

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Кваліфікаційна наукова
праця на правах рукопису

Шиман Анна Павлівна

УДК 004.738.5:004.272.2 (043.3)

ДИСЕРТАЦІЯ
**СИНТЕЗ КЛАСТЕРА ГРАНИЧНОГО ШАРУ КОМП'ЮТЕРНОЇ
СИСТЕМИ ПІДТРИМКИ ІНТЕРНЕТУ РЕЧЕЙ**

Спеціальність 123 – Комп'ютерна інженерія
Галузь знань 12 – Інформаційні технології

Подається на здобуття наукового ступеня доктора філософії

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



А.П. Шиман

Науковий керівник:
Кучук Ніна Георгіївна,
доктор технічних наук, професор

Харків – 2025

АНОТАЦІЯ

Шиман А.П. Синтез кластера граничного шару комп'ютерної системи підтримки Інтернету речей. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії зі спеціальності 123 – Комп'ютерна інженерія. – Національний технічний університет «Харківський політехнічний інститут», Харків, 2025.

Дисертаційна робота присвячена вирішенню актуальної науково-технічної задачі, пов'язаної з підвищенням ефективності використання обчислювальних ресурсів граничного шару при обробці оперативних транзакцій.

Об'єкт дослідження – процес формування кластерів граничного шару, що об'єднують територіально близькі пристрої Інтернету речей.

Предмет дослідження – методи синтезу кластерів граничного шару комп'ютерної системи підтримки Інтернету речей.

Метою дисертаційної роботи є підвищення ефективності використання обчислювальних ресурсів граничного шару при обробці оперативних транзакцій шляхом удосконалення методів синтезу окремих кластерів граничного шару Інтернету речей.

У *вступі* обґрунтовано актуальність підвищення ефективності використання обчислювальних ресурсів граничного шару при обробці оперативних транзакцій. Також представлено зв'язок роботи з науковими програмами, планами і темами; наведено наукову новизну та практичне значення отриманих результатів; надано інформацію щодо особистого внеску здобувача та представлено перелік публікацій за темою дисертації.

У *першому розділі* проведено аналіз сучасного стану розвитку Інтернету речей та хмарної інфраструктури підтримки мережі IoT. Здійснено аналіз існуючих методів дослідження компонент комп'ютерних систем підтримки Інтернету речей. Розглянуто відомі моделі і методи оцінювання ефективності

компонент комп'ютерних систем підтримки режиму реального часу. Сформульовано основну наукову проблему, яка полягає у підвищенні ефективності використання обчислювальних ресурсів граничного шару при обробці оперативних транзакцій шляхом удосконалення методів синтезу окремих кластерів граничного шару Інтернету речей.

Другий розділ присвячено розробці методу побудови функціональних моделей кластера граничного шару, зокрема була визначена послідовність етапів дослідження фрагментів комп'ютерних систем реального часу, розроблено метод оцінки адекватності мережних моделей фрагментів граничного шару.

Третій розділ присвячено розробці методів зменшення часу обробки оперативних транзакцій Інтернету речей. Зокрема, було виявлено особливості часових мереж Петрі, що використовуються при описі мережних моделей комп'ютерних систем підтримки граничного шару мережі IoT. Запропоновано метод редукції кількості станів часової мережі Петрі. Отримав подальший розвиток метод формування кластера граничного шару комп'ютерної системи підтримки Інтернету речей. Удосконалено метод короткострокового прогнозу часових параметрів кластера граничного шару Інтернету речей, який враховує обмежені обчислювальні можливості вузлів граничного шару Інтернету речей та базується на відповідній функціональній моделі

У *четвертому розділі* було проведено синтез функціональної моделі кластера граничного шару та визначено оптимальний підхід до його побудови. Також проведено оцінку ефективності короткострокового прогнозу часових параметрів граничного шару Інтернету речей. Оцінка, що проводилась для реакції класичного та запропонованого методів короткострокового прогнозування, показала перевагу запропонованого методу.

У *висновках* наведено основні результати наукової роботи щодо вирішення поставлених наукових задач дослідження.

За результатами дослідження отримано такі наукові результати:

– удосконалено метод побудови функціональних моделей кластерів

граничного шару комп'ютерної системи підтримки Інтернету речей, який відрізняється від відомих використанням апарату узагальнених часових мереж Петрі, що дозволило отримати значення перехідних ймовірностей між станами мережі та різні рівні деталізації при моделюванні процесу робочого навантаження, за рахунок чого підвищилась адекватність функціональної моделі реальному процесу обробки оперативних транзакцій Інтернету речей;

- отримав подальший розвиток метод формування кластера граничного шару комп'ютерної системи підтримки Інтернету речей шляхом використання функціональної моделі кластера та апарату напівмарковських процесів з можливою редукцією кількості станів мережі Петрі, що дозволило зменшити середній час обробки оперативних транзакцій, що надходять до граничного шару Інтернету речей;

- удосконалено метод короткострокового прогнозу часових параметрів кластера граничного шару Інтернету речей, який враховує обмежені обчислювальні можливості вузлів граничного шару Інтернету речей та базується на відповідній функціональній моделі, що дозволило підвищити точність прогнозу при плануванні оперативного перерозподілу ресурсів системи підтримки Інтернету речей.

Практичне значення отриманих результатів полягає в тому, що розроблені у роботі методи є науково-практичною основою для подальшого удосконалення інфраструктури підтримки мережі Інтернету речей в частині граничного шару.

Представлені на їх основі інженерні методи та алгоритми дають змогу:

- провести побудову функціональної моделі окремого шару інфраструктури підтримки Інтернету речей, адекватної реальним процесам з точністю до введених припущень;

- сформувати кластер граничного шару інфраструктури підтримки Інтернету речей, який дозволяє зменшити середній час обробки оперативних транзакцій;

- здійснити короткостроковий прогноз часових параметрів кластера

граничного шару Інтернету речей, що дозволить проводити оперативний перерозподіл ресурсів комп'ютерної системи підтримки IoT;

– зменшити середній час обслуговуванні оперативних транзакцій на вузлах граничного шару Інтернету речей до 8%.

За результатами дослідження підтверджено практичну та теоретичну цінність розроблених методів, надано практичні рекомендації, щодо застосування розроблених методів та розглянуто перспективи їх подальшого розвитку.

Ключові слова: Інтернет речей, математична модель, оптимізація, інформаційні системи, комп'ютерна система, комп'ютерні компоненти, хмарні обчислення, марківські моделі, мережі Петрі, граничні обчислення, нова інформація, прогноз, базова модель, кластер, оперативна транзакція

Список публікацій здобувача

Наукові праці, в яких опубліковано основні наукові результати:

1. Shyman A., Kuchuk N., Filatova A., Bellorin-Herrera O. Development of a Method for Assessing the Adequacy of a Computer System Model Based on Petri Nets. *Advanced Information Systems*. Kharkiv, 2024. V. 8. № 3. P. 46–52.
2. Кучук Н.Г., Шиман А.П., Філоненко А.М., Бульба С.С. Розрахунок ефективності використання обчислювальних ресурсів самовідновлювальної комп'ютерної системи. *Системи управління, навігації та зв'язку*. Полтава, 2021. № 3 (65). С. 92–95.
3. Кучук Н.Г., Міхаль О.П., Шиман А.П., Науменко М.В. Динамічна відеофіксація транспортних засобів для системи моніторингу дорожнього руху у мегаполісі. *Системи управління, навігації та зв'язку*. Полтава, 2022. № 2 (68). С. 55–58.
4. Kuchuk N., Shyman A. Method for Distributing Transactions to Hybrid Cloud Data Storage. *Control, Navigation and Communication Systems*. Poltava, 2022. № 4 (70). P. 100–103.

5. Kuchuk N., Shyman A. Synthesis of a Computer Network for Controlling Moving Objects. Control, Navigation and Communication Systems. Poltava, 2023. № 4 (74). P. 134–136.

Опубліковані праці апробаційного характеру:

6. Kolodenko V., Shyman A., Kalinin Y., Kuchuk N. Synthesis of The Theory of Motion of Solid Bodies Filled With Bulk Substances For Fault-Tolerant Identification of Their Parameters. *IEEE 13th International Conference on Dependable Systems, Services and Technologies (DESSERT)* (Athens, Greece 13–15 October 2023). – Athens, 2023. – P. 1–6

7. Shyman A., Kuchuk N., Kovalenko A., Kashkevich S., Partyka S., Yankovskyi O. The Method of Dynamic Routing in Self-Healing Networks. *2024 IEEE 5th KhPI Week on Advanced Technology (KhPIWeek)* (Kharkiv, 07 – 11 October 2024). – Kharkiv, 2024. – P. 1–5

8. Гребенюк Д.С., Шиман А.П. Модель транзакційної системи для розгортання у хмарному середовищі. *Проблеми інформатизації: тези доповідей сьомої міжнародної науково-технічної конференції* (м. Харків, 13–15 листопада 2019 р.). Харків, 2019 – С. 74.

9. Кучук Н.Г., Шиман А.П., Гребенюк Д.С. Синтез інформаційної системи на гіперконвергентній платформі. *Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей десятої міжнародної науково-технічної конференції* (м. Харків, 09 – 10 квітня 2020 р.). – Харків, 2020 – С.43.

10. Kuchuk N., Davydov V., Hrebeniuk D., Shyman A. Mathematical Model of the Information System Synthesis Process. *Current Directions of Development of Information and Communication Technologies and Control Tools: proceedings of 11-th International Scientific and Technical Conference* (Kharkiv, 08 – 09 April 2021). – Kharkiv, 2021. – P. 21.

11. Кучук Н.Г., Бульба С.С, Шиман А.П. Знаходження оптимального інформаційного навантаження компонентів СВКС за комплексним показником. *Проблеми інформатизації: тези доповідей дев'ятої міжнародної науково-*

технічної конференції (м. Харків, 18 – 19 листопада, 2021 р.). – Харків, 2021. – С. 45.

12. Shyman A., Kuchuk N., Hrebeniuk D., Davydov V. Analysis of the Level of Automation of Traffic Management Processes. *Current Directions of Development of Information and Communication Technologies and Control Tools*: proceedings of 12-th International Scientific and Technical Conference (Kharkiv, 27 – 28 April 2022). – Kharkiv, 2022. – P. 20.

13. Шиман А.П., Кучук Н.Г., Бульба С.С., Давидов В.В. Інтелектуальні логістичні системи підтримки дорожнього руху / // *Проблеми інформатизації*: тези доповідей десятої міжнародної науково-технічної конференції (м. Харків, 24 – 25 листопада 2022 р.). – Харків, 2022. – С. 115.

14. Кучук Н.Г., Шиман А.П. Інтелектуальні транспортні системи. *Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління*: тези доповідей тринадцятої міжнародної науково-технічної конференції (м. Харків, 26–27 квітня 2023 р.). – Харків, 2023 – С.55.

15. Kuchuk N., Shyman A. Artificial Intelligence for Traffic Data Analysis. *Current Directions of Development of Information and Communication Technologies and Control Tools*: proceedings of 14-th International Scientific and Technical Conference (Kharkiv, 25 – 26 April 2024). – Kharkiv, 2024. – P. 59.

16. Kuchuk N., Shyman A., Shyman M. Development of a Method for Synthesising Models of Real-Time Software Systems. *Problems of Informatization*: proceedings of 12-th International Scientific and Technical Conference (Kharkiv, 21–22 November 2024). – Kharkiv, 2024. – P. 142.

ABSTRACT

Shyman A.P. Synthesis of an Edge Layer Cluster in a Computer System Supporting the Internet of Things. Qualifying scientific work on the rights of a manuscript.

Dissertation for obtaining the scientific degree of Doctor of Philosophy in the specialty 123 – Computer Engineering. – National Technical University ‘Kharkiv Polytechnic Institute’, Kharkiv, 2025.

The dissertation is devoted to solving a current scientific and technical problem related to increasing the efficiency of using the computing resources of the edge layer in the processing of operational transactions.

The object of research is the process of forming edge layer clusters that unite territorially close devices of the Internet of Things.

The subject of research is methods of synthesis of edge layer clusters of a computer system supporting the Internet of Things.

The purpose of the dissertation is to increase the efficiency of using the computing resources of the edge layer in processing operational transactions by improving the methods of synthesis of individual clusters of the edge layer of the Internet of Things.

The introduction substantiates the relevance of increasing the efficiency of using the computing resources of the edge layer in the processing of operational transactions. It also presents the connection of the work with scientific programmes, plans and topics; the scientific novelty and practical significance of the results obtained; provides information on the personal contribution of the applicant and presents a list of publications on the topic of the dissertation.

The first chapter analyses the current state of development of the Internet of Things and the cloud infrastructure supporting the IoT network. The existing methods of researching the components of computer systems supporting the Internet of Things are analysed. The well-known models and methods for evaluating the efficiency of components of computer systems for real-time support are considered.

The main scientific problem is formulated, which is to increase the efficiency of using the computing resources of the edge layer in the processing of operational transactions by improving the methods of synthesis of individual clusters of the edge layer of the Internet of Things.

The second section is devoted to the development of a method for constructing functional models of the edge layer cluster, in particular, the sequence of stages of studying fragments of real-time computer systems was determined, and a method for assessing the adequacy of network models of fragments of the edge layer was developed.

The third section is devoted to the development of methods for reducing the processing time of operational transactions of the Internet of Things. In particular, the features of timed Petri nets used to describe network models of computer systems supporting the IoT edge layer have been identified. A method for reducing the number of states of a timed Petri net has been proposed. The method of forming a cluster of the edge layer of the computer system supporting the Internet of Things has been further developed. The method of short-term forecasting of the time parameters of the IoT edge layer cluster, which takes into account the limited computing capabilities of the IoT edge layer nodes and is based on the corresponding functional model, has been improved.

In the fourth section, the functional model of the edge layer cluster is synthesised and the optimal approach to its construction is determined. The effectiveness of short-term forecasting of the time parameters of the IoT edge layer was also evaluated. The evaluation of the response of the classical and proposed methods of short-term forecasting showed the superiority of the proposed method.

According to the results of the research, the following scientific results were obtained:

- the method of constructing functional models of clusters of the edge layer of the computer system supporting the Internet of Things has been improved, which differs from the known ones using the apparatus of generalised timed Petri nets, which made it possible to obtain the values of transient probabilities between

network states and different levels of detail in modelling the workload process, thereby increasing the adequacy of the functional model to the real process of processing operational transactions of the Internet of Things;

- the method of forming a cluster of the edge layer of the computer system supporting the Internet of Things was further developed by using the functional model of the cluster and the apparatus of semi-Markov processes with a possible reduction in the number of Petri net states, which allowed to reduce the average processing time of operational transactions received by the edge layer of the Internet of Things;

- a method for short-term forecasting of the time parameters of the IoT edge layer cluster, which takes into account the limited computing capabilities of the IoT edge layer nodes and is based on the corresponding functional model, has been improved, which has increased the accuracy of the forecast when planning the operational reallocation of resources of the IoT support system.

The practical significance of the results obtained is that the methods developed in this work are a scientific and practical basis for further improvement of the infrastructure supporting the Internet of Things network in terms of the edge layer.

The engineering methods and algorithms presented on their basis make it possible to

- to build a functional model of a separate layer of the IoT support infrastructure, adequate to real processes with accuracy to the assumptions made;
- to form a cluster of the edge layer of the IoT support infrastructure, which allows to reduce the average processing time of operational transactions;
- to make a short-term forecast of the time parameters of the cluster of the edge layer of the Internet of Things, which will allow for the prompt redistribution of resources of the computer system of IoT support;
- to reduce the average time for servicing operational transactions on the nodes of the IoT edge layer to 8%.

The results of the study confirm the practical and theoretical value of the

developed methods, provide practical recommendations on the application of the developed methods and consider the prospects for their further development.

Keywords: Internet of things, mathematical model, optimisation, information systems, computer system, computer components, cloud computing, Markov models, Petri nets, edge computing, new information, forecast, basic model, cluster, operational transaction.

