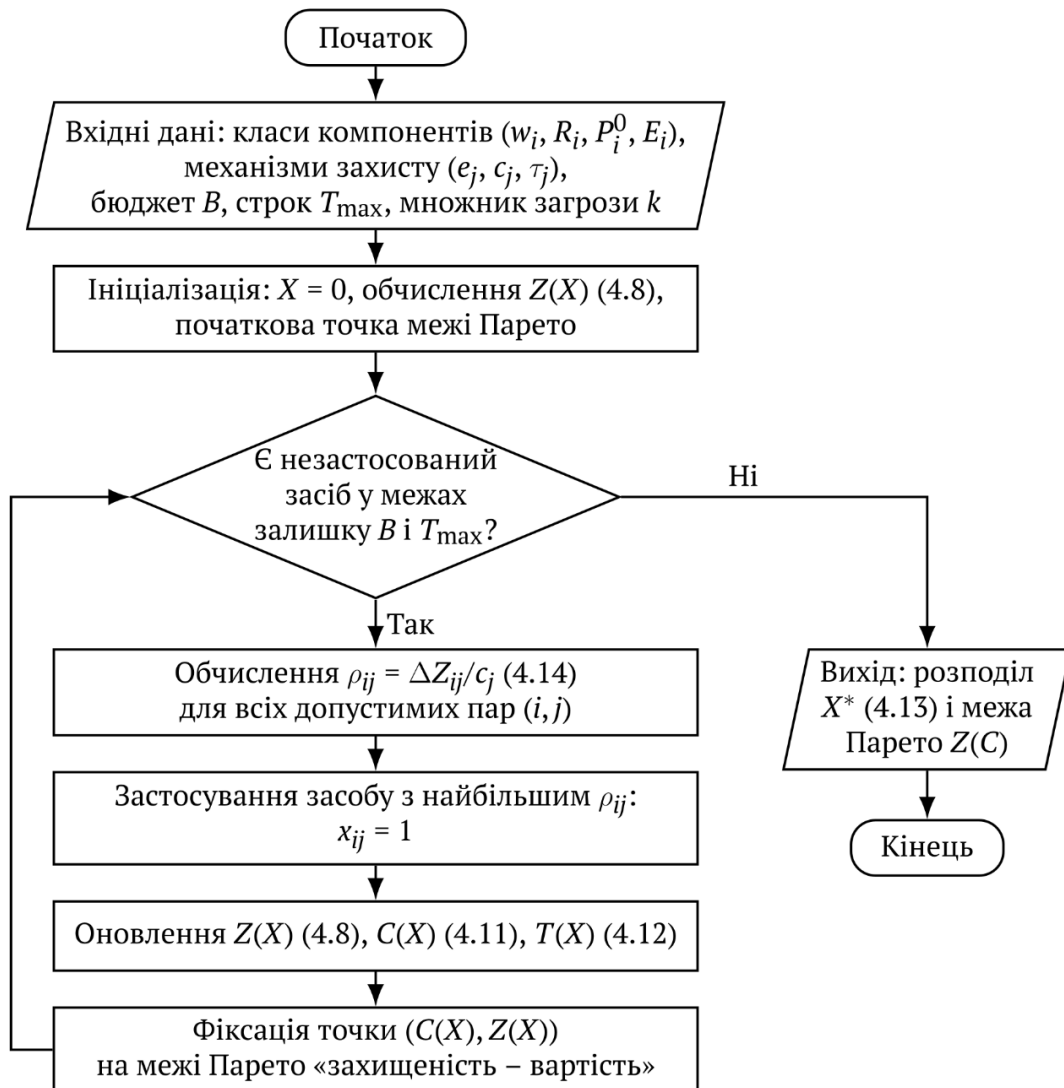


Рис. 4.2. Блок-схема алгоритму оптимізації розподілу ресурсів безпеки за маргіальною ефективністю витрат

Парето-межа має характерну увігнуту форму зі спадною віддачею: перші вкладення в захист дають найбільший приріст захищеності, а подальші – дедалі

менший. Оптимізований розподіл переважає довільний або рівномірний: за однакового бюджету цілеспрямований вибір засобів забезпечує вищий рівень захищеності, а для досягнення заданого рівня захищеності потребує менших витрат. Саме в цьому полягає практична цінність оптимізації розподілу ресурсів безпеки: вона перетворює обмежений бюджет на максимально можливий захисний ефект, спрямовуючи кошти передусім на найкритичніші й найвразливіші компоненти.

Отримана межа компромісу має прозоре економічне тлумачення в термінах оптимальних інвестицій в інформаційну безпеку [62, 144]: вона кількісно виражає «ціну захищеності» – приріст рівня безпеки на одиницю витрат – і допомагає обґрунтувати доцільний обсяг вкладень, за межами якого додаткові витрати вже не виправдані отриманим ефектом [14]. За підвищення рівня загрози, що враховується множителем у моделі ймовірності атаки, межа зміщується: той самий бюджет забезпечує нижчу захищеність, тож у періоди загострення загроз обґрунтованим є перегляд розподілу ресурсів.

У межах методу структурно-параметричного проектування оптимальний розподіл ресурсів безпеки є параметричною складовою проектного рішення. Отриманий розподіл X^* задає рівень захищеності Z , що входить у композитний критерій вибору архітектури (4.5), а через ефективність протидії впливає на інтегральний ризик системи.

Отже, розподіл обмежених ресурсів безпеки між компонентами розподіленої системи сформульовано як багатокритеріальну задачу максимізації захищеності за мінімізації витрат з урахуванням бюджетних і часових обмежень. Її розв'язком є Парето-оптимальна межа компромісу «захищеність – вартість», а наближену побудову цієї межі забезпечує жадібний алгоритм за маргінальною ефективністю витрат, що обґрунтовано спрямовує ресурси на найкритичніші компоненти й формує параметричну складову проектного рішення.

4.3. Модель ймовірності успішної атаки

Ключовим чинником рівня захищеності в цільовій функції (4.8) є ймовірність успішної атаки на компонент. Її оцінювання спирається на дві складові: наскільки компонент вразливий сам по собі та наскільки застосовані засоби захисту знижують цю вразливість за поточного рівня загрози. Базова ймовірність успішної атаки P_i^0 характеризує компонент до впровадження захисту й визначається з аналізу можливих шляхів компрометації. Для її оцінювання застосовують графи атак, що формалізують послідовності експлуатації вразливостей на шляху до цілі [140], дерева атак і захисту, які поєднують дії порушника з контрзаходами [87], а також стандартизовані оцінки критичності вразливостей [48]. За значної невизначеності вихідних даних доцільними є нечітко-логічні моделі оцінювання ризику компрометації [92]. Отримана в такий спосіб базова ймовірність слугує входом для моделі, що враховує захист і рівень загрози.

Ймовірність успішної атаки на i -й клас компонентів за поточного розподілу засобів захисту подається як

$$P_i(X) = \min(1, P_i^0(1 - \pi_i(X))k), \quad (4.15)$$

де P_i^0 – базова ймовірність успішної атаки, $\pi_i(X)$ – частка знятої вразливості засобами захисту за розподілом X згідно з (4.10), а k – множник підвищеної загрози. Множник $(1 - \pi_i(X))$ виражає залишкову вразливість компонента після застосування захисту: що ефективніший набір засобів, то меншою є ймовірність успішної атаки. Обмеження зверху одиницею відображає те, що ймовірність не може перевищувати достовірної події навіть за високого рівня загрози. Модель (4.15) узагальнює раніше обґрунтовану залежність, у якій вплив захисту та підвищеного рівня загрози на ймовірність компрометації подано в аналогічній мультиплікативній формі [164].

Множник підвищеної загрози враховує, що інтенсивність і цілеспрямованість атак змінюються з часом і зовнішніми обставинами. Він подається як зважена сума чинників загрози:

$$k = 1 + \sum_{l=1}^L a_l g_l, g_l \in [0,1], \quad (4.16)$$

де g_l – нормовані показники чинників загрози, a_l – калібровані вагові коефіцієнти, L – кількість чинників. За штатного фоновому рівня всі чинники близькі до нуля, і множник дорівнює одиниці; за загострення обстановки, зокрема під час тривалих цілеспрямованих кампаній, чинники зростають, і множник збільшується, підвищуючи залишковий ризик навіть за наявного захисту.

Ця конструкція узагальнює раніше запропонований множник воєнного часу, знімаючи прив'язку до бойових дій і переносячи її на корпоративне середовище, де роль підсилювальних чинників відіграють інтенсивність загроз безпеці, ступінь відкритості системи та мотивація порушника [164, 168].

Чинники, враховані у моделі множника, наведено в таблиці 4.4.

Таблиця 4.4

Чинники множника підвищеної загрози

Чинник загрози	Позначення	Зміст
Інтенсивність загроз безпеці	g_1	Частота й масштаб атак у поточний період
Ступінь відкритості системи	g_2	Розмір поверхні атаки, доступність ззовні
Мотивація порушника	g_3	Цінність активів організації для зловмисника

Для оцінювання захищеності системи в цілому окремі ймовірності агрегуються у зважену ймовірність успішної атаки:

$$\bar{P}(X) = \sum_{i=1}^n w_i P_i(X), \quad (4.17)$$

де ваги w_i збігаються з вагами важливості класів у цільовій функції захищеності. Показник $\bar{P}(X)$ монотонно зростає з підвищенням множника загрози й монотонно спадає з розширенням застосованого захисту, що відповідає інтуїтивним вимогам до моделі: посилення захисту знижує очікувану

ймовірність компрометації, а загострення загрози її підвищує. Послідовність обчислення ймовірності успішної атаки на компонент від базової оцінки до агрегованого показника подано на рисунку 4.3.

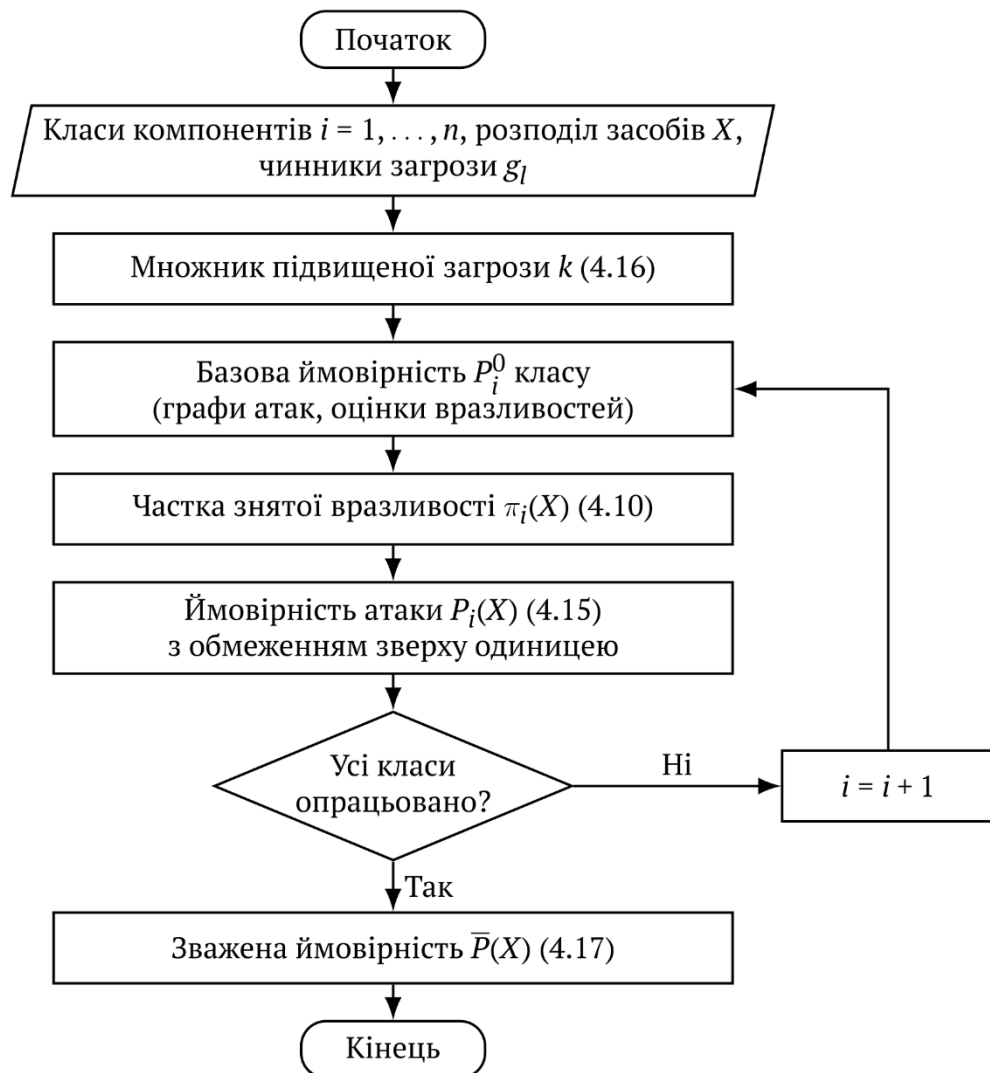


Рис. 4.3. Блок-схема моделі ймовірності успішної атаки на компонент

Модель ймовірності успішної атаки посідає центральне місце в апараті методу. З одного боку, вона визначає рівень захищеності (4.8) і через нього впливає на композитний критерій вибору архітектури; з іншого боку, вона є основою оцінювання інтегрального ризику, розглянутого в підрозділі 4.5, де ймовірність компрометації поєднується з потенційним збитком.

Застосовані засоби захисту знижують ймовірність атаки не лише статично: адаптивні механізми виявлення загроз з часом підвищують ефективність протидії, а отже, знижують ефективну ймовірність успішної компрометації в

періоди активних атак, тож модель ймовірності атаки узгоджується з динамічним характером загроз, за якого захист має не разово, а неперервно адаптуватися до обстановки [14, 112].

Отже, ймовірність успішної атаки на компонент розподіленої системи змодельовано як добуток базової вразливості, залишкової вразливості після застосування захисту та множника підвищеної загрози з подальшим агрегуванням у зважений показник рівня системи. Модель монотонно реагує на посилення захисту й на зміну рівня загрози, що робить її придатною основою для оцінювання захищеності та ризику при структурно-параметричному проєктуванні.

4.4. Моделі живучості, адаптивності та плавної деградації

Здатність розподіленої обчислювальної системи зберігати критичну функціональність за втрати компонентів забезпечується трьома взаємопов'язаними механізмами: структурним резервуванням, поведінковою адаптивністю та функціональною пивною деградацією. Резервування визначає, скільки екземплярів кожного класу компонентів має система й скільки з них потрібно для збереження працездатності класу; адаптивність характеризує здатність системи переналаштовуватися у відповідь на втрати й нові загрози; плавна деградація описує, наскільки поступово знижується функціональність за наростання втрат. Разом ці механізми формують живучість системи як інтегральну властивість.

Структурне резервування описується схемою «k-з-n»: клас зберігає працездатність, якщо справними лишаються щонайменше k_i з n_i його екземплярів. Надійність класу за такої схеми обчислюється як

$$R_i = \sum_{m=k_i}^{n_i} C_{n_i}^m r_i^m (1 - r_i)^{n_i-m}, \quad (4.18)$$

де r_i – надійність окремого екземпляра, n_i – загальна кількість екземплярів класу, k_i – мінімальна кількість справних екземплярів, потрібна для працездатності

класу, а $C_{n_i}^m$ – кількість сполучень. Формула (4.18) показує, що навіть за помірної надійності окремого екземпляра достатня кратність резервування різко підвищує надійність класу, оскільки одночасна відмова багатьох екземплярів малоімовірна [91, 151]. Вибір n_i та k_i для кожного класу є структурною складовою проєктного рішення й безпосередньо впливає на живучість системи.

Системна живучість поєднує структурне резервування з адаптивністю та стійкістю до зовнішніх збурень. Її подано у вигляді

$$S(t) = \prod_{i=1}^n [1 - (1 - R_i(t))^{n_i}] A(t) V(t), \quad (4.19)$$

де $R_i(t)$ – надійність екземплярів i -го класу в момент t , n_i – кратність резервування, $A(t)$ – функція адаптивності системи, а $V(t)$ – функція стійкості до зовнішніх збурень (екранування, резервні канали зв'язку, автономне живлення). Добуток за класами відображає те, що система зберігає живучість лише за працездатності всіх критичних класів, а множники $A(t)$ і $V(t)$ підсилюють живучість завдяки здатності системи переналаштовуватися й протистояти впливам. Ця форма узагальнює раніше обґрунтовану модель інтегрального ризику, знімаючи прив'язку до кіберфізичного моніторингу й переносячи її на розподілену корпоративну обчислювальну систему [164, 168]. Параметри моделей живучості, адаптивності та плавної деградації узагальнено в таблиці 4.5.

Таблиця 4.5

Параметри моделей живучості, адаптивності та плавної деградації

Параметр	Позначення	Зміст
Надійність екземпляра	r_i	Імовірність справності окремого екземпляра класу
Кратність резервування	n_i, k_i	Кількість екземплярів і потрібний мінімум для працездатності
Початкова адаптивність	A_0	Здатність до переналаштування на початку
Коефіцієнт деградації адаптивності	дельта	Швидкість втрати адаптивності з часом
Інтенсивність навчання	бета	Швидкість накопичення досвіду протидії
Швидкість зниження функціональності	лямбда	Темп відключення сервісів за втрат
Темп відновлення	мю	Швидкість повернення функціональності
Критичний рівень функціональності	Φ_{crit}	Поріг збереження найважливіших сервісів

Адаптивність системи описує її здатність пристосовуватися до нових загроз і компенсувати втрати. Її подано як

$$A(t) = A_0 \exp(-\delta t) + (1 - \exp(-\beta E(t))), \quad (4.20)$$

де A_0 – початкова адаптивність, δ – коефіцієнт деградації адаптивності внаслідок зношення й накопичення інформаційного шуму, β – інтенсивність навчання, а $E(t)$ – накопичений досвід протидії загрозам. Перший доданок відображає природне зниження адаптивності з часом, другий – її зростання завдяки навчанню й накопиченню відпрацьованих сценаріїв реагування. Проектні заходи спрямовують на зменшення δ через обслуговування й оновлення моделей та на підвищення β через прискорені цикли оновлення й розподілене навчання, що узгоджується з концепцією самоадаптивних та автономних систем [82, 32].

Плавність зниження функціональності за наростання втрат характеризується коефіцієнтом плавної деградації

$$D(t) = \frac{F_{min}}{F_{max}} \exp(-\lambda t) + (1 - \exp(-\mu R_{rate}(t))), \quad (4.21)$$

де F_{min} і F_{max} – мінімальний і максимальний рівні функціональності, λ – швидкість відключення некритичних сервісів, μ – темп відновлення, а $R_{rate}(t)$ – миттєва швидкість повернення сервісів. Співвідношення параметрів λ і μ визначає характер деградації: за переважання швидкості відключення над темпом відновлення система «зривається» у різкий обвал функціональності, тоді як достатній темп відновлення завдяки резервним ресурсам і локальній аналітиці забезпечує контрольовану, плавну деградацію [139].

Значення функцій живучості $S(t)$, адаптивності $A(t)$ та плавної деградації $D(t)$ обмежуються зверху одиницею (нормуються до відрізка $[0,1]$), оскільки описують нормовані частки збереженої спроможності, а додатний доданок навчання не виводить їх за цю межу. Мінімальний рівень функціональності F_{min} дають так, щоб гарантувати виконання найважливіших сервісів навіть за втрати значної частки компонентів, що відповідає умові збереження критичної функціональності

$$\Phi(f) \geq \Phi_{\text{crit}}, f \leq f_{\text{max}}, \quad (4.22)$$

де $\Phi(f)$ – функціональність за частки втрачених компонентів f , Φ_{crit} – критичний рівень, а f_{max} гранична частка втрат, за якої критична функціональність ще зберігається. Саме показник f_{max} ражає відмовостійкість архітектури в термінах допустимих втрат. Послідовність оцінювання живучості, адаптивності та плавної деградації на етапі проектування подано на рисунку 4.4.

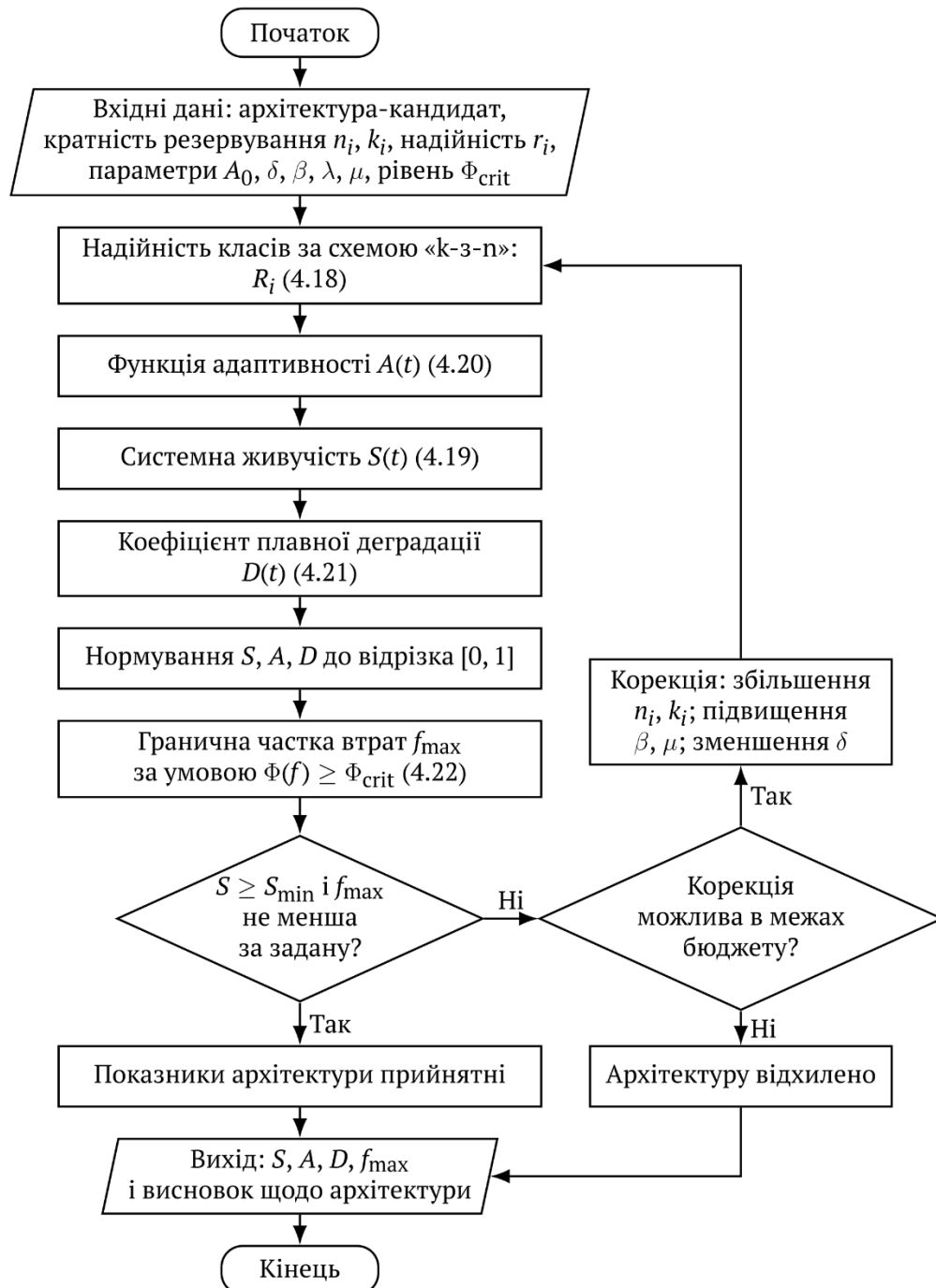


Рис. 4.4. Блок-схема оцінювання живучості, адаптивності та плавної деградації архітектури на етапі проектування

Три розглянуті механізми поєднуються при структурно-параметричному проєктуванні. Резервування «к-з-п» є структурною складовою: збільшення кратності резервування критичних класів підвищує їхню надійність за (4.18) і зсуває граничну частку втрат f_{max} бік більших значень. Параметри адаптивності та плавної деградації є параметричною складовою: налаштування темпів навчання й відновлення визначає, наскільки швидко система відновлює функціональність і наскільки плавно її втрачає. Спільно вони забезпечують виконання вимоги до мінімального рівня живучості, закладеної в обмеженнях задачі проєктування. Важливо, що надмірне резервування підвищує живучість ціною зростання вартості, тому кратність резервування підбирають узгоджено з бюджетними обмеженнями, а не максимізують безумовно.

Отже, живучість розподіленої обчислювальної системи змодельовано як поєднання структурного резервування за схемою «к-з-п», поведінкової адаптивності та функціональної плавної деградації, що спільно забезпечують збереження критичної функціональності за втрати значної частки компонентів. Подані моделі дають кількісні показники для порівняння архітектур за відмовостійкістю й задають структурні та параметричні важелі, якими проєктувальник керує рівнем живучості в межах допустимих витрат.

4.5. Модель оцінювання ризиків

Ризик слугує узагальненим показником, що поєднує ймовірність несприятливих подій з їх наслідками й дозволяє порівнювати проєктні рішення за очікуваними втратами. У розподіленій обчислювальній системі джерелом ризику є як кіберзагрози, так і відмови компонентів, тому оцінювання ризику спирається на усталені підходи до аналізу інформаційних ризиків [75, 122] та кількісні моделі очікуваних втрат [50]. Внесок окремого класу компонентів в інтегральний ризик подається як добуток імовірності реалізації загрози, потенційного збитку та частки нейтралізованого впливу:

$$r_i = q_i I_i (1 - M_i), \quad (4.23)$$

де q_i – ймовірність реалізації загрози для i -го класу, I_i – потенційний збиток від його компрометації, а M_i – ефективність протидії, що зменшує наслідки. Множник $(1 - M_i)$ виражає залишкову частку ризику після впровадження захисних заходів: чим вища ефективність протидії, тим меншим є очікуваний збиток за тієї самої ймовірності загрози.

Ефективність протидії пов'язана з розподілом ресурсів безпеки: застосування захисних механізмів до компонента підвищує його захищеність і водночас знижує наслідки можливої компрометації. Цей зв'язок подано як

$$M_i = M_i^0 + (M_i^{max} - M_i^0) \pi_i(X), \quad (4.24)$$

де M_i^0 – базова ефективність протидії за відсутності цілеспрямованого захисту, M_i^{max} – гранично досяжна ефективність, а $\pi_i(X)$ – частка знятої вразливості за розподілом засобів згідно з (4.10). Таким чином, чим повніше захищено компонент, тим ближча ефективність протидії до граничної, і тим меншим є його внесок у ризик. Інтегральний ризик системи обчислюється як сума внесків усіх класів з урахуванням поточного рівня загрози:

$$R(X) = k \sum_{i=1}^n r_i, \quad (4.25)$$

де k – множник підвищеної загрози згідно з (4.16). Форма (4.25) узагальнює модель інтегрального ризику, поєднуючи ймовірності загроз, збитки та ефективність протидії в єдиному показнику, чутливому до рівня загрози [164]. Складові моделі оцінювання ризику узагальнено в таблиці 4.6.

Таблиця 4.6

Складові моделі оцінювання інтегрального ризику

Складова	Позначення	Зміст
Ймовірність загрози	q_i	Ймовірність реалізації загрози для i -го класу
Потенційний збиток	I_i	Втрати від компрометації i -го класу
Ефективність протидії	M_i	Частка нейтралізованого впливу загрози
Множник загрози	k	Підсилення ризику за загострення обстановки
Інтегральний ризик	R	Сукупні очікувані втрати системи

Отриманий показник ризику виконує в методі подвійну роль. По-перше, він входить із від'ємним знаком у композитний критерій вибору архітектури (4.5), урівноважуючи прагнення підвищити живучість і захищеність необхідністю утримувати очікувані втрати в прийнятних межах. По-друге, він задає проєктне обмеження на гранично допустимий рівень ризику:

$$R(X) \leq R_{max}, \quad (4.26)$$

де R_{max} максимально прийнятний рівень ризику для конкретної корпоративної структури. Обмеження (4.26) відсікає архітектури, що навіть за формально високих показників живучості й захищеності лишають неприйнятно високі очікувані втрати. Економічний вимір ризику – співвідношення залишкового ризику й витрат на безпеку, оптимальний обсяг вкладень та віддача від них; тут ризик інтегровано безпосередньо в критерій проєктування, що дозволяє обґрунтовувати проєктні рішення за критерієм мінімізації очікуваних втрат за обмежених ресурсів [62, 144, 167]. Послідовність оцінювання інтегрального ризику подано на рисунку 4.5.