List of publications of the applicant

Scientific works in which the main scientific results are published:

1. Shyman A., Kuchuk N., Filatova A., Bellorin-Herrera O. Development of a Method for Assessing the Adequacy of a Computer System Model Based on Petri Nets. *Advanced Information Systems*. Kharkiv, 2024. V. 8. № 3. P. 46-52.
2. Kuchuk N.G., Shyman A.P., Filonenko A.M., Bulba S.S. Calculation of the Efficiency of Using Computing Resources of a Self-healing Computer System. *Control, navigation and communication systems*. Poltava, 2021. № 3 (65). P. 92-95.
3. Kuchuk N.G., Mikhal O.P., Shyman A.P., Naumenko M.V. Dynamic video recording of vehicles for a traffic monitoring system in a metropolis. *Control, navigation and communication systems*. Poltava, 2022. № 2 (68). P. 55-58.
4. Kuchuk N., Shyman A. Method for Distributing Transactions to Hybrid Cloud Data Storage. *Control, Navigation and Communication Systems*. Poltava, 2022. № 4 (70). P. 100-103.
5. Kuchuk N., Shyman A. Synthesis of a Computer Network for Controlling Moving Objects. *Control, Navigation and Communication Systems*. Poltava, 2023. № 4 (74). P. 134-136.

Published papers of an approbation nature:

6. Kolodenko V., Shyman A., Kalinin Y., Kuchuk N. Synthesis of The Theory of Motion of Solid Bodies Filled With Bulk Substances For Fault-Tolerant Identification of Their Parameters. *IEEE 13th International Conference on Dependable Systems, Services and Technologies (DESSERT)* (Athens, Greece 13-

15 October 2023). - Athens, 2023. - P. 1-6.

7. Shyman A., Kuchuk N., Kovalenko A., Kashkevich S., Partyka S., Yankovskyi O. The Method of Dynamic Routing in Self-Healing Networks. 2024 *IEEE 5th KhPI Week on Advanced Technology (KhPIWeek)* (Kharkiv, 07 - 11 October 2024). - Kharkiv, 2024. - P. 1-5.

8. Hrebeniuk D., Shyman A. A model of a transactional system for deployment in the cloud environment. *Problems of informatisation: proceedings of 7-th international scientific and technical conference* (Kharkiv, 13-15 November 2019). Kharkiv, 2019 - P. 74.

9. Kuchuk N., Shyman A., Hrebeniuk D. Synthesis of an information system on a hyperconverged platform. *Modern directions of development of information and communication technologies and management tools: proceedings of 10-th international scientific and technical conference* (Kharkiv, 09 - 10 April 2020). - Kharkiv, 2020 - P. 43.

10. Kuchuk N., Davydov V., Hrebeniuk D., Shyman A.. Mathematical Model of the Information System Synthesis Process. Current Directions of Development of Information and Communication Technologies and Control Tools: proceedings of the 11th International Scientific and Technical Conference (Kharkiv, 08 - 09 April 2021). - Kharkiv, 2021. - P. 21.

11. Kuchuk N., Bulba S., Shyman A. Finding the optimal information load of IPSS components by a complex indicator. *Problems of informatisation: proceedings of -th international scientific and technical conference* (Kharkiv, 18-19 November, 2021). - Kharkiv, 2021. - P. 45.

12. Shyman A., Kuchuk N., Hrebeniuk D., Davydov V.. Analysis of the Level of Automation of Traffic Management Processes. *Current Directions of Development of Information and Communication Technologies and Control Tools: proceedings of the 12th International Scientific and Technical Conference* (Kharkiv, 27 - 28 April 2022). - Kharkiv, 2022. - P. 20.

13. Shyman A., Kuchuk N., Bulba S., Davydov V. Intelligent logistics systems for road traffic support. *Problems of informatisation: proceedings of 10-th*

international scientific and technical conference (Kharkiv, 24 - 25 November 2022).
- Kharkiv, 2022. - P. 115.

14. Kuchuk N., Shyman A. Intelligent transport systems. *Modern directions of development of information and communication technologies and management tools*: proceedings of 13-th international scientific and technical conference (Kharkiv, 26-27 April 2023). - Kharkiv, 2023 - P. 55.

15. Kuchuk N., Shyman A. Artificial Intelligence for Traffic Data Analysis. *Current Directions of Development of Information and Communication Technologies and Control Tools*: proceedings of the 14th International Scientific and Technical Conference (Kharkiv, 25 - 26 April 2024). - Kharkiv, 2024. - P. 59.

16. Kuchuk N., Shyman A., Shyman M. Development of a Method for Synthesising Models of Real-Time Software Systems. *Problems of Informatisation*: proceedings of the 12th International Scientific and Technical Conference (Kharkiv, 21-22 November 2024). - Kharkiv, 2024. - P. 142.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	4
ВСТУП.....	7
РОЗДІЛ 1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ КОМП'ЮТЕРНОЇ СИСТЕМИ ПІДТРИМКИ ІНТЕРНЕТУ РЕЧЕЙ.....	15
1.1 Особливості концепції Інтернету речей.....	15
1.2 Аналіз хмарної інфраструктури підтримки мережі IoT	21
1.3 Аналіз існуючих методів дослідження компонент комп'ютерних систем підтримки Інтернету речей	26
1.4 Огляд відомих моделей і методів оцінювання ефективності компонент комп'ютерних систем підтримки режиму реального часу	33
1.5 Постановка наукової задачі та окремих завдань дослідження	36
Висновки за розділом 1	38
РОЗДІЛ 2 РОЗРОБКА МЕТОДУ ПОБУДОВИ ФУНКЦІОНАЛЬНИХ МОДЕЛЕЙ КЛАСТЕРІВ ГРАНИЧНОГО ШАРУ	40
2.1 Визначення послідовності етапів дослідження фрагментів комп'ютерних систем реального часу.....	40
2.2 Розробка методу побудови функціональної моделі кластера граничного шару.....	46
2.2.1 Опис методу побудови функціональних моделей.....	47
2.2.2 Приклад застосування методу побудови функціональних моделей	50
2.2.3 Переваги та недоліки запропонованого методу побудови функціональних моделей	53
2.3 Розробка методів оцінки адекватності мережних моделей фрагментів граничного шару	55
2.4 Агрегативний опис функціональних моделей процесів у кластерах граничного шару	63

Висновки за розділом 2.....	69
РОЗДІЛ 3 РОЗРОБКА МЕТОДІВ ЗМЕНШЕННЯ ЧАСУ ОБРОБКИ ОПЕРАТИВНИХ ТРАНЗАКЦІЙ ІНТЕРНЕТУ РЕЧЕЙ.....	71
3.1 Розробка методу формування кластера граничного шару комп'ютерної системи підтримки Інтернету речей	71
3.2 Редукція кількості станів часових мереж Петрі	80
3.3 Розробка методу короткострокового прогнозу часових параметрів кластера граничного шару Інтернету речей.....	89
Висновки за розділом 3.....	103
РОЗДІЛ 4 ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ ГРАНИЧНОГО ШАРУ КОМП'ЮТЕРНОЇ СИСТЕМИ ПІДТРИМКИ ІНТЕРНЕТУ РЕЧЕЙ.....	105
4.1 Опис вхідних даних для побудови функціональної моделі кластера граничного шару системи підтримки IoT	105
4.2 Синтез функціональної моделі кластера граничного шару	108
4.3 Вибір оптимального варіанту формування кластера граничного шару.....	112
4.4 Оцінка ефективності короткострокового прогнозу часових параметрів кластера граничного шару Інтернету речей.....	116
Висновки за розділом 4.....	122
ВИСНОВКИ.....	124
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	127
ДОДАТОК А СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА.....	143
ДОДАТОК Б АКТИ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ НАУКОВИХ ДОСЛІДЖЕНЬ ДИСЕРТАЦІЙНОЇ РОБОТИ.....	147

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

- AI – Artificial Intelligence (штучний інтелект);
- API – Application Programming Interface (прикладний програмний інтерфейс);
- ARM – Advanced RISC Machine (поліпшена RISC машина);
- AWS – Amazon Web Services;
- CAN – Controller Area Network (локальна мережа контролерів);
- CCPA – California Consumer Privacy Act (закон Каліфорнії про конфіденційність споживачів);
- CoAP – Constrained Application Protocol (Протокол обмеженого застосування);
- CRM – Customer Relationship Management (система управління взаємовідносинами з клієнтами);
- DC – Direct Current (постійний струм);
- DI – Digital Input (цифровий вхід);
- DIN – Deutsches Institut für Normung e.V (Німецький інститут зі стандартизації);
- DLL – Dynamic Link Library (динамічно приєднувана бібліотека);
- DO – Digital Output (цифрових вихід);
- ECU – Electronic Control Unit (блок керування двигуном);
- ERP – Enterprise resource planning (планування ресурсів підприємства);
- GDPR – General Data Protection Regulation (загальний регламент про захист даних);
- GERT – Graphical Evaluation and Review Technique (метод графічної оцінки й аналізу);
- GPS – Global Positioning System (Система глобального позиціонування)
- GSPN – Generalized Stochastic Petri Nets (узагальнена стохастична мережа Петрі);

- HTTP – Hypertext Transfer Protocol (протокол передачі гіпертексту);
- HTTPS – Hyper Text Transfer Protocol Secure (безпечний протокол передачі гіпертексту);
- IBM – International Business Machines;
- IEC – International Electrotechnical Commission (міжнародна електротехнічна комісія);
- IoT – Internet of Things (Інтернет речей)
- ISO – International Organization for Standardization (міжнародна організація зі стандартизації);
- LoRaWAN – Long Range Wide Area Network (глобальна мережа далекого радіусу дії);
- LPWAN – Low-power Wide-area Network (мережі з низьким енергоспоживанням і великим радіусом дії)
- LTE – Long Term Evolution (довготерміновий розвиток);
- ML – Machine Learning (Машинне навчання);
- MQTT – Message Queuing Telemetry Transport (протокол черги повідомлень телеметрії);
- NB-IoT – Narrow Band Internet of Things (вузькосмуговий Інтернет речей);
- NoSQL - not only Structured Query Language;
- OPC UA – Open Platform Communications Unified Architecture (уніфікована архітектура комунікацій відкритої платформи);
- OTA – Over The Air (оновлення «по повітрю»);
- RAM – Random Access Memory (пам'ять з довільним доступом);
- RFID – Radio Frequency Identification (радіочастотна ідентифікація);
- RISC – Reduced Instruction Set Computer (обчислення зі скороченим набором команд);
- RS – Recommended Standard (рекомендований стандарт);
- RTU – Remote Terminal Unit (віддалене кінцевий пристрій);

- SCADA – Supervisory Control and Data Acquisition (диспетчерське управління і збір даних);
- SDK – Software Development Kit (набір з засобів розробки);
- SM-SPN – Semi-Markov Stochastic Petri Nets (напівмарковські стохастичні мережі Петрі);
- SPN – Stochastic Petri Nets (стохастичної мережі Петрі);
- SSL – Secure Sockets Layer (рівень захищених сокетів);
- TCP – Transmission Control Protocol (Протокол управління передачею);
- TLS – Transport Layer Security (протокол безпеки транспортного рівня);
- USB – Universal Serial Bus (універсальна послідовна шина);
- VPN – Virtual Private Network (віртуальна приватна мережа);
- КС – комп'ютерна система;
- КСРЧ – комп'ютерна система реального часу;
- МНК – метод найменших квадратів;
- НМП – напівмарковський процес;
- ОС – операційна система;
- РРЧ – режим реального часу;
- СДЧМП – стохастично-детермінована часова мережа Петрі;
- СМО – системи масового обслуговування;
- СРЧ – система реального часу;
- ССКПП – середня сума квадратів помилок прогнозу;
- УЧСМП – узагальнені часові стохастичні мережі Петрі.

ВСТУП

Актуальність теми. З розвитком інформаційного суспільства змінюються очікування користувачів щодо інформаційних сервісів. Важливішими стають не технічні параметри, а якість обслуговування: швидкість виконання завдань, зручність використання та своєчасне задоволення потреб бізнесу.

На сьогодні, швидкого розвитку набуває концепція Інтернету речей (IoT), що є мережею взаємопов'язаних пристроїв, які обмінюються даними без прямого втручання людини. Завдяки IoT можливе дистанційне керування технікою, моніторинг виробничих процесів і автоматизація повсякденного життя, а розвиток Інтернету речей сприяє підвищенню ефективності бізнесу, зниженню витрат і створенню розумних екосистем.

Інфраструктура підтримки IoT є комплексом технологій, обладнання та програмного забезпечення, необхідних для забезпечення стабільного функціонування IoT-систем. Вона містить мережну інфраструктуру (Wi-Fi, 5G, LPWAN, Bluetooth) для з'єднання пристроїв, обчислювальні ресурси хмарного, туманного та граничного шарів обробки та зберігання даних і програмні платформи для управління пристроями та аналізу інформації. Різниця між хмарним (Cloud), туманним (Fog) та граничним (Edge) шарами IoT полягає в рівні обробки даних та віддаленості від кінцевих пристроїв. Хмарний шар (Cloud Computing) обробляє дані у віддалених дата-центрах, які забезпечують достатні ресурси для глибокої аналітики, машинного навчання та довготривалого зберігання. Проте він має високу затримку передачі даних, оскільки потребує підключення до Інтернету. Туманний шар (Fog Computing) обробляє дані на локальних серверах та розподілених вузлах, розташованих ближче до кінцевих пристроїв (наприклад, у мережах підприємств або базових станціях). Він розподіляє обчислювальні навантаження між хмарою та граничним рівнем, забезпечуючи менші затримки, ніж у хмарі, хоча все ще залежить від мережі. Граничний шар обробляє дані безпосередньо на IoT-

пристроях або шлюзах (камери, сенсори, мікроконтролери, локальні сервери). Він забезпечує мінімальні затримки, оскільки обробка відбувається на пристрої, проте не завжди має достатню потужність для виконання складних обчислень.

У інформаційних потоках, що надходять від пристроїв IoT до інфраструктури, особливої уваги потребують оперативні транзакції, які вимагають виконання в режимі реального часу або близькому до нього. Тому, найбільш сприятливі умови для їхнього виконання надають пристрої граничного шару. Але водночас треба враховувати обмеженість обчислювальних ресурсів цього шару. Внаслідок цього виникають непередбачувані затримки в процесі обробки оперативних транзакцій, пов'язані або з чергами на обробку, або з пошуком пристроїв, що задовольняють вимогам, які необхідні для виконання поточної транзакції.

Таким чином, **актуальною науково-технічною задачею** є підвищення ефективності використання обчислювальних ресурсів граничного шару при обробці оперативних транзакцій шляхом удосконалення методів синтезу окремих кластерів граничного шару Інтернету речей, що об'єднують низку територіально близьких пристроїв IoT. При цьому мають враховуватися як вимоги, що необхідні для виконання оперативних транзакцій, так і показники використання наявних ресурсів граничного шару Інтернету речей.

Зв'язок роботи з науковими програмами, планами, темами. Дослідження виконані на кафедрі «Комп'ютерна інженерія та програмування» НТУ «Харківський політехнічний інститут» в рамках науково-дослідної теми: «Моделі і методи обробки даних і розподілу мережних ресурсів в комп'ютерних системах» (ДР №0122U200527, компанія «LineUp», м. Харків), в котрій здобувачка брала участь у якості виконавиці окремого розділу.

Об'єкт дослідження – процес формування кластерів граничного шару, що об'єднують територіально близькі пристрої Інтернету речей.

Предмет дослідження – методи синтезу кластерів граничного шару комп'ютерної системи підтримки Інтернету речей.

Мета дослідження – підвищення ефективності використання обчислювальних ресурсів граничного шару при обробці оперативних транзакцій шляхом удосконалення методів синтезу окремих кластерів граничного шару Інтернету речей.

Відповідно до поставленої мети визначені наступні задачі:

- 1) проведення аналізу існуючих методів підвищення ефективності комп'ютерної системи підтримки Інтернету речей з метою виявлення проблем, що виникають при задіянні пристроїв граничного шару для виконання оперативних транзакцій;
- 2) розроблення методу побудови функціональних моделей кластерів граничного шару комп'ютерної системи підтримки Інтернету речей;
- 3) розроблення методу формування кластера граничного шару комп'ютерної системи підтримки Інтернету речей;
- 4) розроблення методу короткострокового прогнозу часових параметрів кластера граничного шару комп'ютерної системи підтримки Інтернету речей;
- 5) проведення порівняльної оцінки розроблених та існуючих методів синтезу кластера граничного шару комп'ютерної системи підтримки Інтернету речей.