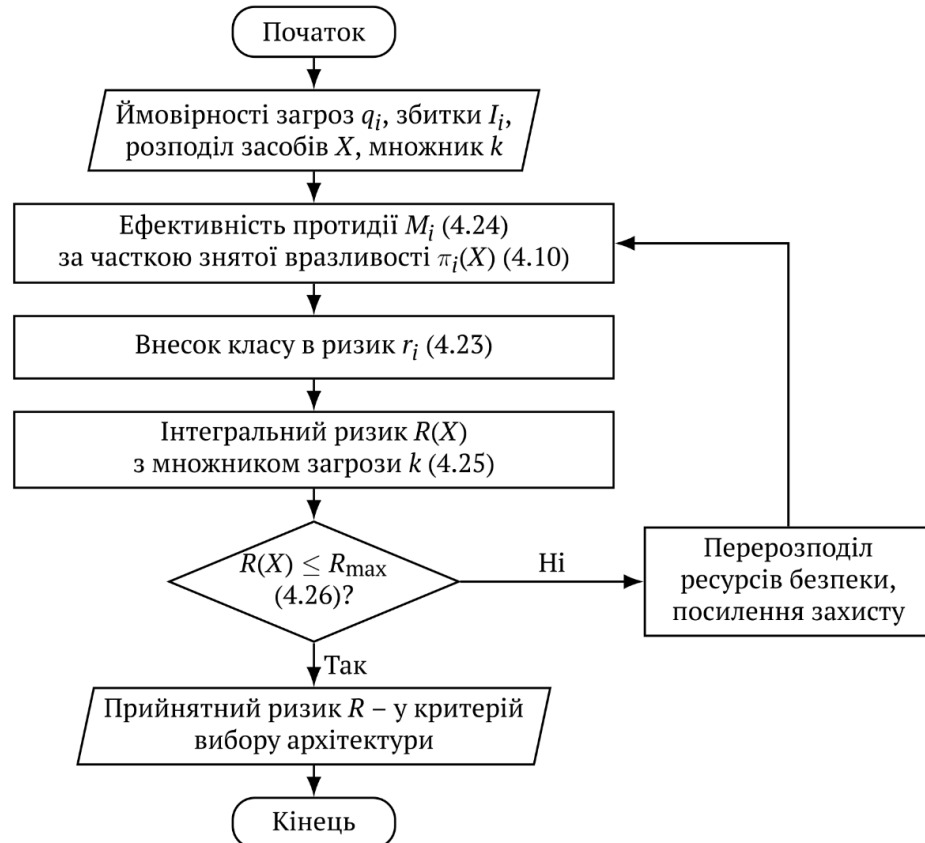


Рис. 4.5. Блок-схема оцінювання інтегрального ризику розподіленої системи

Модель ризику узгоджує решту складових методу між собою. Підвищення захищеності через розподіл ресурсів безпеки (4.24) знижує ризик, але потребує витрат; посилення резервування підвищує живучість, проте також збільшує вартість; надмірні вкладення в захист зменшують ризик усе повільніше. Через це саме показник ризику разом із бюджетним обмеженням стримує як недостатнє, так і надмірне інвестування в безпеку, спрямовуючи проєктне рішення до збалансованого варіанта.

Отже, оцінювання ризику розподіленої обчислювальної системи зведено до інтегрального показника очікуваних втрат, що поєднує ймовірності загроз, потенційні збитки та ефективність протидії з урахуванням рівня загрози. Цей показник входить у критерій проєктування й у проєктні обмеження, забезпечуючи обґрунтування архітектурних рішень за критерієм мінімізації очікуваних втрат у межах доступних ресурсів.

4.6. Механізми забезпечення захищеності

Захищеність розподіленої обчислювальної системи забезпечується не окремим засобом, а сукупністю взаємодоповнювальних механізмів, інтегрованих в усі рівні архітектури за принципом багаторівневого захисту. Три такі механізми складають основу захисту: постквантова криптографія, архітектура нульової довіри та адаптивне машинне навчання для виявлення загроз. Кожен із них закриває власний клас загроз, а спільно вони формують ешелоновану оборону, за якої подолання одного рубежу не відкриває зловмиснику доступу до системи в цілому. Залишкова вразливість компонента, захищеного набором механізмів, за принципом ешелонованої оборони визначається добутком часток непокритих вразливостей:

$$\eta_i = \prod_{j \in J_i} (1 - e_j), \quad (4.27)$$

де J_i – множина механізмів, застосованих до i -го компонента, а e_j – ефективність j -го механізму. Величина η_i пов'язана з часткою знятої вразливості

співвідношенням $\pi_i = 1 - \eta_i$, тож нарощування набору механізмів послідовно зменшує залишкову вразливість, хоча й зі спадною віддачею. Призначення й основу кожного механізму захисту узагальнено в таблиці 4.7.

Таблиця 4.7

Механізми забезпечення захищеності розподіленої системи

Механізм	Призначення	Основа
Постквантова криптографія	Довгострокова конфіденційність і цілісність даних	Стандарти FIPS 203, 204, 205
Архітектура нульової довіри	Мінімізація довіри, обмеження бічного руху	NIST SP 800-207
Адаптивне виявлення загроз ML-	Виявлення нових аномалій і атак у реальному часі	Поведінкова аналітика, машинне навчання

Постквантова криптографія забезпечує довгострокову стійкість захисту даних до атак із застосуванням квантових обчислень. Класичні алгоритми з відкритим ключем, стійкість яких ґрунтується на складності факторизації чи дискретного логарифмування, втрачають надійність із появою достатньо потужних квантових комп'ютерів, тоді як постквантові алгоритми спираються на задачі, стійкі до квантових атак [20]. У 2024 році відповідні алгоритми стандартизовано: механізм інкапсуляції ключів на основі ґраток, цифровий підпис на основі ґраток та цифровий підпис на основі геш-функцій [119–121]. Для розподіленої корпоративної системи постквантовий захист критичний передусім для даних тривалого зберігання й довгоживучих каналів, перехоплення яких сьогодні може бути використане після появи квантових засобів дешифрування.

Архітектура нульової довіри ґрунтується на принципі відсутності неявної довіри: кожен запит на доступ, незалежно від його джерела, розташування чи статусу користувача, проходить автентифікацію, авторизацію та перевірку контексту безпеки [133]. Мікросегментація мережі, надання мінімально необхідних прав і перевірка кожної взаємодії обмежують бічний рух злоумисника: навіть компрометувавши один вузол, він не отримує безперешкодного доступу до інших сегментів. Для розподіленої системи з багатьма вузлами й каналами такий підхід суттєво знижує ймовірність

поширення атаки територією системи й узгоджується з моделлю ймовірності успішної атаки, у якій захист компонента знижує його вразливість незалежно від інших.

Адаптивне машинне навчання забезпечує виявлення загроз, зокрема раніше невідомих, шляхом неперервного аналізу поведінки системи, трафіку та подій безпеки [15, 135]. На відміну від сигнатурних методів, поведінкова аналітика виявляє відхилення від нормального профілю роботи й здатна реагувати на нові типи атак без ручного втручання [112]. Важливою властивістю адаптивних методів є те, що їхня здатність до виявлення зростає з накопиченням досвіду: у міру донавчання моделей імовірність вчасного виявлення підвищується від початкового рівня до асимптотичного, що описується залежністю

$$d(t) = d_{max} - (d_{max} - d_0) \exp(-\gamma t), \quad (4.28)$$

де d_0 – початкова здатність до виявлення, d_{max} асимптотично досяжний рівень, а γ – інтенсивність навчання. Ця залежність узгоджується з функцією адаптивності (4.20): накопичення досвіду протидії підвищує як адаптивність системи, так і ефективність виявлення загроз. Інтеграцію трьох механізмів у єдиний контур обробки запиту подано на рисунку 4.6.

У межах методу структурно-параметричного проєктування розглянуті механізми виступають засобами захисту, розподіл яких між компонентами оптимізується в задачі підрозділу 4.2. Кожен механізм характеризується власною ефективністю та вартістю, а їх спільне застосування до компонента формує ешелоновану оборону з залишковою вразливістю (4.27). Вибір складу механізмів для кожного компонента є частиною параметричного проєктного рішення, тоді як інтеграція підсистеми безпеки в усі рівні архітектури належить до структурних рішень.

Узгоджене застосування постквантової криптографії, нульової довіри та адаптивного виявлення забезпечує стійкість системи до різномірних загроз – від перехоплення даних до цілеспрямованих кампаній, зокрема в координації з національними службами реагування на інциденти [157, 57].

Отже, захищеність розподіленої обчислювальної системи забезпечується інтеграцією постквантової криптографії, архітектури нульової довіри та адаптивного машинного навчання в багаторівневий захист, залишкова вразливість якого спадає з нарощуванням ешелонів оборони. Ці механізми є засобами, розподіл яких оптимізується при проєктуванні, а їх поєднання закриває взаємодоповнювальні класи загроз і підвищує загальний рівень захищеності системи.

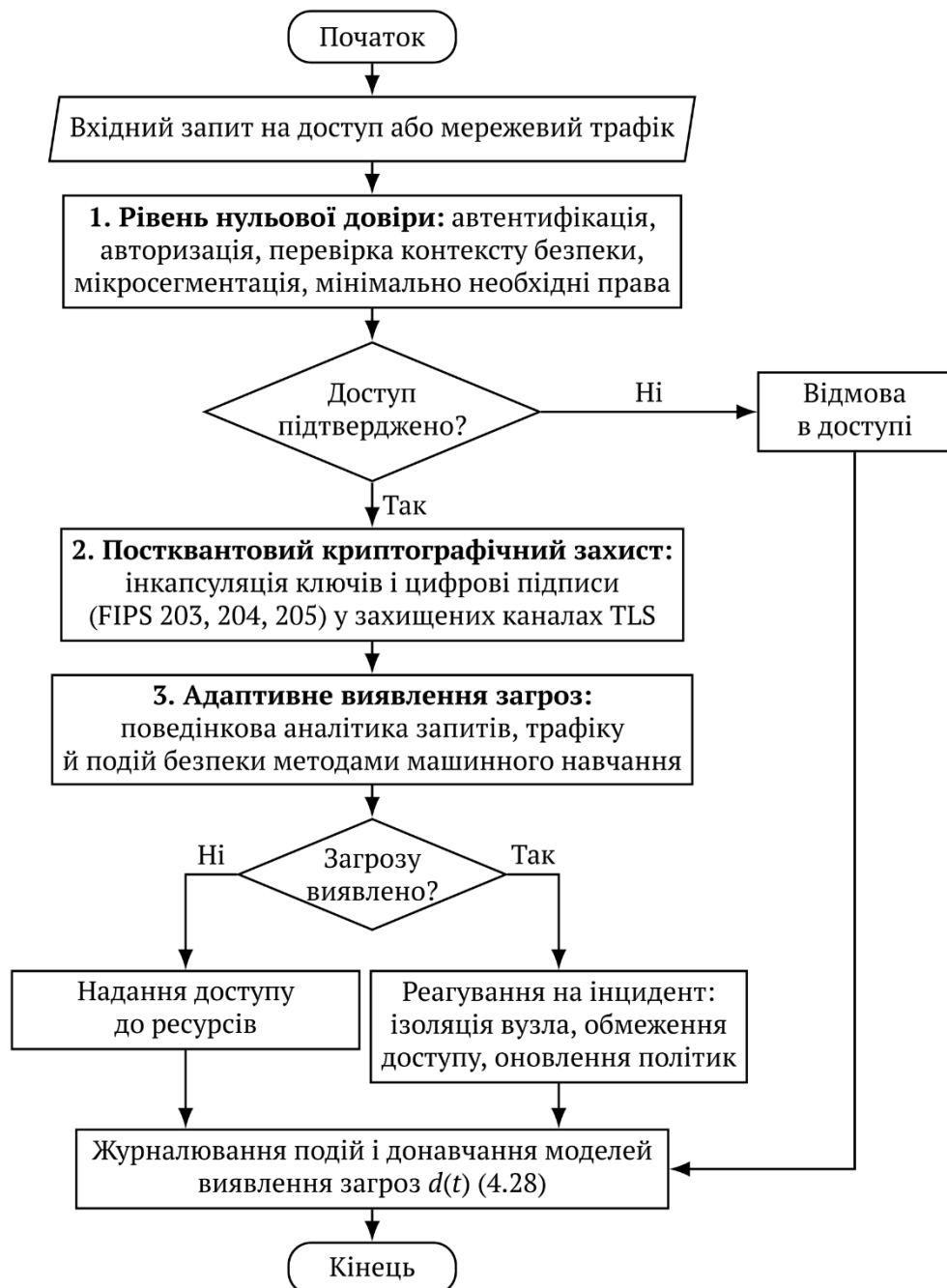


Рис. 4.6. Блок-схема багаторівневого захисту: нульова довіра, постквантова криптографія та адаптивне ML-виявлення загроз

4.7. Метод та інформаційна технологія структурно-параметричного проєктування відмовостійких і захищених систем

Розглянуті в попередніх підрозділах складові поєднуються в цілісний метод структурно-параметричного проєктування, який на етапі проєктування обирає архітектуру розподіленої обчислювальної системи з наперед заданими показниками живучості й захищеності. Метод спирається на постановку задачі (підрозділ 4.1), у якій архітектуру подано структурно-параметричною парою, а якість рішення оцінюють композитним критерієм. Для кожної архітектури-кандидата оптимізують розподіл ресурсів безпеки (підрозділ 4.2), обчислюють рівень захищеності через модель ймовірності успішної атаки (підрозділ 4.3), оцінюють живучість і плавну деградацію за втрати компонентів (підрозділ 4.4), визначають інтегральний ризик (підрозділ 4.5) та закладають механізми захисту (підрозділ 4.6). Отримані показники зводять до єдиної оцінки, за якою й обирають архітектуру.

Оскільки показники мають різну природу й одиниці, перед згорткою їх нормують до спільної шкали. Для кожного показника-вигоди нормоване значення обчислюється як

$$\hat{c}_l = \frac{c_l - c_l^{\min}}{c_l^{\max} - c_l^{\min}}, \quad (4.29)$$

де c_l – значення l -го показника для архітектури, а $c_l^{\min}$ та $c_l^{\max}$ його найменше й найбільше значення серед кандидатів; для показників-витрат нормування виконують у зворотному напрямі, щоб більшій вигоді відповідало більше нормоване значення. Композитний критерій вибору подається як зважена сума нормованих показників:

$$G(\mathcal{A}) = \sum_{l=1}^L \lambda_l \hat{c}_l(\mathcal{A}), \quad \sum_{l=1}^L \lambda_l = 1, \quad (4.30)$$

де λ_l – вагові коефіцієнти пріоритетності показників, що відображають значущість кожного критерію для конкретної корпоративної структури. Критерії вибору архітектури та напрям їх оптимізації узагальнено в таблиці 4.8.

Критерії багатокритеріального вибору архітектури

Критерій	Напрямок оптимізації	Роль у проектуванні
Захищеність Z	Максимізація	Стійкість до загроз безпеці
Живучість	Максимізація	Збереження функціональності за втрат
Плавність деградації	Максимізація	Контрольоване зниження функціональності
Інтегральний ризик	Мінімізація	Обмеження очікуваних втрат
Вартість	Мінімізація	Економічність рішення

Остаточне проєктне рішення визначається як архітектура з найбільшим значенням композитного критерію серед скінченної множини кандидатів:

$$\mathcal{A}^* = \arg \max_{\mathcal{A} \in \{\mathcal{A}_1, \dots, \mathcal{A}_M\}} G(\mathcal{A}), \quad (4.31)$$

де $\{\mathcal{A}_1, \dots, \mathcal{A}_M\}$ – множина кандидатних архітектур, сформована варіюванням структурних рішень (топологія, кратність резервування, розподіл за рівнями) та параметричних рішень (розподіл ресурсів безпеки, порогові значення). Такий вибір урівноважує суперечливі критерії: архітектура з надмірним резервуванням і захистом дає високу живучість і захищеність, проте програє за економічністю, тоді як спрощена архітектура економна, але вразлива; композитний критерій відбирає збалансований варіант.

Перед остаточним вибором кожен архітектур-кандидат перевіряють на цифровому двійнику – імітаційній моделі системи, що відтворює її поведінку за різних сценаріїв втрат і загроз без ризику для реальної інфраструктури [44, 52, 149]. У запропонованому методі цифровий двійник побудовано з двох попередніх результатів дисертації. Продуктивнісну складову двійника становить сурогатна модель гаусівської процесової регресії: для кожного сценарію втрат вона прогнозує р95-затримку за навантаження, перерозподіленого на справні обчислювальні вузли, разом із верхньою довірчою межею (2.23). Складову стійкості становить інформаційна технологія забезпечення функціональної стійкості: вона відтворює траєкторію показника якості (3.1) кандидата за сценаріями збурень із перемаршрутизацією інформаційних потоків і контролем загроз і дає індекс функціональної стійкості (3.3). Кандидат вважається допустимим, якщо виконуються умови

$$P_f(\mu(A, \omega) + z_\gamma \sigma(A, \omega) \leq L_{\max}) \geq P_{\min}, FSI(A) \geq FSI_{\min}, \quad (4.32)$$

де $\mu(A, \omega)$ і $\sigma(A, \omega)$ – прогнозне середнє (2.11) і середньоквадратичне відхилення (2.12) р95-затримки архітектури A у сценарії ω , z_γ – квантиль стандартного нормального розподілу, $L_{\max}$ – гранично допустима затримка, P_f – частка сценаріїв утрати частки f компонентів, за яких умова на затримку виконується, $P_{\min}$ – мінімально прийнятне значення цієї частки, $FSI(A)$ – індекс функціональної стійкості кандидата, $FSI_{\min}$ – його мінімально прийнятне значення. Перша умова гарантує, що за втрати частини вузлів продуктивність не вийде за допустимі межі з урахуванням невизначеності прогнозу, друга – що система з такою архітектурою зберігатиме критичну функціональність під збуреннями в експлуатації. Кандидатів, які не задовольняють умови (4.32), вилучають до обчислення композитного критерію, тож вибір (4.31) виконується лише серед допустимих архітектур.

Метод реалізується як послідовність кроків, що поєднують оптимізацію, оцінювання й вибір:

1. Формалізація вимог і побудова моделі системи згідно з постановкою.
2. Формування множини кандидатних архітектур варіюванням структурних і параметричних рішень та обґрунтування конфігурації обчислювальних вузлів сурогатною моделлю продуктивності.
3. Оптимізація розподілу ресурсів безпеки для кожної архітектури та обчислення рівня захищеності.
4. Оцінювання живучості, плавної деградації та інтегрального ризику кожної архітектури.
5. Перевірка кандидатів на цифровому двійнику, побудованому із сурогатної моделі продуктивності та інформаційної технології забезпечення функціональної стійкості, за умовами (4.32).
6. Нормування показників, обчислення композитного критерію та вибір архітектури з найбільшим його значенням.
7. Ухвалення архітектури з наперед заданими показниками живучості й захищеності як проектного рішення.

Повну послідовність методу подано на рисунку 4.7.

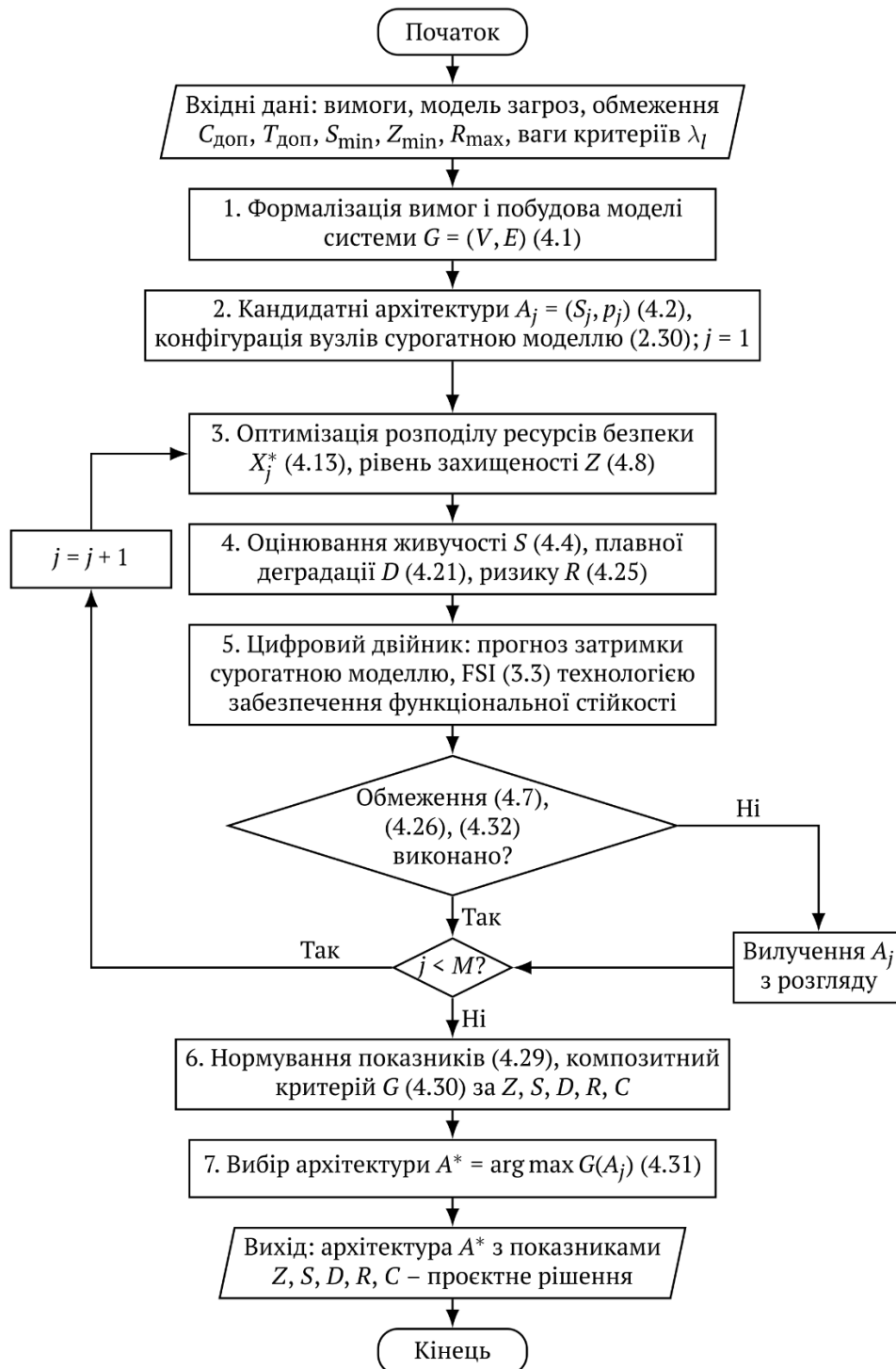


Рис. 4.7. Блок-схема методу структурно-параметричного проєктування відмовостійких і захищених розподілених обчислювальних систем

Ітерація за множиною кандидатних архітектур забезпечує повне оцінювання простору проєктних рішень, а композитний критерій зводить різномірні показники до єдиної підстави для вибору. Метод чітко розмежовує

структурні та параметричні важелі: структурними рішеннями (топологія, резервування, розподіл за рівнями) керують живучістю й відмовостійкістю, параметричними (розподіл ресурсів безпеки, порогові значення) – захищеністю й ризиком, а їх узгоджене налаштування дає архітектуру, збалансовану за всіма критеріями.

Для застосування в практиці проектування метод реалізовано у вигляді інформаційної технології структурно-параметричного проектування, яка перетворює вихідні вимоги до розподіленої обчислювальної системи на обґрунтоване проектне рішення засобами взаємопов'язаних програмних модулів, спільного сховища проектних даних і панелі підтримки ухвалення рішень. Якщо метод визначає послідовність математичних операцій, то інформаційна технологія визначає склад програмних засобів, вхідні та вихідні дані кожного з них, спосіб зберігання проміжних результатів і форму їх подання проектувальникові, тобто те, як метод виконується в автоматизованому режимі.

Інформаційну технологію подано як упорядковану сукупність семи модулів, кожен з яких реалізує окремий етап методу. Модуль формалізації перетворює вимоги, модель загроз і ресурсні обмеження на граф системи з класами компонентів і проектними обмеженнями, а модуль генерування формує множину кандидатних архітектур, варіюючи топологію, кратність резервування та розподіл компонентів за рівнями, і обґрунтовує конфігурацію обчислювальних вузлів сурогатною моделлю продуктивності.

Модулі оптимізації розподілу засобів захисту, оцінювання живучості та оцінювання ризику обчислюють для кожного кандидата рівень захищеності, ймовірність збереження критичної функціональності за статистичним моделюванням втрат та інтегральний ризик. Модуль цифрового двійника відтворює поведінку кандидата за сценаріями відмов, перевантажень і атак засобами сурогатної моделі продуктивності та інформаційної технології забезпечення функціональної стійкості й перевіряє умови (4.32), а модуль багатокритеріального вибору нормує показники, обчислює композитний

критерій і формує звіт для особи, яка приймає рішення. Склад модулів, їхні вхідні та вихідні дані наведено в таблиці 4.9.

Таблиця 4.9

Модулі інформаційної технології структурно-параметричного проектування

Модуль	Вхідні дані	Вихідні дані
М1. Формалізація вимог і моделі системи	Вимоги до продуктивності й доступності, модель загроз, бюджет, строки	Граф системи, класи компонентів, ваги важливості, проєктні обмеження
М2. Генерування кандидатних архітектур	Граф системи, допустимі структурні рішення, сурогатна модель продуктивності	Множина кандидатних архітектур, конфігурації обчислювальних вузлів
М3. Оптимізація розподілу засобів захисту	Класи компонентів, механізми захисту, бюджет, множник загрози	Розподіл засобів, рівень захищеності, межа Парето
М4. Оцінювання живучості й плавної деградації	Схема резервування, надійність екземплярів	Живучість, гранична частка втрат, коефіцієнт деградації
М5. Оцінювання ризику	Ймовірності загроз, збитки, ефективність протидії	Інтегральний ризик
М6. Цифровий двійник	Кандидатна архітектура, сценарії відмов і загроз, сурогатна модель продуктивності, технологія забезпечення функціональної стійкості	Прогноз затримки, індекс функціональної стійкості, допустимість кандидата
М7. Багатокритеріальний вибір і звітність	Показники кандидатів, ваги критеріїв	Рекомендована архітектура, рейтинг, дані для панелі

Програмну реалізацію модулів наведено в додатку Б, приклад панелі вибору архітектури – у додатку Г, а структуру інформаційної технології з модулями, сховищем проєктних даних і контуром погодження рішення – на рисунку 4.8.

Модулі обмінюються даними через спільне сховище проєктних даних, у якому зберігаються параметри класів компонентів і механізмів захисту, сценарії збурень та результати оцінювання кандидатів, завдяки чому кожен модуль можна перезапустити окремо за зміни вихідних даних без повторення всього розрахунку.

Результати подаються проєктувальникові на панелі вибору архітектури, що відображає ключові показники кандидатів, межу компромісу «захищеність –

вартість», криві живучості та рейтинг за композитним критерієм, а особа, яка приймає рішення, може скоригувати ваги критеріїв і проєктні обмеження й отримати перераховане рішення без повторного оцінювання кандидатів.

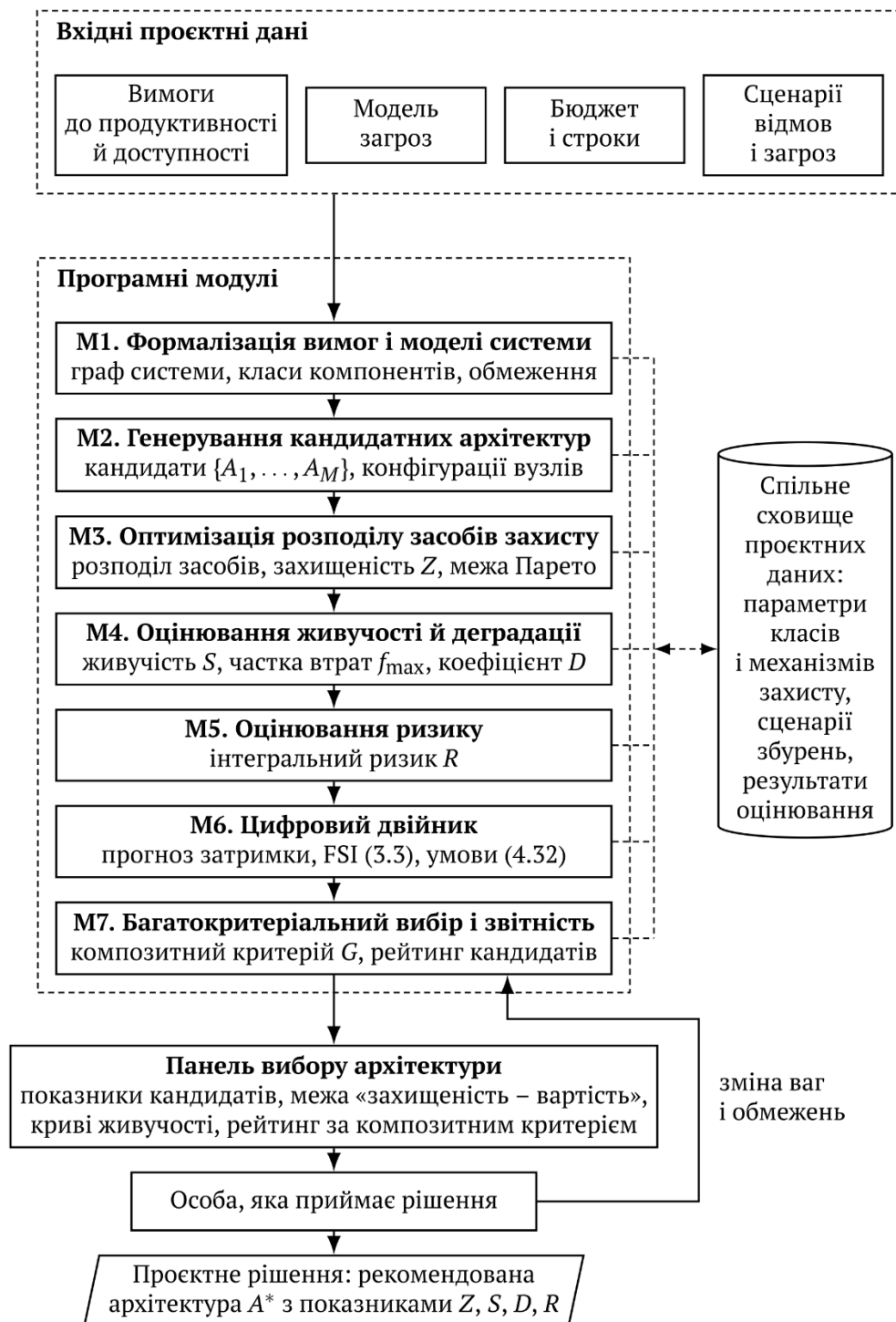


Рис. 4.8. Структура інформаційної технології структурно-параметричного проєктування відмовостійких і захищених розподілених обчислювальних систем

Взаємозв'язок методу та інформаційної технології структурно-параметричного проєктування з двома попередніми результатами дисертації подано на рисунку 4.9: сурогатна модель продуктивності забезпечує модуль генерування кандидатів обґрунтованими конфігураціями обчислювальних вузлів, а модуль цифрового двійника – прогнозом затримки з верхньою довірчою межею, тоді як інформаційна технологія забезпечення функціональної стійкості надає модулю цифрового двійника індекс функціональної стійкості кандидата, за яким перевіряють умови (4.32).

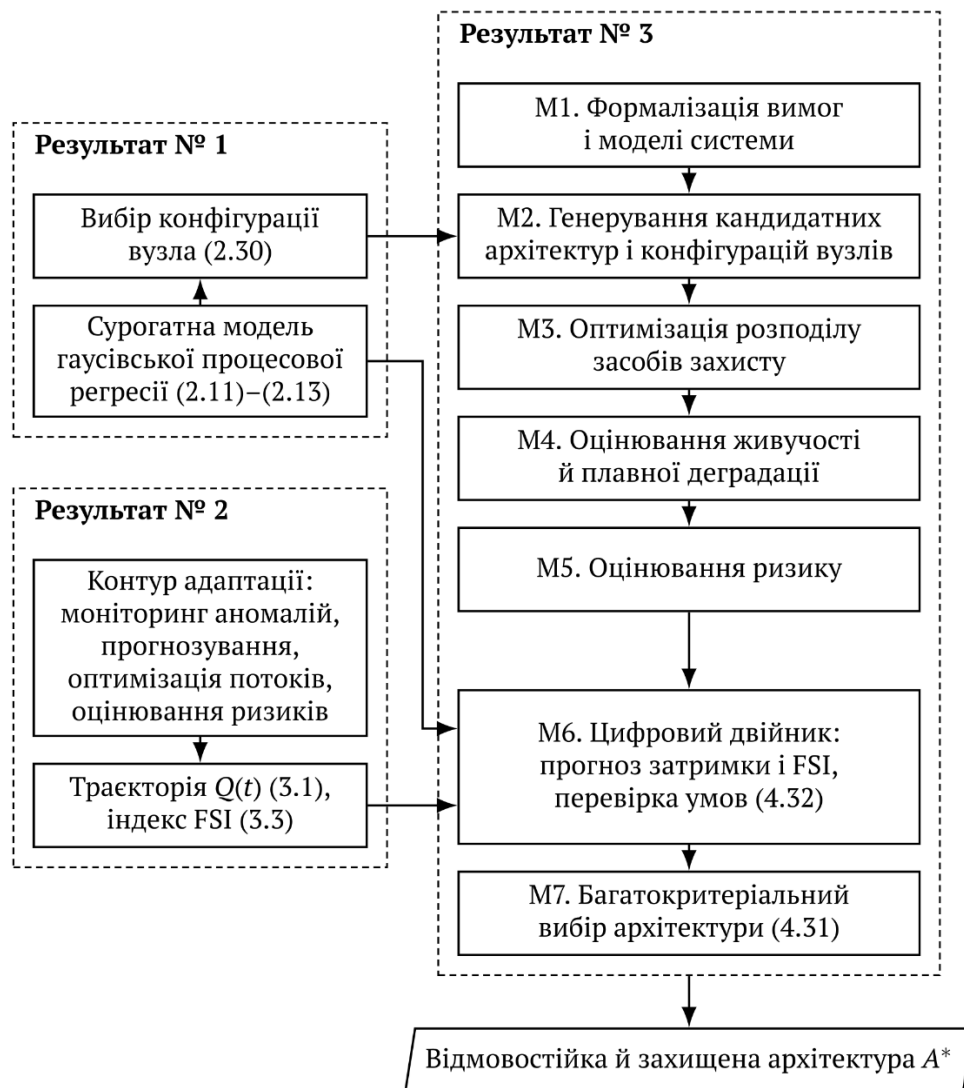


Рис. 4.9. Взаємозв'язок наукових результатів у методі та інформаційній технології структурно-параметричного проєктування відмовостійких і захищених розподілених обчислювальних систем

Отже, метод структурно-параметричного проектування поєднує обґрунтування конфігурації сурогатною моделлю продуктивності, оптимізацію розподілу ресурсів безпеки, оцінювання живучості, захищеності та ризику, механізми захисту й перевірку функціональної стійкості в єдину процедуру вибору архітектури за композитним критерієм, а інформаційна технологія реалізує цю процедуру як сукупність програмних модулів зі спільним сховищем проєктних даних і панеллю підтримки рішень. Разом вони дозволяють на етапі проектування обґрунтовано обирати архітектуру, стійку до втрати значної частки компонентів і до загроз безпеці функціонування, узгоджуючи відмовостійкість і захищеність у межах доступних ресурсів.

Висновки до розділу 4

У розділі розвинено метод та інформаційну технологію структурно-параметричного проектування відмовостійких і захищених розподілених обчислювальних систем корпоративних структур з використанням результатів сурогатного моделювання продуктивності та оцінювання функціональної стійкості в поєднанні з багатокритеріальною оптимізацією розподілу обмежених ресурсів безпеки з урахуванням функцій живучості, адаптивності та плавної деградації, що дозволяє на етапі проектування обирати архітектуру, стійку до втрати значної частки компонентів і до загроз безпеці функціонування.

Інформаційна технологія реалізує метод як сукупність семи програмних модулів – формалізації вимог, генерування кандидатних архітектур, оптимізації розподілу засобів захисту, оцінювання живучості, оцінювання ризику, цифрового двійника на основі сурогатної моделі продуктивності й технології забезпечення функціональної стійкості та багатокритеріального вибору – зі спільним сховищем проєктних даних і панеллю підтримки рішень, що дозволяє виконувати проектування в автоматизованому режимі та коригувати ваги критеріїв без повторного оцінювання кандидатів.

Новизна результату полягає в тому, що, на відміну від підходів, у яких надійність і захищеність забезпечують окремо, а засоби захисту додають до вже розгорнутої системи, метод та інформаційна технологія використовують результати методу автоматизованого проєктування на основі сурогатної моделі гаусівської процесової регресії для обґрунтування параметрів конфігурації та результати інформаційної технології забезпечення функціональної стійкості для врахування вимог до стійкості системи, поєднуючи їх із багатокритеріальною оптимізацією розподілу ресурсів безпеки з урахуванням живучості, адаптивності та плавної деградації, що дозволяє на етапі проєктування обирати відмовостійку й захищену архітектуру системи.

Практичне значення результату полягає в можливості ще до розгортання обирати архітектуру розподіленої обчислювальної системи з наперед заданими показниками живучості й захищеності в межах доступних ресурсів і перевіряти проєктні рішення на цифровому двійнику за сценаріями відмов і загроз.

РОЗДІЛ 5. ЕКСПЕРИМЕНТАЛЬНА ВЕРИФІКАЦІЯ РЕЗУЛЬТАТІВ

5.1. Програма й методика експериментальної верифікації

Розділ присвячено експериментальній верифікації трьох наукових результатів дисертації, що утворюють єдину тріаду: сурогатного моделювання продуктивності, забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи та структурно-параметричного проектування відмовостійких і захищених розподілених обчислювальних систем. Мета верифікації полягає в тому, щоб для кожного результату кількісно підтвердити ті теоретичні положення, що виносяться в наукову новизну, за допомогою відтворюваних числових експериментів, а не декларативних оцінок. Підрозділ задає спільну для всіх трьох верифікацій частину: програму досліджень, програмно-інструментальне середовище, характер вихідних даних, систему метрик і критерії оцінювання.

Верифікацію побудовано на трьох спільних принципах: відтворюваність, фізична обґрунтованість вихідних даних і валідність інтерпретації [137]. Відтворюваність забезпечено детермінованим формуванням усіх наборів даних із фіксованим початковим значенням генератора випадкових чисел ($SEED = 42$) на всіх стадіях – від генерації даних до розбиття вибірок, ініціалізації моделей і статистичного моделювання методом Монте-Карло; за такого підходу будь-який запуск відтворює ті самі числа й рисунки. Фізична обґрунтованість означає, що синтетичні дані породжуються не довільно, а за змістовними моделями предметної області: моделлю черги М/М/с для латентності веб-стека (результат № 1), імітаційною моделлю корпоративної інформаційної системи із сезонним попитом і збуреннями (результат № 2) та структурною моделлю розподіленої системи з класами компонентів і механізмами захисту (результат № 3); свідоме документування припущень і спрощень такої імітаційної моделі є визнаною умовою її валідності [132].

Синтетичний бенчмарк із заданою нелінійністю, збуреннями чи структурою відмов дозволяє точно знати еталонну відповідь і кількісно виміряти, наскільки метод її відновлює, чого натурний експеримент із неконтрольованими сторонніми чинниками на цьому етапі не забезпечив би. Реальні апаратні й телеметричні дані промислових корпоративних систем, крім того, є комерційною таємницею й недоступні для відкритої публікації та незалежного відтворення, тому параметри й діапазони бенчмарків (склад ресурсів, профілі навантаження, характер збурень, класи компонентів) задано відповідно до типових режимів експлуатації корпоративних систем, відомих з інженерної практики та фахової літератури.

Загальну програму верифікації трьох результатів, із зазначенням об'єкта перевірки, складових і головного критерію успіху для кожного, наведено в таблиці 5.1.

Таблиця 5.1

Загальна програма експериментальної верифікації тріади результатів

Результат	Об'єкт верифікації	Складові перевірки	Головний критерій успіху
№ 1	Сурогатна GPR-модель продуктивності	Точність апроксимації, відкаліброваність довірчих інтервалів, порівняння з адитивними та деревними моделями, крива навчання, важливість ознак	Висока точність ($R^2 \geq 0,90$) за скорочення обсягу вимірювань близько 90% при відкалібрований невизначеності
№ 2	Інтегрована технологія функціональної стійкості	Виявлення аномалій, прогнозування навантаження, оптимізація потоків, ризик-економіка, композитний індекс стійкості	Приріст індексу функціональної стійкості та цільової функції технології над базовою реактивною конфігурацією
№ 3	Метод проєктування відмовостійких захищених систем	Оптимізація розподілу ресурсів безпеки, ймовірність атаки й ризик, живучість і плавна деградація, вибір архітектури, перевірка на цифровому двійнику з використанням результатів № 1 і № 2	Вибір архітектури з найвищим композитним критерієм за зниженого ризику й підвищеної живучості

Програмно-інструментальне середовище є єдиним для всіх трьох експериментів. Обчислення виконано мовою Python (версія 3.14) із

застосуванням усталеного стеку з відкритим кодом – бібліотек NumPy для векторизованих чисельних обчислень, SciPy для латинського гіперкубічного планування експерименту й статистичних розподілів та scikit-learn для реалізації гаусівської процесової регресії, ансамблевих моделей, градієнтного бустингу й обчислення метрик якості [65, 126, 153]. Аналіз важливості ознак виконано методом адитивних пояснень, моделювання мережі обслуговування й пошук максимального потоку – засобами бібліотеки NetworkX, а побудову рисунків – засобами бібліотеки Matplotlib [64, 72, 99]. Такий вибір інструментарію забезпечує повну відтворюваність і незалежну перевірюваність результатів, оскільки всі використані компоненти є загальнодоступними й широко апробованими в наукових обчисленнях.

Оскільки три результати перевіряють різні за природою твердження – регресійну точність, стійкість під збуреннями та проєктну оптимізацію, – систему метрик згруповано за призначенням. Точність апроксимації та прогнозування оцінено коефіцієнтом детермінації, середньою абсолютною та середньою абсолютною відсотковою похибками за формулою (3.25), а також середньоквадратичною похибкою, що посилено штрафує великі відхилення прогнозу. Відкаліброваність довірчих інтервалів сурогатної моделі оцінено емпіричним покриттям 95-відсоткового інтервалу за формулою (2.24) та очікуваною похибкою калібрування за формулою (2.25), яка слугує скалярною мірою якості діаграми надійності [40]. Якість виявлення аномалій оцінено точністю, повнотою та їхнім гармонічним середнім за формулою (3.19).

Крім наведених базових метрик, для результатів № 2 і № 3 використано інтегральні показники предметної області, означені у теоретичних розділах: індекс функціональної стійкості та цільову функцію, а також індекс безпеки, інтегральний ризик, функції живучості й плавної деградації та композитний критерій вибору архітектури. Їх обчислюють безпосередньо за формулами відповідних розділів, тому тут вони не переозначаються.

Для результату № 1 успіхом вважається одночасне виконання трьох умов: висока точність апроксимації нелінійної поверхні відгуку, відкаліброваність

довірчих інтервалів (емпіричне покриття, близьке до номінального, за низької очікуваної похибки калібрування) та досягнення цієї точності за істотно скороченого обсягу вимірювань. Для результату № 2 критерієм є перевищення інтегрованою технологією базової реактивної конфігурації за композитним індексом функціональної стійкості та цільовою функцією, підкріплене вимірним внеском кожної складової – детектора, прогнозу, оптимізації потоків і ризик-економіки. Для результату № 3 успіхом є здатність методу на етапі проєктування обрати архітектуру з найвищим композитним критерієм, яка водночас забезпечує знижений інтегральний ризик і підвищену живучість за прийнятною вартості. Такі критерії зумисно прив'язані до порівняння з коректно визначеною базовою альтернативою, аби показати не просто працездатність методу, а його перевагу.

Порівняння з базовою альтернативою є наскрізним методичним прийомом усіх трьох експериментів. У результаті № 1 сурогатну GPR-модель зіставлено з адитивною моделлю впливу та з деревними ансамблевими моделями, у результаті № 2 інтегровану технологію – з реактивною конфігурацією без проактивного керування, у результаті № 3 спроектовану захищену розподілену архітектуру – з централізованою нерезервованою та проміжними варіантами. Для коректності зіставлення базові й запропоновані конфігурації прогонялися на тих самих даних, і з тим самим початковим значенням генератора, тому спостережувані відмінності зумовлені саме методом.

Отже, програма верифікації охоплює три результати за єдиною методикою з детермінованими фізично обґрунтованими синтетичними даними, спільним інструментарієм, узгодженою системою метрик і критеріями, з коректно визначеними базовими альтернативами.

5.2. Верифікація методу сурогатного моделювання продуктивності

Підрозділ присвячено експериментальній верифікації результату № 1 – методу автоматизованого проєктування обчислювальних систем на основі сурогатної моделі гаусівської процесової регресії. Верифікація має підтвердити

чотири положення, що виносяться в наукову новизну: сурогатна модель точно апроксимує нелінійну поверхню відгуку у просторі «конфігурація – навантаження»; вона надає відкалібровані довірчі інтервали; адитивні моделі впливу для цієї задачі принципово непридатні; сурогатний підхід скорочує обсяг натурних вимірювань приблизно на 90% за збереження точності. Додатково перевірено придатність предиктивної невизначеності для активного вибору наступних вимірювань. Дослідження є сфокусованим відтворенням раніше проведеної серії експериментів, звуженим до тих тверджень, що безпосередньо обґрунтовують наукову новизну [162].

Об'єктом моделювання є корпоративний веб-стек (сервер застосунків із реляційною базою даних) у просторі шести параметрів, з яких чотири описують конфігурацію ресурсів (обсяг оперативної пам'яті, кількість ядер процесора, розмір пулу з'єднань до бази даних і кількість робочих процесів), а два – характер навантаження (кількість віртуальних користувачів і частку операцій запису). Діапазони й тип параметрів відповідають таблиці 2.1; довідковий повний перебір фіксованих дискретних рівнів дає 720 конфігурацій, що слугує знаменником для оцінки скорочення обсягу вимірювань.

Цільовою метрикою продуктивності є 95-й перцентиль часу відповіді (p95-затримка, мс) – показник, чутливий до хвостових затримок і тому інформативніший за середнє. Детерміновану поверхню відгуку задано фізичною моделлю черги М/М/с: базовий час обслуговування залежить від частки операцій запису (операції запису повільніші приблизно в 1,4 раза) та штрафу за брак пам'яті; ефективна кількість каналів обслуговування визначається обчислювальною потужністю й пулом з'єднань; коефіцієнт завантаженості дорівнює відношенню інтенсивності надходження запитів до сумарної спроможності обслуговування. Ключова нелінійність зосереджена саме в цьому коефіцієнті завантаженості: p95-затримка зростає як степенева функція завантаженості, унаслідок чого поверхня відгуку має характерне «коліно» насичення, недосяжне для адитивного опису. Отримана латентність обмежена таймаутом 2000 мс.

Навчальну вибірку з 420 точок сформовано за планом латинського гіперкубічного вибору, реалізованим засобами SciPy [106, 153]. Такий план забезпечує рівномірне заповнення шестивимірному простору параметрів; цілочислові параметри дискретизовано після масштабування. До детермінованої латентності додано мультиплікативний логнормальний шум із коефіцієнтом варіації близько 5%, який імітує природну варіативність натурного вимірювання. Отриманий набір сильно асиметричний: більшість конфігурацій дає десятки мілісекунд, а меншість поблизу насичення – сотні й тисячі мілісекунд (медіана 43,0 мс, 90-й перцентиль 212,2 мс, 99-й перцентиль 1275,4 мс, максимум 2083,8 мс). Ця асиметрія відповідає реальному характеру р95-затримки й обґрунтовує логарифмічне перетворення цільової змінної перед навчанням. До шести первинних параметрів додано чотири конструйовані ознаки моделі черги, найважливіша з яких – інтенсивність навантаження (відношення кількості користувачів до інтегральної потужності оброблення), тож простір ознак є десятивимірним; ознаки стандартизовано за параметрами навчальної частини.

Модель навчали за такою методикою. Вибірку розбито у пропорції 70/15/15 на навчальну, валідаційну й тестову частини з фіксованим початковим значенням генератора. Сурогатну модель побудовано у трьох варіантах коваріаційного ядра – RBF, Матерна 5/2 і раціонального квадратичного – з ізотропною параметризацією (єдиний масштаб довжини з обмеженими межами) та окремим компонентом шуму; гіперпараметри визначено максимізацією маргінальної правдоподібності.

Вибір ізотропного ядра, а не анізотропного з окремим масштабом на кожну ознаку, є свідомим методологічним рішенням: за малих навчальних вибірок анізотропна параметризація давала вироджені підгонки з масштабами на межах діапазону й нестійкі прогнози на важкому хвості розподілу, тоді як ізотропне ядро з обмеженими межами забезпечує стійкість і відтворюваність за незначної втрати пікової точності.

Як бази для порівняння взято градієнтний бустинг і випадковий ліс на логарифмованій цілі, а також адитивну лінійну модель впливу без взаємодій за

первинними параметрами, що відповідає класу моделей продуктивності з довірчими інтервалами [41, 42, 51]. Якість оцінено метриками точності та відкаліброваності, описаними в підрозділі 5.1.

Нелінійний характер поверхні відгуку, який має відтворити сурогатна модель, ілюструє рисунок 5.1: зрізи p95-затримки за кількістю користувачів для різної обчислювальної потужності демонструють чітке «коліно» насичення, що настає раніше за меншої потужності, а ізолінії затримки у площині «користувачі – ядра процесора» показують різко неоднорідний рельєф.

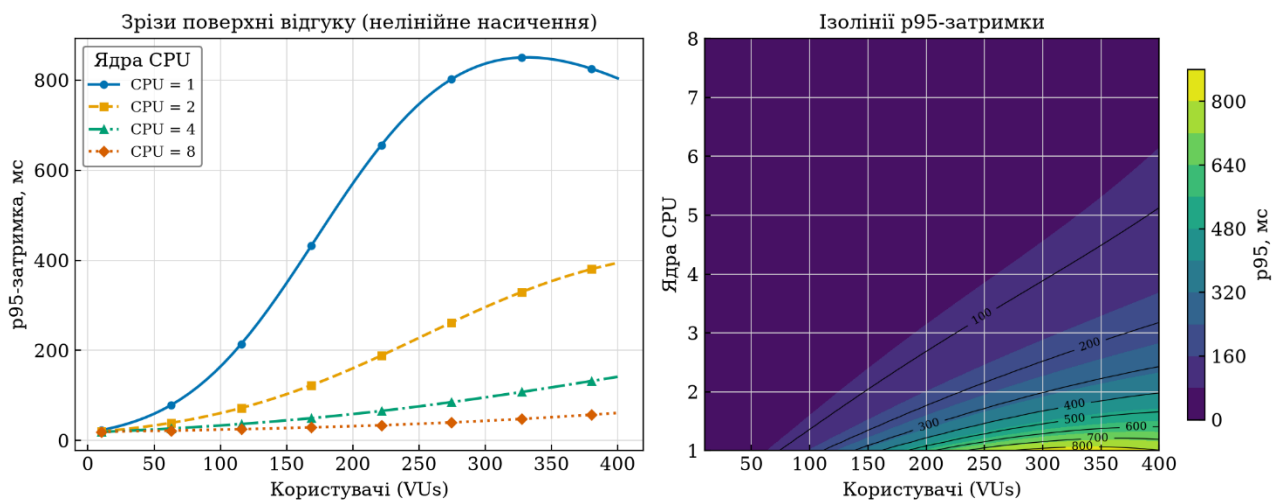


Рис. 5.1. Поверхня відгуку p95-затримки: зрізи за кількістю користувачів для різної обчислювальної потужності (ліворуч) та ізолінії затримки у площині «користувачі – ядра процесора» (праворуч)

Саме така форма поверхні робить адитивний опис принципово неадекватним.

Результати порівняння моделей на тестовій вибірці наведено в таблиці 5.2. Усі три ядра сурогатної моделі дають високу й практично однакову точність (коефіцієнт детермінації близько 0,97 за середньої абсолютної похибки близько 11 мс); за основне обрано ядро Матерна 5/2 з міркувань фізичної обґрунтованості (скінченна гладкість функції відгуку), а не лише емпірично. Сурогатна модель істотно перевершує деревні ансамблі (градієнтний бустинг дає коефіцієнт детермінації 0,88, випадковий ліс – 0,76) і, на відміну від них, надає довірчі інтервали. Адитивна модель впливу показує коефіцієнт детермінації лише 0,38

за середньої абсолютної похибки 101,9 мс – майже на порядок гіршої за сурогатну (11,2 мс), що кількісно підтверджує теоретичне положення розділу 2 про принципову непридатність адитивних моделей для нелінійної поверхні «навантаження – потужність».

Таблиця 5.2

Порівняння моделей на тестовій вибірці (63 точки)

Модель	RMSE, мс	MAE, мс	R ²	Покриття 95%	ECE	Довірчі інтервали
Сурогатна, ядро RBF	29,73	11,55	0,968	1,000	0,055	так
Сурогатна, ядро Матерна 5/2	28,99	11,18	0,970	1,000	0,055	так
Сурогатна, раціональне квадратичне ядро	28,99	11,23	0,970	1,000	0,057	так
Гradientний бустинг	57,52	23,24	0,881	–	–	ні
Випадковий ліс	80,73	29,47	0,765	–	–	ні
Адитивна модель впливу	130,94	101,87	0,381	–	–	ні

Прогноз сурогатної моделі з довірчим інтервалом показано на рисунку 5.2: для відсортованої за зростанням тестової вибірки 95-відсотковий інтервал охоплює фактичні значення й розширюється в зоні насичення, де затримка чутливіша до параметрів, а діаграма «прогноз проти факту» розташовує точки вздовж ідеальної діагоналі, причому рівень предиктивної невизначеності маркує складні для прогнозу конфігурації, тож модель не лише точна, а й адекватно оцінює власну невпевненість.

Відкаліброваність довірчих інтервалів перевірено окремо, оскільки для задач проєктування важлива не лише точкова оцінка, а й достовірність оголошеного інтервалу. Емпіричне покриття 95-відсоткового інтервалу для всіх трьох ядер дорівнює одиниці: усі фактичні значення тестової вибірки

потрапляють в інтервал, тобто модель дещо консервативна, що безпечно для практичних рішень.

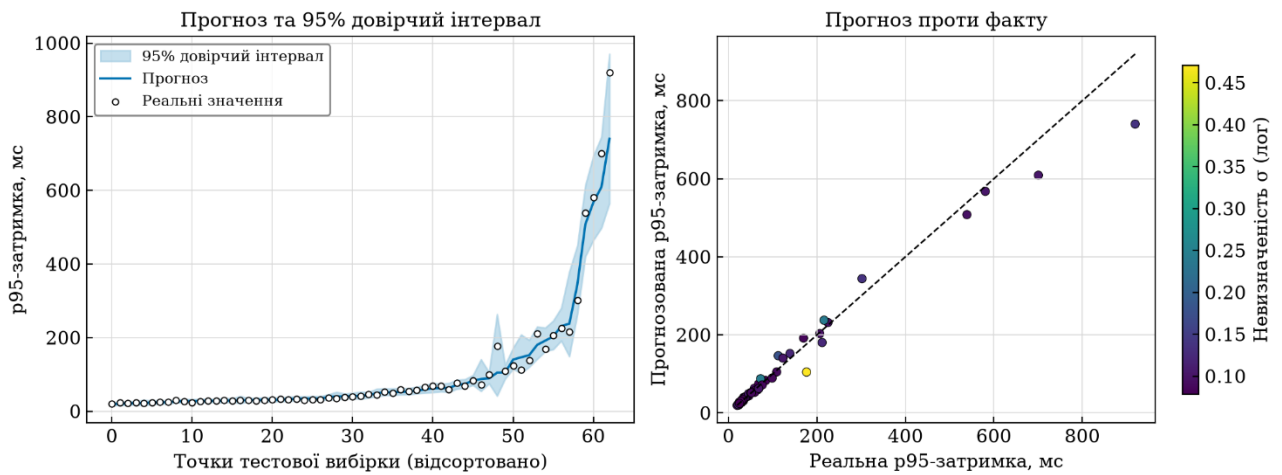


Рис. 5.2. Прогноз сурогатної моделі з 95-відсотковим довірчим інтервалом на відсортованій тестовій вибірці (ліворуч) та діаграма «прогноз проти факту» з кольоровим кодуванням рівня невизначеності (праворуч)

Очікувана похибка калібрування низька (0,055–0,057), а криві надійності близькі до ідеальної діагоналі (рисунок 5.3). На противагу цьому адитивна й деревні моделі довірчих інтервалів не надають узагалі, тому діаграма надійності та очікувана похибка калібрування вводяться в роботу як стандарт перевірки якості невизначеності ймовірнісних прогнозів [40].

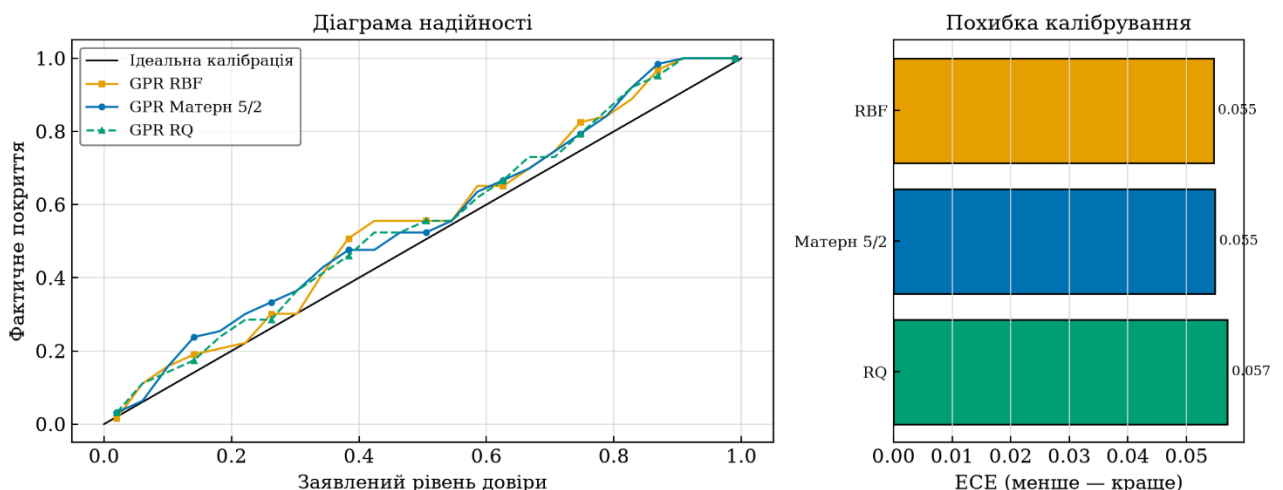


Рис. 5.3. Діаграма надійності трьох ядер сурогатної моделі (криві близькі до ідеальної діагоналі) та стовпчикова діаграма очікуваної похибки калібрування

Стійкість моделі підтверджено п'ятикратною крос-валідацією (таблиця 5.3): середній коефіцієнт детермінації становить $0,877 \pm 0,064$ без жодного провального фолда (мінімум 0,766), а середнє покриття лишається стабільно високим ($0,952 \pm 0,020$).

Помітна варіативність середньоквадратичної похибки між фолдами пояснюється важким хвостом розподілу затримки, за якого поодинокі конфігурації поблизу насичення дають великі абсолютні похибки; водночас стабільність покриття свідчить, що відкаліброваність зберігається на всіх підвбірках, а не лише на основному розбитті.

Таблиця 5.3

Результати п'ятикратної крос-валідації (ядро Матерна 5/2)

Фолд	RMSE, мс	MAE, мс	R ²	Покриття 95%	ECE
1	13,83	6,63	0,960	0,988	0,065
2	103,83	25,89	0,904	0,929	0,062
3	78,10	19,06	0,896	0,952	0,041
4	105,20	24,61	0,859	0,940	0,034
5	80,30	19,18	0,766	0,952	0,025
Середнє	76,25 ± 33,21	19,07 ± 6,81	0,877 ± 0,064	0,952 ± 0,020	0,046 ± 0,016

Центральне для методу твердження про скорочення обсягу вимірювань перевірено кривою навчання (рисунок 5.4). Порогове значення коефіцієнта детермінації 0,90 на фіксованому тесті досягається вже за 75 навчальних точок, тоді як повний перебір фіксованих рівнів налічує 720 конфігурацій, тож відносне скорочення обсягу вимірювань за формулою (2.28) становить близько 89,6%.