Методи дослідження. При розв'язанні науково-технічної задачі використано широкий спектр методів. Так, при розробці методу побудови функціональних моделей кластерів граничного шару комп'ютерної системи підтримки Інтернету речей використовувалися: теоретико-множинний підхід; математичний апарат узагальнених часових мереж Петрі; теорія подійних графів; теорія марковських процесів. При розробці методів зменшення часу обробки оперативних транзакцій Інтернету речей додатково використовувалися методи теорії масового обслуговування, теорія напівмарковських процесів та теорія графової редукції. Оцінка експериментальних даних, отриманих у ході роботи, проводилася на основі методів теорії ймовірності та математичної статистики.

Вибір методів досліджень забезпечив достовірність отриманих

результатів і висновків, що підтверджується збіжністю результатів експериментальних досліджень, отриманих при програмній реалізації алгоритмів запропонованих методів з теоретичними і практичними результатами.

Наукова новизна отриманих результатів обумовлена розробленими методами обробки оперативних транзакцій на граничному шарі комп'ютерної системи підтримки Інтернету речей, в межах яких отримані такі нові наукові результати:

- удосконалено метод побудови функціональних моделей кластерів граничного шару комп'ютерної системи підтримки Інтернету речей, який відрізняється від відомих використанням апарату узагальнених часових мереж Петрі, що дозволило отримати значення перехідних ймовірностей між станами мережі та різні рівні деталізації при моделюванні процесу робочого навантаження, за рахунок чого підвищилась адекватність функціональної моделі реальному процесу обробки оперативних транзакцій Інтернету речей;

- отримав подальший розвиток метод формування кластера граничного шару комп'ютерної системи підтримки Інтернету речей шляхом використання функціональної моделі кластера та апарату напівмарковських процесів з можливою редукцією кількості станів мережі Петрі, що дозволило зменшити середній час обробки оперативних транзакцій, що надходять до граничного шару Інтернету речей;

- удосконалено метод короткострокового прогнозу часових параметрів кластера граничного шару Інтернету речей, який враховує обмежені обчислювальні можливості вузлів граничного шару Інтернету речей та базується на відповідній функціональній моделі, що дозволило підвищити точність прогнозу при плануванні оперативного перерозподілу ресурсів системи підтримки Інтернету речей.

Практичне значення отриманих результатів полягає в тому, що розроблені у роботі методи є науково-практичною основою для подальшого удосконалення інфраструктури підтримки мережі Інтернету речей в частині

граничного шару.

Представлені на їх основі інженерні методи та алгоритми дають змогу:

- провести побудову функціональної моделі окремого шару інфраструктури підтримки Інтернету речей, адекватної реальним процесам з точністю до введених припущень;
- сформувати кластер граничного шару інфраструктури підтримки Інтернету речей, який дозволяє зменшити середній час обробки оперативних транзакцій;
- здійснити короткостроковий прогноз часових параметрів кластера граничного шару Інтернету речей, що дозволить проводити оперативний перерозподіл ресурсів комп’ютерної системи підтримки IoT;
- зменшити середній час обслуговування оперативних транзакцій на вузлах граничного шару Інтернету речей до 8%.

Результати досліджень впроваджено (див. додаток Б):

- на підприємстві EURO RAIL SP. Z O.O. (місто Краків) в ході дослідницьких робіт при проектуванні комплексної рекомендаційної системи (акт впровадження від 30.12.2024 р.);
- у навчальний процес кафедри “Комп’ютерна інженерія та програмування” Національного технічного університету “ХПІ” (акт впровадження від 17.12.2024 р.);
- у навчальний процес кафедри “Системи інформації ім. В.О. Кравця” Національного технічного університету “ХПІ” (акт впровадження від 25.01.2025 р.);

Особистий внесок здобувача полягає в розробленні нових методів і рішень, які забезпечують виконання поставлених в дисертаційній роботі завдань.

Результати дисертаційної роботи отримані здобувачем особисто та в співаторстві опубліковані у публікаціях (додаток А):

1) фахові видання України:

[23] – здобувачем розроблено метод розрахунку ефективності

використання обчислювальних ресурсів комп'ютерної системи, орієнтованого на особливості самовідновлювальних структур. Співавторами: Кучук Н.Г. було запропоновано спосіб знаходження оптимального інформаційного навантаження компонентів СВКС за комплексним показником; Філоненко А.М. було проведено синтез інформаційної структури СВКС та отримано числові значення відповідних характеристик технічних засобів; Бульбою С.С. було розроблено алгоритм пошуку прийнятних значень ступеню завантаженості пучка каналів компонент.

[20] – здобувачем було проведено аналіз технічного забезпечення, що використовується для фіксації об'єктів дорожнього руху. Співавторами: Кучук Н.Г. було розглянуті принципи дії детекторів транспортних засобів, що визначають інтенсивність і склад руху, їх недоліки і переваги; Міхалем О.П. та Науменко Н.В. було вибрано потрібні програмні засоби та розроблено відповідну програму в обраному середовищі.

[78] – здобувачем було розроблено метод оптимального розподілу транзакцій до гібридного хмарного сховища. Співавтором Кучук Н.Г. було запропоновано ітераційний алгоритм побудови максимального допустимого потоку.

[80] – здобувачем було розроблено методу синтезу базової мережі комп'ютерної системи управління, яка враховує рухомість об'єктів управління. Співавтор Кучук Н.Г. провела синтез алгоритмів адаптивної маршрутизації, що забезпечують оптимальний розподіл потоків по КЗ мережі.

[108] – здобувачем було проаналізовано поведінку моделі систем, побудованою за допомогою мереж Петрі. Співавторами: Кучук Н.Г. було проведено дослідження алгоритму управління транспортною системою мегаполісу; Бельорін-Еррерою О.М. було запропоновані відповідні статистичні критерії; Філатовою Г.Є. було розглянуто варіант перевірки гіпотези щодо марковського характеру процесів зміни станів системи.

Апробація результатів. Основні наукові положення й результати, отримані автором при виконанні дисертаційної роботи, доповідалися на:

VII Міжнародній науково-технічній конференції «Проблеми інформатизації». Черкаси – Баку – Бельсько-Бяла – Харків, 2019;

IX Міжнародній науково-технічній конференції «Проблеми інформатизації». Черкаси – Баку – Бельсько-Бяла – Харків, Т. 2. 2021;

X Міжнародній науково-технічній конференції «Проблеми інформатизації». Черкаси – Баку – Бельсько-Бяла – Харків, Т. 2. 2022;

XII Міжнародній науково-технічній конференції «Проблеми інформатизації». Черкаси – Баку – Бельсько-Бяла – Харків, Т. 2. 2024;

X Міжнародній науково-технічній конференції «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління». Баку – Харків – Жиліна, 2020;

XI Міжнародній науково-технічній конференції. Баку – Харків – Жиліна, Т. 1. «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління» Баку – Харків – Жиліна, 2021 р.;

XII Міжнародній науково-технічній конференції. Баку – Харків – Жиліна, Т. 1. «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління» Баку – Харків – Жиліна, 2022 р.;

XIII Міжнародній науково-технічній конференції. Баку – Харків – Жиліна, Т. 1. «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління» Баку – Харків – Жиліна, 2023 р.;

XIV Міжнародній науково-технічній конференції «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління». Баку – Харків – Жиліна, 2024 р.;

XIII Міжнародній науково-технічній конференції «Dependable Systems, Services and Technologies», Афіни, Греція, 2023 р. (SCOPUS);

V Міжнародній науково-технічній конференції «IEEE 3rd KhPI Week on Advanced Technology Conference» Україна, Харків, 2024 р. (SCOPUS).

Публікації. Результати наукових досліджень відображено в 16 друкованих працях, зокрема в 5 статтях у наукових фахових виданнях України (з яких 1 індексована у SCOPUS), 9 публікаціях в матеріалах міжнародних

наукових конференцій, 2 публікаціях в матеріалах міжнародних наукових конференцій, що внесені до міжнародної наукометричної бази SCOPUS.

Структура і обсяг дисертації. Дисертаційна робота складається з анотації двома мовами, вступу, чотирьох розділів, висновків, списку використаних джерел та 2 додатків. Робота містить 142 сторінки основного тексту, серед них: 32 рисунки, 9 таблиць, 114 найменувань у списку використаних джерел, 7 сторінок додатків. Загальний обсяг роботи викладено на 149 сторінках.

РОЗДІЛ 1

АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ КОМП'ЮТЕРНОЇ СИСТЕМИ ПІДТРИМКИ ІНТЕРНЕТУ РЕЧЕЙ

1.1 Особливості концепції Інтернету речей

Концепція Інтернету речей пройшла шлях від простих RFID-систем до складних мереж, що підтримуються системами штучного інтелекту [33]. Передумовою її появи була запропонована у 1969 році Кевіном Ештоном концепція ідентифікації об'єктів за допомогою RFID (Radio Frequency Identification) [83]. Через 30 років він вперше вводить термін "Internet of Things" (IoT) для опису мережного зв'язку фізичних об'єктів. Спочатку концепція базувалася на RFID, що використовувався для автоматизованого збору даних. Подальше масове поширення Інтернету, зростання кількості підключених пристроїв, розвиток бездротових мереж, хмарних обчислень та розумних сенсорів призвели до того, що у 2024 році кількість підключених пристроїв втричі перевищила населення Землі [113].

На сьогодні IoT є ключовим елементом цифрової трансформації суспільства та продовжує активно розвиватися.

IoT є технологічною концепцією, яка передбачає з'єднання різноманітних фізичних пристроїв, таких як сенсори, датчики, побутова техніка, транспортні засоби, промислове обладнання, тощо, в єдину мережу, котра здійснює збір, обмін та аналіз даних, виключаючи безпосереднє втручання людини [91]. Пристрої IoT підключаються до мережі через Wi-Fi, Bluetooth, 5G, LPWAN, Zigbee та інші бездротові технології. Дані передаються від пристроїв до хмарних сервісів для подальшої обробки [48]. Пристрої можуть працювати самостійно, використовуючи алгоритми штучного інтелекту (AI) та машинного навчання (ML) для прийняття рішень, наприклад, розумні термостати можуть регулювати без втручання людини температуру приміщення. Низка пристроїв IoT займається постійним збиранням даних про

навколишнє середовище та передачею їх для обробки. Також використання Big Data та аналітики в реальному часі дозволяє прогнозувати події та оптимізувати роботу систем IoT [37].

IoT-системи можуть миттєво реагувати на зміни умов, використовуючи оперативні транзакції, наприклад, сигналізувати про несправності обладнання або аварійні ситуації в розумному будинку чи на виробництві. IoT-екосистема може включати від кількох до мільйонів пристроїв, що працюють разом. Різні платформи та стандарти дозволяють налаштовувати систему під конкретні потреби.

Архітектура екосистеми IoT складається з таких рівнів (рис. 1.1, [49]):

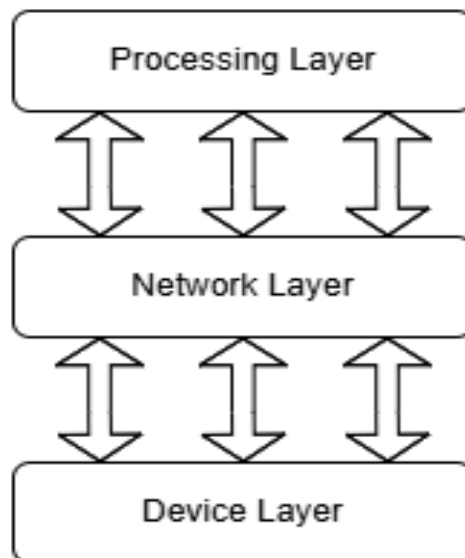


Рисунок 1.1 – Узагальнена архітектура екосистеми IoT

- рівень пристроїв (Device Layer), котрий включає датчики, сенсори, мікроконтролери, розумні пристрої та відповідає за збір інформації;
- рівень мережі зв'язку (Network Layer), на котрому зібрані дані передаються через дротові та бездротові мережі та забезпечується зв'язок між пристроями, шлюзами та хмарними сервісами;
- рівень обчислень (Processing Layer) або інфраструктура підтримки IoT, котра базується на граничних, туманних та хмарних обчисленнях і

використовує Big Data, AI, машинне навчання для обробки інформації та прийняття рішень.

На сьогодні концепція Інтернету речей застосовується у багатьох сферах. Серед них є такі:

1. Розумні міста (Smart Cities), містять, наприклад, системи моніторингу транспорту, освітлення, екології, енергоспоживання; інтелектуальні світлофори, що регулюють трафік залежно від завантаженості доріг; датчики якості повітря, що аналізують рівень забруднення в режимі реального часу [88].

2. Промисловість (Industry 4.0/5.0), наприклад, віддалений моніторинг та управління виробничими процесами; автоматичний контроль за станом обладнання та передбачуване обслуговування (Predictive Maintenance); використання роботизованих систем та автономних дронів [96].

3. Медицина (IoT Healthcare), наприклад, портативні пристрої для моніторингу здоров'я (розумні годинники, фітнес-браслети, кардіомонітори); віддалена діагностика пацієнтів через підключені медичні пристрої; смарт-лікарні з автоматизованими системами управління [99].

4. Розумні будинки (Smart Home), наприклад, інтелектуальні системи освітлення, опалення, відеоспостереження та безпеки; автоматизовані помічники (Google Home, Amazon Alexa, Apple HomeKit); енергозберігаючі рішення на основі поведінкового аналізу користувачів [97].

5. Сільське господарство (Smart Agriculture), наприклад, автоматизовані системи зрошення на основі вологості ґрунту; відстеження стану полів та прогнозування врожайності за допомогою дронів; моніторинг стану худоби та оптимізація годування [105].

6. Транспорт і логістика (Smart Transport & Logistics), наприклад, інтелектуальні системи керування рухом (автономні автомобілі, дрони); відстеження вантажів у реальному часі через GPS та RFID; оптимізація маршрутів для зменшення витрат на паливо [107].

Визначимо основні властивості мереж IoT [12, 34, 50, 82].

1. Масштабованість та взаємопов'язаність: пристрої IoT формують єдину мережу, де обмінюються даними без прямого втручання людини; мережа може містити мільйони пристроїв, що взаємодіють через Інтернет; мережа IoT припускає горизонтальне масштабування.
2. Автоматизація та автономність: пристрої можуть працювати в автономному режимі, приймаючи рішення на основі сенсорних даних та відповідних алгоритмів; використовуються розумні системи керування, що зменшують потребу в ручному управлінні.
3. Обробка та аналіз великих даних: IoT генерує величезні обсяги даних, які можуть оброблятися як локально (Edge, Fog Computing), так і віддалено (у хмарному середовищі); застосовуються штучний інтелект (AI) та технологія Big Data для аналізу інформації.
4. Реальний час та низькі затримки: багато IoT-додатків працюють у режимі реального часу (автономні автомобілі, медицина, промислові датчики); використання граничних і периферійних обчислень зменшує затримки передачі даних.
5. Різноманітність мережних технологій: IoT-пристрої підключаються через Wi-Fi, 5G, LPWAN, Bluetooth, Zigbee, LoRaWAN та інші технології; вибір технології залежить від потреб у швидкості, енергоспоживанні та дальності зв'язку; постійно збільшується відсоток підключення мобільних IoT пристроїв за відповідними технологіями.
6. Безпека та конфіденційність: IoT-пристрої вразливі до кібератак, тому важливі шифрування даних, автентифікація та оновлення прошивок; обробка персональних даних вимагає відповідності стандартам GDPR, ISO 27001 та іншим, необхідним для забезпечення безпеки.
7. Енергоефективність та низьке енергоспоживання: багато IoT-пристроїв працюють на батареях або альтернативних джерелах енергії; використовуються енергоефективні протоколи зв'язку; особливо важливо енергозбереження для мобільних пристроїв IoT.

8. Гнучкість застосування: IoT використовується у розумних містах, медицині, транспорті, виробництві, сільському господарстві та інших галузях; гнучка архітектура концепції IoT дозволяє адаптувати відповідну технологію під конкретні задачі.