Пікова точність на основному тестовому розбитті (коефіцієнт детермінації 0,970) відповідає повній навчальній вибірці; плато кривої навчання (0,94–0,96) і середнє значення крос-валідації (0,877) нижчі внаслідок важкого хвоста розподілу затримки й меншого обсягу підвбірок, що не суперечить досягненню порогу 0,90 за 75 вимірювань.

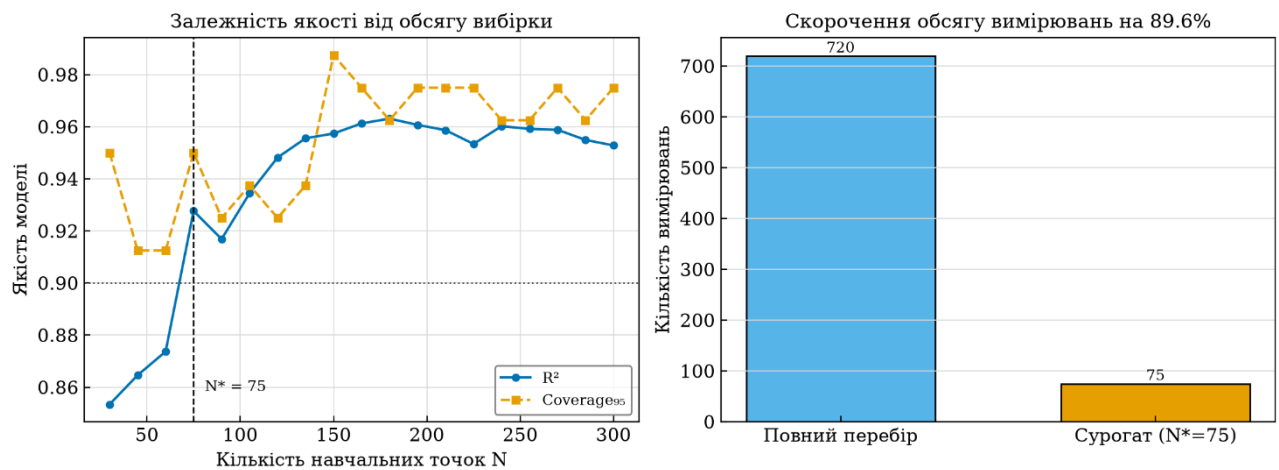


Рис. 5.4. Крива навчання – залежність коефіцієнта детермінації та покриття від обсягу навчальної вибірки (ліворуч) та порівняння обсягу вимірювань повного перебору й сурогатного підходу (праворуч)

Придатність предиктивної невизначеності для активного планування вимірювань ілюструє рисунок 5.5. Карта прогнозової дисперсії у площині «користувачі – розмір пулу» показує, що невизначеність мінімальна у щільно заповнених центральних областях і зростає на розріджених екстремумах, насамперед у зоні високого навантаження й малого пулу з'єднань, тобто в області насичення.

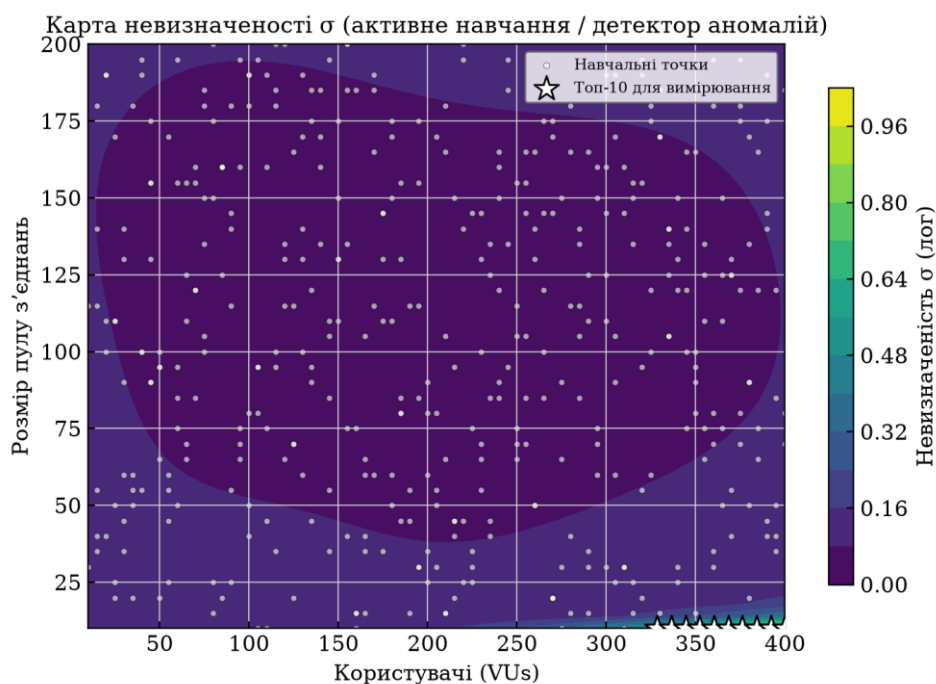


Рис. 5.5. Карта прогнозової дисперсії сурогатної моделі у площині «користувачі – розмір пулу»

Реалізований критерій вибору за максимальною дисперсією зосереджує рекомендовані наступні вимірювання саме в цій зоні, що узгоджується з теоретичним обґрунтуванням активного навчання: додаткові дані потрібні там, де модель найменш упевнена [95].

Нарешті, аналіз важливості ознак методом адитивних пояснень кількісно підтверджує, що поведінку системи описує саме відношення навантаження до потужності, а не окремі первинні параметри [99]. Провідною ознакою є інтенсивність навантаження із середнім абсолютним внеском 0,655, що приблизно вп'ятеро перевищує внесок наступної ознаки (пам'ять на процес, 0,132) і більш ніж у двадцять разів – внесок кількості користувачів окремо (0,032). Ранжування ознак наведено в таблиці 5.4 та проілюстровано на рисунку 5.6.

Таблиця 5.4

Важливість ознак за середнім абсолютним внеском

Ознака	Середній абсолютний внесок
Інтенсивність навантаження	0,655
Пам'ять на процес	0,132
Розмір пулу з'єднань	0,054
Оперативна пам'ять	0,042
Ядра процесора	0,032
Віртуальні користувачі	0,032
Частка операцій запису	0,024
Робочі процеси	0,012
Конкуренція за пул з'єднань	0,010
Інтегральна потужність	0,005

Домінування конструйованої ознаки відношення навантаження до потужності узгоджується з фізичною моделлю черги й підтверджує теоретичне положення розділу 2 про визначальну роль коефіцієнта завантаженості.

Отже, експериментальна верифікація підтвердила всі положення результату № 1. Сурогатна модель гаусівської процесової регресії апроксимує нелінійну поверхню відгуку продуктивності з високою точністю (коефіцієнт детермінації 0,970, середня абсолютна похибка 11,2 мс) і надає відкалібровані довірчі інтервали (покриття 1,00, очікувана похибка калібрування 0,055), тоді як

адитивна модель впливу дає майже на порядок гіршу точність, що підтверджує її принципову непридатність для цієї задачі.

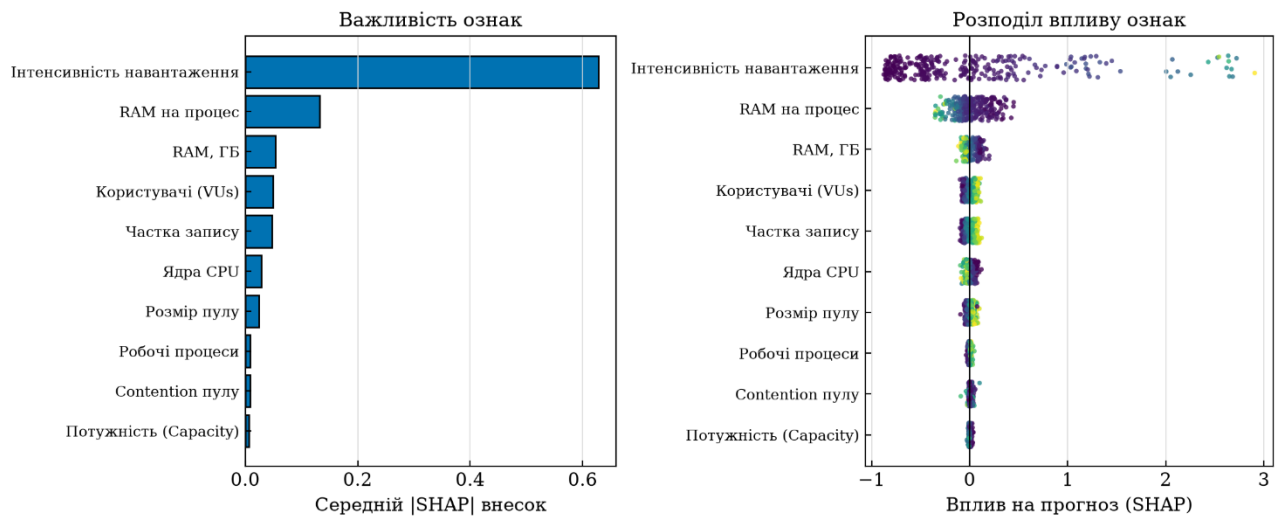


Рис. 5.6. Важливість ознак сурогатної моделі: середній абсолютний внесок за спаданням (ліворуч) та розподіл впливу ознак на прогноз (праворуч)

Модель стійка за крос-валідацією ($0,877 \pm 0,064$ без провальних фолдів) і перевершує деревні ансамблі, додатково надаючи оцінку власної невизначеності. Сурогатний підхід досягає цільової точності за 75 вимірювань замість 720, тобто скорочує їх обсяг приблизно на 90%, а провідною ознакою поверхні відгуку є відношення навантаження до потужності. Тим самим кількісно підтверджено переваги методу, за коректного застереження про синтетичний характер даних і винесення натурних вимірювань у подальші дослідження.

5.3. Верифікація інформаційної технології функціональної стійкості

Підрозділ присвячено експериментальній верифікації результату № 2 – інформаційної технології забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи, теоретичний опис якої подано в розділі 3. Мета верифікації полягає в тому, щоб підтвердити: інтегрована технологія, що поєднує статистичний моніторинг аномалій, прогнозування навантаження методами машинного навчання, оптимізацію інформаційних потоків та ризик-економічне керування, підвищує здатність системи зберігати критичну

функціональність під збуреннями порівняно з базовою реактивною конфігурацією, і кількісно оцінити внесок кожної складової. Функціональну стійкість тут розуміють як здатність системи підтримувати задану функціональність попри збурення й часткові відмови, що відповідає прийнятому в роботі понятійному апарату надійності та стійкості [17, 93].

Об'єктом моделювання є корпоративна інформаційна система під змінним навантаженням, подана синтетичною детермінованою телеметрією. Горизонт спостереження становить 21 добу з кроком дискретизації 30 хв, що дає 1008 відліків; попит формується добовою й тижневою сезонністю з висхідним трендом і поведінковим шумом, а час відповіді зростає нелінійно з коефіцієнтом використання потужності та рівнем загрози. Параметри імітаційної моделі наведено в таблиці 5.5. Свідоме документування таких припущень і спрощень моделі є визнаною умовою її валідності й відтворюваності [132].

Таблиця 5.5

Параметри імітаційної моделі корпоративної інформаційної системи

Параметр	Значення
Горизонт спостереження	21 доба
Крок дискретизації	30 хв (1008 відліків)
Номінальна потужність	420 користувач-еквівалентів
Базове / пікове навантаження	150 / 300 користувачів
Тренд зростання за горизонт	+18%
Шум поведінки (логнормальний)	$\sigma = 0,035$
Латентність у ненавантаженому режимі	60 мс
Поріг якості обслуговування	300 мс
Кількість збурень	10 (57 аномальних кроків)

На горизонт спостереження внесено десять збурень трьох типів, що відтворюють характерні порушення штатного режиму корпоративної системи: сплески навантаження, кіберінциденти й відмови компонентів. Їхню кількість і параметри впливу наведено в таблиці 5.6; разом вони охоплюють 57 аномальних кроків, що становить близько 5,7% горизонту й формує реалістичне

співвідношення штатних та позаштатних станів для перевірки складових технологій.

Таблиця 5.6

Типи внесених збурень

Тип збурення	Кількість	Параметр впливу
Сплеск навантаження	3	множник попиту 1,6...2,6
Кіберінцидент	4	частка успішності загрози 0,5...0,95
Відмова компонента	3	втрата потужності вузла 35...60%

Порівняння виконано для двох конфігурацій системи на однакових даних і збуреннях. Базова реактивна конфігурація має статичну потужність, слабкий контроль загроз і повільне реагування з характерним часом 12 год, тоді як запропонована інтегрована технологія застосовує проактивне резервування за прогнозом навантаження, активний контроль загроз, швидке реагування з характерним часом 1 год і перемаршрутизацію інформаційних потоків. Оскільки обидві конфігурації прогналися за того самого початкового значення генератора, спостережувані відмінності зумовлені саме технологією, а не випадковими чинниками.

Першу складову технології – статистичний моніторинг і виявлення аномалій – реалізовано на основі сезонного контрольного графіка залишку відповіді з робастним масштабуванням, контрольною межею три сигми й правилом серій, доповненого дисперсійним аналізом як допоміжним детектором зсуву режиму, що відповідає класичному апарату статистичного контролю процесів і виявлення аномалій [30, 114]. Результат подано на рисунку 5.7: детектор виявляє всі десять внесених подій із високими показниками якості (точність 0,887, повнота 0,825, гармонічне середнє 0,855) за середньої затримки виявлення близько 0,15 год.

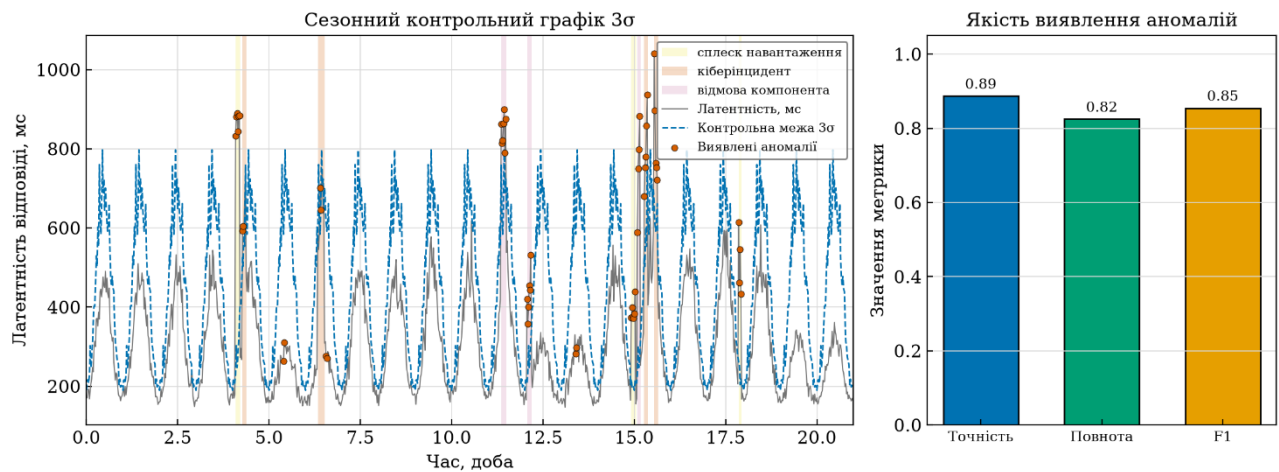


Рис. 5.7. Сезонний контрольний графік латентності з контрольною межею три сигми й виявленими аномаліями та стовпчикова діаграма точності, повноти й гармонічного середнього детектора

Другу складову – прогнозування навантаження на найближчий горизонт – реалізовано методом градієнтного бустингу за ознаками з лагів, календарних гармонік і нормованого часу, що відповідає сучасній практиці прогнозування часових рядів засобами машинного навчання [51, 73]. На штатних кроках прогноз досягає коефіцієнта детермінації 0,805 за середньої абсолютної відсоткової похибки 5,2%, що показано на рисунку 5.8. Така точність прогнозу на один крок уперед є достатньою для проактивного резервування потужності до настання піка навантаження.

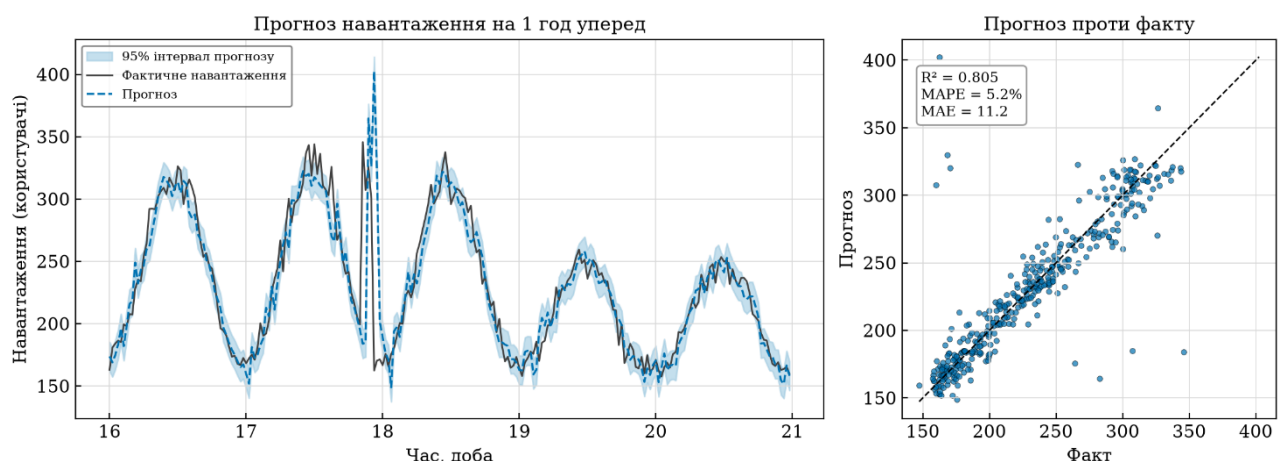


Рис. 5.8. Прогноз навантаження на найближчий горизонт із 95-відсотковим інтервалом (ліворуч) та діаграма «прогноз проти факту» на штатних кроках (праворуч)

Третю складову – оптимізацію інформаційних потоків – реалізовано як задачу максимального потоку в мережі обслуговування з джерелом, трьома проміжними вузлами й стоком, де відмова вражає один вузол; статичну рівномірну маршрутизацію порівняно з максимальним потоком за алгоритмом Форда–Фалкерсона [49, 64]. Результат наведено на рисунку 5.9.



Рис. 5.9. Обслуговане навантаження під час відмов за статичної маршрутизації та за оптимізації максимального потоку; частка необслугованого попиту знижується з 6,0% до 0,0%

За статичної маршрутизації під час відмов необслуговується близько 6,0% попиту, тоді як оптимізація потоків повністю усуває недообслуговування, знижуючи його частку до 0,0%, тобто перемаршрутизація потоків зберігає пропускну спроможність системи навіть за втрати окремого вузла.

Четверту складову – ризик-економічне керування – побудовано на раніше обґрунтованій моделі залишкового ризику, сигмоїди стійкості від витрат на безпеку та монотонної функції фінансових втрат від часу реагування [167]. Ці залежності проілюстровано на рисунку 5.10, а сценарні операційні точки наведено в таблиці 5.7.

Таблиця 5.7

Сценарії безпеки та час реагування

Сценарій	Залишковий ризик	Час реагування
Без активного захисту	85%	12 год
Лише резервування	60%	4 год
Комплексний захист	25%	1 год

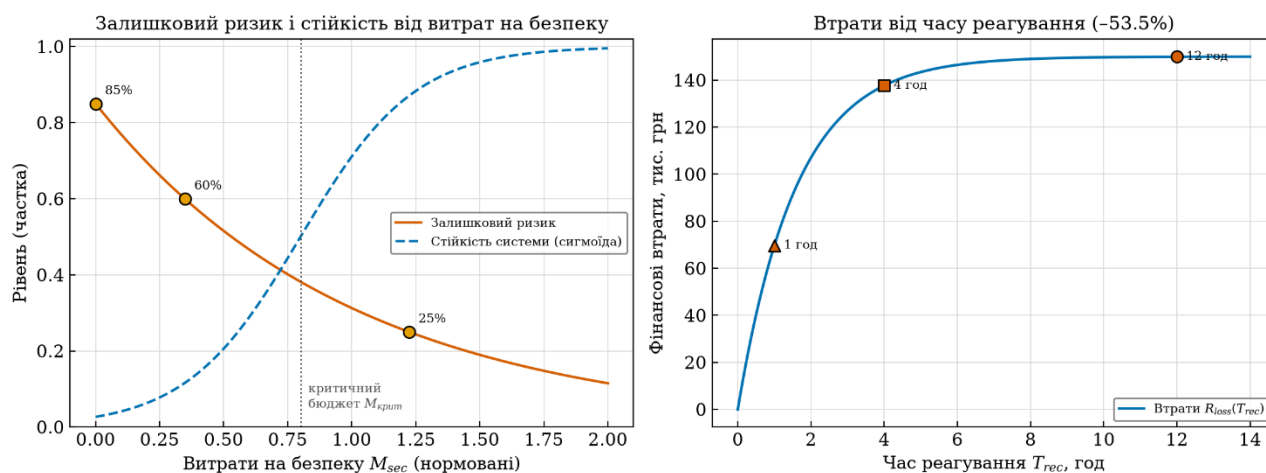


Рис. 5.10. Залишковий ризик і сигмоїда стійкості від витрат на безпеку з трьома сценарними точками (ліворуч) та фінансові втрати від часу реагування зі зниженням на 53,5% (праворуч)

Модель відтворює три сценарні операційні точки із залишковим ризиком 85%, 60% і 25% за відповідного часу реагування 12, 4 і 1 год, як показано в таблиці 5.7. Скорочення часу реагування з 12 до 1 год завдяки активному захисту зменшує очікувані фінансові втрати на 53,5%, що узгоджується з економічним обґрунтуванням оптимального рівня інвестицій у безпеку [50, 62].

Сукупний ефект технології оцінено композитним індексом функціональної стійкості, що є усередненим за горизонтом добутком продуктивнісної й безпекової складових якості обслуговування, та цільовою функцією від функціональності, безпеки, ризику й адаптивності. Часовий ряд індексу стійкості для обох конфігурацій наведено на рисунку 5.11: базова реактивна конфігурація дає глибокі й тривалі провали під час збурень, тоді як інтегрована технологія тримається біля одиниці, оскільки швидке реагування скорочує тривалість деградації.

Інтегральна оцінка стійкості як площі під кривою функціональності відповідає прийнятому підходу до кількісного вимірювання стійкості як функції часу [27, 67].



Рис. 5.11. Індекс функціональної стійкості двох конфігурацій на горизонті 21 доба зі смугами збурень; базова конфігурація дає глибокі провали, технологія тримається біля одиниці

Складові цільової функції для двох конфігурацій проілюстровано на рисунку 5.12.

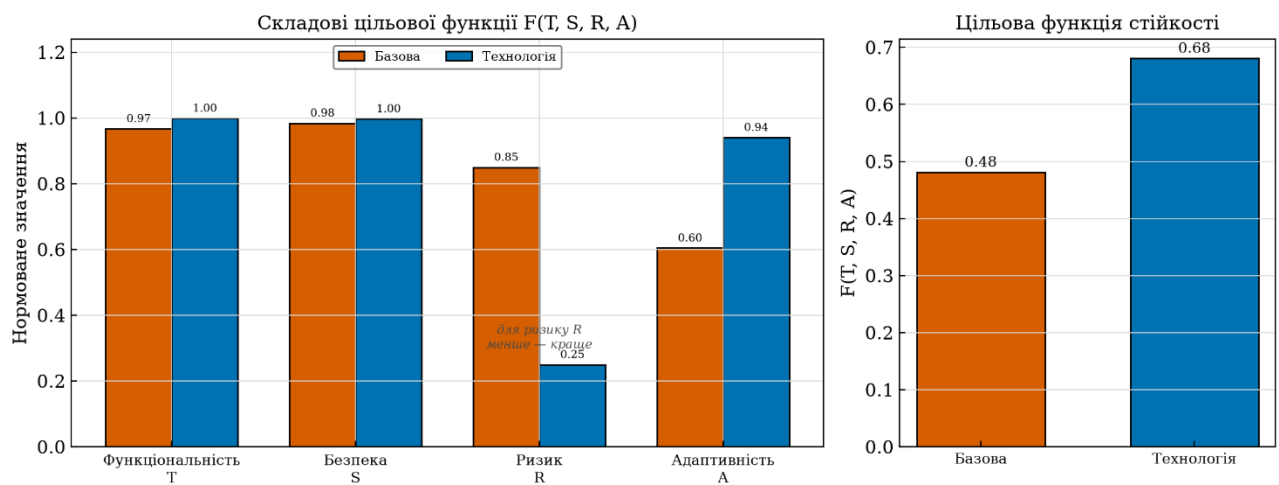


Рис. 5.12. Складові цільової функції для базової конфігурації та інтегрованої технології й підсумкове значення цільової функції

Інтегрована технологія перевершує базову конфігурацію за всіма складовими: функціональність зростає з 0,967 до 0,999, безпека – з 0,984 до 0,997, адаптивність – з 0,604 до 0,940, а ризик знижується з 0,850 до 0,250. Композитний індекс функціональної стійкості зростає з 0,917 до 0,996, а цільова функція – з 0,481 до 0,681, тобто приріст становить 0,079 і 0,200 відповідно. Слід зауважити,

що функціональність є середнім лише продуктивнісної складової якості, тоді як індекс стійкості враховує ще й безпекову складову; тому для базової конфігурації зі слабким контролем загроз індекс стійкості помітно нижчий за функціональність (0,917 проти 0,967), а для технології з активним контролем безпеки ці величини майже збігаються (0,996 проти 0,999), що є змістовним наслідком усунення безпекових провалів, а не артефактом обчислення.

Отже, експериментальна верифікація підтвердила положення результату № 2. Кожна зі складових технології показала вимірну ефективність: детектор аномалій виявив усі внесені події з гармонічним середнім 0,855, прогноз навантаження досяг коефіцієнта детермінації 0,805 за відсоткової похибки 5,2%, оптимізація потоків усунула недообслуговування під час відмов, а ризик-економічне керування знизило очікувані фінансові втрати на 53,5%. Інтегрована технологія перевершила базову реактивну конфігурацію за композитним індексом функціональної стійкості (0,996 проти 0,917) та цільовою функцією (0,681 проти 0,481), забезпечивши підтримання критичної функціональності під збуреннями.

5.4. Верифікація методу проєктування відмовостійких і захищених систем

Підрозділ присвячено експериментальній верифікації результату № 3 – методу та інформаційної технології структурно-параметричного проєктування відмовостійких і захищених розподілених обчислювальних систем корпоративних структур, теоретичний опис якого подано в розділі 4. Верифікація має підтвердити, що метод дозволяє вже на етапі проєктування обрати архітектуру, стійку до втрати значної частки компонентів і до загроз безпеці функціонування, та кількісно оцінити переваги спроектованої архітектури над неоптимізованими альтернативами, а також перевірити, що використання результатів № 1 і № 2 у цифровому двійнику впливає на проєктне рішення. Апарат, обґрунтований у раніше запропонованій концептуальній

моделі захисту критичної інфраструктури, узагальнено з початкового прикладного контексту на розподілену корпоративну обчислювальну систему: множник підвищеної загрози застосовано в узагальненому вигляді, а об'єктом слугує набір класів обчислювальних компонентів [164, 168]. Оскільки первинна модель подавала лише математичний апарат, наскрізний обчислювальний експеримент побудовано заново.

Об'єктом моделювання є розподілена корпоративна обчислювальна система, подана як набір п'яти класів компонентів з вагами важливості, сума яких дорівнює одиниці, базовою надійністю, базовою ймовірністю успішної атаки та потенційним збитком. Параметри класів наведено в таблиці В.2 додатка В. Механізми забезпечення захищеності – постквантову криптографію, архітектуру нульової довіри, адаптивне виявлення загроз машинним навчанням і резервування із самовідновленням – задано ефективністю зниження вразливості, нормованою вартістю та часом упровадження (таблиця В.3 додатка В); ефект застосування набору механізмів прийнято мультиплікативним зі спадною віддачею, тобто залишкова вразливість дорівнює добутку часток невідвернутого впливу застосованих механізмів згідно з формулою (4.10), що уникає нереалістичного повного усунення вразливості [20, 133, 135].

Рівень підвищеної загрози задано множником: штатний фоновий рівень дорівнює одиниці, а загострений (кампанії складних цілеспрямованих загроз, сплеск атак) – 1,8. Резервування задано схемою «k-з-n»: нерезервована архітектура містить вісім екземплярів компонентів, відмовостійка – шістнадцять з надмірністю за критичними класами. Щоб уникнути змішування різних опорних конфігурацій, у підрозділі послідовно вживаються точні назви: «без захисних засобів» – відмовостійка основа з нульовим розподілом ресурсів безпеки; «базова протидія» – слабка базова ефективність протидії без оптимізованого розподілу засобів; «базова (нерезервована) архітектура» – вісім екземплярів без резервування.

Першу складову методу – оптимізацію розподілу ресурсів безпеки – реалізовано через індекс безпеки за формулою (4.8), що є зваженою за

важливістю сумою добутків надійності, відвернутої ймовірності атаки та потенційного збитку класів. Оптимальний розподіл обмеженого бюджету засобів захисту знайдено жадібним алгоритмом, який на кожному кроці додає застосування з найбільшим приростом індексу безпеки на одиницю вартості, наближаючись до Парето-оптимальної межі [39, 45]. Результат подано на рисунку 5.13: жадібний розподіл лежить на межі Парето й помітно переважає хмару неоптимізованих випадкових розподілів.