Але швидкі темпи розвитку IoT сприяють появі нових викликів та обмежень. Розглянемо деякі із них [6, 31, 86].

1. Безпека та конфіденційність: велика кількість підключених пристроїв створює більше точок для кібератак, але при цьому відмічається недостатній рівень шифрування даних під час передачі, уразливі прошивки та відсутність регулярних оновлень, проблеми з автентифікацією та контролем доступу, тобто збільшуються ризики кіберзагроз; тому використання шифрування, автентифікації та оновлення прошивок є критично важливим.

2. Енергоефективність: більшість IoT-пристроїв працює на батареях, що обмежує їхній час роботи та потребує постійної заміни або заряджання батарей у великій кількості пристроїв, тому потрібно використання енергоефективних процесорів та компонентів, застосування протоколів LPWAN (Low Power Wide Area Network) для зменшення витрат енергії, використання відновлюваних джерел енергії (сонячна, кінетична енергія).

3. Стандартизація: відсутність єдиних стандартів призводить до того, що пристрої різних виробників не можуть працювати разом, використання різних протоколів зв'язку (MQTT, CoAP, HTTP, LoRaWAN, Zigbee, Z-Wave) створює проблеми для інтеграції, закриті екосистеми виробників обмежують можливості підключення сторонніх пристроїв, тому потрібно розвивати загальні стандарти (наприклад, Matter – спільний стандарт від Apple, Google, Amazon), використовувати універсальні API та платформи для керування IoT-пристроями та підтримувати відкриті протоколи для полегшення інтеграції.

4. Управління великими обсягами даних: IoT-мережі генерують терабайти даних, які потрібно зберігати та обробляти в реальному часі, або наближеному до нього, внаслідок чого виникає складність фільтрації

корисних даних серед великого обсягу інформації та затримки у передачі та обробці даних, що може впливати на швидкість реакції системи; отже, потрібно використовувати технології Big Data та штучного інтелекту для аналізу IoT-даних та впроваджувати локальну обробку даних (Fog Computing, Edge Computing) або проводити оптимізацію протоколів передачі даних (наприклад, MQTT замість HTTP для зменшення трафіку).

5. Надійність та стійкість до збоїв: втрата зв'язку у мережах IoT може призвести до збоїв у критичних системах, таких як, наприклад, автоматизоване виробництво, транспорт, медицина; поломка одного вузла може впливати на роботу всієї мережі, крім того, є висока залежність від хмарних сервісів; тому шляхами вирішення є розробка механізмів автономної роботи пристроїв у разі втрати зв'язку, використання дублюючих систем для зменшення ризиків відмови та перехід на розподілені обчислення для зменшення залежності від центрального сервера.

6. Правові та етичні виклики: дані проблеми виникають при регулюванні збору, зберігання та використання персональних даних, розслідуваннях у разі збою або аварії, наприклад, у випадку самокерованих автомобілів, використанні IoT для стеження за людьми без їхньої згоди, дотриманні міжнародних норм, таких як GDPR, CCPA, для вирішення яких треба впроваджувати прозорі механізми керування даними, забезпечувати відкритість алгоритмів прийняття рішень у критичних системах та посилювати законодавчий контроль за безпекою IoT.

7. Вартість впровадження: попри зниження вартості сенсорів та мережних рішень, комплексні IoT-системи все ще потребують значних інвестицій.

8. Масштабованість та продуктивність: проблеми пов'язані із збільшення кількості пристроїв створює навантаження на мережу, що може призводити до затримок та збоїв, також з високим рівнем енергоспоживання пристроїв із постійним підключенням до інтернету та обмеженими можливостями обчислювальної потужності та пам'яті у багатьох IoT-

пристроях; шляхами вирішення є проведення обробки даних безпосередньо на пристроях або шлюзах перед передачею в хмару, оптимізація енергоспоживання за рахунок втілення ефективних алгоритмів та низькоенергетичних протоколів зв'язку та впровадження 5G/6G для покращення пропускної здатності мережі зв'язку IoT.

Особливо чутливим даний виклик є при опрацюванні оперативних транзакцій IoT, котрі повинні виконуватися у режимі реального часу або наближеному до нього [4, 107]. Одним із варіантів його вирішення є розширення хмарної інфраструктури підтримки IoT шарами, наближеними до пристроїв IoT [118]. Затримки в передачі даних зменшуються за рахунок використання граничних (Edge Computing) та туманних (Fog Computing) обчислень. Звісно, найменші затримки при передачі даних будуть при задіянні вузлів граничних обчислень. Але треба враховувати, що всі задіяні пристрої мають обмежені обчислювальні можливості.

1.2 Аналіз хмарної інфраструктури підтримки мережі IoT

Хмарні технології відіграють важливу роль у розвитку Інтернету речей (IoT), забезпечуючи гнучке масштабування, зберігання та ефективну обробку великих масивів даних. Вони дозволяють керувати величезною кількістю пристроїв, здійснювати аналіз інформації в режимі реального часу та впроваджувати розумні рішення [1, 5, 39].

Хмарна екосистема IoT складається з кількох рівнів: пристрої IoT (датчики, контролери, розумні пристрої), які збирають дані та передають їх до хмари; шлюзи (Gateways), які є проміжним рівнем між пристроями IoT та хмарними сервісами, який може фільтрувати та агрегувати дані перед передачею; хмарна платформа, що забезпечує зберігання, обробку, аналіз та керування пристроями, аналітика та штучний інтелект, що забезпечує механізми штучного інтелекту для аналізу даних, виявлення аномалій та автоматизації рішень [25, 35].

Популярні хмарні платформи підтримки IoT забезпечують інструменти для управління пристроями, збору даних та аналітики. Особливості основних хмарних платформ наведені в табл. 1.1 [46, 67].

Таблиця 1.1 – Основні можливості популярних хмарних платформ

Хмарна платформа	Особливості
AWS IoT Core	Масштабованість, інтеграція з AWS Lambda, безпека на рівні пристроїв
Microsoft Azure IoT Hub	Потужна аналітика, підтримка Edge Computing, інтеграція з Power BI
Google Cloud IoT	Машинне навчання, BigQuery для аналізу даних, управління цифровими двійниками
IBM Watson IoT	Аналітика, штучний інтелект, корпоративна безпека
Oracle IoT Cloud	Висока продуктивність, автоматизація виробничих процесів

Структура хмарної IoT-інфраструктури може бути різною залежно від застосування, але типовий підхід включає такі складові [98]:

- збір даних – пристрої IoT (сенсори, камери, контролери) надсилають дані через MQTT, HTTP або CoAP;
- обробка на периферії (Edge Computing) – частина даних фільтрується на шлюзах, щоб зменшити навантаження на хмару;
- передача в хмару – дані потрапляють у брокери повідомлень (Kafka, RabbitMQ) та сховища (AWS S3, Google Cloud Storage);
- аналіз та візуалізація – AI/ML алгоритми аналізують інформацію, визначають тренди та відхилення;
- автоматизація рішень – наприклад, у розумному місті система може автоматично перемикає світлофори для оптимізації трафіку.

Головні перевагами використання хмарної інфраструктури для IoT є такими [27, 100]:

- масштабованість – можна легко підключати нові пристрої та збільшувати обчислювальні ресурси;
- гнучкість – можливість обробки даних на різних рівнях (Edge, Fog, Cloud);
- автоматизація – AI та машинне навчання допомагають виявляти проблеми та приймати рішення в реальному часі;
- оптимізація витрат – компанії можуть використовувати різні хмарні моделі, сплачуючи лише за використані ресурси.

Але є проблеми, пов'язані з використанням хмарної інфраструктури [94]:

- великі обсяги даних – необхідно ефективне зберігання та обробка з використанням технології Big Data;
- затримки передачі – важливо оптимізувати процес маршрутизації на периферійних рівнях;
- безпека – захист від кібератак, автентифікація пристроїв, шифрування;
- навантаження на мережу – IoT-пристрої генерують величезний трафік, що може призводити до перевантаження каналів мережі зв'язку.

Отже, хмарна інфраструктура є критично важливою для IoT, адже вона забезпечує зберігання, аналіз та безпечне управління величезними обсягами даних. Для мінімізації затримок та підвищення ефективності треба використовувати периферійні обчислення, а для виконання оперативних транзакцій IoT бажано максимально задіяти вузли Edge Computing [61].

Розглянемо більш детально архітектуру хмарної інфраструктури підтримки Інтернету речей [36, 93].

Така архітектура є багаторівневою системою, яка включає в себе пристрої, мережні шлюзи, платформи обробки даних та аналітичні сервіси. Вона має забезпечувати ефективну комунікацію між великою кількістю пристроїв, швидку обробку великих обсягів даних і безпечне зберігання інформації та містить такі ключові рівні (рис. 1.2) [57]:

– рівень пристроїв (Perception Layer): фізичний рівень, що містить датчики, сенсори, пристрої збору даних, розумні об'єкти (Smart Devices) та контролери; на цьому рівні проводиться збір інформації із зовнішнього середовища (температура, вологість, освітлення, рух, рівень CO₂, GPS-координати тощо) та передаються дані до шлюзів або безпосередньо у хмару через мережні технології (Wi-Fi, Lora WAN, 5G, Bluetooth, Zigbee, NB-IoT); в якості пристроїв можуть використовуватися розумні лічильники електроенергії; сенсори руху у системах безпеки; GPS-трекери у транспорті, біометричні датчики у медицині тощо;

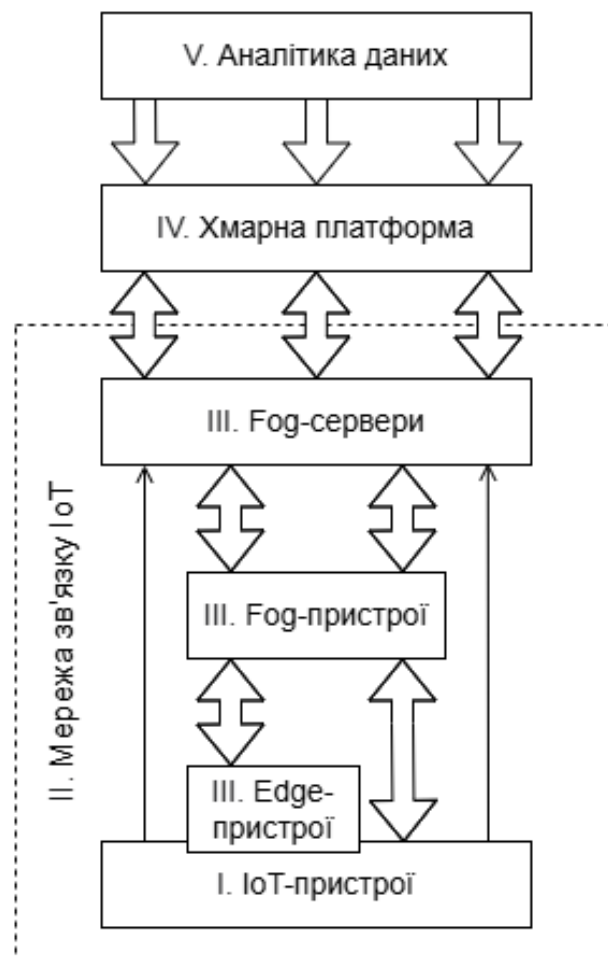


Рисунок 1.2 – Архітектура хмарної інфраструктури підтримки IoT

– рівень мережної взаємодії (Network Layer): відповідає за передачу даних від пристроїв до хмарної платформи через дротові та бездротові мережі; забезпечує з'єднання між пристроями та хмарними сервісами; фільтрацію та

маршрутизацію трафіку; використовує захищені канали зв'язку (VPN, TLS/SSL); задіює такі типи мереж як LPWAN (Low Power Wide Area Network) – Lora WAN, NB-IoT (для пристроїв з низьким енергоспоживанням), мобільні мережі 3G, 4G, 5G, локальні мережі Wi-Fi, Zigbee, Bluetooth, промислові протоколи – Modbus, OPC UA (у виробництві);

– рівень обробки даних (Edge/Fog Computing): використовується для попередньої обробки даних перед передачею в хмару та включає Edge Computing (обчислення на пристроях) та Fog Computing (обробка на проміжних вузлах, наприклад, на шлюзах або серверах біля джерела даних); здійснюються фільтрація та попередня обробка даних, щоб зменшити навантаження на хмару, збереження даних для автономної роботи у разі втрати з'єднання та виконання AI/ML алгоритмів на пристроях, наприклад, виявлення аномалій у датчиках без необхідності передачі всіх даних в хмару; застосовується у автономних транспортних системах, які використовують штучний інтелект для аналізу ситуації на дорозі, камерах відеоспостереження з вбудованими алгоритмами розпізнавання обличчя, розумні фабрики, де обробка даних відбувається локально для швидкого реагування на несправності;

– рівень хмарної платформи (Cloud Layer): центральний рівень, де відбувається управління пристроями, зберігання великих обсягів даних та їх подальший аналіз; на цьому рівні збираються та зберігаються дані у хмарних сховищах; проводиться управління підключеними пристроями (Device Management), здійснюється аналітика великих даних (Big Data Analytics), виявляються аномалії та прогнозування поломок (Predictive Maintenance); використовуються такі популярні хмарні платформи для IoT, як AWS IoT Core, Microsoft Azure IoT Hub, Google Cloud IoT, IBM Watson IoT, Oracle IoT Cloud, а для зберігання даних використовуються такі технології, як NoSQL бази даних (MongoDB, Firebase, Cassandra), об'єктне сховище (Amazon S3, Google Cloud Storage) та інструменти для потокової обробки (Apache Kafka, AWS Kinesis);

– рівень аналітики та застосувань (Application Layer): на цьому рівні відбувається візуалізація та використання IoT-даних у реальних бізнес-процесах, зокрема, проводяться аналітика та прогнозування (Big Data, AI, Machine Learning), інтеграція IoT-даних у бізнес-додатки (ERP, CRM, SCADA) та розгортання інтерфейсів для керування IoT-пристроями (мобільні додатки, веб-панелі); прикладами використання є розумні міста (контроль трафіку та освітлення вулиць), промисловість 4.0 (моніторинг обладнання та автоматичне прогнозування несправностей), охорона здоров'я (віддалений контроль за пацієнтами через пристрої, що носяться).

Комунікація між рівнями IoT-архітектури здійснюється через IoT-протоколи, такі як MQTT (Message Queuing Telemetry Transport) – малий розмір пакетів, підходить для пристроїв з низьким енергоспоживанням, CoAP (Constrained Application Protocol) – використовується в бездротових сенсорних мережах та HTTP/HTTPS – для традиційних веб-запитів [106, 107].

Отже, хмарна архітектура системи IoT – це складна багаторівнева система, яка забезпечує збір, передачу, обробку та аналіз даних від пристроїв IoT. Одним із найважливіших аспектів цієї архітектури є використання Edge Computing для отримання рішень в режимі реального часу та зменшення навантаження на туманний шар та хмарні центри обробки даних.

1.3 Аналіз існуючих методів дослідження компонент комп'ютерних систем підтримки Інтернету речей

Під час оцінювання ефективності комп'ютерних систем (КС) підтримки Інтернету речей, окремі компоненти котрих, що розташовані на граничному шарі, орієнтовані на режим реального часу (РРЧ) або наближений до нього, моделювання є найбільш доцільним і практично єдиним методом їх дослідження [64]. Моделювання як засіб для прийняття рішень під час проєктування таких систем дає змогу зробити висновок про принципову працездатність системи, оцінити її потенційні характеристики, встановити

їхню залежність від різних параметрів і змінних, визначити оптимальні значення параметрів, передбачити поведінку системи тощо [70, 71].

Крім того, модель, як специфікований опис системи, значно полегшує її подальший супровід (модифікацію, вдосконалення).

Моделювання систем базується на низці принципів, що забезпечують коректність, точність і достовірність одержуваних результатів, а саме [2, 107]:

- системний підхід під час розв'язання задач аналізу та синтезу;
- принцип ієрархічного багаторівневого моделювання;
- принцип множинності моделей.