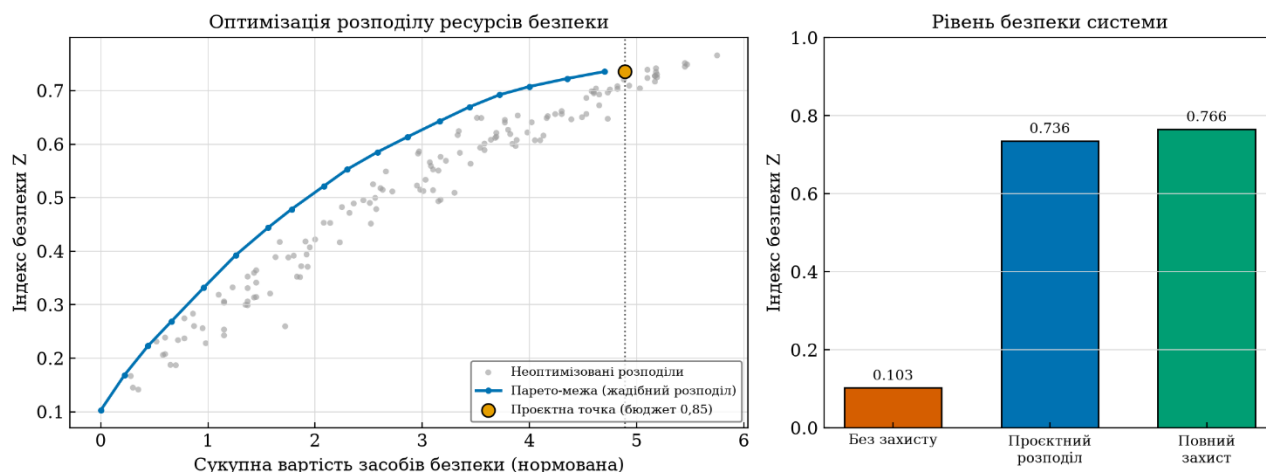


Рис. 5.13. Оптимізація розподілу ресурсів безпеки: Парето-межа індексу безпеки за вартістю та хмара неоптимізованих розподілів (ліворуч) і рівень безпеки для відмовостійкої основи – без захисних засобів, проєктний розподіл, повний захист (праворуч)

Таблиця 5.8

Індекс безпеки та ймовірність атаки (загострена загроза, множник 1,8)

Показник (відмовостійка основа, 16 екземплярів)	Значення
Індекс безпеки, без захисних засобів	0,103
Індекс безпеки, проєктний розподіл (бюджет 0,85)	0,736
Індекс безпеки, повний захист	0,766
Середньозважена ймовірність атаки, без захисних засобів	0,893
Середньозважена ймовірність атаки, оптимізований розподіл	0,218

Залежність захищеності від рівня загрози досліджено окремо. На рисунку 5.14 показано, що зі зростанням множника підвищеної загрози середньозважена ймовірність успішної атаки для системи без захисних засобів швидко

наближається до одиниці, тоді як за оптимізованого розподілу вона залишається помірною в усьому діапазоні загрози, що підтверджує стійкість спроектованого захисту до загострення обстановки [48, 86].

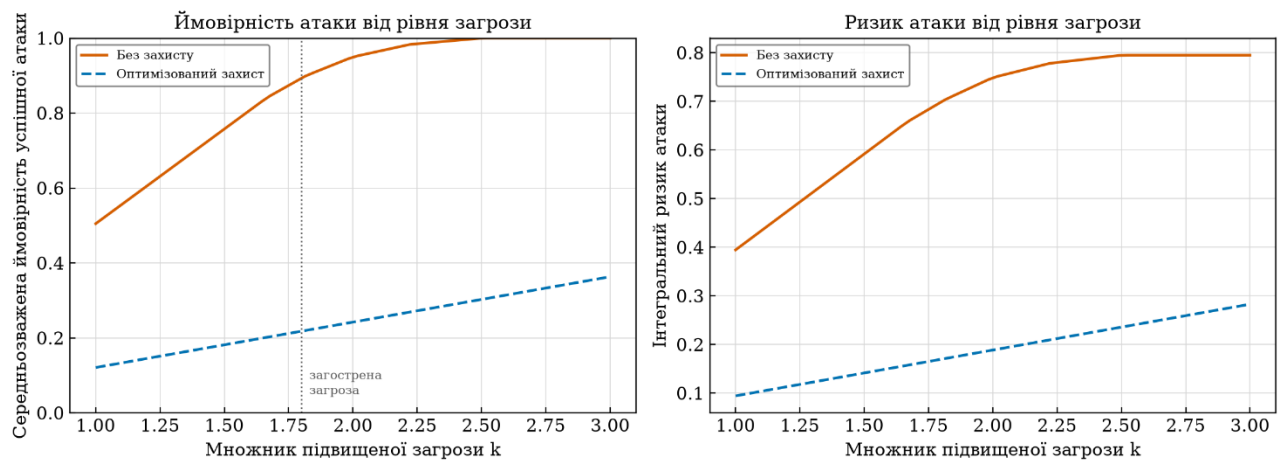


Рис. 5.14. Середньозважена ймовірність успішної атаки залежно від множника підвищеної загрози: без захисних засобів та за оптимізованого розподілу ресурсів безпеки

Інтегральний ризик системи, який є сумою очікуваних збитків за компонентами з урахуванням ефективності протидії та множника загрози, наведено на рисунку 5.15 і в таблиці 5.7: за однакового загостреного рівня загрози перехід від базової протидії до спроектованої знижує ризик з 1,769 до 0,722, тобто на 59% [8, 122].

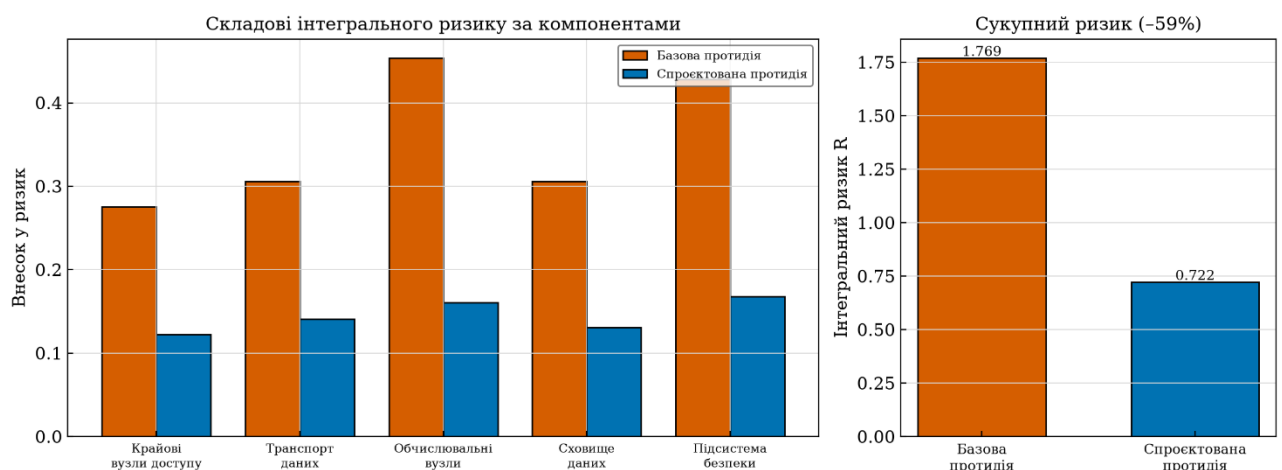


Рис. 5.15. Складові інтегрального ризику за компонентами та сукупний інтегральний ризик системи: базова протидія проти спроектованої протидії зі зниженням на 59%

Таблиця 5.9

Інтегральний ризик системи (однаковий множник загрози 1,8 для обох)

Показник	Базова протидія	Спроектована протидія
Інтегральний ризик	1,769	0,722
Зниження ризику (1,769 → 0,722)	–	59%

Другу складову методу – забезпечення живучості й плавної деградації – перевірено статистичним моделюванням методом Монте-Карло. За кожної частки втрати компонентів випадково вилучається відповідна кількість екземплярів, після чого обчислюється функціональність системи як зважена сума одиничних індикаторів працездатності класів за схемою «k-з-n» (формула (4.3)), а живучість визначається як імовірність того, що функціональність не падає нижче половини (формула (4.4)); оцінку усереднено за кількома тисячами повторень [124].

Результати моделювання живучості подано на рисунку 5.16. Характер деградації функціональності показано на рисунку 5.17.

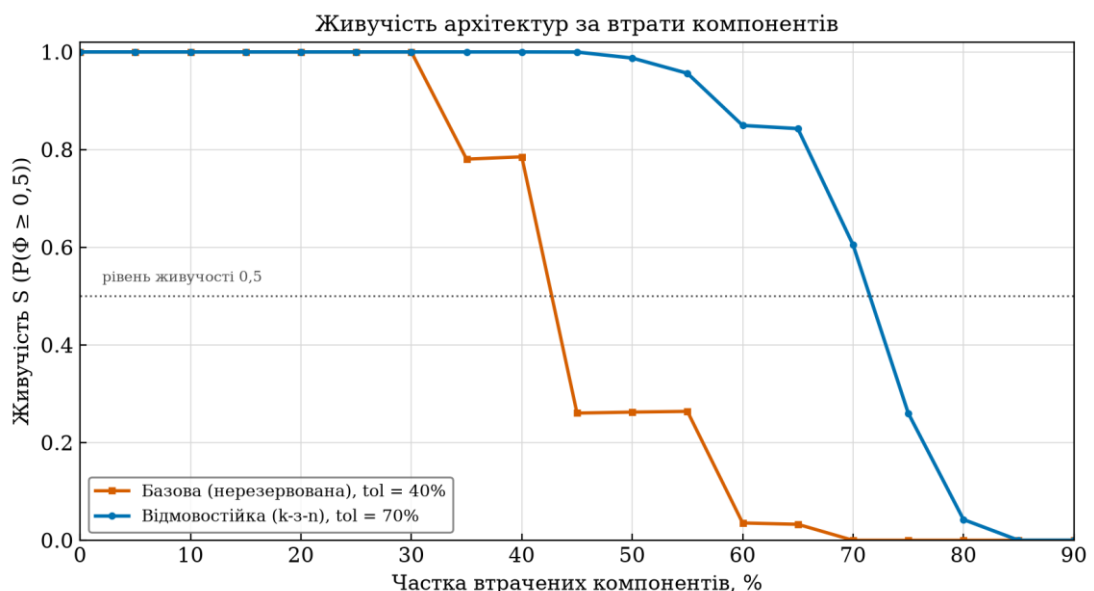


Рис. 5.16. Живучість базової нерезервованої та відмовостійкої архітектур залежно від частки втрати компонентів; допустима втрата 40% проти 70% за критичним рівнем живучості

Базова нерезервована архітектура зберігає прийнятний рівень живучості до втрати близько 40% компонентів, тоді як відмовостійка архітектура витримує втрату до 70% за тим самим критерієм, що майже вдвічі підвищує допустиму частку відмов [46, 84].

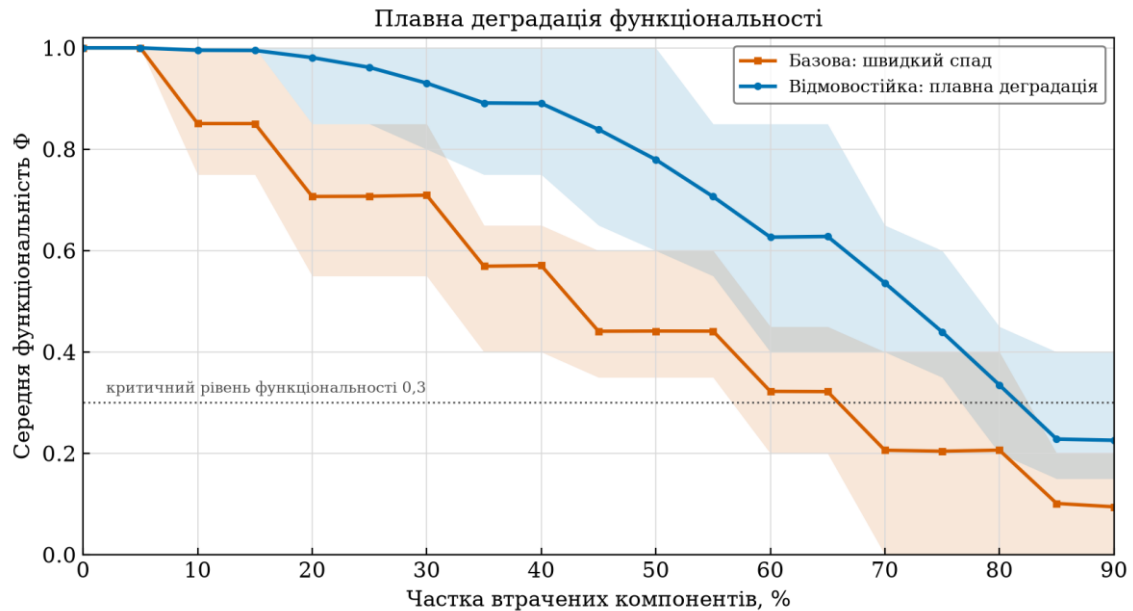


Рис. 5.17. Плавна деградація середньої функціональності за втрати компонентів: різкий спад базової нерезервованої архітектури проти плавної деградації відмовостійкої (критичний рівень 0,3)

Середня функціональність базової нерезервованої архітектури спадає різко й перетинає критичний рівень 0,3 вже за втрати близько 70% компонентів, натомість відмовостійка архітектура деградує плавно й утримується вище критичного рівня до втрати близько 85% компонентів [139].

Саме плавний, а не обвальний характер втрати функціональності є практично значущою властивістю спроектованої архітектури.

Підсумкову складову методу – вибір архітектури на етапі проєктування – реалізовано як багатокритеріальне оцінювання чотирьох кандидатних архітектур за п'ятьма нормованими критеріями: індексом безпеки, живучістю, плавністю деградації, низьким ризиком та економічністю. Композитний критерій обчислено як зважену суму цих показників за формулою (4.30), а рекомендованою вважається архітектура з його максимумом за формулою (4.31)

[45, 109]. Профілі кандидатів і їхній рейтинг наведено на рисунку 5.18 та в таблиці 5.10.

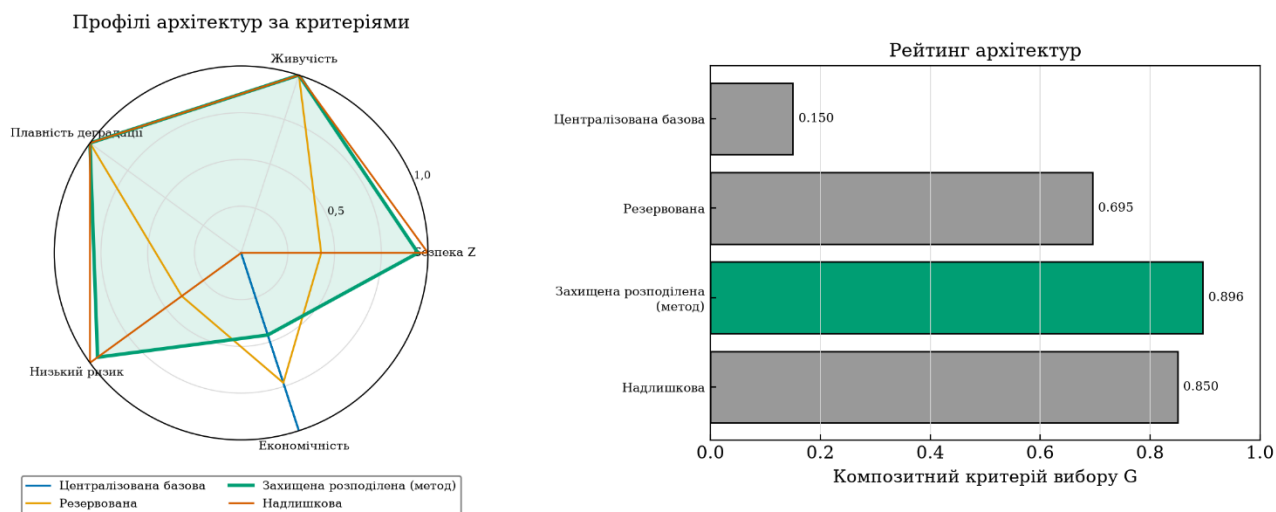


Рис. 5.18. Багатокритеріальні профілі кандидатних архітектур і їхній рейтинг за композитним критерієм; рекомендовано архітектуру «Захищена розподілена» з критерієм 0,896

Метод обирає архітектуру «Захищена розподілена» з найвищим композитним критерієм 0,896: вона забезпечує індекс безпеки 0,736, допустиму втрату 70% компонентів і на 59% нижчий інтегральний ризик за помірної вартості 1,70. Показово, що надлишкова архітектура має дещо вищий індекс безпеки (0,766) і трохи нижчий ризик, проте її композитний критерій нижчий (0,850) саме через надмірні витрати (вартість 2,30), а централізована базова й суто резервована архітектури поступаються за сукупністю критеріїв, тож метод обирає збалансоване, а не найдорожче рішення.

Таблиця 5.10

Багатокритеріальний вибір архітектури

Архітектура	Індекс безпеки	Інтегральний ризик	Допустима втрата	Вартість	Композитний критерій
Централізована базова	0,203	1,596	40%	1,00	0,150
Резервована	0,444	1,234	70%	1,35	0,695
Захищена розподілена (метод)	0,736	0,722	70%	1,70	0,896
Надлишкова	0,766	0,676	70%	2,30	0,850

Завершальну складову методу – перевірку кандидатів на цифровому двійнику за умовами (4.32) – реалізовано засобами двох попередніх результатів. Конфігурацію обчислювального вузла обґрунтовано сурогатною моделлю гаусівської процесової регресії за задачею (2.30): обирали найдешевшу конфігурацію, верхня довірча межа р95-затримки якої не перевищує 300 мс за проєктного пікового навантаження 360 користувачів, що припадає на два справні обчислювальні вузли (мінімум за схемою «k-з-n»), а обрану конфігурацію перевіряли натурним вимірюванням, роль якого виконувала фізична модель черги (додаток В). Сурогатна модель знайшла придатну конфігурацію (4 ГБ оперативної пам'яті, одне ядро процесора, пул 50 з'єднань, 16 робочих процесів) за дві перевірки: перший вибір на межі допустимої області перевищив поріг, а після додавання цього вимірювання до навчальної вибірки уточнена модель обрала конфігурацію з фактичною затримкою 222 мс. Адитивна модель впливу за тих самих умов упродовж 20 перевірок так і не запропонувала придатної конфігурації: її вибір щоразу мав фактичну затримку 327 мс, оскільки модель не відтворює нелінійності поверхні відгуку. Конфігурація з максимальними ресурсами задовольняє поріг із великим запасом, проте в 4,5 раза дорожча (52,8 проти 11,6 умовної одиниці вартості для чотирьох вузлів).

Далі для кожної з чотирьох кандидатних архітектур цифровий двійник перевіряв обидві умови (4.32). Дотримання порогу затримки за втрати компонентів оцінювали статистичним моделюванням: за втрати 40% компонентів, що відповідає допустимій втраті базової архітектури, навантаження перерозподілялося на справні обчислювальні вузли, а сурогатна модель прогнозувала верхню довірчу межу р95-затримки; мінімально прийнятну частку сценаріїв із дотриманням порогу прийнято рівною 0,5, як і для живучості. Функціональну стійкість оцінювали інформаційною технологією забезпечення функціональної стійкості на тій самій 21-добовій траєкторії зі збуреннями, що в підрозділі 5.3, з потужністю, визначеною кількістю обчислювальних вузлів і їхньою конфігурацією; мінімально прийнятне значення індексу функціональної стійкості прийнято 0,95. Результати перевірки наведено в таблиці 5.11.

Перевірка кандидатних архітектур на цифровому двійнику

Архітектур а	Обчисл ювальн их вузлів	Частка сценаріїв із дотриманням порогу затримки (втрата 40%)	Індекс функціонал ьної стійкості з технологіє ю	Індекс функціонал ьної стійкості без технології	Умов и (4.32)	Композит ний критерій
Централізо вана базова	2	0,35	0,983	0,940	не викон ано	0,150
Резервован а	4	0,88	0,989	0,946	викон ано	0,695
Захищена розподілен а (метод)	4	0,88	0,996	0,946	викон ано	0,896
Надлишков а	4	0,88	0,996	0,946	викон ано	0,850

Цифровий двійник вилучив з розгляду централізовану базову архітектуру: на проєктному навантаженні вона поріг затримки дотримує, проте за втрати 40% компонентів у 65% сценаріїв справним лишається лише один обчислювальний вузол і затримка виходить за допустимі межі. Три резервовані архітектури задовольняють обидві умови, тож вибір (4.31) серед них дав архітектуру «Захищена розподілена» з композитним критерієм 0,896. Без інформаційної технології забезпечення функціональної стійкості, тобто за статичної маршрутизації та реагування впродовж 12 год, індекс функціональної стійкості жодного кандидата не досягає 0,95 (не більше 0,946), що кількісно обґрунтовує використання цієї технології для оцінювання кандидатів. Однакова частка сценаріїв для трьох резервованих архітектур пояснюється тим, що вони мають однаковий обчислювальний шар і різняться лише бюджетом безпеки, який на продуктивність не впливає.

Отже, експериментальна верифікація підтвердила положення результату № 3. Оптимізований розподіл ресурсів безпеки підняв індекс безпеки відмовостійкої основи з 0,103 до 0,736 і знизив середньозважену ймовірність успішної атаки за загостреної загрози з 0,893 до 0,218, а спроектована протидія зменшила інтегральний ризик на 59%. Відмовостійка архітектура підвищила

допустиму частку втрати компонентів з 40% до 70% і забезпечила плавну, а не обвальну деградацію функціональності. Багатокритеріальний вибір на етапі проєктування рекомендував збалансовану архітектуру «Захищена розподілена» з найвищим композитним критерієм за помірної вартості, відкинувши як недостатньо захищені, так і надмірно дорогі альтернативи. Цифровий двійник, побудований із сурогатної моделі продуктивності та інформаційної технології забезпечення функціональної стійкості, вилучив архітектуру, що не дотримує порогу затримки за втрати компонентів, а обґрунтована сурогатною моделлю конфігурація вузла виявилася придатною, тоді як адитивна модель придатної конфігурації не знайшла. Тим самим кількісно підтверджено, що метод дозволяє на етапі проєктування обрати архітектуру, стійку до втрати значної частки компонентів і до загроз безпеці функціонування, за коректного застереження про синтетичний характер вихідних даних.

5.5. Зведений аналіз результатів верифікації

Кожен із трьох попередніх підрозділів підтвердив положення відповідного результату за власними метриками й критеріями; тут ці підтвердження зведено докупи, щоб показати цілісність тріади та методологічну спільність верифікації. Зведені кількісні результати верифікації, що підкріплюють положення наукової новизни, наведено в таблиці 5.12.

Взаємозв'язок трьох результатів утворює послідовний ланцюг життєвого циклу корпоративної обчислювальної системи і має не лише змістовий, а й функціональний характер. Результат № 1 діє на етапі проєктування за критерієм продуктивності: сурогатна модель дозволяє передбачити відгук системи й обрати конфігурацію ресурсів за мінімуму натурних вимірювань. Результат № 2 діє на етапі експлуатації: інтегрована технологія забезпечує функціональну стійкість під змінним навантаженням і збуреннями. Результат № 3 діє на етапі структурного проєктування й безпосередньо використовує обидва попередні: сурогатна модель обґрунтовує конфігурацію обчислювальних вузлів кандидатів

і прогнозує їхню затримку за втрати компонентів, а технологія забезпечення функціональної стійкості оцінює стійкість кандидатів у складі цифрового двійника. Верифікація показала, що без цих складових проектне рішення було б гіршим: адитивна модель не знаходить придатної конфігурації вузла, а без технології забезпечення функціональної стійкості жоден кандидат не досягає потрібного рівня стійкості.

Таблиця 5.12

Зведені підтверджені показники верифікації тріади

Результат	Ключовий показник	Базова альтернатива	Запропонований метод
№ 1	Коефіцієнт детермінації сурогатної моделі	0,381 (адитивна)	0,970 (сурогатна)
№ 1	Обсяг натурних вимірювань	720 конфігурацій	75 конфігурацій (-89,6%)
№ 1	Відкаліброваність (покриття / ECE)	не надається	1,00 / 0,055
№ 2	Індекс функціональної стійкості	0,917	0,996
№ 2	Цільова функція	0,481	0,681
№ 2	Необслуговано під час відмов	6,0%	0,0%
№ 3	Індекс безпеки (відмовостійка основа)	0,103	0,736
№ 3	Інтегральний ризик	1,769	0,722 (-59%)
№ 3	Допустима частка втрати компонентів	40%	70%
№ 1 → № 3	Фактична затримка обраної конфігурації вузла	327 мс	222 мс
№ 2 → № 3	Індекс функціональної стійкості кандидатів	не більше 0,946	0,989–0,996

Верифікацію всіх трьох результатів виконано за єдиною методикою. Спільними є детерміновані фізично обґрунтовані синтетичні дані з фіксованим початковим значенням генератора, єдиний науковий інструментарій, узгоджена система метрик і, головне, наскрізний прийом порівняння з коректно визначеною базовою альтернативою на тих самих даних. Завдяки цьому спостережувані переваги в таблиці 5.12 зумовлені саме запропонованими методами, а не відмінностями умов експерименту. Перевага методу в кожному випадку є якісною: майже на порядок вища точність за непридатності адитивної моделі (результат № 1), стійкість біля номінального рівня проти глибоких провалів

(результат № 2), майже двократне підвищення допустимої частки втрати компонентів за зниження ризику на 59% (результат № 3).

Отже, зведений аналіз показує, що три результати підтверджено кожен за власними критеріями, вони узгоджено покривають життєвий цикл корпоративної обчислювальної системи від проєктування за продуктивністю до безпечної відмовостійкої архітектури, а спільна відтворювана методика верифікації надає цим підтвердженням цілісності та зіставності.

Висновки до розділу 5

У розділі експериментально верифіковано три наукові результати дисертації за єдиною відтворюваною методикою на фізично обґрунтованих синтетичних даних із порівнянням кожного методу з коректно визначеною базовою альтернативою.

Підтверджено, що сурогатна модель гаусівської процесової регресії точно апроксимує нелінійну поверхню відгуку продуктивності й надає відкалібровані довірчі інтервали, скорочуючи обсяг натурних вимірювань приблизно на 90% за непридатності адитивних моделей впливу. Підтверджено, що інтегрована технологія функціональної стійкості, поєднуючи моніторинг аномалій, прогнозування, оптимізацію потоків і ризик-економічне керування, зберігає критичну функціональність під збуреннями й перевершує базову реактивну конфігурацію за композитним індексом стійкості та цільовою функцією.

Підтверджено, що метод структурно-параметричного проєктування дозволяє на етапі проєктування обрати відмовостійку й захищену архітектуру, яка витримує втрату значної частки компонентів за плавної деградації та істотно нижчого інтегрального ризику. Перевірка на цифровому двійнику показала, що результати сурогатного моделювання продуктивності та оцінювання функціональної стійкості безпосередньо використовуються в методі структурно-параметричного проєктування й впливають на проєктне рішення.

Отримані підтвердження узгоджені між собою й охоплюють життєвий цикл корпоративної обчислювальної системи від проєктування за продуктивністю через забезпечення функціональної стійкості в експлуатації до проєктування безпечної відмовостійкої архітектури. Достовірність результатів забезпечено детермінованістю й відтворюваністю всіх експериментів, а межі застосовності окреслено застереженням про синтетичний характер вихідних даних і винесенням натурних вимірювань у напрям подальших досліджень.

ВИСНОВКИ

Дисертаційну роботу присвячено підвищенню ефективності автоматизованого проєктування обчислювальних систем для корпоративних структур на основі сурогатного моделювання продуктивності, інтегрованого забезпечення функціональної стійкості та багатокритеріального проєктування відмовостійких і захищених розподілених систем. Основні наукові та практичні результати роботи полягають у такому.

1. На основі аналізу сучасного стану автоматизованого проєктування обчислювальних систем для корпоративних структур встановлено, що продуктивність, функціональну стійкість, надійність і безпеку таких систем оцінюють окремими моделями на різних етапах життєвого циклу, оскільки відповідні напрями сформувалися відокремлено: прогнозування продуктивності спирається на адитивні моделі впливу без відкаліброваної оцінки невизначеності, функціональна стійкість забезпечується реактивно, а відмовостійку й захищену архітектуру обґрунтовують якісно. Звідси виникають пізнє виявлення архітектурних помилок і неможливість кількісно обґрунтувати проєктне рішення до розгортання системи, що зумовило потребу в узгодженому апараті автоматизованого проєктування.

2. Вперше розроблено метод автоматизованого проєктування обчислювальних систем на основі сурогатної моделі гаусівської процесової регресії, яка, на відміну від адитивних моделей впливу, апроксимує нелінійну поверхню відгуку у просторі «конфігурація – навантаження», забезпечує оцінювання невизначеності прогнозу за допомогою відкаліброваних довірчих інтервалів і цілеспрямоване поповнення навчальної вибірки за критерієм предиктивної невизначеності, що дозволяє обґрунтовано визначати раціональну конфігурацію системи за змінних умов навантаження. Експериментальна верифікація показала коефіцієнт детермінації 0,970 проти 0,381 в адитивній моделі, покриття довірчих інтервалів 1,00 за очікуваної похибки калібрування

0,055 і досягнення заданого порогу точності за 75 натурних вимірювань замість 720, тобто скорочення їх обсягу на 89,6%.

3. Удосконалено інформаційну технологію забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи, яка, на відміну від відомих реактивних рішень, інтегрує статистичний моніторинг аномалій, прогнозування поведінки методами машинного навчання, оптимізацію інформаційних потоків та оцінювання ризиків у єдиний контур керування, що забезпечує адаптацію системи в режимі реального часу та підтримання критичної функціональності в умовах збурень. Верифікація підтвердила підвищення індексу функціональної стійкості з 0,917 до 0,996 і значення цільової функції з 0,481 до 0,681, повне усунення недообслуговування інформаційних потоків під час відмов (з 6,0% до 0,0%), прогнозування навантаження із середньою абсолютною відсотковою похибкою 5,2% та зниження втрат від часу реагування на 53,5%.