В основі дослідження систем РРЧ з використанням моделювання лежить системний підхід, за якого її розглядають як складну систему, що володіє великою кількістю взаємодіючих елементів, властивостями неоднорідності, емерджентності, ієрархічності, багатофункціональності, стійкості, адаптації тощо [62, 81]. Системний підхід починається з визначення цілей для всієї системи, при цьому аналізуються дані та наявні знання про зовнішнє середовище, особливості побудови та вимоги до системи загалом, її компонентів і функціональної схеми їхньої взаємодії.

Принцип ієрархічного багаторівневого моделювання базується на такому описі досліджуваної системи, за якого її поведінку подають у вигляді сімейства моделей, кожна з яких описує процеси, що протікають у системі, з точки зору різних рівнів деталізації, що враховують низку характерних для цього рівня особливостей та параметрів [84].

Принцип множинності моделей полягає в можливості відображення багатьох різних систем і процесів за допомогою однієї й тієї самої моделі. Використання цього принципу дає змогу розробляти абстрактні моделі, шляхом грамотної параметризації котрих можливо досліджувати системи різних класів [66].

Моделі комплексів програм поділяють на структурні та функціональні. Структурні моделі характеризують складові частини системи, взаємозв'язки та інтерфейси між ними. Функціональні моделі описують поведінку системи у

формі, придатній для дослідження математичними методами. Відповідно до апарату, який покладено в їхню основу, функціональні моделі комплексів програм можна розбити на п'ять груп [64]:

- графові моделі;
- кінцеві автомати;
- моделі на основі ланцюгів Маркова;
- моделі з чергами;
- мережі.

В основі теорії графів [55], широко використовуваної при моделюванні комп'ютерних систем [8, 26], лежать поняття вершина і ребро (або дуга). Наприклад, у моделі для оцінювання часу виконання окремої транзакції вершини можуть представляти окремі застосунки, модулі або оператори, а дуги - передачу управління між ними. Знання значень часу виконання фрагментів транзакції та ймовірностей переходів у точках розгалуження дає змогу проводити кількісний аналіз процесів виконання програми загалом та окремих її частин.

Однак обчислення ймовірностей переходів у складних графах із достатнім ступенем точності надзвичайно трудомістке. Це пов'язано з тим, що функціонування системи розглядають як послідовність переходів у глобальному просторі станів системи, кількість яких стрімко зростає зі зростанням складності модельованої системи [29]. Витрати на обчислення ймовірностей навіть у послідовних програмах стають настільки великими, що це змушує або спрощувати модель, або вдаватися до евристичних алгоритмів, що знижує точність одержуваних оцінок.

Моделі, побудовані на основі теорії автоматів [15], представляють комплекс програм як кінцевий автомат, що перетворює дискретну інформацію і має кінцеву пам'ять.

Перевагами таких моделей є близькість до теорії графів і булевої алгебри, наочність і простота структурного аналізу. Однак вони придатні для дослідження суто послідовних і знову ж таки порівняно простих програмних

систем. Також, як і для графових моделей, для них потрібне знання ймовірностей переходів між локальними станами, обчислення яких для складних систем доволі проблематичне.

Математичні моделі систем, що використовують ланцюги Маркова [84], є перехідними між детермінованими і суто випадковими моделями. Для таких моделей ймовірність перебування системи в конкретному стані визначається, виходячи з відомостей про попередній стан, при цьому припускається, що система перебуває в стаціонарному режимі.

Для марковських моделей розроблено відповідний математичний апарат, що використовує диференціальні рівняння в приватних похідних та інтегрально-диференціальні рівняння. Він добре описує найпростіші ситуації, наприклад, коли вхідний потік вимог є пуасонівським потоком, а розподіл тривалості обслуговування вимоги підпорядковується експоненціальному розподілу. Однак реальні процеси, що протікають у складних системах, не є марковськими [87]. Способи ж апроксимації реальних процесів багатовимірними марковськими процесами або не дають потрібної точності, або пов'язані зі значними труднощами під час отримання формул.

Моделі з чергами базуються на теорії масового обслуговування [17, 28]. Основними компонентами в таких моделях є джерела, що генерують заявки на обробку, черги заявок і обслуговуючі пристрої. Аналітичні вирази для основних показників ефективності функціонування обслуговуючих пристроїв: коефіцієнт завантаження, пропускна здатність, час реакції тощо, можуть бути отримані з використанням різноманітних розширень апарату марковських ланцюгів: напівмарковські моделі, кусково-лінійні марковські моделі.

Застосування систем масового обслуговування (СМО) для розв'язання задач аналізу та синтезу систем РРЧ [8] передбачає низку припущень, пов'язаних із видом законів розподілу часу між надходженнями заявок на обслуговування та часу їх обробки. Як правило, використовують гіпотези про експоненціальний розподіл цього часу [10], що, як і в попередньому випадку, є неприйнятним для функціонального опису периферійних шарів хмарної

інфраструктури, які обробляють нестационарні, неоднорідні потоки оперативних транзакцій за заздалегідь визначеними алгоритмами.

Крім того, навіть за таких припущень одержувані для реальних систем формули настільки складні для практичного використання, що потребують застосування спеціальних методів спрощення (апроксимації) [103]. Однак це негативно позначається на точності одержуваних результатів.

Мережі, що використовуються для побудови функціональних моделей, є одним із сучасних засобів формалізації, призначених для моделювання та аналізу паралельних асинхронних розподілених систем. Найвідомішими формальними апаратами є мережі Петрі та їхні розширення [56]. Менш поширеними, що володіють подібними моделювальними можливостями, є мережні моделі зі стохастичною структурою [110], агрегатні моделі [122], моделі Керка [43], PS-мережі [44], поточкові моделі або М-мережі [41].

У моделях на основі мереж Петрі основні труднощі моделювання складних систем долаються за рахунок трьох основних положень [52]:

1. Один крок у розвитку системи визначається не на загальному або глобальному стані системи, а тільки на стані тих її компонентів, від яких він залежав, і де відбувалися якісь зміни. Тим самим досягається можливість опису паралелізму.

2. Моделюється управління. Модель абстрагується від тих змін, які не впливають на потік управління, або моделюються ці зміни за рахунок введення недетермінізму.

3. Модель графічна: всі можливі властивості задаються або графами, або розміткою на графах.

Доведено, що кінцеві автомати є окремим випадком мереж Петрі, причому наочність динаміки та композиційні можливості останніх значно вищі [58].

Класична мережа Петрі задається п'ятіркою [58]:

$$NP = \{P, T, F, H, M_0\},$$

де P – множина позицій, що мають властивість зберігання маркерів;

T – множина переходів, що мають властивість переміщення маркерів («спрацьовування»), при цьому $P \cap T = \emptyset$ і $P \cup T \neq \emptyset$;

$F : P \times T \rightarrow \{0, 1\}$ – функція передумови виконання подій;
 T , що визначає передумови виконання подій;

$H : T \times P \rightarrow \{0, 1\}$ – функція слідування множини позицій P і переходів T , що визначає постумови виконання подій;

$M_0 : P \rightarrow \{0, 1, \dots\}$ – початкове маркування (розмітка).

Функціонування мережі Петрі полягає в аналізі поточного маркування з метою визначення збудженого переходу (для якого виконуються передумови спрацьовування) і подальшого переміщення маркера (за правилами постумов).

В класичній інтерпретації мереж Петрі за існування кількох збуджених переходів вибір переходу, який у даній розмітці має спрацювати, ніяк не регламентується. Таким чином, поведінка мережі Петрі в загальному випадку має недетермінований характер, у якому відображаються лише причинно-наслідкові зв'язки між подіями, що моделюються спрацьовуванням окремих переходів.

З метою збільшення моделювальних можливостей мереж Петрі запропоновано значну кількість їхніх розширень [42].

Деяка їхня частина стосується структурних удосконалень (введення кратності дуги, заборонних дуг, пріоритетів переходу, гіпердуг, розфарбовування міток, навантажених переходів, групових переходів, вирішальних позицій. Для аналізу процесів управління в умовах чіткої та нечіткої вхідної інформації запропоновано розширення мереж Петрі, що визначає нечіткі переходи, нечіткі функції інцидентності та нечіткий вектор маркувань [119].

Особливої уваги заслуговують так звані часові мережі Петрі, у яких окремим переходам приписують час затримки їхнього спрацьовування [42]. Для простих часових мереж цей час або детермінований, або заданий

інтервалом. Відомий також підхід, пов'язаний із приписуванням часу перебування міток позиціям мережі [52].

У [122] визначено поняття стохастичної мережі Петрі (Stochastic Petri Nets, SPN), в якій час спрацьовування переходів розглядають як експоненціально розподілену випадкову величину. Розширення моделювальних можливостей SPN у [58] досягається допущенням миттєвих переходів із часом спрацьовування, що дорівнює нулю. Відповідний клас таких мереж названо узагальненими стохастичними мережами Петрі (Generalized Stochastic Petri Nets, GSPN). Слід зазначити, що аналітичне дослідження процесів у мережах SPN і GSPN зводиться (у разі їх обмеженості) до аналізу однорідних ланцюгів Маркова з безперервним часом.

Наступним кроком у розвитку апарату часових мереж Петрі стало введення так званих напівмарковських SPN (Semi-Markov SPN, SM-SPN) [42]. У цих мережах допустимо використовувати часові переходи з довільною функцією розподілу часу спрацьовування тільки у двох можливих випадках:

- 1) при збудженні такого переходу жоден інший перехід не може бути збуджений;
- 2) якщо одночасно збуджено кілька таких переходів, то спрацьовування одного з них спричиняє зникнення умов збудження інших.

Під час аналізу SM-SPN випадковий процес змін розміток можна представити напівмарковським процесом із фазовим простором, що являє собою множину досяжних розміток.

Наразі в багатьох галузях науки і техніки часові мережі Петрі знаходять широке застосування. Зокрема, з їхньою допомогою проводять моделювання комп'ютерних систем і мереж, їхнього програмного забезпечення, систем обробки даних [43, 44]. Крім того, останнім часом створено спеціальні програмні засоби моделювання часових мереж Петрі, котрі в низці випадків інтегровані в потужні засоби підтримки проєктних рішень [84].

Таким чином, з огляду на особливості проєктування та реалізації хмарних інфраструктур підтримки мережі IoT, а також вимоги, які

висуваються оперативними транзакціями IoT до компонентів комп'ютерних систем підтримки режиму реального часу, вибір теорії мереж Петрі в якості базового апарату математичного моделювання обчислювальних процесів видається найбільш виправданим.

Практика використання часових мереж Петрі, призначених для вивчення якісних властивостей і кількісних характеристик систем реального часу, демонструє актуальність розв'язання питань, пов'язаних із проблемами аналітичного дослідження випадкових процесів у таких моделях.

1.4 Огляд відомих моделей і методів оцінювання ефективності компонент комп'ютерних систем підтримки режиму реального часу

Основою детального аналізу часової ефективності компонент комп'ютерних систем підтримки режиму реального часу є метод декомпозиції, за допомогою якого визначається множина окремих станів, у яких може перебувати система під час свого функціонування [74]. Метою аналізу є отримання конкретних значень показників, що характеризують процес виконання транзакції, серед яких найчастіше обирають:

- числові характеристики (математичне очікування і дисперсія) часу виконання транзакції;
- ймовірність виконання транзакції за заданий час;
- об'єм пам'яті, необхідної для реалізації транзакції.

Методи оцінки продуктивності поділяються на аналітичні, математичного моделювання та натурних (напівнатурних) випробувань.

Аналітичні методи базуються на представленні процесів обробки транзакцій ймовірнісною моделлю [114], для якої визначаються ймовірності реалізації маршрутів обробки інформації по групах задіяних застосунків і тривалості їхнього виконання, які інколи розглядаються як деякі випадкові величини. Для аналітичного розрахунку, заснованого на СМО, приймають гіпотезу про експоненціальний закон розподілу цих величин [114], що дає

змогу одержати верхні оцінки тривалості очікування заявок, які надходять на обробку. Для нижніх оцінок використовують детерміноване значення часу обслуговування, що визначається за найвірогіднішим маршрутом реалізації.

У [115] запропоновано методи перетворення елементів процесу на алгебраїчні вирази, що описують час обробки та ґрунтуються на наборах оцінки часу виконання команд процесорів та ймовірності проходження відповідних ділянок транзакції. Зазначено, що визначення цих ймовірностей є окремою і значною проблемою, а використання гіпотез про вид розподілу часу виконання окремої підпрограми вимагає великої акуратності.

Для визначення щільності ймовірності часу виконання послідовно-паралельних програм, що описуються ГЕРТ-мережами [114], використовують характеристичні функції для різних блочно-ієрархічних рівнів графа програми, що описуються безперервними законами розподілу часу їх виконання. Обернення обчисленої характеристичної функції всієї мережі інтерполюється багаточленом Лагранжа в комплексній області [113].

Відомі й інші моделі, які використовують під час опису процесів функціонування транзакцій [9, 14, 75]:

- суміші команд;
- еталони і тести;
- стохастичні моделі;
- трасові моделі.

Суміші застосовуються, як правило, при імітаційному моделюванні процесу виконання транзакцій, і являють собою послідовності команд, типових для застосунків, що реалізуються на досліджуваних обчислювальних засобах. Склад суміші визначається через функцію розподілу частоти виконання команд за певною схемою [72]. Цю функцію отримують методами математичної статистики, наприклад, за допомогою регресійного аналізу. Найвідомішими є стандартні суміші Гібсона і Флінна, що застосовуються для наукових і технічних застосунків.

Цей метод моделювання використовують, коли проведення аналізу реальних транзакцій неможливе. Крім низької точності, такий підхід має істотний недолік, який полягає в тому, що він не враховує внутрішні зв'язки між окремими складовими застосунку. Це не дозволяє використовувати такі моделі для точного й адекватного представлення процесу реалізації оперативних транзакцій IoT на вузлах граничного шару.

Еталони і тести замінюють реальні застосунки прикладними процесами, якими виступають існуючі окремі аналогічні застосунки або стандартизовані пакети тестів (benchmarks), що базуються на прикладних програмах широкого кола користувачів [90]. Цей метод здебільшого використовують для порівняння різних існуючих систем у плані використання обчислювальних ресурсів.

Однак застосування будь-якого стандартного тесту не може замінити виконання завдань у реальних умовах. Використання ж як еталона компонент комп'ютерних систем підтримки режиму реального часу в повному обсязі пов'язане з великими витратами і в кращому разі не відрізняється від натурного експерименту, а практично і просто неможливе.

Стохастичні моделі описують поведінку програм за допомогою набору випадкових величин, що визначають динаміку використання процесора, пам'яті, каналів передачі даних тощо. Розподіли цих випадкових величин підбираються на основі статистичного аналізу результатів вимірювань потоку запитів на відповідні ресурси реально діючої мережі. Для збору таких даних використовують спеціальні засоби - апаратні та програмні (або гібридні) монітори [73].

Стохастичні моделі являють собою найбільш компактні, проте недостатньо точні описи транзакцій. Під час їхньої побудови використовується гіпотеза про незалежність різних потоків запитів на ресурси, а отже, не враховуються зв'язки з управління між задіяними процесами, які реалізуються в туманному шарі. Досить важким завданням є визначення форми та параметрів розподілів випадкових величин. Крім того, статистичні

дані завжди несуть на собі відбиток конкретної системи, на якій вони отримані, і можуть бути викривлені вимірювальними моніторами, оскільки вони самі використовують ресурси системи.

У трасових моделях процеси виконання транзакцій представляють у вигляді множини впорядкованих записів, що містять дані про події, які визначають переходи між станами обчислювального процесу. Ці дані, зібрані за допомогою моніторів, після спеціального оброблення використовують як функціональну модель процесу [18]. Важливою перевагою цього підходу, на відміну від розглянутих вище, є те, що він зберігає всі взаємозалежності між різними процесами під час функціонування системи. Однак, як і в попередньому підході, вимірювання мають системно-залежний характер і можуть впливати на роботу самої системи. Досить трудомістким є процес обробки траси та підготовки її до використання у вигляді моделі. Трасувальна модель досить об'ємна і вузькоспрямована на систему збору трас і модель аналізованої системи.

Таким чином, для отримання достовірних оцінок продуктивності необхідне обґрунтоване застосування таких моделей і методів їх дослідження, що дають змогу:

- врахувати особливості реалізації оперативних транзакцій IoT у туманному шарі хмарної інфраструктури, а також вимоги, які висуваються до них;
- описати часові процеси асинхронної паралельної обробки інформації з необхідним ступенем адекватності;
- застосувати ефективні методи як якісного, так і кількісного аналізу моделей, що дають змогу отримати результати з необхідним рівнем точності.

1.5 Постановка наукової задачі та окремих завдань дослідження

Розглянуті вище практичні проблеми та особливості проектування, розробки та експлуатації окремих компонент комп'ютерних систем підтримки

режиму реального часу, що розміщені на граничному шарі та задовольняють заданим показникам якості, показали необхідність подальшого розвитку теоретичних методів моделювання та аналізу процесів виконання оперативних транзакцій у хмарній інфраструктурі Інтернету речей.

Серед відомих апаратів формалізованого опису функціонування складних комп'ютерних систем та їхніх компонентів найбільш переважними виглядають часові мережі Петрі, безпосередньо розроблені для моделювання поведінки паралельних асинхронно взаємодіючих процесів. Водночас недостатньо опрацьованими залишаються питання, пов'язані з методами створення мережних моделей і обґрунтуванням їхньої адекватності. Крім того, потребують подальшого розвитку аналітичні методи отримання ймовірнісно-часових характеристик випадкових процесів, що протікають у мережах Петрі. Такі характеристики при моделюванні компонент комп'ютерних систем підтримки режиму реального часу дадуть змогу оцінити процес зміни станів системи та їхню продуктивність – найбільш важливий параметр, що впливає на більшість інших показників якості систем реального часу. Окремого розгляду потребують і проблеми забезпечення необхідного рівня продуктивності в процесі паралельної обробки оперативних транзакцій та визначення моментів нестачі ресурсів граничного шару.

Таким чином, виникає необхідність розв'язання науково-технічної задачі, яка полягає в підвищенні ефективності використання обчислювальних ресурсів граничного шару при обробці оперативних транзакцій шляхом удосконалення методів синтезу окремих кластерів граничного шару Інтернету речей, що об'єднують низку територіально близьких пристроїв IoT. При цьому мають враховуватися як вимоги, що необхідні для виконання оперативних транзакцій, так і показники використання наявних ресурсів граничного шару Інтернету речей. Таке розв'язання включає:

- розроблення методу побудови функціональних моделей кластерів граничного шару комп'ютерної системи підтримки Інтернету речей;
- розроблення методу формування кластера граничного шару

комп'ютерної системи підтримки Інтернету речей;

- розроблення методу короткострокового прогнозу часових параметрів кластера граничного шару комп'ютерної системи підтримки Інтернету речей;

- проведення порівняльної оцінки розроблених та існуючих методів синтезу кластера граничного шару комп'ютерної системи підтримки Інтернету речей.

Висновки за розділом 1

1. Проведений аналіз сучасного стану розвитку концепції Інтернету речей, свідчить про якісні зміни принципів побудови комп'ютерних систем підтримки IoT та методів обробки і зберігання інформації, що надходить з датчиків і пристроїв IoT. При цьому темпи розвитку IoT сприяють появі нових викликів та обмежень, серед яких особливо чутливими є проблеми, пов'язані зі збільшенням кількості пристроїв, що створює суттєве додаткове навантаження на мережу та призводить до непередбачуваних затримок і збоїв, а також високого рівня енергоспоживання пристроїв IoT.

2. Аналіз хмарної інфраструктури підтримки мережі IoT показав, що внаслідок збільшення навантаження та посилення вимог до оперативних транзакцій IoT виникла необхідність перенесення частини обчислень на периферійні шари: хмарний і граничний. При цьому граничний шар забезпечує найменший час обробки, але має обмежені обчислювальні ресурси, тому виникає потреба у задіянні вузлів туманного, а іноді й хмарного шарів відповідної архітектури. Рішення цієї проблеми визначає необхідність розвитку теоретичної бази дослідження процесів виконання оперативних транзакцій IoT у граничному шарі хмарної інфраструктури та повинно ґрунтуватися на принципах системного підходу до аналізу поточних запитів і за можливістю найбільш повного врахування суттєвих факторів, що впливають на показники ефективності всієї системи загалом.

3. Для отримання кількісних оцінок продуктивності компонент комп'ютерних систем, що підтримують режим реального часу для оперативних транзакцій IoT, з огляду на особливості їхнього проєктування та реалізації, необхідне використання моделей комплексів програм, що володіють розширеними можливостями з аналізу асинхронних паралельних процесів. Таким вимогам задовольняють моделі програмних систем реального часу, що використовують як апарат формалізації часові мережі Петрі.

4. Дослідження якісних властивостей моделей компонент комп'ютерних систем, що підтримують режим реального часу, і кількісних ймовірісно-часових характеристик процесів оперативних транзакцій IoT передбачає одержання необхідних значень із заданим рівнем точності та достовірності. Це зумовлює необхідність дослідження питань, пов'язаних з оцінкою адекватності моделей, що описуються мережами Петрі, а також розвитком методів аналізу часових мереж Петрі.

Зазначені питання потребують окремого опрацювання і розглядаються в наступних розділах роботи.

РОЗДІЛ 2

РОЗРОБКА МЕТОДУ ПОБУДОВИ ФУНКЦІОНАЛЬНИХ МОДЕЛЕЙ КЛАСТЕРІВ ГРАНИЧНОГО ШАРУ

2.1 Визначення послідовності етапів дослідження фрагментів комп'ютерних систем реального часу

Однією з відмінностей комп'ютерних систем реального часу (КСРЧ) є наявність жорстких вимог до обробки інформації в реальному часі. Тому задача оцінки часової ефективності КСРЧ є особливо важливою [8]. При цьому показники продуктивності КСРЧ можуть виявитися недостатніми при вирішенні необхідної сукупності задач у реальних умовах, що вимагають змін як і структури, так і способів реалізації програмних та й апаратних засобів. У зв'язку з цим застосування різноманітних методів аналізу комп'ютерних систем необхідне на всіх фазах їх життєвого циклу, визначеного як сукупність окремих етапів робіт, починаючи з моменту вирішення питання щодо розробки КСРЧ і продовжуючи при його експлуатації [40]:

- системний аналіз та специфікація вимог;
- проєктування;
- розробка (реалізація);
- тестування (випробування);
- супровід.

Практика показує, що в загальному випадку фазові межі вищезазначених етапів нечіткі, фази є взаємозалежними, а процес експлуатації має ітераційний характер [45], причому проєктні зміни частіше відбуваються паралельно з реалізацією, що зумовлюється неможливістю повного і детального опису як вихідних вимог, так і самої системи [38]. Істотну частину всього життєвого циклу КСРЧ (рис. 2.1) складають верифікація та валідація (перевірка правильності).

Враховуючи зростаючі вимоги до якості верифікації програмних системи реального часу, виникає необхідність найбільш об'єктивної та повної оцінки, що ґрунтуються на формуванні системи відношень до КСРЧ:

- між вимогами до системи і вимогами до КСРЧ;
- між вимогами до КСРЧ та функціями КСРЧ;
- між функціями КСРЧ та методиками випробування (тестування);
- між випробуванням та звітами з верифікації.

При цьому через недостатні, неповні або помилкові специфікації систем відношень комп'ютерна система, навіть та, що пройшла етап валідації та тестування, в процесі експлуатації може показати незадовільну продуктивність. Це відбувається внаслідок того, що при створенні й виконанні тестів необхідно враховувати паралелізм процесів, що відбуваються в комп'ютерній системі, можливість переривань ходу обчислень, а також час надходження вхідних даних як додатковий параметр, що робить задачу інтерпретації тестових результатів досить складною для виконання [53]. Окрім цього, високий рівень вимог до надійності системи реального часу може вимагати неприйнятно великих об'ємів тестування.

З метою підвищення впевненості в коректності системи [89] висувається вимога паралельного (з розробкою самих програм) створення системи тестування. Таким же паралельним процесом, що дозволяє вчасно виявляти вагомі особливості поведінки системи має бути створення обґрунтованої математичної моделі КСРЧ та її подальший аналіз на різноманітних стадіях життєвого циклу (рис. 2.2).

Вихідними даними для моделювання КСРЧ є [85]:

- характеристики КСРЧ (структурні, ймовірні та часові);
- характеристики фізичного середовища (апаратні компоненти комп'ютерної системи та мережі зв'язку між ними);
- характеристики операційного середовища для системи організації обчислювального процесу, що забезпечує виконання програм користувачів (ОС, бібліотеки, файлова система тощо);

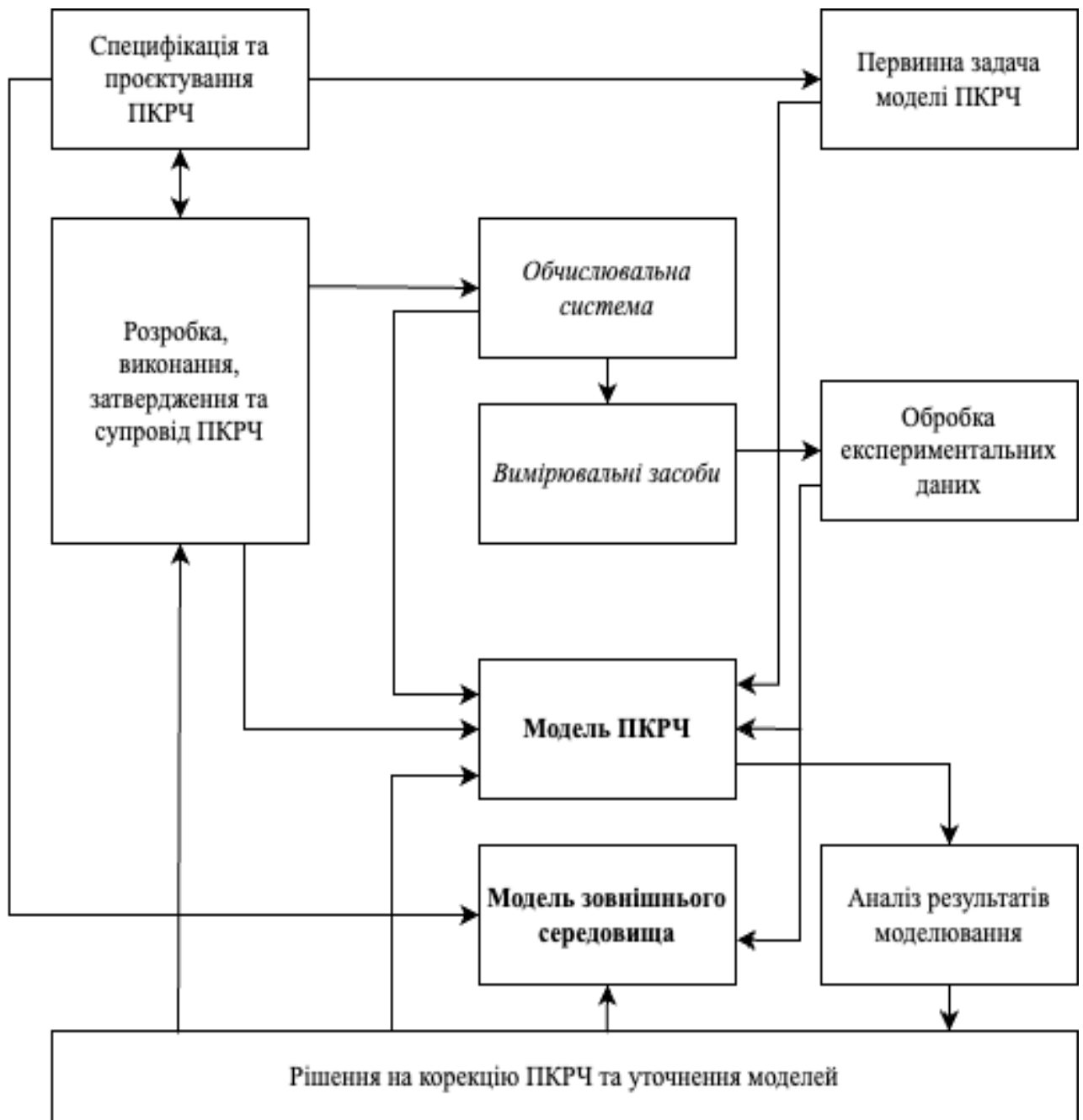


Рисунок 2.2 – Схема модельного аналізу продуктивності КСРЧ

- характеристики зовнішнього середовища (параметри потоків різнотипних заяв на обслуговування тощо).

При оцінці продуктивності в якості конкретних значень характеристик КСРЧ використовують статичні дані або задаються опорні значення, котрі уточнюються в ході розробки та тестування конкретних програм (рис. 2.3) [8].

Основна перевага використання формальної моделі КСРЧ полягає в можливості передбачення її поведінки [124]. Наприклад, до практичної

реалізації багатопотокових систем можуть бути реалізовані попередні оцінки їх часової ефективності, відсутності глухих кутів та кількості використаних ресурсів (особливо процесора та пам'яті).

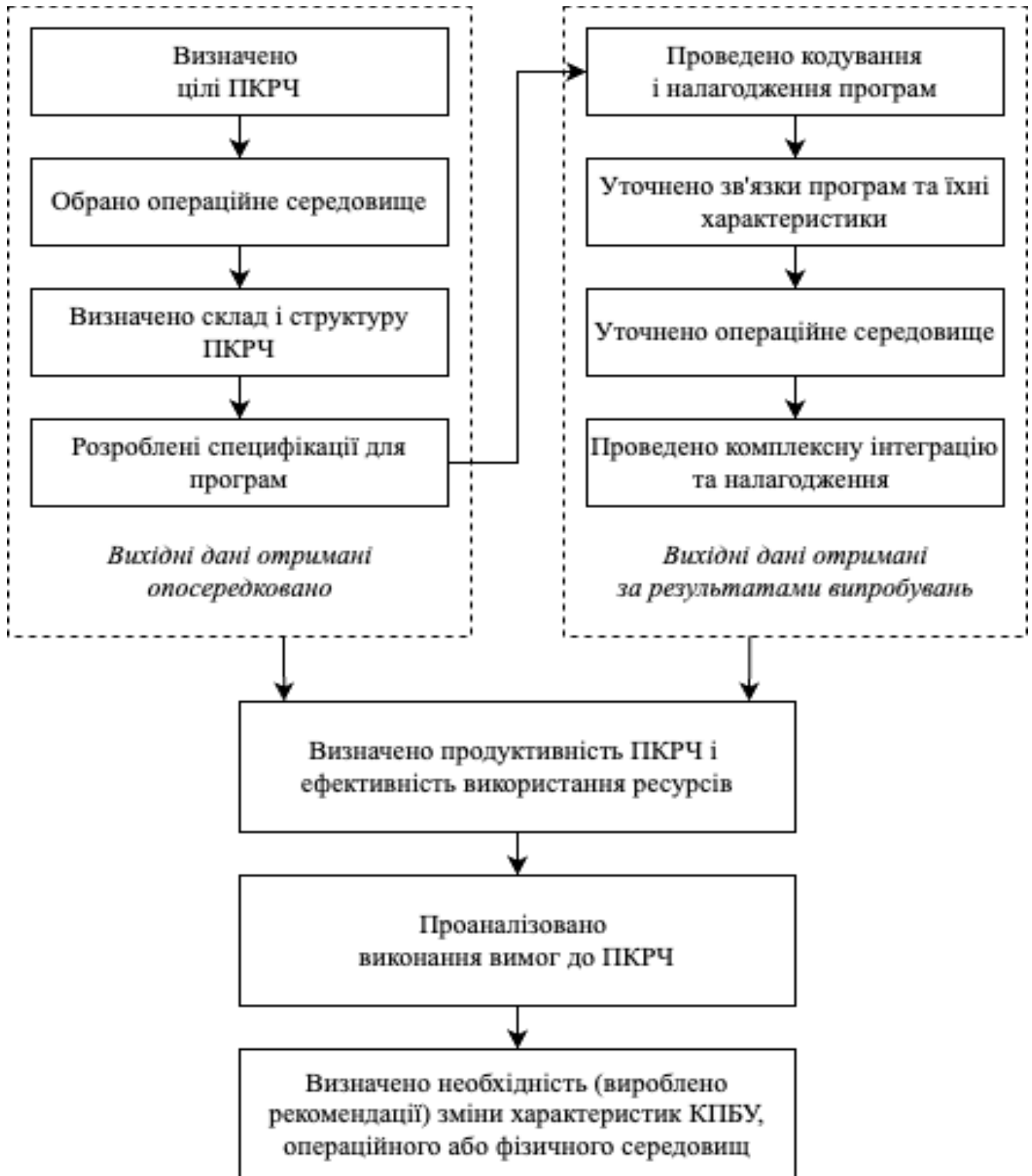


Рисунок 2.3 – Етапи аналізу продуктивності КСРЧ під час його розробки

Критичні за часом обробки компоненти можуть бути виявлені до того, як вони стануть вузьким місцем реальної системи, після чого під час їхнього кодування можуть бути обрані відповідні засоби.