4. Набули подальшого розвитку метод та інформаційна технологія структурно-параметричного проєктування відмовостійких і захищених розподілених обчислювальних систем корпоративних структур, які відрізняються від існуючих використанням результатів методу автоматизованого проєктування на основі сурогатної моделі гаусівської процесової регресії для обґрунтування параметрів конфігурації та результатів інформаційної технології забезпечення функціональної стійкості для врахування вимог до стійкості системи, у поєднанні з багатокритеріальною оптимізацією розподілу ресурсів безпеки з урахуванням живучості, адаптивності та плавної деградації; інформаційна технологія реалізує метод сукупністю програмних модулів зі сховищем проєктних даних, цифровим двійником і панеллю підтримки рішень. Верифікація підтвердила підвищення допустимої частки втрати компонентів із 40% до 70% за плавної деградації, зниження інтегрального ризику з 1,769 до 0,722, тобто на 59%, підвищення індексу безпеки відмовостійкої основи з 0,103 до 0,736 та вибір збалансованої архітектури за

найвищим композитним критерієм 0,896, а перевірка на цифровому двійнику вилучила архітектуру, що не дотримує порогу затримки за втрати компонентів.

5. Експериментальну верифікацію запропонованих методів та інформаційних технологій виконано засобами комп'ютерного моделювання за єдиною відтворюваною методикою на фізично обґрунтованих синтетичних даних із порівнянням кожного рішення з коректно визначеною базовою альтернативою на тих самих даних, що за фіксованого початкового значення генератора й узгодженої системи метрик дозволило віднести спостережувані переваги саме до запропонованих рішень.

Практичне значення результатів полягає в тому, що метод сурогатного моделювання, інформаційну технологію забезпечення функціональної стійкості, а також метод та інформаційну технологію структурно-параметричного проєктування реалізовано у вигляді програмних засобів і панелей підтримки рішень, придатних для конфігурування ресурсів корпоративних обчислювальних систем за мінімуму натурних випробувань, підтримання їх функціональної стійкості службою експлуатації та обґрунтованого вибору відмовостійкої й захищеної архітектури ще до розгортання. Результати роботи впроваджено в Інституті програмних систем НАН України, у навчальний процес Державного університету інформаційно-комунікаційних технологій, що підтверджено актами впровадження та в ТОВ «Байт Орчард Консалтінг», яка надала статистичне підтвердження отриманих результатів.

Отримані результати свідчать про досягнення мети дослідження. Перспективи подальших досліджень пов'язані з відтворенням верифікації на реальних апаратних і телеметричних даних корпоративних систем, розробленням механізмів автоматичного переналаштування конфігурації та рівня захисту за поточним станом системи й поширенням запропонованого підходу на багатокластерні та географічно розподілені обчислювальні середовища.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ільїна І., Артюх Р., Зимогляд М. Аналіз даних та машинне навчання у хмарних та туманних платформах для ефективної передачі даних. Системи управління, навігації та зв'язку. 2025. № 2 (80). С. 132–136. DOI: 10.26906/SUNZ.2025.2.132.
2. Гнатієнко О., Дружинін В. Сучасні тенденції забезпечення функціональної стійкості складних організаційних систем. Управління розвитком складних систем. 2024. № 60. С. 13–17. DOI: 10.32347/2412-9933.2024.60.13-17.
3. Додонов О. Г., Кузнєцова М. Г., Горбачик О. С. Моделювання і оцінювання функціональної стійкості інформаційних систем. Реєстрація, зберігання і обробка даних. 2025. Т. 27, № 1. С. 76–88. DOI: 10.35681/1560-9189.2025.27.1.335752.
4. Замрій І. В., Собчук А. В., Лаптев С. О., Лаптева Т. О., Копитко С. Б. Алгоритм контролю та прогнозування функціональної стійкості складних інформаційно-технічних систем. Телекомунікаційні та інформаційні технології. 2022. № 1 (74). С. 4–14. DOI: 10.31673/2412-4338.2022.010414.
5. Будзинський О., Щавінський Ю., Мужанова Т., Якименко Ю., Примаченко Д. Методика інтелектуальної оцінки ризиків інформаційної безпеки корпоративних баз даних. Кібербезпека: освіта, наука, техніка. 2026. Т. 1, № 33. С. 399–413. DOI: 10.28925/2663-4023.2026.33.1221.
6. Красій Д. М., Ларін О. О. Фізично-керована ланцюгова гаусівська регресія процесу для прогнозування прогресії атеросклеротичної бляшки. Науковий вісник Ужгородського університету. Серія «Математика і інформатика». 2026. Т. 48, № 1. С. 153–161. DOI: 10.24144/2616-7700.2026.48(1).153-161.
7. Олімпієва Ю. Забезпечення функціональної стійкості виробничих процесів промислових підприємств на основі нейроадаптивної системи. Системи

управління, навігації та зв'язку. 2024. Т. 3, № 77. С. 46–54. DOI: 10.26906/SUNZ.2024.3.046.

8. Aghazadeh Ardebili A., Lezzi M., Pourmadadkar M. Risk assessment for cyber resilience of critical infrastructures: methods, governance, and standards. *Applied Sciences*. 2024. Т. 14, № 24. Ст. 11807. DOI: 10.3390/app142411807.

9. Ahmed M., Naser Mahmood A., Hu J. A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*. 2016. Т. 60. С. 19–31. DOI: 10.1016/j.jnca.2015.11.016.

10. Ahuja R. K., Magnanti T. L., Orlin J. B. *Network flows: theory, algorithms, and applications*. Englewood Cliffs, NJ : Prentice Hall, 1993. 846 с.

11. Alaeifar P., Pal S., Jadidi Z., Hussain M., Foo E. Current approaches and future directions for cyber threat intelligence sharing: a survey. *Journal of Information Security and Applications*. 2024. Т. 83. Ст. 103786. DOI: 10.1016/j.jisa.2024.103786.

12. Ali S., Abuhmed T., El-Sappagh S., Muhammad K., Alonso-Moral J. M., Confalonieri R., Guidotti R., Del Ser J., Diaz-Rodriguez N., Herrera F. Explainable artificial intelligence (XAI): what we know and what is left to attain trustworthy artificial intelligence. *Information Fusion*. 2023. Т. 99. Ст. 101805. DOI: 10.1016/j.inffus.2023.101805.

13. Alotaibi B. та ін. A survey on Industrial Internet of Things security: requirements, attacks, AI-based solutions, and edge computing opportunities. *Sensors*. 2023. Т. 23, № 17. Ст. 7470. DOI: 10.3390/s23177470.

14. Anderson R. *Security engineering: a guide to building dependable distributed systems*. 3-тє вид. Indianapolis : Wiley, 2020. 1232 с.

15. Apruzzese G., Colajanni M., Ferretti L., Guido A., Marchetti M. On the effectiveness of machine and deep learning for cyber security. *Proceedings of the 10th International Conference on Cyber Conflict (CyCon)*. Tallinn : IEEE, 2018. С. 371–390. DOI: 10.23919/CYCON.2018.8405026.

16. Arora S., Rani R., Saxena N. A systematic review on detection and adaptation of concept drift in streaming data using machine learning techniques.

WIREs Data Mining and Knowledge Discovery. 2024. T. 14, № 4. Ст. e1536. DOI: 10.1002/widm.1536.

17. Avizienis A., Laprie J.-C., Randell B., Landwehr C. Basic concepts and taxonomy of dependable and secure computing. IEEE Transactions on Dependable and Secure Computing. 2004. T. 1, № 1. С. 11–33. DOI: 10.1109/TDSC.2004.2.

18. Barabash O. V., Svynchuk O. V., Salanda I. P., Mashkov V. A., Myroniuk M. M. Ensuring the functional stability of the information system of the power plant on the basis of monitoring the parameters of the working condition of computer devices. Advanced Information Systems. 2024. T. 8, № 2. С. 107–117. DOI: 10.20998/2522-9052.2024.2.12.

19. Barabash O., Sobchuk V., Musienko A., Laptiev O., Bohomia V., Kopytko S. System analysis and method of ensuring functional sustainability of the information system of a critical infrastructure object. System Analysis and Artificial Intelligence : Studies in Computational Intelligence. Cham : Springer, 2023. Т. 1107. С. 177–192. DOI: 10.1007/978-3-031-37450-0_11.

20. Bernstein D. J., Lange T. Post-quantum cryptography. Nature. 2017. Т. 549, № 7671. С. 188–194. DOI: 10.1038/nature23461.

21. Bertsimas D., Tsitsiklis J. N. Introduction to linear optimization. Belmont, MA : Athena Scientific, 1997. 608 с.

22. Binois M., Wycoff N. A survey on high-dimensional Gaussian process modeling with application to Bayesian optimization. ACM Transactions on Evolutionary Learning and Optimization. 2022. Т. 2, № 2. С. 1–26. DOI: 10.1145/3545611.

23. Bischl B., Binder M., Lang M., Pielok T., Richter J. та ін. Hyperparameter optimization: foundations, algorithms, best practices, and open challenges. WIREs Data Mining and Knowledge Discovery. 2023. Т. 13, № 2. Ст. e1484. DOI: 10.1002/widm.1484.

24. Bolanowski M., Cmil M., Starzec A. New model for defining and implementing performance tests. Future Internet. 2024. Т. 16, № 10. Ст. 366. DOI: 10.3390/fi16100366.

25. Borisut P., Nuchitprasittichai A. Adaptive Latin hypercube sampling for a surrogate-based optimization with artificial neural network. *Processes*. 2023. T. 11, № 11. Ст. 3232. DOI: 10.3390/pr11113232.

26. Box G. E. P., Jenkins G. M., Reinsel G. C., Ljung G. M. *Time series analysis: forecasting and control*. 5-те вид. Hoboken, NJ : Wiley, 2015. 712 с.

27. Bruneau M., Chang S. E., Eguchi R. T. та ін. A framework to quantitatively assess and enhance the seismic resilience of communities. *Earthquake Spectra*. 2003. T. 19, № 4. С. 733–752. DOI: 10.1193/1.1623497.

28. Burn M. J., Popelier P. L. A. FEREBUS: a high-performance modern Gaussian process regression engine. *Digital Discovery*. 2023. T. 2, № 1. С. 152–164. DOI: 10.1039/D2DD00082B.

29. Carrion C. Kubernetes scheduling: taxonomy, ongoing issues and challenges. *ACM Computing Surveys*. 2022. T. 55, № 7. С. 1–37. DOI: 10.1145/3539606.

30. Chandola V., Banerjee A., Kumar V. Anomaly detection: a survey. *ACM Computing Surveys*. 2009. T. 41, № 3. Ст. 15. DOI: 10.1145/1541880.1541882.

31. Chelak V., Hornostal O., Chelak Y., Gavrylenko S. Deep learning-based intrusion detection in computer systems and networks: advances, hybrids, and challenges 2022–2026. *Advanced Information Systems*. 2026. T. 10, № 3. С. 80–96. DOI: 10.20998/2522-9052.2026.3.09.

32. Cheng B. H. C., de Lemos R., Giese H. та ін. *Software engineering for self-adaptive systems: a research roadmap*. *Software Engineering for Self-Adaptive Systems (LNCS 5525)*. Berlin : Springer, 2009. С. 1–26. DOI: 10.1007/978-3-642-02161-9_1.

33. Chen P., Chen T. PromiseTune: unveiling causally promising and explainable configuration tuning. *Proceedings of the 48th IEEE/ACM International Conference on Software Engineering (ICSE 2026)*. Piscataway, NJ : IEEE, 2026. (y друці).

34. Cheng J., Gao C., Zheng Z. HINNPerf: hierarchical interaction neural network for performance prediction of configurable systems. *ACM Transactions on*

Software Engineering and Methodology. 2023. T. 32, № 2. C. 1–30. DOI: 10.1145/3528100.

35. Cherkaoui Dekkaki K., Tasic I., Cano M.-D. Exploring post-quantum cryptography: review and directions for the transition process. *Technologies*. 2024. T. 12, № 12. Ст. 241. DOI: 10.3390/technologies12120241.

36. Conti P., Guo M., Manzoni A., Frangi A., Brunton S. L., Kutz J. N. Multi-fidelity reduced-order surrogate modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*. 2024. T. 480, № 2283. Ст. 20230655. DOI: 10.1098/rspa.2023.0655.

37. Cross E. J., Rogers T. J., Pitchforth D. J., Gibson S. J., Zhang S., Jones M. R. A spectrum of physics-informed Gaussian processes for regression in engineering. *Data-Centric Engineering*. 2024. T. 5. Ст. e8. DOI: 10.1017/dce.2024.2.

38. De Alwis C., Porambage P., Dev K., Gadekallu T. R., Liyanage M. A survey on network slicing security: attacks, challenges, solutions and research directions. *IEEE Communications Surveys & Tutorials*. 2024. T. 26, № 1. C. 534–570. DOI: 10.1109/COMST.2023.3312349.

39. Deb K., Pratap A., Agarwal S., Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*. 2002. T. 6, № 2. C. 182–197. DOI: 10.1109/4235.996017.

40. DeGroot M. H., Fienberg S. E. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*. 1983. T. 32, № 1–2. C. 12–22. DOI: 10.2307/2987588.

41. Dorn J., Apel S., Siegmund N. Mastering uncertainty in performance estimations of configurable software systems. *Empirical Software Engineering*. 2023. T. 28. Ст. 33. DOI: 10.1007/s10664-022-10250-2.

42. Dorn J., Apel S., Siegmund N. Mastering uncertainty in performance estimations of configurable software systems. *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering (ASE 2020)*. New York : ACM, 2020. C. 684–696. DOI: 10.1145/3324884.3416620.

43. Dovbysh A., Liubchak V., Shelehov I., Simonovskiy J., Tenytska A. Information-extreme machine learning of a cyber attack detection system. Radioelectronic and Computer Systems. 2022. № 3. С. 121–131. DOI: 10.32620/reks.2022.3.09.

44. Eckhart M., Ekelhart A. Digital twins for cyber-physical systems security: state of the art and outlook. Security and Quality in Cyber-Physical Systems Engineering. Cham : Springer, 2019. С. 383–412. DOI: 10.1007/978-3-030-25312-7_14.

45. Ehrgott M. Multicriteria optimization. 2-ге вид. Berlin : Springer, 2005. 323 с.

46. Ellison R. J., Fisher D. A., Linger R. C., Lipson H. F., Longstaff T., Mead N. R. Survivable network systems: an emerging discipline : Technical Report CMU/SEI-97-TR-013. Pittsburgh : Software Engineering Institute, Carnegie Mellon University, 1997 (rev. 1999). 42 с.

47. Farooq M. S., Riaz S., Alvi A. Security and privacy issues in software-defined networking (SDN): a systematic literature review. Electronics. 2023. Т. 12, № 14. Ст. 3077. DOI: 10.3390/electronics12143077.

48. FIRST. Common Vulnerability Scoring System version 3.1: specification document. Cary, NC : Forum of Incident Response and Security Teams, 2019. 24 с.

49. Ford L. R., Fulkerson D. R. Maximal flow through a network. Canadian Journal of Mathematics. 1956. Т. 8. С. 399–404. DOI: 10.4153/CJM-1956-045-5.

50. Freund J., Jones J. Measuring and managing information risk: a FAIR approach. Waltham, MA : Butterworth-Heinemann, 2014. 408 с.

51. Friedman J. H. Greedy function approximation: a gradient boosting machine. Annals of Statistics. 2001. Т. 29, № 5. С. 1189–1232. DOI: 10.1214/aos/1013203451.

52. Fuller A., Fan Z., Day C., Barlow C. Digital twin: enabling technologies, challenges and open research. IEEE Access. 2020. Т. 8. С. 108952–108971. DOI: 10.1109/ACCESS.2020.2998358.

53. Ganapathi A. Predicting and optimizing system utilization and performance via statistical machine learning : PhD dissertation. Berkeley : University of California, 2009. 138 c.

54. Gao S., Lin R., Fu Y., Li H., Cao J. Security threats, requirements and recommendations on creating 5G network slicing system: a survey. *Electronics*. 2024. T. 13, № 10. Ст. 1860. DOI: 10.3390/electronics13101860.

55. Gawlikowski J., Tassi C. R. N., Ali M. та ін. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*. 2023. T. 56, № S1. С. 1513–1589. DOI: 10.1007/s10462-023-10562-9.

56. Genton M. G. Classes of kernels for machine learning: a statistics perspective. *Journal of Machine Learning Research*. 2001. T. 2. С. 299–312.

57. Gnatyuk S. та ін. Method for managing IT incidents in critical information infrastructure facilities. *Cybersecurity Providing in Information and Telecommunication Systems II. CEUR Workshop Proceedings*. 2024. T. 3826. С. 326–333.

58. Gnatyuk S., Berdibayev R., Sydorenko V., Zhyharevych O., Smirnova T. System for cyber security events correlation and incident management in critical infrastructure objects. *Кибербезпека: освіта, наука, техніка*. 2023. № 3 (19). С. 176–196. DOI: 10.28925/2663-4023.2023.19.176196.

59. Gong J., Chen T. Deep configuration performance learning: a systematic survey and taxonomy. *ACM Transactions on Software Engineering and Methodology*. 2024. T. 34, № 1. С. 1–62. DOI: 10.1145/3702986.

60. Gong J., Chen T. Predicting configuration performance in multiple environments with sequential meta-learning. *Proceedings of the ACM on Software Engineering*. 2024. T. 1, № FSE. С. 359–382. DOI: 10.1145/3643743.

61. Gong J., Chen T. Predicting software performance with divide-and-learn. *Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2023)*. New York : ACM, 2023. С. 858–870. DOI: 10.1145/3611643.3616334.

62. Gordon L. A., Loeb M. P. The economics of information security investment. *ACM Transactions on Information and System Security*. 2002. T. 5, № 4. C. 438–457. DOI: 10.1145/581271.581274.

63. Guru A., Mohanta B. K., Mohapatra H., Al-Turjman F., Altrjman C., Yadav A. A survey on consensus protocols and attacks on blockchain technology. *Applied Sciences*. 2023. T. 13, № 4. CT. 2604. DOI: 10.3390/app13042604.

64. Hagberg A. A., Schult D. A., Swart P. J. Exploring network structure, dynamics, and function using NetworkX. *Proceedings of the 7th Python in Science Conference (SciPy 2008)* / eds. G. Varoquaux, T. Vaught, J. Millman. Pasadena, CA, 2008. C. 11–15.

65. Harris C. R., Millman K. J., van der Walt S. J. та ін. Array programming with NumPy. *Nature*. 2020. T. 585, № 7825. C. 357–362. DOI: 10.1038/s41586-020-2649-2.

66. He C., Zhang Y., Gong D., Ji X. A review of surrogate-assisted evolutionary algorithms for expensive optimization problems. *Expert Systems with Applications*. 2023. T. 217. CT. 119495. DOI: 10.1016/j.eswa.2022.119495.

67. Henry D., Ramirez-Marquez J. E. Generic metrics and quantitative approaches for system resilience as a function of time. *Reliability Engineering & System Safety*. 2012. T. 99. C. 114–122. DOI: 10.1016/j.ress.2011.09.002.

68. Hlybovets A., Paprotskyi I. Increasing the fault tolerance in microservice architecture. *Cybernetics and Systems Analysis*. 2024. T. 60, № 3. C. 480–488. DOI: 10.1007/s10559-024-00689-0.

69. Hosseini S., Barker K., Ramirez-Marquez J. E. A review of definitions and measures of system resilience. *Reliability Engineering & System Safety*. 2016. T. 145. C. 47–61. DOI: 10.1016/j.ress.2015.08.006.

70. Hu X., Ma P., Li S., Li W., Yang M. The optimization of space-filling and orthogonality for Latin hypercube design using a local search-based NSGA-II. *Journal of Statistical Theory and Practice*. 2025. T. 19, № 1. CT. 12. DOI: 10.1007/s42519-025-00430-w.

71. Huang M., Mao P., Li K. Rethinking performance analysis for configurable software systems: a case study from a fitness landscape perspective. Proceedings of the ACM on Software Engineering. 2025. T. 2, № ISSTA. C. 1748–1771. DOI: 10.1145/3728954.

72. Hunter J. D. Matplotlib: a 2D graphics environment. Computing in Science & Engineering. 2007. T. 9, № 3. C. 90–95. DOI: 10.1109/MCSE.2007.55.

73. Hyndman R. J., Athanasopoulos G. Forecasting: principles and practice. 3-тє вид. Melbourne : OTexts, 2021. 442 с.

74. Idouglid L., Tkatek S., Elfayq K., Guezzaz A. A novel anomaly detection model for the industrial Internet of Things using machine learning techniques. Radioelectronic and Computer Systems. 2024. № 1. C. 143–151. DOI: 10.32620/reks.2024.1.12.

75. ISO/IEC 27005:2022. Information security, cybersecurity and privacy protection – Guidance on managing information security risks. 4-тє вид. Geneva : ISO, 2022. 62 с.

76. Jamshidi P., Siegmund N., Velez M., Kästner C., Patel A., Agarwal Y. Transfer learning for performance modeling of configurable systems: an exploratory analysis. Proceedings of the 32nd IEEE/ACM International Conference on Automated Software Engineering (ASE 2017). Piscataway, NJ : IEEE, 2017. C. 497–508. DOI: 10.1109/ASE.2017.8115661.

77. Javadpour A., Ja'fari F., Taleb T., Shojafar M., Benzaid C. A comprehensive survey on cyber deception techniques to improve honeypot performance. Computers & Security. 2024. T. 140. Ст. 103792. DOI: 10.1016/j.cose.2024.103792.

78. Jeffrey N., Tan Q., Villar J. R. Using ensemble learning for anomaly detection in cyber–physical systems. Electronics. 2024. T. 13, № 7. Ст. 1391. DOI: 10.3390/electronics13071391.

79. Jeremiah S. R., El Azzaoui A., Xiong N. N., Park J. H. A comprehensive survey of digital twins: applications, technologies and security challenges. Journal of Systems Architecture. 2024. T. 151. Ст. 103120. DOI: 10.1016/j.sysarc.2024.103120.

80. Jin Y., Wang H., Chugh T., Guo D., Miettinen K. Data-driven evolutionary optimization: an overview and case studies. *IEEE Transactions on Evolutionary Computation*. 2019. T. 23, № 3. C. 442–458. DOI: 10.1109/TEVC.2018.2869001.

81. Katal A., Dahiya S., Choudhury T. Energy efficiency in cloud computing data centers: a survey on software technologies. *Cluster Computing*. 2023. T. 26. C. 1845–1875. DOI: 10.1007/s10586-022-03713-0.

82. Kephart J. O., Chess D. M. The vision of autonomic computing. *Computer*. 2003. T. 36, № 1. C. 41–50. DOI: 10.1109/MC.2003.1160055.

83. Kleinrock L. Queueing systems. Volume 1: Theory. New York : Wiley-Interscience, 1975. 417 c.

84. Knight J. C., Sullivan K. J. On the definition of survivability : Technical Report CS-TR-33-00. Charlottesville : University of Virginia, 2000. 10 c.

85. Kong X., Chen Z., Liu W., Ning K., Zhang L., Marier S. M., Liu Y., Chen Y., Xia F. Deep learning for time series forecasting: a survey. *International Journal of Machine Learning and Cybernetics*. 2025. T. 16, № 7–8. C. 5079–5112. DOI: 10.1007/s13042-025-02560-w.

86. Konsta A.-M., Lluch Lafuente A., Spiga B., Dragoni N. Survey: automatic generation of attack trees and attack graphs. *Computers & Security*. 2024. T. 137. Ст. 103602. DOI: 10.1016/j.cose.2023.103602.

87. Kordy B., Piètre-Cambacédès L., Schweitzer P. DAG-based attack and defense modeling: don't miss the forest for the attack trees. *Computer Science Review*. 2014. T. 13–14. C. 1–38. DOI: 10.1016/j.cosrev.2014.07.001.

88. Koren I., Krishna C. M. Fault-tolerant systems. 2-ге вид. Cambridge, MA : Morgan Kaufmann, 2020. 424 c.

89. Kreuzberger D., Kuhl N., Hirschl S. Machine learning operations (MLOps): overview, definition, and architecture. *IEEE Access*. 2023. T. 11. C. 31866–31879. DOI: 10.1109/ACCESS.2023.3262138.

90. Kuleshov V., Fenner N., Ermon S. Accurate uncertainties for deep learning using calibrated regression. *Proceedings of the 35th International Conference on Machine Learning (ICML 2018)*. PMLR, 2018. T. 80. C. 2796–2804.

91. Kuo W., Zuo M. J. Optimal reliability modeling: principles and applications. Hoboken, NJ : Wiley, 2003. 544 c.
92. Lakhno V., Kasatkin D. A fuzzy-logical model for assessing the status of an information system during DDoS attacks. Information technologies in economics and environmental sciences. 2025. № 2. C. 56–62. DOI: 10.31548/itees.2025.02.056.
93. Laprie J.-C. From dependability to resilience. Proceedings of the 38th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2008) : Fast Abstracts. Piscataway, NJ : IEEE, 2008. C. G8–G9.
94. Lesoil L., Acher M., Blouin A., Jezequel J.-M. Input sensitivity on the performance of configurable systems: an empirical study. Journal of Systems and Software. 2023. T. 201. Cт. 111671. DOI: 10.1016/j.jss.2023.111671.
95. Li D., Wang Z., Chen Y., Jiang R., Ding W., Okumura M. A survey on deep active learning: recent advances and new frontiers. IEEE Transactions on Neural Networks and Learning Systems. 2025. T. 36, № 4. C. 5879–5899. DOI: 10.1109/TNNLS.2024.3396463.
96. Li H., Ge L., Tian L. Survey: federated learning data security and privacy-preserving in edge-Internet of Things. Artificial Intelligence Review. 2024. T. 57, № 5. Cт. 130. DOI: 10.1007/s10462-024-10774-7.
97. Li Z., Guo L., Cheng J., Chen Q., He B., Guo M. The serverless computing survey: a technical primer for design architecture. ACM Computing Surveys. 2022. T. 54, № 10s. C. 1–34. DOI: 10.1145/3508360.
98. Liang J., Lou Y., Yu M., Bi Y., Yu K. A survey of surrogate-assisted evolutionary algorithms for expensive optimization. Journal of Membrane Computing. 2024. T. 7, № 2. C. 108–127. DOI: 10.1007/s41965-024-00165-w.
99. Lundberg S. M., Lee S.-I. A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems 30 (NeurIPS 2017). Red Hook, NY : Curran Associates, 2017. C. 4765–4774.
100. Lyu C., Liu X., Mihaylova L. Review of recent advances in Gaussian process regression methods. Advances in Computational Intelligence Systems :

Advances in Intelligent Systems and Computing. Cham : Springer, 2024. T. 1454. C. 226–237. DOI: 10.1007/978-3-031-55568-8_19.

101. Ma H., Zhang Y., Sun S., Liu T., Shan Y. A comprehensive survey on NSGA-II for multi-objective optimization and applications. Artificial Intelligence Review. 2023. T. 56, № 12. C. 15217–15270. DOI: 10.1007/s10462-023-10526-z.

102. Machiraju V. S., Kumar V., Sharma S. ML-based autoscaling for elastic cloud applications: taxonomy, frameworks, and evaluation. Mathematical and Computational Applications. 2026. T. 31, № 2. Cт. 49. DOI: 10.3390/mca31020049.

103. Marrel A., Iooss B. Probabilistic surrogate modeling by Gaussian process: a review on recent insights in estimation and validation. Reliability Engineering & System Safety. 2024. T. 247. Cт. 110094. DOI: 10.1016/j.res.2024.110094.

104. Martinez K., Claudio D. Expanding fundamental boundaries between resilience and survivability in systems engineering: a literature review. Sustainability. 2023. T. 15, № 6. Cт. 4811. DOI: 10.3390/su15064811.

105. Martinez Saucedo A., Rodriguez G., Gomes Rocha F., Pereira dos Santos R. Migration of monolithic systems to microservices: a systematic mapping study. Information and Software Technology. 2025. T. 177. Cт. 107590. DOI: 10.1016/j.infsof.2024.107590.

106. McKay M. D., Beckman R. J., Conover W. J. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics. 1979. T. 21, № 2. C. 239–245. DOI: 10.1080/00401706.1979.10489755.

107. Melara M. S., Torres-Arias S. A viewpoint on software supply chain security: are we getting lost in translation? IEEE Security & Privacy. 2023. T. 21, № 6. C. 55–58. DOI: 10.1109/MSEC.2023.3316568.

108. Memon Q. A., Al Ahmad M., Pecht M. Quantum computing: navigating the future of computation, challenges, and technological breakthroughs. Quantum Reports. 2024. T. 6, № 4. Cт. 39. DOI: 10.3390/quantum6040039.

109. Miettinen K. Nonlinear multiobjective optimization. Boston : Kluwer Academic Publishers, 1999. 298 c.

110. Mohale V. Z., Obagbuwa I. C. A systematic review on the integration of explainable artificial intelligence in intrusion detection systems to enhancing transparency and interpretability in cybersecurity. *Frontiers in Artificial Intelligence*. 2025. T. 8. Ст. 1526221. DOI: 10.3389/frai.2025.1526221.

111. Mohammed K., Shanmugam B., El-Den J. Evolution of DevSecOps and its influence on application security: a systematic literature review. *Technologies*. 2025. T. 13, № 12. Ст. 548. DOI: 10.3390/technologies13120548.

112. Momand A., Jan S. U., Ramzan N. A systematic and comprehensive survey of recent advances in intrusion detection systems using machine learning: deep learning, datasets, and attack taxonomy. *Journal of Sensors*. 2023. T. 2023. Ст. 6048087. DOI: 10.1155/2023/6048087.