Ступінь надмірності в системах, котрі мають бути ефективними при наявності збоїв, відмов та навіть помилок в самих програмах, також визначається при аналізі моделі, причому може бути виявлено як недостатня, так і перевищена надмірність.

Дослідження моделі КСРЧ на ранніх стадіях проєктування дозволяє [92]:

- визначити чи буде ця система задовольняти закладеним вимогам, а також можливість реалізації на визначеній технічній базі;
- здійснити контроль цілісності системи;
- визначити підсистеми, котрі можуть працювати паралельно для досягнення оптимальної продуктивності.

На більш пізніх стадіях структурні методи аналізу моделі комп'ютерної системи та її окремі компоненти або сукупності компонентів застосовуються для визначення таких умов [92]:

- коректність реалізації;
- досяжність всіх віддалених реалізацій;
- виходи з усіх проблемних програмних циклів та тупиків (відсутність глухих кутів);
- коректне звільнення всіх зайнятих ресурсів.

Метою додаткових методів аналізу може бути перевірка таких моментів:

- правильності послідовності обробки даних;
- отримання результатів в заданий інтервал часу, що є суттєвим для СРЧ або систем, наближених до режиму реального часу;
- заданих значень ймовірності збою (відмови) системи та втрат (наприклад, часових), пов'язаних з їх обробкою.

Нарешті, моделювання комп'ютерних систем реального часу дозволяє вирішити питання, що стосується адаптивності програм (здатності до перелаштування при зміні зовнішніх умов або функціональних вимог) та

ефективного реінжинірингу (здатності до внесення модифікацій з метою модернізації) [101].

Таким чином, забезпечення якості КСРЧ та отримання необхідних кількісних показників, в тому числі продуктивності, вимагає застосування адекватної формальної моделі та відповідних методів її аналізу на всіх етапах життєвого циклу комп'ютерної системи реального часу. При цьому така модель має володіти підвищеним ступенем гнучкості, розширюваності та модифікованості.

2.2 Розробка методу побудови функціональної моделі кластера граничного шару

Загальним підходом до аналізу складних систем є оцінка поведінки її моделі під впливом події, що генеруються моделлю зовнішніх впливів. Стосовно процесів, що протікають у комп'ютерній системі, множина вхідних завдань (заявок на обслуговування), оброблюваних за певними правилами, називається робочим навантаженням [111].

Залежно від цілей дослідження під робочим навантаженням можуть розумітися як вхідні дані, так і транзакції IoT. Для всіх відомих методів моделювання комп'ютерних систем досить складним є опис моделі робочого навантаження, яка є сумісною і відповідає за ступенем деталізації моделі системи. При цьому необхідно розв'язувати окреме завдання, пов'язане з оцінкою їхньої точності й особливо адекватності.

Наприклад, найбільш широко розповсюджені графові моделі будуються на основі вивчення статичних структур, а не реальної динаміки поведінки процесів, і по суті відображають погляд дослідника або розробника на ймовірну поведінку системи. Внаслідок цього при моделюванні існує ймовірність врахування маршрутів, котрі насправді не реалізуються [111], що знижує рівень довіри до достовірності одержуваних під час моделювання результатів. Натомість найбільш представницькі трасові моделі, що базуються

на вимірюваних даних реальних систем надмірно об'ємні та недостатньо гнучкі.

2.2.1 Опис методу побудови функціональних моделей

Для усунення наведених недоліків пропонується метод побудови мережних моделей, котрий може застосовуватись для створення як моделі робочого навантаження, так і моделі КСРЧ [8, 11].

На відміну від інших він дозволяє компактно, на основі єдиного апарату формалізації, описати різноманітні часові процеси, що відбуваються в комп'ютерній системі.

Метод ґрунтується на представленні трас подій, що спостерігаються в системі, стохастично-детермінованими часовими мережами Петрі (СДЧМП) [43]. Це розширення мереж Петрі характеризується використанням випадкових величин, які описуються законами розподілу загального вигляду для визначення затримки між активацією та спрацюванням переходу, та введенням поняття групових переходів, що описують повну групу несумісних подій.

Кожному переходу в складі групи відповідає ймовірність відповідної події. Реалізація такого групового переходу дозволяє організувати недетермінований вибір напрямку розвитку процесу.

Пропонується метод побудови стохастично-детермінованої часової мережі Петрі з використанням трасувальних даних, що складається з п'яти етапів, наведених на рис. 2.4.

Розглянемо детальніше етапи запропонованого методу.

1. На основі концептуальної моделі функціонування комп'ютерної системи здійснюється декомпозиція КС на N типів елементарних аналізованих часових подій T , вагомих з точки зору цілей моделювання, які описуються множинами:

$$T^{(поточне)} = \{t_1, t_2, \dots, t_N\}.$$

Для побудови моделі робочого навантаження в якості подій можуть розглядатись надходження різнорідних заявок на обробку, для моделі КСРЧ – етапи обробки оперативних транзакцій за заданими алгоритмами.

2. За допомогою системи вимірювання організовується збір інформації щодо послідовності, частоти та часу виконання елементарних подій.

3. На основі аналізу отриманої траси виділяються M станів комп'ютерної системи, які визначаються передісторією подій, що відбулися:

$$S = \{s_1, s_2, \dots, s_M\}, s_i = (t_j, \dots, t_k).$$

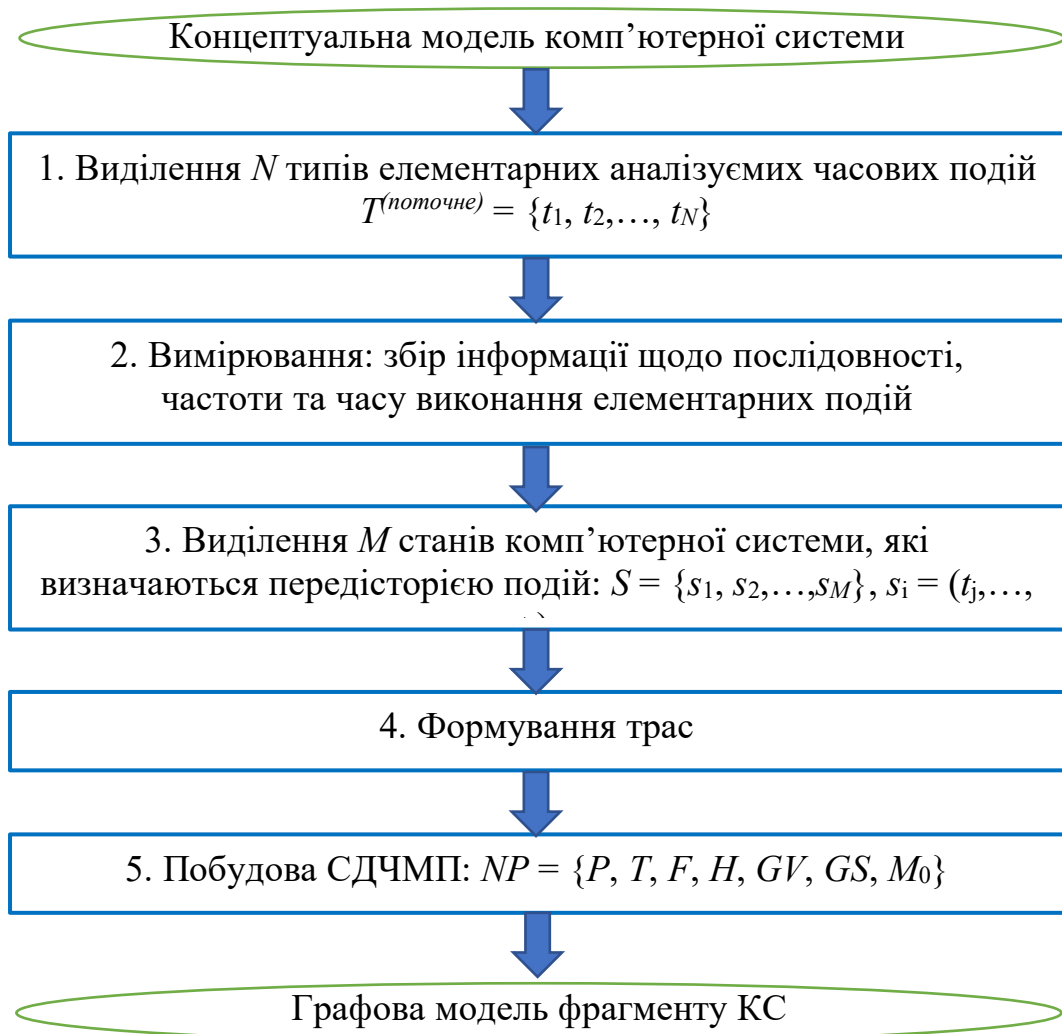


Рисунок 2.4 – Структура методу синтезу моделі фрагменту комп'ютерної системи підтримки режиму реального часу з використанням трасувальних даних

4. Траса є спрямованим виваженим подієвим графом, вершинами T якого є виділені стани, а дуги з кожною вершиною відображають послідовність переходів від одного стану до іншого. Вага кожної дуги цього подієвого графу визначається як статистична ймовірність відповідного переходу. Можна відзначити, що таке представлення подієвого графу відповідає марковському ланцюгу, яке виражає структуру управління процесу, що моделюється.

5. На основі подієвого графу будується СДЧМП такого вигляду:

$$NP = \{P, T, F, H, GV, GS, M_0\},$$

де $P = \{p_1, p_2, \dots, p_L\}$ – множина позицій умов виконання подій, що визначається наявністю входних дуг подієвого графу;

T – множина переходів, що відповідає множині подій;

$F : P \times T \rightarrow \{0, 1\}$ – функція передумови множини позицій P і переходів T , що визначається передумовами виконання подій, тобто входними дугами подієвого графу;

$H : T \times P \rightarrow \{0, 1\}$ – функція слідування множини позицій P і переходів T , що визначається вихідними дугами подієвого графу;

$GV : T \times R_V = f(Z_V)$ – функція відповідності між множиною переходів T і множиною стохастичних величин часу виконання подій R_V , розподілених за випадковим законом Z_V ; вид розподілу і його параметри визначаються на основі даних траси з використанням методів математичної статистики;

$GS : T \times P_S \rightarrow [0, 1]$ – функція відповідності між множиною групових переходів T , що утворюють повну групу несумісних подій, і множиною ймовірностей їхнього спрацьовування P_S , що дорівнюють вагам подієвого графа;

$M_0 : P \rightarrow \{0, 1, \dots\}$ – початкове маркування;

$$P \cap T = \emptyset \quad \text{та} \quad P \cup T \neq \emptyset.$$

Вага кожної дуги дорівнює 1, тобто отримана мережа є ординарною.

Необхідними умовами застосування розглянутого підходу до побудови

мережних моделей компонент реального часу у граничному шарі інфраструктури підтримки IoT є [120, 123]:

- можливість отримання точної інформації про послідовність і тривалість подій, що відбуваються у відповідному фрагменті граничного шару, що припускає наявність, по-перше, реалізованих у програмних кодах самих застосунків системи (або, принаймні, їхніх основних частин), по-друге, спеціальних засобів вимірювання для отримання таких оцінок - вимірювальних моніторів;
- припущення про відсутність помилок і збоїв під час виконання застосунків або наявність спеціально розроблених механізмів, що усувають їхні наслідки;
- припущення щодо стаціонарності імовірностей переходів системи за обраними станами.

Крім того, для точного визначення виду і параметрів випадкових величин, що характеризують процеси виконання оперативних транзакцій IoT, слід забезпечити відповідний обсяг трасувальних даних і певні умови їх збору. Очевидно, що для оцінки максимальної продуктивності фрагменту необхідно проводити вимірювання при максимальному (визначеному технічним завданням або іншими умовами) навантаженні на комп'ютерну систему.

2.2.2 Приклад застосування методу побудови функціональних моделей

Розглянемо приклад застосування запропонованого методу для моделювання фрагмента найпростішої комп'ютерної системи.

Нехай для оцінювання продуктивності фрагменту компоненти реального часу граничного шару хмарного середовища виділено 5 груп застосунків, що обробляють однотипові оперативні транзакції IoT. Метою аналізу є отримання ймовірнісно-часових характеристик виконання кожної групи застосунків, які пов'язують із відповідними подіями з множини

$T = \{t_1, t_2, \dots, t_5\}$. Припустимо, що за допомогою спеціальних вимірювальних засобів проведено оцінку часу, частоти та послідовності виконання в даному фрагменті відповідних застосунків, причому отримана траса подій має такий вигляд:

$$t_2, t_3, t_1, t_2, t_3, t_4, t_2, t_1, t_5, t_2, t_3, t_4.$$

Зробимо спрощувальне припущення, що настання наступної події залежить тільки від поточної, тобто дана послідовність подій може бути представлена марковським ланцюгом. Це означає, що стани системи визначаються історією процесу, тривалістю в одну подію, а тому множина S станів програмної системи збігається з множиною T .

На основі аналізу траси будується подієвий граф (рис. 2.5).

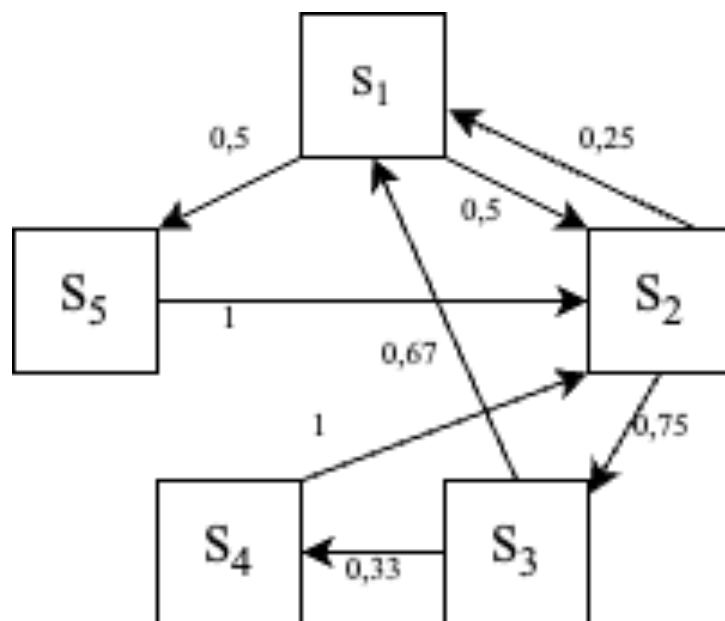


Рисунок 2.5 – Вигляд подієвого графа для траси, що аналізується

Внаслідок відсутності передісторії всі дуги графа, що виходять з i -ої вершини, означають подію t_i , тому відповідні позначення подій на дугах опущені.

Дуги графа з імовірністю, що дорівнює одиниці, які виходять із вершин s_4 і s_5 , означають, що відповідні події реалізуються звичайними часовими

переходами, а решта подій – груповими переходами. Кількість переходів у групі визначається кількістю вихідних дуг, а ймовірності їхнього спрацьовування – вагою відповідних дуг, причому сума ймовірностей подій групового переходу за принципом побудови подієвого графа дорівнює одиниці.

Відповідно до правил перетворення подієвого графа на стохастично-детерміновану мережу Петрі, з огляду на апіорну інформацію, що подія t_2 означає початок обробки заявки, яка надходить на обслуговування, будують мережу (рис. 2.6) з початковим маркуванням у вершині P_2 .