113. Montgomery D. C. Design and analysis of experiments. 10-е вид. Hoboken, NJ : Wiley, 2019. 688 с.

114. Montgomery D. C. Introduction to statistical quality control. 8-е вид. Hoboken, NJ : Wiley, 2019. 768 с.

115. Mozaffari M., Dignos A., Gamper J., Storl U. Self-tuning database systems: a systematic literature review of automatic database schema design and tuning. *ACM Computing Surveys*. 2024. T. 56, № 11. С. 1–37. DOI: 10.1145/3665323.

116. Muhlbauer S., Apel S., Siegmund N. Accurate modeling of performance histories for evolving software systems. *Proceedings of the 34th IEEE/ACM International Conference on Automated Software Engineering (ASE 2019)*. Piscataway, NJ : IEEE, 2019. С. 640–652. DOI: 10.1109/ASE.2019.00065.

117. Nahar N., Andersson K., Schelen O., Saguna S. A survey on zero trust architecture: applications and challenges of 6G networks. *IEEE Access*. 2024. T. 12. С. 94753–94764. DOI: 10.1109/ACCESS.2024.3425350.

118. Ngo D., Park H.-C., Kang B. Edge intelligence: a review of deep neural network inference in resource-limited environments. *Electronics*. 2025. T. 14, № 12. Ст. 2495. DOI: 10.3390/electronics14122495.

119. NIST. FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard. Gaithersburg, MD : National Institute of Standards and Technology, 2024. DOI: 10.6028/NIST.FIPS.203.

120. NIST. FIPS 204: Module-Lattice-Based Digital Signature Standard. Gaithersburg, MD : National Institute of Standards and Technology, 2024. DOI: 10.6028/NIST.FIPS.204.

121. NIST. FIPS 205: Stateless Hash-Based Digital Signature Standard. Gaithersburg, MD : National Institute of Standards and Technology, 2024. DOI: 10.6028/NIST.FIPS.205.

122. NIST. SP 800-30 Rev. 1: Guide for conducting risk assessments. Gaithersburg, MD : National Institute of Standards and Technology, 2012. 95 c. DOI: 10.6028/NIST.SP.800-30r1.

123. Noack M. M., Luo H., Risser M. D. A unifying perspective on non-stationary kernels for deeper Gaussian processes. APL Machine Learning. 2024. T. 2, № 1. CT. 010902. DOI: 10.1063/5.0176963.

124. Oszczypala M., Konwerski J., Ziolkowski J., Malachowski J. Reliability analysis and redundancy optimization of k-out-of-n systems with random variable k using continuous time Markov chain and Monte Carlo simulation. Reliability Engineering & System Safety. 2024. T. 242. CT. 109780. DOI: 10.1016/j.ress.2023.109780.

125. Pahl C., Sezen O. C., Hofer F. Artificial intelligence for infrastructure-as-code – a systematic literature review. Electronics. 2026. T. 15, № 4. CT. 755. DOI: 10.3390/electronics15040755.

126. Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B. та ін. Scikit-learn: machine learning in Python. Journal of Machine Learning Research. 2011. T. 12. C. 2825–2830.

127. Poyser M., Breckon T. P. Neural architecture search: a contemporary literature review for computer vision applications. Pattern Recognition. 2024. T. 147. CT. 110052. DOI: 10.1016/j.patcog.2023.110052.

128. Qi X., Mei G. Network resilience: definitions, approaches, and applications. *Journal of King Saud University – Computer and Information Sciences*. 2024. T. 36, № 1. СТ. 101882. DOI: 10.1016/j.jksuci.2023.101882.

129. Rasmussen C. E., Williams C. K. I. Gaussian processes for machine learning. Cambridge, MA : MIT Press, 2006. 248 c.

130. Ribeiro L. G., Parente Jr. E., de Melo A. M. C. Alternative variable-fidelity acquisition functions for efficient global optimization of black-box functions. *Structural and Multidisciplinary Optimization*. 2023. T. 66, № 7. СТ. 147. DOI: 10.1007/s00158-023-03607-8.

131. Riedl R., Stieninger M., Muehlburger M., Koch S., Hess T. What is digital transformation? A survey on the perceptions of decision-makers in business. *Information Systems and e-Business Management*. 2024. T. 22. C. 61–95. DOI: 10.1007/s10257-023-00660-0.

132. Robinson S., Brooks R. Assumptions and simplifications in discrete-event simulation modelling. *Journal of Simulation*. 2024. T. 19, № 6. C. 639–656. DOI: 10.1080/17477778.2024.2407369.

133. Rose S., Borchert O., Mitchell S., Connelly S. Zero Trust Architecture : NIST Special Publication 800-207. Gaithersburg, MD : National Institute of Standards and Technology, 2020. 59 c. DOI: 10.6028/NIST.SP.800-207.

134. Saeed M. M., Saeed R. A., Hasan M. K., Ali E. S. та иҲ. A comprehensive survey on 6G-security: physical connection and service layers. *Discover Internet of Things*. 2025. T. 5, № 1. СТ. 28. DOI: 10.1007/s43926-025-00123-7.

135. Sarker I. H. Deep cybersecurity: a comprehensive overview from neural network and deep learning perspective. *SN Computer Science*. 2021. T. 2, № 3. СТ. 154. DOI: 10.1007/s42979-021-00535-6.

136. Scyphers M. E., Missik J. E. C., Kujawa H., Paulson J. A., Bohrer G. Bayesian optimization for anything (BOA): an open-source framework for accessible, user-friendly Bayesian optimization. *Environmental Modelling & Software*. 2024. T. 182. СТ. 106191. DOI: 10.1016/j.envsoft.2024.106191.

137. Semmelrock H., Ross-Hellauer T., Kopeinik S., Theiler D., Haberl A., Thalmann S., Kowald D. Reproducibility in machine-learning-based research: overview, barriers, and drivers. *AI Magazine*. 2025. T. 46, № 2. Ст. e70002. DOI: 10.1002/aaai.70002.

138. Shahriari B., Swersky K., Wang Z., Adams R. P., de Freitas N. Taking the human out of the loop: a review of Bayesian optimization. *Proceedings of the IEEE*. 2016. T. 104, № 1. С. 148–175. DOI: 10.1109/JPROC.2015.2494218.

139. Shelton C. P., Koopman P., Nace W. A framework for scalable analysis and design of system-wide graceful degradation in distributed embedded systems. *Proceedings of the 8th IEEE International Workshop on Object-Oriented Real-Time Dependable Systems (WORDS 2003)*. Los Alamitos, CA : IEEE Computer Society, 2003. С. 156–163. DOI: 10.1109/WORDS.2003.1218078.

140. Sheyner O., Haines J., Jha S., Lippmann R., Wing J. M. Automated generation and analysis of attack graphs. *Proceedings of the 2002 IEEE Symposium on Security and Privacy*. Los Alamitos, CA : IEEE Computer Society, 2002. С. 273–284. DOI: 10.1109/SECPRI.2002.1004377.

141. Shyaa M. A., Ibrahim N. F., Zainol Z., Abdullah R., Anbar M., Alzubaidi L. Evolving cybersecurity frontiers: a comprehensive survey on concept drift and feature dynamics aware machine and deep learning in intrusion detection systems. *Engineering Applications of Artificial Intelligence*. 2024. T. 137. Ст. 109143. DOI: 10.1016/j.engappai.2024.109143.

142. Siegmund N., Grebhahn A., Apel S., Kästner C. Performance-influence models for highly configurable systems. *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering (ESEC/FSE 2015)*. New York : ACM, 2015. С. 284–294. DOI: 10.1145/2786805.2786845.

143. Sluijterman L., Cator E., Heskes T. How to evaluate uncertainty estimates in machine learning for regression? *Neural Networks*. 2024. T. 173. Ст. 106203. DOI: 10.1016/j.neunet.2024.106203.

144. Sonnenreich W., Albanese J., Stout B. Return on security investment (ROSI): a practical quantitative model. *Journal of Research and Practice in Information Technology*. 2006. Т. 38, № 1. С. 45–56.

145. Stein M. L. *Interpolation of spatial data: some theory for kriging*. New York : Springer, 1999. 247 с.

146. Steinmann P., Tobi H., van Voorn G. A. K. Resilience metrics for socio-ecological and socio-technical systems: a scoping review. *Systems*. 2024. Т. 12, № 9. Ст. 357. DOI: 10.3390/systems12090357.

147. Sun R., Zhu Y., Fei J., Chen X. A survey on moving target defense: intelligently affordable, optimized and self-adaptive. *Applied Sciences*. 2023. Т. 13, № 9. Ст. 5367. DOI: 10.3390/app13095367.

148. Tanenbaum A. S., Van Steen M. *Distributed systems*. 4-те вид. London : Pearson, 2023. 736 с.

149. Tao F., Zhang H., Liu A., Nee A. Y. C. Digital twin in industry: state-of-the-art. *IEEE Transactions on Industrial Informatics*. 2019. Т. 15, № 4. С. 2405–2415. DOI: 10.1109/TII.2018.2873186.

150. Tharwat A., Schenck W. A survey on active learning: state-of-the-art, practical challenges and research directions. *Mathematics*. 2023. Т. 11, № 4. Ст. 820. DOI: 10.3390/math11040820.

151. Trivedi K. S., Bobbio A. *Reliability and availability engineering: modeling, analysis, and applications*. Cambridge : Cambridge University Press, 2017. 730 с.

152. Vano R., Lacalle I., Sowinski P., S-Julian R., Palau C. E. Cloud-native workload orchestration at the edge: a deployment review and future directions. *Sensors*. 2023. Т. 23, № 4. Ст. 2215. DOI: 10.3390/s23042215.

153. Virtanen P., Gommers R., Oliphant T. E. та ін. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*. 2020. Т. 17, № 3. С. 261–272. DOI: 10.1038/s41592-019-0686-2.

154. Wang X., Jin Y., Schmitt S., Olhofer M. Recent advances in Bayesian optimization. *ACM Computing Surveys*. 2023. Т. 55, № 13s. С. 1–36. DOI: 10.1145/3582078.

155. Wang Y., Sun T., Li S., Yuan X., Ni W., Hossain E., Poor H. V. Adversarial attacks and defenses in machine learning-empowered communication systems and networks: a contemporary survey. *IEEE Communications Surveys & Tutorials*. 2023. T. 25, № 4. С. 2245–2298. DOI: 10.1109/COMST.2023.3319492.

156. Wong T., Wagner M., Treude C. Self-adaptive systems: a systematic literature review across categories and domains. *Information and Software Technology*. 2022. T. 148. Ст. 106934. DOI: 10.1016/j.infsof.2022.106934.

157. Yevseiev S., Ponomarenko V., Laptiev O., Milov O. та ін. Synergy of building cybersecurity systems : monograph. Kharkiv : PC Technology Center, 2021. 188 с. DOI: 10.15587/978-617-7319-31-2.

158. Zahynei A. та ін. Method for calculating the residual resource of fog node elements of distributed information systems of critical infrastructure facilities. *Cybersecurity Providing in Information and Telecommunication Systems. CEUR Workshop Proceedings*. 2024. T. 3654. С. 432–439.

159. Zehra S., Faseeha U., Syed H. J., Samad F., Ibrahim A. O., Abulfaraj A. W., Nagmeldin W. Machine learning-based anomaly detection in NFV: a comprehensive survey. *Sensors*. 2023. T. 23, № 11. Ст. 5340. DOI: 10.3390/s23115340.

160. Zhang L., Jia T., Jia M., Wu Y., Liu A., Yang Y., Wu Z., Hu X., Yu P., Li Y. A survey of AIOps in the era of large language models. *ACM Computing Surveys*. 2025. T. 58, № 2. С. 1–35. DOI: 10.1145/3746635.

161. Zhong W., Yang C., Liang W., Cai J., Chen L., Liao J., Xiong N. Byzantine fault-tolerant consensus algorithms: a survey. *Electronics*. 2023. T. 12, № 18. Ст. 3801. DOI: 10.3390/electronics12183801.

162. Жебка В. В., Читулян В. О. Модель на основі гаусівської регресії для прогнозування продуктивності корпоративних веб-систем за конфігурованих умов навантаження. *Телекомунікаційні та інформаційні технології*. 2026. № 2 (91). С. 215–222. DOI: 10.31673/2412-4338.2026.029118.

2. Ананченко О. Є., Миронюк П. Я., Нестеренко К. С., Читулян В. О. Розробка адаптивної корпоративної освітньої інформаційної системи з

використанням методів машинного навчання. Телекомунікаційні та інформаційні технології. 2025. № 4 (89). С. 189–195. DOI: 10.31673/2412-4338.2025.048922.

3. Гангало І. М., Читулян В. О. Кіберфізичні системи екологічного моніторингу: концептуальні засади захисту критичної інфраструктури в умовах війни. Кібербезпека: освіта, наука, техніка. 2025. № 4 (28). С. 688–700. DOI: 10.28925/2663-4023.2025.28.843.

4. Читулян В. О., Олейніков І. А., Ананченко О. Є., Дзюба В. В. Web-програмування і мобільна розробка: UX/UI-рішення для захисту персональних даних користувачів. Кібербезпека: освіта, наука, техніка. 2025. Спецвипуск № 2 (30). С. 543–554. DOI: 10.28925/2663-4023.2025.30.992.

5. Читулян В. О., Поперешняк С. В. Інформаційні технології автоматизованого проєктування обчислювальних систем для корпоративних структур: сучасний стан та перспективи. Кібербезпека: освіта, наука, техніка. 2026. Т. 2, № 34. С. 831–846. DOI: 10.28925/2663-4023.2026.34.

6. Hanhalo I., Chytulian V., Zhebka V., Bebeshko B., Khorolska K. Adaptive approach to ensuring the functional stability of corporate educational platforms under dynamic cyber threats. Cybersecurity Providing in Information and Telecommunication Systems (CPITS 2025) : proc. of the Workshop. CEUR Workshop Proceedings. 2025. Vol. 3991. P. 481–491.

7. Chytulian V., Kolodiuk A., Oleinikov I., Zhebka V., Balatska V. Formation of a conceptual model for cyber-physical monitoring of critical infrastructure environmental objects. Cybersecurity Providing in Information and Telecommunication Systems II (CPITS-II 2025) : proc. of the Workshop. CEUR Workshop Proceedings. 2025. Vol. 4145. P. 330–341.

Додаток А. Акти впровадження

ЗАТВЕРДЖУЮ

Перший проректор Державного
університету
інформаційно-комунікаційних
технологій

Корченко О. Г.
“13” 2026 року



А К Т

впровадження в навчальний процес Державного університету інформаційно-комунікаційних технологій наукових положень і результатів дисертаційної роботи Читуляна Вадима Олеговича на тему «Інформаційні технології автоматизованого проектування обчислювальних систем для корпоративних структур»

Науково-педагогічна комісія у складі голови – директора навчально-наукового інституту Інформаційних технологій доктора технічних наук, професора Нестеренко К.С. та членів комісії: завідувача кафедри Технологій цифрового розвитку доктора технічних наук, професора Жебки В.В., завідувача кафедри Інформаційних систем та технологій доктора технічних наук, професора Сторчак К.П. склала даний акт про те, що наукові положення і результати дисертаційної роботи на здобуття наукового ступеня доктора філософії Читуляна В. О. на тему «Інноваційні технології автоматизованого проектування обчислювальних систем для корпоративних структур», а саме:

1. Метод автоматизованого проектування на основі сурогатної моделі гаусівської процесової регресії, який апроксимує нелінійну поверхню відгуку продуктивності та надає відкалібровані довірчі інтервали, введено в навчальний процес кафедри Комп'ютерної інженерії з дисципліни «Технології проектування

комп'ютерних систем» для студентів спеціальності F7 «Комп'ютерна інженерія» денної форми навчання.

2. Інформаційну технологію забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи, яка інтегрує моніторинг аномалій, прогнозування, оптимізацію потоків та оцінювання ризиків у єдиний контур керування, введено в навчальний процес кафедри Інформаційних систем та технологій з дисципліни «Архітектура інформаційних систем» для студентів спеціальності F6 «Інформаційні системи та технології» денної форми навчання.

3. Метод структурно-параметричного проєктування відмовостійких і захищених розподілених систем на основі багатокритеріальної оптимізації розподілу ресурсів безпеки з урахуванням живучості та плавної деградації введено в навчальний процес кафедри Комп'ютерної інженерії з дисципліни «Паралельні та розподілені обчислення» для студентів спеціальності F7 «Комп'ютерна інженерія» денної форми навчання.

Голова комісії:

Директор навчально-наукового інституту Інформаційних технологій

доктор технічних наук, професор



Нестеренко К.С.

Члени комісії:

Завідувач кафедри Технологій цифрового розвитку

доктор технічних наук, професор



Жебка В.В.

доцент кафедри Технологій цифрового розвитку

кандидат технічних наук



Аронов А.О.

ЗАТВЕРДЖУЮ

Директор Інституту програмних систем
Національної академії наук України



І.П. Сініцин
2026 р.

АКТ

впровадження наукових результатів дисертації
аспіранта Державного університету інформаційно-комунікаційних технологій
ЧИТУЛЯНА Вадима Олеговича

Комісія у складі:

Голова: завідувач відділу організації та супроводження наукових досліджень Чистяков В.А., к.т.н.

члени комісії:

завідувач відділу Федоренко Р.М., к.е.н., ст.досл.

заступник завідувача відділу Ігнатенко П.П., к.т.н., с.н.с.

розглянула матеріали дисертаційного дослідження аспіранта Читуляна В.О. на тему «Інформаційні технології автоматизованого проектування обчислювальних систем для корпоративних структур».

Комісією підтверджено, що реалізація поставленого наукового завдання сприяє підвищенню ефективності автоматизованого проектування обчислювальних систем для корпоративних структур за одночасного врахування продуктивності, функціональної стійкості, відмовостійкості та захищеності, що досягається завдяки розробленому комплексу методів і інформаційної технології автоматизованого проектування.

У рамках дослідження розроблено метод автоматизованого проектування обчислювальних систем на основі сурогатної моделі гаусівської процесової регресії, який, на відміну від адитивних моделей впливу, апроксимує нелінійну поверхню відгуку у просторі «конфігурація – навантаження» та надає відкалібровані довірчі інтервали. Практичне застосування методу дозволяє скоротити обсяг натурних вимірювань на 89,6% за збереження точності прогнозу, що суттєво знижує витрати на конфігурування високонавантажених корпоративних систем.

Для підтримання безперервної роботи інформаційних систем застосовано удосконалену інформаційну технологію забезпечення функціональної стійкості адаптивної корпоративної інформаційної системи, яка, на відміну від відомих реактивних рішень, інтегрує статистичний моніторинг аномалій, прогнозування поведінки методами машинного навчання, оптимізацію інформаційних потоків та оцінювання ризиків у єдиний контур керування. Використання технології забезпечує адаптацію системи в реальному часі та підтримання критичної функціональності за збурень, підвищує індекс функціональної стійкості з 0,917 до 0,996 і повністю усуває недообслуговування інформаційних потоків під час відмов.

Для обґрунтування архітектури обчислювальної інфраструктури застосовано розвинений метод структурно-параметричного проєктування відмовостійких і захищених розподілених обчислювальних систем корпоративних структур на основі багатокритеріальної оптимізації розподілу ресурсів безпеки з урахуванням функцій живучості, адаптивності та плавної деградації, який дозволяє на етапі проєктування обирати архітектуру, стійку до втрати значної частки компонентів і до кіберзагроз. Застосування методу забезпечує підвищення допустимої частки втрати компонентів із 40% до 70% за умови плавної деградації та зниження інтегрального ризику системи на 59%.

Практичне значення отриманих результатів полягає в тому, що впровадження розроблених у дисертаційному дослідженні методів та інформаційної технології дозволяє проєктувати, модернізувати та експлуатувати обчислювальні системи корпоративних структур з урахуванням вимог до продуктивності, функціональної стійкості, відмовостійкості та захищеності.

Отримані результати використано для обґрунтування проєктних рішень щодо конфігурування, забезпечення функціональної стійкості та захищеності обчислювальної інфраструктури інформаційно-аналітичних систем установи.

Голова комісії

Члени комісії



Валерій ЧИСТЯКОВ

Руслан ФЕДОРЕНКО

Петро ІГНАТЕНКО

АКТ

про впровадження результатів дисертаційної роботи

«10» 02 2026 р.

м. Київ

Результати дисертаційної роботи Читуляна Вадима Олеговича на тему «Інформаційні технології автоматизованого проектування обчислювальних систем для корпоративних структур» були використані в діяльності компанії Byte Orchard Consulting у процесі дослідження та розроблення програмних рішень для автоматизованого проектування, аналізу та забезпечення функціонування обчислювальних систем корпоративних структур. У межах використання результатів дослідження застосовано метод автоматизованого проектування на основі сурогатної моделі гаусівської процесової регресії, інформаційну технологію забезпечення функціональної стійкості адаптивних корпоративних інформаційних систем, а також метод структурно-параметричного проектування відмовостійких і захищених розподілених обчислювальних систем.

Практичне використання результатів дисертаційного дослідження дозволило підвищити ефективність автоматизованого проектування обчислювальних систем за рахунок прогнозування їх продуктивності, оцінювання невизначеності та оптимізації проектних рішень. Застосування запропонованої сурогатної моделі дозволяє скоротити обсяг натурних вимірювань на 89,6%, а використання інформаційної технології забезпечення функціональної стійкості сприяє підвищенню індексу функціональної стійкості з 0,917 до 0,996 та усуненню недообслуговування інформаційних потоків під час відмов. Запропонований метод структурно-параметричного проектування забезпечує можливість врахування відмовостійкості, захищеності, живучості та плавної деградації під час вибору архітектурних рішень.

Отримані результати мають практичне значення для проектування та розвитку сучасних корпоративних обчислювальних систем, інформаційно-аналітичних систем і систем підтримки прийняття рішень.

Запропоновані методи та інформаційні технології можуть бути використані для автоматизованого проєктування обчислювальних систем, прогнозування їх продуктивності, забезпечення функціональної стійкості, підвищення відмовостійкості та захищеності розподілених систем, а також для подальшого розвитку програмних рішень у сфері інформаційних технологій.

Директор
ТОВ «Байт Орчард Консалтінг»



В. В. Рудич

Додаток Б. Програмне забезпечення експериментальної верифікації тріади

Експериментальну верифікацію трьох наукових результатів реалізовано у вигляді відтворюваних програмних стендів мовою Python. Кожному результату відповідає окремий каталог із генератором вихідних даних, наскрізним обчислювальним конвеєром, файлом підсумкових метрик у форматі JSON та супровідним описом. Усі експерименти є детермінованими: початкове значення генератора псевдовипадкових чисел зафіксовано (SEED = 42), тому повторний запуск відтворює наведені у розділі 5 числові результати без відхилень.

Науковий інструментарій є переважно спільним для трьох стендів і охоплює бібліотеки чисельних обчислень, аналізу, машинного навчання, роботи з графами та побудови графіки.

Конвеєр кожного стенду виконує єдину послідовність кроків: генерування фізично обґрунтованого синтетичного набору даних, застосування запропонованого методу та коректно визначеної базової альтернативи на тих самих даних, обчислення метрик якості й побудову рисунків розділу 5.

Запуск стендів результатів № 1 і № 2 виконується послідовним викликом генератора даних і конвеєра штатним інтерпретатором Python. Стенд результату № 3 запускається з попереднім установленням кодування виводу у UTF-8, оскільки у службовому виводі використовуються символи сумування.

Детальний опис методики калібрування генераторів, повний перелік проміжних результатів та історію налаштування стендів наведено у супровідних файлах Історія_і_результати_експерименту.md кожного каталогу, а призначену для написання розділу 5 зведену форму – у файлах Протокол_експерименту_для_Розділу_5.md результатів № 2 і № 3. Наведений склад програмного забезпечення забезпечує повну відтворюваність числових результатів, винесених на захист.

Цифровий двійник методу структурно-параметричного проєктування реалізовано окремим сценарієм стенду результату № 3, який без зміни стендів результатів № 1 і № 2 повторно використовує їхні функції: сурогатну модель і вибір конфігурації вузла за задачею (2.30) з натурною перевіркою, а також моделювання траєкторії якості функціонування з перемаршрутизацією інформаційних потоків; результати зберігаються у файлі підсумкових метрик стенду результату № 3. Сценарії експорту повторно викликають функції стендів результатів № 2 і № 3 без їх зміни й передають отримані дані сторінкам панелей, реалізованим мовою JavaScript із візуалізацією у форматі SVG, тож значення на панелях збігаються з результатами розділу 5.

Конструювання ознак на основі фізичної моделі системи та побудову сурогатної моделі гаусівської процесової регресії з ядром Матерна і метрики відкаліброваності реалізовано згідно з лістингом Б.1.

Лістинг Б.1. Побудова сурогатної моделі та метрики відкаліброваності (результат № 1)

```
def add_features(df):
    df = df.copy()
    df["capacity_score"] = df["cpu_cores"] * df["workers"] * np.log2(df["pool_size"])
    df["load_intensity"] = df["vus"] / df["capacity_score"]
    df["pool_contention"] = df["vus"] / df["pool_size"]
    df["ram_per_worker"] = df["ram_gb"] / df["workers"]
    return df
```

```

def make_gpr(kind):
    lsb = (1e-1, 1e2) # межі масштабу довжини (ізотропне ядро)
    nlb = (1e-5, 1e1) # межі рівня шуму (nugget)
    if kind == "Matern":
        k = C(1.0, (1e-2, 1e2)) * Matern(length_scale=1.0, length_scale_bounds=lsb, nu=2.5) \
            + WhiteKernel(1e-2, noise_level_bounds=nlb)
    return GaussianProcessRegressor(kernel=k, normalize_y=True,
                                     n_restarts_optimizer=3, random_state=SEED)

def ece_score(y_log, mu_log, sigma_log, levels=None):
    """Expected Calibration Error за набором рівнів довіри."""
    if levels is None:
        levels = np.linspace(0.05, 0.99, 20)
    errs = []
    for a in levels:
        z = norm.ppf(0.5 + a / 2.0)
        lo, hi = mu_log - z * sigma_log, mu_log + z * sigma_log
        emp = np.mean((y_log >= lo) & (y_log <= hi))
        errs.append(abs(emp - a))
    return float(np.mean(errs))

def coverage(y_log, mu_log, sigma_log, a=0.95):
    z = norm.ppf(0.5 + a / 2.0)
    return float(np.mean(np.abs(y_log - mu_log) <= z * sigma_log))

```

Статистичний детектор аномалій на основі сезонного контрольного графіка з правилом серій та оптимальну маршрутизацію інформаційних потоків методом максимального потоку реалізовано згідно з лістингом Б.2.

Лістинг Б.2. Виявлення аномалій та оптимізація потоків (результат № 2)

```

def seasonal_baseline(latency):
    """Робастний добовий профіль (медіана за слотом доби) і залишок."""
    n = len(latency); slot = np.arange(n) % SPD
    expected = np.zeros(n); resid_std = np.zeros(n)
    for s in range(SPD):
        vals = latency[slot == s]
        med = np.median(vals)
        mad = np.median(np.abs(vals - med)) + 1e-9
        expected[slot == s] = med
        resid_std[slot == s] = 1.4826 * mad # MAD -> sigma
    return expected, latency - expected, resid_std

def detect_anomalies_3sigma(latency, k=3.0):
    """Сезонний контрольний графік з межею k·sigma і правилом серій:
    сигнал підтверджується щонайменше двома послідовними перевищеннями."""
    expected, resid, resid_std = seasonal_baseline(latency)
    raw = np.abs(resid) > k * resid_std
    prev = np.concatenate([[False], raw[:-1]])
    nxt = np.concatenate([raw[1:], [False]])
    flags = ((raw & prev) | (raw & nxt)).astype(int) # серія довжини >= 2

```

```
def served_maxflow(demand, node_caps):
    """Оптимальна маршрутизація: попит перерозподіляється на справні вузли."""
    G = nx.DiGraph()
    for i, c in enumerate(node_caps):
        G.add_edge("s", f"n{i}", capacity=float(demand))
        G.add_edge(f"n{i}", "t", capacity=float(c))
    val, _ = nx.maximum_flow(G, "s", "t")
    return float(min(val, demand))
```

```
dz = security_index(x, k_threat, R_eff) - z0
x[i, j] = 0
ratio = dz / gen.C_MECH[j]
if ratio > best_ratio + 1e-12:
    best, best_ratio = (i, j), ratio
if best is None:
    break
x[best] = 1
path.append((alloc_cost(x), security_index(x, k_threat, R_eff)))
return x, path
```

Додаток В. Специфікація синтетичних наборів даних і калібрувальних моделей

Вихідними даними для верифікації є фізично обґрунтовані синтетичні набори, побудовані так, щоб відтворювати ключові нелінійні залежності відповідної предметної області за керованої, наперед відомої складності. Такий підхід коректний для верифікації методів, оскільки перевіряється здатність кожного методу розв'язати задачу за відомої поверхні відгуку, структури збурень чи відмов, а всі порівняння виконуються за єдиних відтворюваних умов. Нижче наведено специфікацію набору даних і калібрувальної моделі для кожного результату тріади.

Набір даних результату № 1 моделює продуктивність корпоративної веб-системи у просторі «конфігурація – навантаження». Основою є модель системи масового обслуговування з обмеженою кількістю обслуговуючих приладів, у якій час відгуку зростає нелінійно з підвищенням коефіцієнта завантаження. Час відгуку обчислюється як $resp = s_{eff}(1 + 18\rho^{1.8})$, де ρ – коефіцієнт завантаження, а інтенсивність надходження запитів пропорційна кількості віртуальних користувачів. Верхню межу часу відгуку обмежено таймаутом 2000 мс, а на детермінований відгук накладено логнормальний шум із параметром $\sigma = 0,05$. Простір конфігурацій утворює регулярну сітку з 720 варіантів. Сурогатною моделлю є гаусівська процесова регресія з ізотропним ядром Матерна з параметром гладкості 5/2; ізотропне ядро з єдиним масштабом довжини обрано як критичну умову стійкості моделі на малих вибірках порівняно з ядром з окремими масштабами за ознаками.

Набір даних результату № 2 моделює роботу адаптивної корпоративної інформаційної системи під змінним навантаженням із збуреннями. Горизонт моделювання становить 21 добу з кроком 30 хвилин, що дає 1008 відліків. Попит формується як сума добової й тижневої сезонності, лінійного тренду зростання близько 18 % та випадкового шуму з параметром $\sigma = 0,035$. На горизонт накладено 10 збурень трьох типів – три сплески навантаження, чотири кіберінциденти та три відмови компонентів, – що разом утворюють 57 аномальних відліків. Час відгуку системи задано як $lat = 60(1 + 14 util^{2.2})(1 + 1,2 threat)$, де $util$ – використання ресурсу, $threat$ – рівень загрози, ξ – мультиплікативний шум. Порівнюються базова реактивна конфігурація з часом реагування 12 годин та інтегрована технологія з часом реагування 1 година.

Набір даних результату № 3 моделює структуру розподіленої корпоративної обчислювальної системи з п'яти класів компонентів – крайових, транспортних, обчислювальних, компонентів сховища та компонентів безпеки – з нормованими вагами вкладу. Захист описано чотирма механізмами: постквантовою криптографією, архітектурою

нульової довіри, адаптивним машинним навчанням і структурним резервуванням. Ефект захисту є мультиплікативним: залишкова вразливість дорівнює добутку залишкових вразливостей окремих механізмів, що унеможлиблює нефізичне занулення ризику. Множник підвищеної загрози дорівнює 1,8. Структурне резервування реалізовано за схемою «к з п»: базова конфігурація містить 8 екземплярів, відмовостійка – 16. Параметри наборів даних і калібрувальних моделей зведено в таблиці В.1.

Таблиця В.1

Параметри синтетичних наборів даних і калібрувальних моделей

Параметр	Результат № 1	Результат № 2	Результат № 3
Обсяг / розмірність	сітка 720 конфігурацій	1008 відліків (21 доба × 30 хв)	5 класів компонентів
Джерело нелінійності	час відгуку черги, $1 + 18\rho^{1,8}$	час відгуку, $1 + 14, util^{2,2}$	мультиплікативний ефект захисту
Збурення	логнормальний шум $\sigma = 0,05$	10 збурень, 57 аномальних відліків	множник загрози 1,8
Базова альтернатива	адитивна модель впливу	реактивна конфігурація (12 год)	базове резервування (8 екземплярів)
Запропонований підхід	гаусівська процесова регресія (Матерн 5/2)	інтегрована технологія (1 год)	відмовостійка основа (16 екземплярів)
SEED	42	42	42

Нижче наведено програмні реалізації калібрувальних моделей вихідних даних, які повністю визначають наведені у таблиці В.1 залежності. Кожна модель є детермінованою за фіксованого початкового значення генератора.

Фізичну модель часу відгуку корпоративної веб-системи на основі системи масового обслуговування з обмеженою кількістю каналів та межі простору параметрів наведено в лістингу В.1.

Лістинг В.1. Модель часу відгуку у просторі «конфігурація – навантаження» (результат № 1)

```

BOUNDS = {
    "ram_gb": (2.0, 16.0),
    "cpu_cores": (1.0, 8.0),
    "pool_size": (10.0, 200.0),
    "workers": (2.0, 16.0),
    "vus": (10.0, 400.0),
    "write_ratio": (0.0, 0.6),
}

def p95_latency(ram_gb, cpu_cores, pool_size, workers, vus, write_ratio):
    """p95-затримка відповіді (мс) за моделлю М/М/с."""
    s_read, s_write = 18.0, 1.4 * 18.0

```

```

s_base = s_read * (1.0 - write_ratio) + s_write * write_ratio
ram_factor = 1.0 + 0.9 * np.exp(-ram_gb / 4.0)    # штраф за брак пам'яті
s_eff = s_base * ram_factor
mu = 1000.0 / s_eff                                # req/s на канал
c_raw = cpu_cores * np.sqrt(workers)              # CPU-обмежена конкурентність
c_eff = int(max(1, round(min(c_raw, pool_size / 2.0))))
lam = vus * 0.5                                    # інтенсивність надходження
rho = lam / (c_eff * mu)                           # коефіцієнт завантаженості
resp_p95 = s_eff * (1.0 + 18.0 * rho ** 1.8)       # ключова нелінійність
return float(np.clip(resp_p95, 0.0, 2000.0))       # обмеження таймаутом 2 с

```

Профіль навантаження з добовою і тижневою сезонністю та трендом, генерування збурень трьох типів і модель показника якості обслуговування наведено в лістингу В.2.

Лістинг В.2. Модель телеметрії адаптивної корпоративної системи (результат № 2)

```

def _seasonal_demand(t_idx):
    """Профіль навантаження: добова + тижнева сезонність + тренд +18%. """
    day_phase = 2.0 * np.pi * (t_idx % STEPS_PER_DAY) / STEPS_PER_DAY
    daily = np.clip(0.55 + 0.45 * np.clip(np.sin(day_phase - np.pi / 2.2), -1, 1), 0.12, 1.0)
    dow = (t_idx // STEPS_PER_DAY) % 7
    weekly = np.where(dow >= 5, 0.55, 1.0)          # нижча активність у вихідні
    trend = 1.0 + 0.18 * (t_idx / N)
    return 150.0 + (300.0 - 150.0) * daily * weekly * trend

def _make_events(rng):
    """Три сплески, чотири кіберінциденти, три відмови компонентів. """
    events = []
    for etype, count in [("surge", 3), ("attack", 4), ("failure", 3)]:
        for _ in range(count):
            start = int(rng.integers(STEPS_PER_DAY, N - STEPS_PER_DAY))
            if etype == "surge":
                dur, mag = int(rng.integers(3, 8)), float(rng.uniform(1.6, 2.6))
            elif etype == "attack":
                dur, mag = int(rng.integers(4, 12)), float(rng.uniform(0.5, 0.95))
            else:
                dur, mag = int(rng.integers(3, 9)), float(rng.uniform(0.35, 0.6))
            events.append({"type": etype, "start": start, "dur": dur, "mag": mag})
    return sorted(events, key=lambda e: e["start"])

# Показник якості: латентність як нелінійна функція завантаженості й загрози
util = demand / cap_eff
latency = BASE_LATENCY * (1.0 + 14.0 * np.clip(util, 0, 0.98) ** 2.2)

```

latency *= (1.0 + 1.2 * threat) # деградація від кіберінциденту

latency *= rng.lognormal(mean=0.0, sigma=0.04, size=N) # шум вимірювань

Структуру розподіленої корпоративної обчислювальної системи задано класами компонентів (таблиця В.2), механізмами захисту (таблиця В.3) та кандидатними архітектурами для вибору на етапі проєктування (таблиця В.4).

Таблиця В.2

Класи компонентів розподіленої системи та параметри резервування

Клас компонента	Вага w	Надійність R0	Базова ймовірність атаки P0	Збиток I	Резервування база (k-3-n)	Резервування відмовостійка (k-3-n)
Крайові вузли доступу	0,20	0,90	0,55	0,6	1 з 2	1 з 4
Транспорт даних	0,20	0,93	0,45	0,8	1 з 1	1 з 3
Обчислювальні вузли	0,25	0,92	0,50	0,9	2 з 2	2 з 4
Сховище даних	0,15	0,95	0,40	1,0	2 з 2	2 з 3
Підсистема безпеки	0,20	0,90	0,60	0,7	1 з 1	1 з 2

Таблиця В.3

Механізми захисту та їхні параметри

Механізм захисту	Ефективність e	Вартість	Час упровадження
Постквантова криптографія	0,35	0,28	0,30
Архітектура нульової довіри	0,30	0,22	0,25
Адаптивне ML-виявлення загроз	0,40	0,30	0,35
Резервування та самовідновлення	0,25	0,35	0,40

Таблиця В.4

Кандидатні архітектури для вибору на етапі проєктування

Архітектура	Резервування	Частка бюджету безпеки	Сукупна вартість (відносно базової)
Централізована базова	немає	0,10	1,00
Резервована	є	0,30	1,35
Захищена розподілена (метод)	є	0,85	1,70
Надлишкова	є	1,00	2,30

Множник підвищеної загрози дорівнює 1,0 у штатному фоновому режимі та 1,8 в умовах загострення, а загальний бюджет безпеки нормовано як максимально можливу суму вартостей застосування всіх механізмів до всіх компонентів. Наведені моделі й параметри повністю визначають вихідні дані верифікації результатів тріади.

Додаток Г. Панелі підтримки рішень інформаційних технологій

Для подання результатів роботи двох інформаційних технологій розроблено панелі підтримки рішень: рядок змінних, що задають умови аналізу, тематичні рядки панелей, панелі ключових показників, часові ряди з позначками подій і таблиці з кольоровим кодуванням значень.

Панель моніторингу функціональної стійкості призначена для служби експлуатації корпоративної інформаційної системи. Її перший фрагмент (рисунок Г.1) містить рядок змінних, зведені показники – індекс функціональної стійкості, значення цільової функції, кількість виявлених збурень, похибку прогнозу навантаження, частку необслугованого попиту під час відмов і залишковий ризик – та часовий ряд показника якості функціонування для інтегрованої технології й базової конфігурації з позначками збурень.

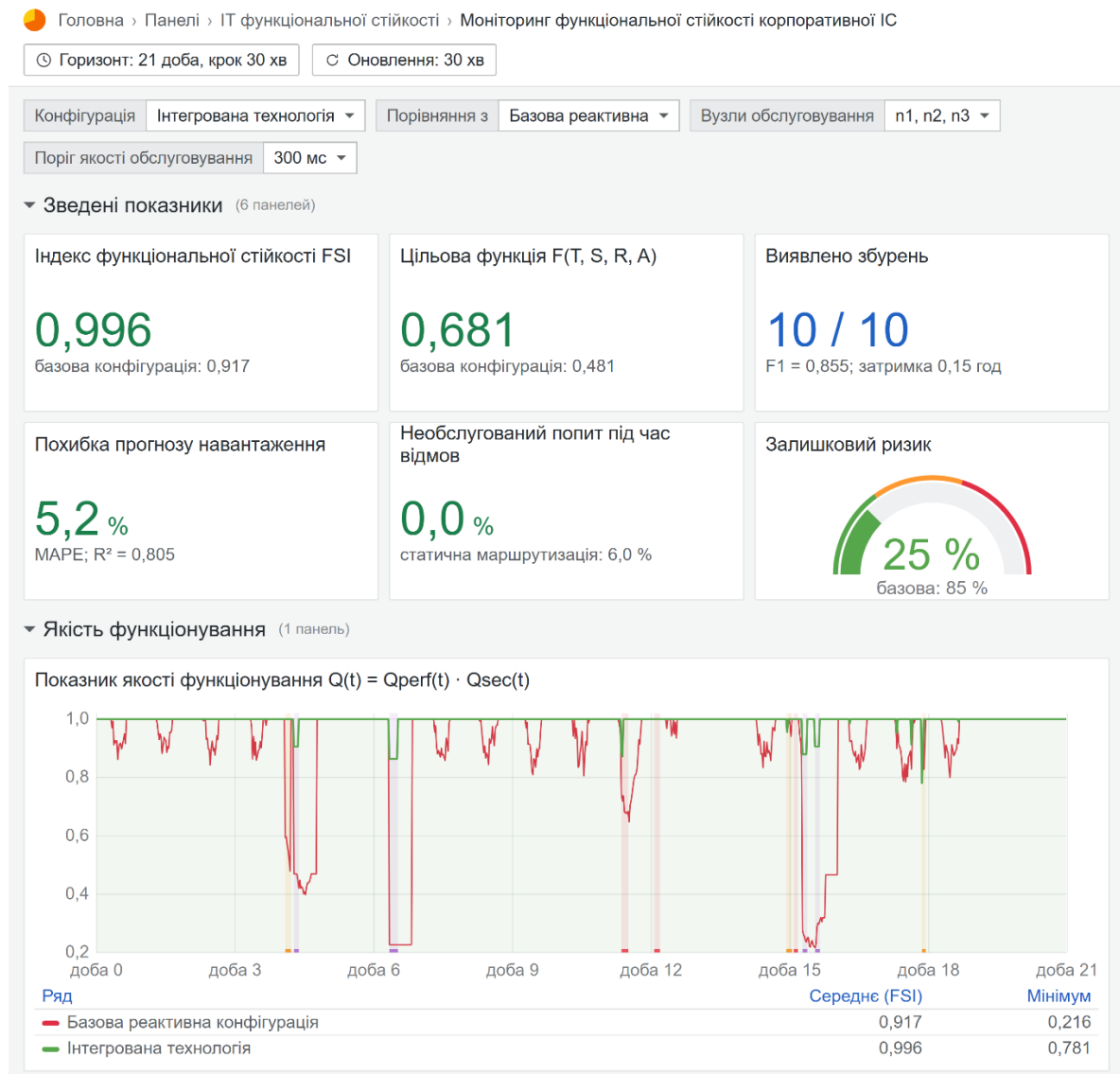


Рис. Г.1. Панель моніторингу функціональної стійкості: зведені показники та якість функціонування

Другий фрагмент панелі (рисунок Г.2) подає контрольний графік затримки відповіді з межею 3σ і виявленими аномаліями та прогноз навантаження з інтервалом, а третій (рисунок Г.3) – складові цільової функції порівняно з базовою конфігурацією та журнал збурень із часом виявлення й реакцією технології.

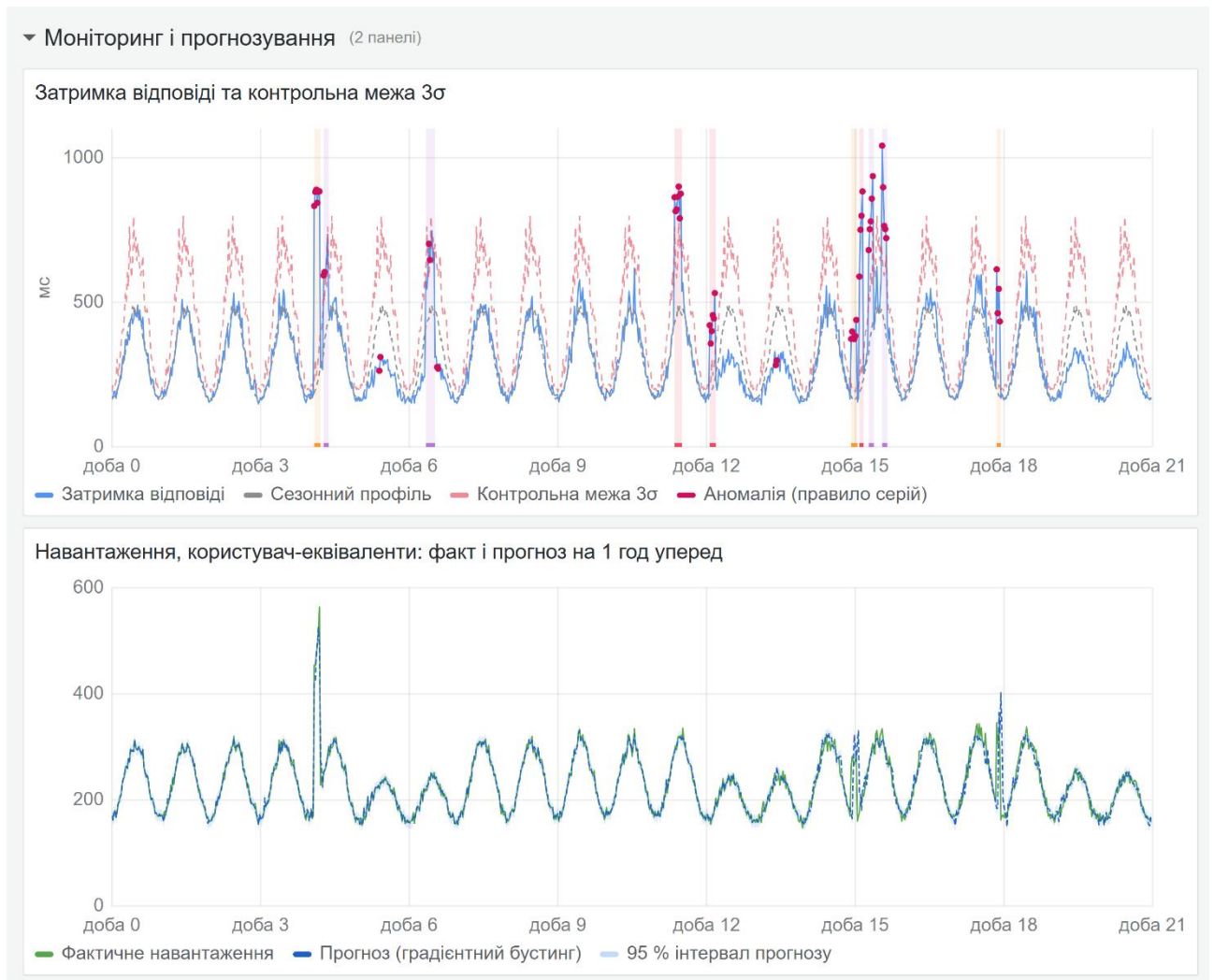


Рис. Г.2. Панель моніторингу функціональної стійкості: моніторинг і прогнозування

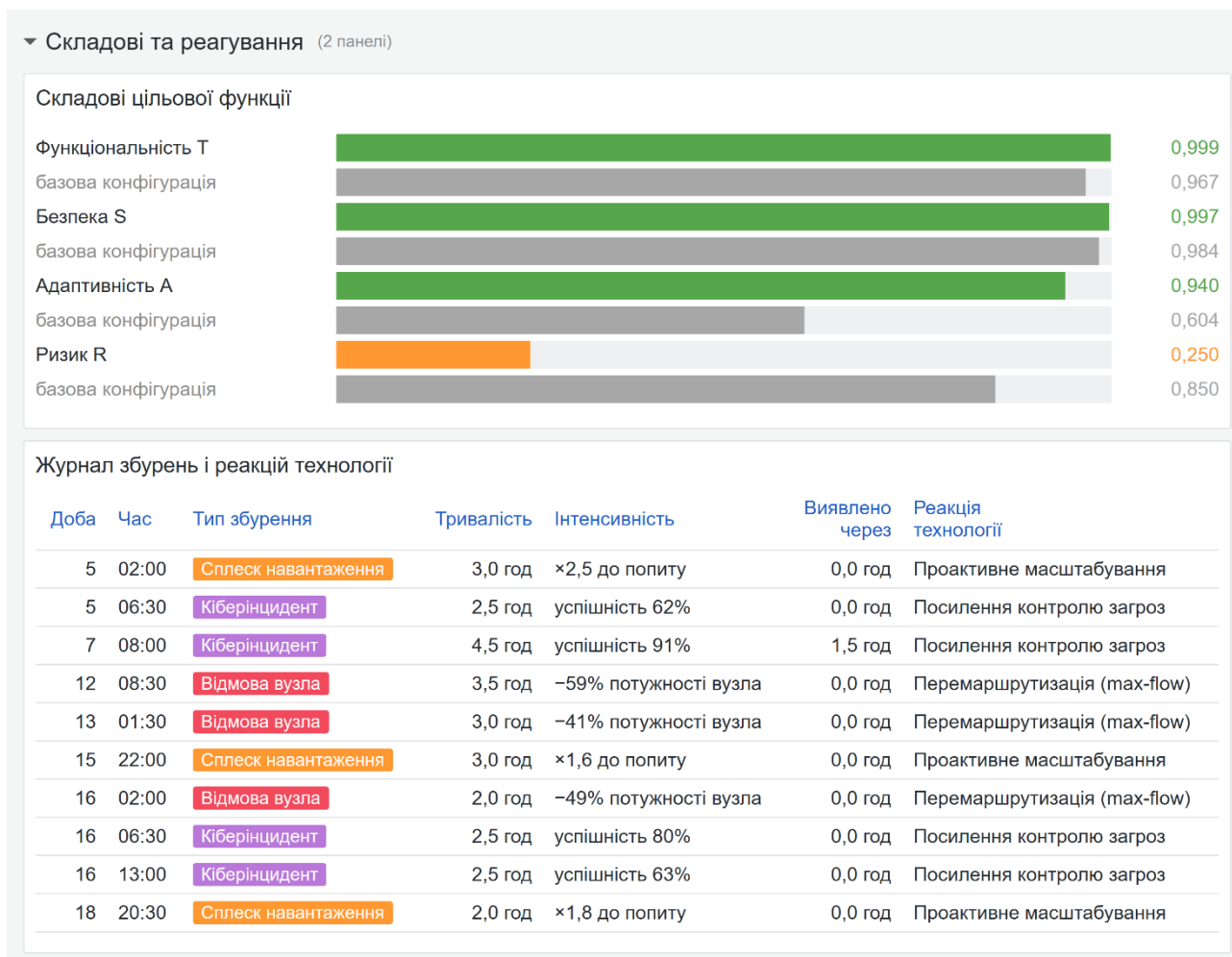


Рис. Г.3. Панель моніторингу функціональної стійкості: складові цільової функції та журнал збурень

Панель вибору архітектури є вихідною формою модуля багатокритеріального вибору інформаційної технології структурно-параметричного проєктування й призначена для проєктувальника та особи, яка приймає рішення. Її перший фрагмент (рисунок Г.4) містить рядок змінних – множник загрози, бюджет безпеки, ваги критеріїв і сценарії збурень, – зведені показники рекомендованої архітектури та таблицю показників кандидатних архітектур із рейтингом за композитним критерієм, а зміна ваг або обмежень у рядку змінних відповідає контуру погодження рішення в інформаційній технології.

Проект: розподілена КОС, варіант 4 кандидати

Сформувати звіт

Множник загрози k 1,8 (загострений)

Бюджет безпеки 0,85 · Bmax

Ваги критеріїв λ Z 0,25 · живучість 0,25 · деградація 0,15 · ризик 0,20 · вартість 0,15

Сценарії Ω відмови, атаки, перевантаження

Рекомендоване рішення (6 панелей)

Рекомендована архітектура

Захищена розподілена

найвищий критерій G серед 4 кандидатів

Композитний критерій G

0,896

наступний: Надлишкова, 0,850

Індекс безпеки Z

0,736

без засобів захисту: 0,103

Інтегральний ризик R

0,722

базова протидія: 1,769 (–59 %)

Допустима втрата компонентів

70 %

нерезервована: 40 %

Відносна вартість

1,70

базова = 1,00

Порівняння кандидатних архітектур (2 панелі)

Показники кандидатів (до нормування)

Архітектура	Резервування к-з-п	Бюджет безпеки	Індекс Z	Ризик R	Допустима втрата	Вартість	Критерій G
Централізована базова	ні (8 екз.)	0,10	0,203	1,596	40 %	1,00	0,150
Резервована	так (16 екз.)	0,30	0,444	1,234	70 %	1,35	0,695
Захищена розподілена рекомендовано	так (16 екз.)	0,85	0,736	0,722	70 %	1,70	0,896
Надлишкова	так (16 екз.)	1,00	0,766	0,676	70 %	2,30	0,850

Рейтинг за композитним критерієм G

Захищена розподілена		0,896
Надлишкова		0,850
Резервована		0,695
Централізована базова		0,150

Рис. Г.4. Панель вибору архітектури: рекомендоване рішення та порівняння кандидатів

Другий фрагмент панелі (рисунок Г.5) подає межу компромісу «захищеність – вартість» із проектною точкою та розподіл засобів захисту за класами компонентів, а третій (рисунок Г.6) – криві живучості й плавної деградації та внесок класів компонентів в інтегральний ризик.

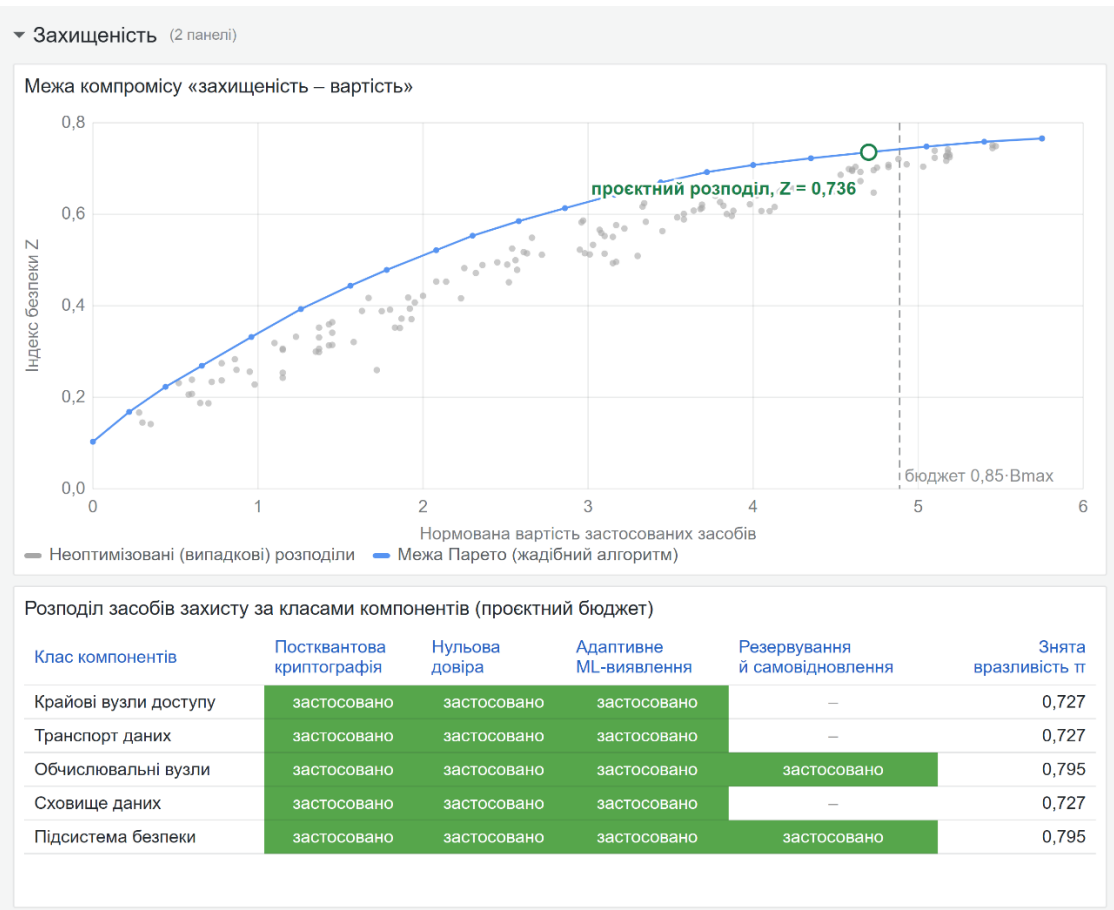


Рис. Г.5. Панель вибору архітектури: захищеність і розподіл засобів захисту

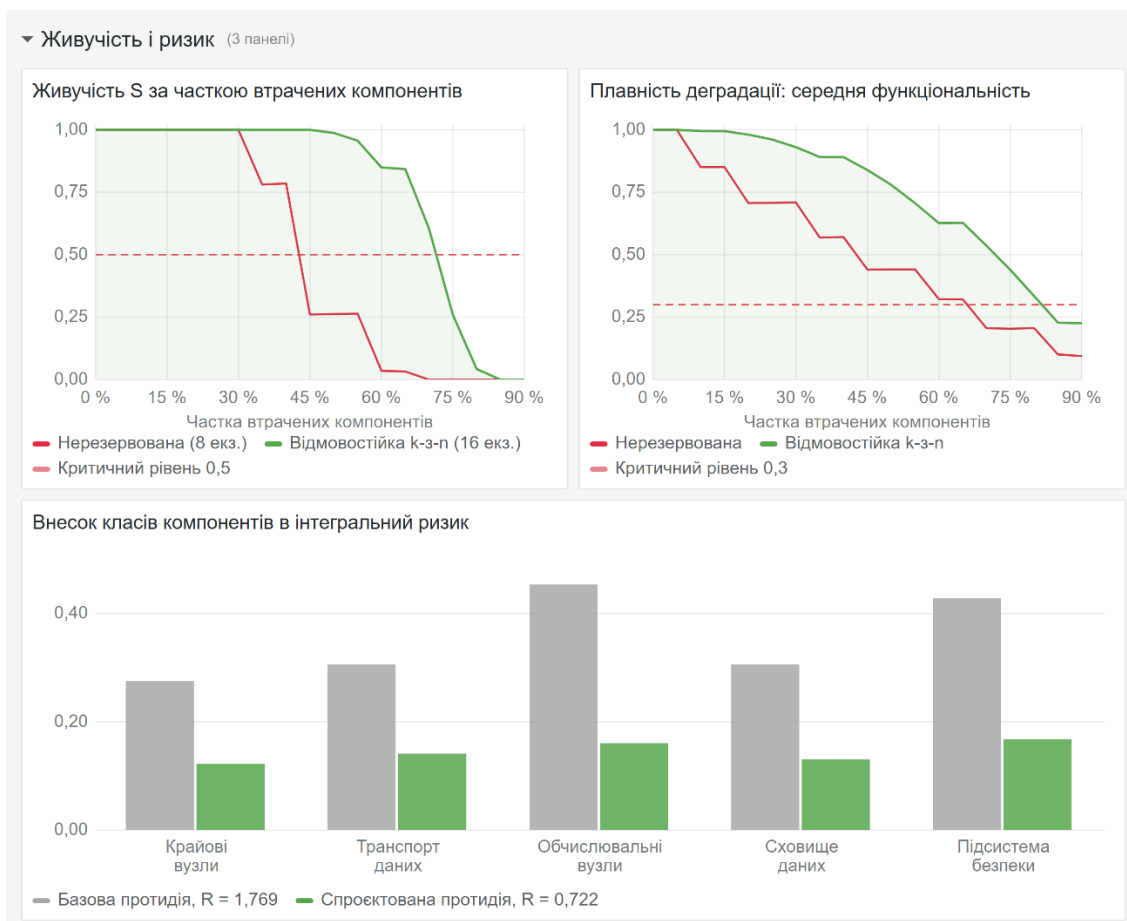


Рис. Г.6. Панель вибору архітектури: живучість, плавна деградація та ризик