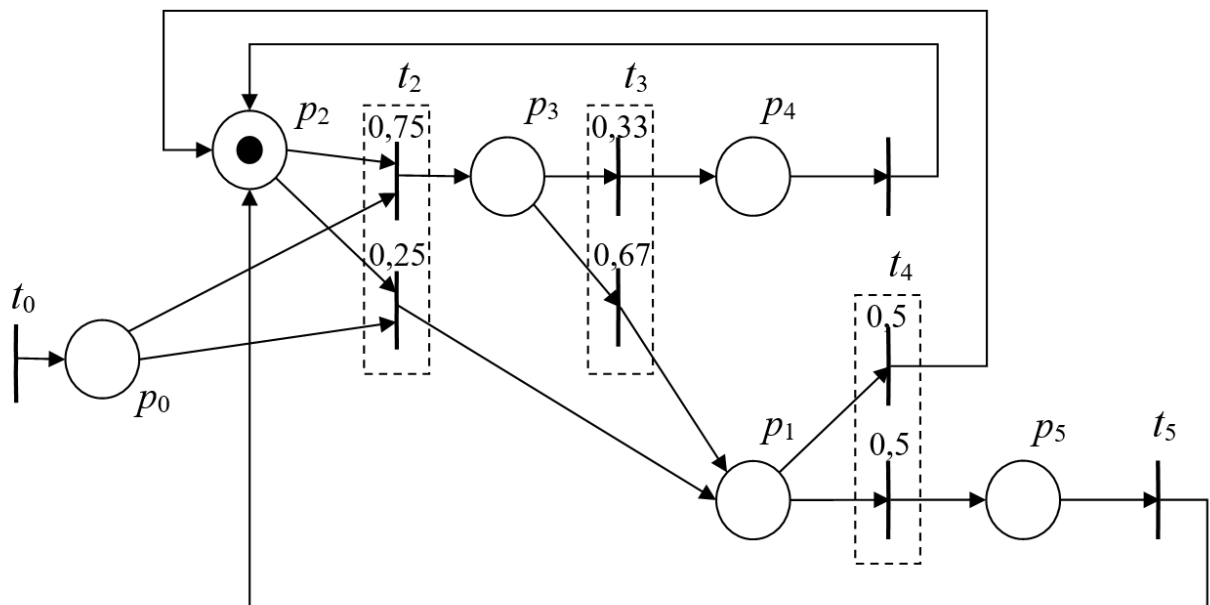


Рисунок 2.6 – Стохастично-детермінована часова мережа Петрі, що моделює функціонування групи оперативних транзакцій IoT

Додатково введені позиція p_0 , що відповідає умові «заявка надійшла», і перехід t_0 , що генерує заявки через випадковий час, у даному прикладі виконують роль найпростішої моделі робочого навантаження та дозволяють проаналізувати поведінку програмного комплексу за різних характеристиках потоку заявок, що надходять на обробку.

Без додатково введених позиції для умови «заявка надійшла» і переходу, що генерує заявки через випадковий час, стохастично-детермінована часова

мережа Петрі є ординарною активною мережею Петрі, що дає змогу досліджувати замкнуті системи, до яких можуть бути приведені, наприклад, системи зі сталим режимом завантаження.

2.2.3 Переваги та недоліки запропонованого методу побудови функціональних моделей

У зв'язку з вищевикладеним, основний недолік, притаманний побудованим за запропонованим методом мережним моделям, – це їхня певна спрямованість на ті обчислювальні системи, на яких здійснюється збір вимірювальної інформації.

Перевагами запропонованого методу є такі:

1. Можливість використання різних рівнів деталізації при вивченні різних властивостей КСРЧ. Оскільки переходи мережі Петрі відповідають етапам обробки інформації, кожен із них, у міру проектування конкретних компонент (а потім і більш детальних компонент), може уточнюватися шляхом побудови відповідних підмереж. Тим самим реалізується принцип ієрархічного моделювання, що дає змогу підвищити розуміння модельованої системи і зосередитися на вивченні її обраних властивостей.

2. Застосування єдиної нотації для опису як моделі робочого навантаження, так і моделі самого компоненту. Узгодженість підмоделей (на рівні апарату формалізації) дає змогу розробляти їх окремо і надає широкі можливості з дослідження поведінкових властивостей КСРЧ за різних умов і станів зовнішнього середовища.

3. Використання найрізноманітніших видів розподілень випадкових величин, що характеризують тривалість обраних для аналізу подій, що дає змогу врахувати в моделі основні фактори, що впливають на процес виконання КСРЧ у багатозадачному середовищі.

Час виконання оперативної транзакції та задіяних застосунків залежить від таких факторів [121]:

- обсягу і значень вихідних даних;
- труднощів (часової складності) алгоритму обчислень і способу його практичної реалізації;
- характеристик апаратних засобів, задіяних у вузлах граничного шару комп'ютерної системи – тип і структура процесора, обсяг і час доступу до оперативної пам'яті, зовнішніх пристроїв тощо;
- характеристик програмних засобів граничного шару комп'ютерної системи.

Значний вплив на цей час мають і особливості організації обчислювального процесу в граничному шарі комп'ютерних систем. Дійсно, псевдопаралельно виконувані процеси перебувають у стані постійної конкуренції не тільки за процесор, а й за оперативну пам'ять, що може спричиняти додаткові часові витрати активної програми на завантаження необхідної частини виконуваного застосунку або даних із файлу підкачки. Часову неоднозначність можуть викликати і звернення до стандартних процедур та до інших підпрограм, що динамічно завантажуються (наприклад, DLL-файлах).

Оператори розгалуження, циклу і безумовного переходу також можуть істотно впливати на час виконання застосунків, залежно від того, чи перебувають вони в межах конвеєра процесора, його кеш-пам'яті першого чи другого рівня або у звичайній оперативній пам'яті [7, 11].

Унаслідок впливу розглянутих чинників час виконання конкретної оперативної транзакції (або окремої її частини) навіть на одному й тому самому обчислювальному вузлі КСКП являє собою випадкову величину специфічного закону розподілу, відмінного від експоненціального або нормального розподілу [7, 15].

Таким чином, використання розподілів загального виду в СДЧМП дає змогу найточніше охарактеризувати особливості виконання транзакцій КСРЧ, а запропонований метод синтезу мережних моделей – автоматизувати процеси обробки вимірюваних даних і забезпечити:

- єдиний підхід до формального опису моделей робочого навантаження і фрагменту граничного шару комп'ютерної системи, що дає змогу компактно й наочно представити комплексну модель за допомогою одного з найпотужніших апаратів моделювання – часових мереж Петрі, які оперують із випадковими величинами загального виду;
- врахування передісторії розвитку процесів, що дає змогу виявити взаємозалежності між подіями, що відбуваються в обчислювальній системі;
- можливість визначення не тільки якісних (причинно-наслідкових), а й кількісних (насамперед часових) характеристик процесів, що аналізуються, що дає змогу моделювати різноманітні системи реального часу, зокрема фрагменти граничного шару КСРЧ.

Крім того, подальше дослідження побудованих моделей, як буде показано в розділі 3, може бути проведене не тільки шляхом безпосередньої імітації, а й аналітичним способом.

2.3 Розробка методів оцінки адекватності мережних моделей фрагментів граничного шару

Однією з основних вимог, що висуваються до моделей, є їхня адекватність реальним системам, під якою розуміють здатність правильно передбачати різноманітні властивості процесів, що відповідають дійсності [16]. Процеси функціонування реальних комп'ютерних систем реального часу неможливо описати повно й детально, що зумовлено їхньою значною складністю. Будь-яка модель, унаслідок своєї формальності, адекватна лише за деяких умов.

Підвищення ступеня адекватності можна досягти завдяки використанню різних рівнів деталізації, що залежать від особливостей структурно-функціональної організації системи та задач дослідження, врахування додаткових факторів, що впливають на досліджуваний процес, а також уточнення моделі під час проєктування системи.

Питання оцінювання адекватності моделей і достовірності одержуваних за їхньою допомогою результатів із точки зору важливості та ступеня врахування факторів, що використовуються при моделюванні, докладно розглянуто в [24]. У цій дисертаційній роботі запропоновано метод оцінювання адекватності, заснований на оцінюванні ступеня відповідності поведінки моделі системі, інакше кажучи, адекватності опису динаміки досліджуваного процесу.

Синтезована запропонованим вище методом модель комплексу програм має важливий параметр, що налаштовується, - число M станів системи, яке визначається тривалістю передісторії D (числом переходів за станами системи, що враховуються). Використання гіпотези про марковість досліджуваних процесів ($D = 0$) призводить до отримання найбільш компактної, аналогічної за своїми можливостями стохастичної моделі, а розгляд довжини історії всієї траси призводить до найбільш точної, але надмірно громіздкої трасувальної моделі.

За великої кількості аналізованих подій можливе число станів системи, верхня межа якого визначається як $M = C_N^D \cdot D! = N!/(N-D)!$, зі збільшенням D швидко зростає. Таким чином, під час моделювання комплексу програм необхідно розв'язати задачу вибору конкретного значення тривалості врахованої передісторії розвитку процесів, а також оцінити достовірність представлення поведінки системи трасовою мережною моделлю.

Для вирішення цього завдання в якості початкового припущення про природу аналізованого процесу висунемо гіпотезу про можливість його представлення подієвим графом, використовуючи множину $S = \{S_0^{(0)}, S_1^{(0)}, \dots, S_n^{(0)}\}$ можливих станів системи, що збігається з множиною T зареєстрованих подій ($D = 0$).

Подальший аналіз адекватності синтезованої моделі та її уточнення пропонується проводити за таким ітеративним алгоритмом:

1. Проведення перевірки гіпотези «Процес переходів системи за станами із S_D - марковський».

2. Якщо гіпотезу прийнято, то закінчуємо алгоритм, якщо гіпотеза відхилена, то переходимо до наступного пункту 3.

3. Збільшуємо D на одиницю.

4. Визначаємо множину S_D нових станів системи, що формуються передісторією процесу.

5. Якщо розмірність моделі перевищує задану межу, то закінчуємо виконання алгоритму, інакше переходимо до пункту 1.

Зауважимо, що, незважаючи на властивість відсутності післядії, гіпотезу про марковську поведінку системи не можна звести до гіпотези випадковості послідовностей спостережуваних станів системи, для перевірки якої, зокрема, може бути застосовано статистичний критерій серій Вальда-Вольфовица [59]. Наприклад, для моделі системи, наведеної на рис. 2.7, траса її поведінки може мати такий вигляд:

$$S_0, S_2, S_0, S_2, S_0, S_1, S_0, S_2, S_0, S_1, S_0, S_1, S_0, \dots,$$

з якої навіть без застосування будь-якого критерію очевидно, що поява стану s_0 у трасі не випадкова.

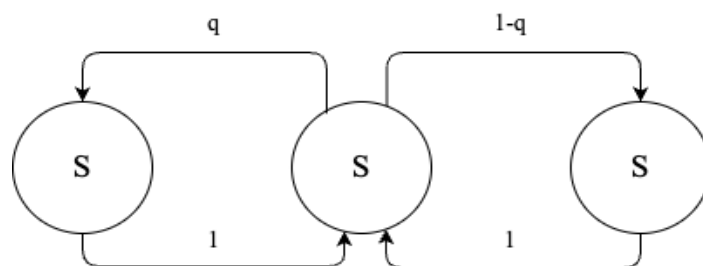


Рисунок 2.7 – Приклад поведінкової моделі системи, що описується марковським ланцюгом

Тому перевірку гіпотези про марковський характер процесів переходу системи пропонується здійснювати, використовуючи теоретичне (відповідне марковській моделі) значення ймовірності досягнення системою деякого обраного стану за $1, 2, \dots, n$ кроків, і експериментальне значення частоти таких подій, обчислене за даними траси.

Визначимо для мережної моделі, подієвий граф якої є ланцюгом Маркова без поглинаючих станів, імовірність $P_{ij}^{(n)}$ переходу системи з i -го в j -й стан за n кроків ($n \geq 1$).

Введемо виробляючу функцію

$$\tilde{P}_{ij}(s) = \sum_{n=0}^{\infty} P_{ij}^{(n)} s^n, \quad |s| < 1.$$

Якщо матриця $Q = \|q_{ij}\|$ перехідних імовірностей марковського ланцюга має розмірність ρ , то, множачи обидві частини рівності на $s q_{ki}$ та сумуючи за $i = \overline{1, \rho}$, отримуємо співвідношення

$$s \sum_{i=1}^{\rho} q_{ki} \tilde{P}_{ij}(s) = \sum_{n=0}^{\infty} \sum_{i=1}^{\rho} q_{ki} P_{ij}^{(n)} s^{n+1} = \tilde{P}_{kj}(s) - P_{kj}^{(0)},$$

котре визначає такі системи рівнянь:

$$\tilde{P}_{kj}(s) - s \sum_{i=1}^{\rho} q_{ki} \tilde{P}_{ij}(s) = P_{kj}^{(0)}, \quad k = \overline{1, \rho}, \quad j = \overline{1, \rho}.$$

Їхніми розв'язками за фіксованих k і $s \in$ функції вигляду $\tilde{P}_{ij}(s) = G_{ij}(s) / D(s)$, які можуть бути розкладені на прості дроби, і, як показано в [65],

$$\tilde{P}_{ij}(s) = \sum_{\lambda=1}^{\rho} \frac{g_{ij}^{(\lambda)}}{1 - s \cdot r_{\lambda}}, \quad (2.1)$$

де r_{λ} - ненульові характеристичні числа (або власні значення) матриці Q перехідних імовірностей; $g_{ij}^{(\lambda)}$ - деякі коефіцієнти.

Нехай h - довільний лівий власний вектор матриці Q , тобто

$$hQ = r_{\lambda} h.$$

Тоді сума елементів векторів лівої та правої частини рівності розраховується таким чином:

$$\sum_{j=1}^{\rho} \sum_{i=1}^{\rho} h_i q_{ij} = \sum_{j=1}^{\rho} h_i \sum_{i=1}^{\rho} q_{ij} = \sum_{j=1}^{\rho} h_i = r_{\lambda} \sum_{j=1}^{\rho} h_i.$$

За умови, що всі h_i позитивні, $r_{\lambda} = 1$ визначає спектральний радіус матриці Q , отже, всі її власні значення задовольняють нерівності $|r_{\lambda}| \leq 1$.

Тому отримаємо

$$\frac{1}{1 - s \cdot r_{\lambda}} = \sum_{i=0}^{\infty} s^i r_{\lambda}^i,$$

а з розкладення на прості дробі випливає, що

$$P_{ij}^{(n)} = \sum_{\lambda=1}^{\rho} g_{ij}^{(\lambda)} r_{\lambda}^n.$$

Для знаходження значень коефіцієнтів $g_{ij}^{(\lambda)}$ скористаємося тим, що

$$P_{ij}^{(n+1)} = \sum_{\lambda=1}^{\rho} g_{ij}^{(\lambda)} r_{\lambda}^{n+1} = \sum_{k=1}^{\rho} q_{ik} P_{kj}^{(n)} = \sum_{k=1}^{\rho} q_{ik} \sum_{\lambda=1}^{\rho} g_{kj}^{(\lambda)} r_{\lambda}^n = \sum_{\lambda=1}^{\rho} r_{\lambda}^n \sum_{k=1}^{\rho} q_{ik} g_{kj}^{(\lambda)},$$

звідки

$$g_{ij}^{(\lambda)} r_{\lambda} = \sum_{k=1}^{\rho} q_{ik} g_{kj}^{(\lambda)}. \quad (2.2)$$

З іншого боку,

$$P_{ij}^{(n+1)} = \sum_{\lambda=1}^{\rho} g_{ij}^{(\lambda)} r_{\lambda}^{n+1} = \sum_{k=1}^{\rho} P_{ik}^{(n)} q_{kj} = \sum_{k=1}^{\rho} \sum_{\lambda=1}^{\rho} g_{ik}^{(\lambda)} r_{\lambda}^n q_{kj} = \sum_{\lambda=1}^{\rho} r_{\lambda}^n \sum_{k=1}^{\rho} g_{ik}^{(\lambda)} q_{kj},$$

звідки

$$g_{ij}^{(\lambda)} r_{\lambda} = \sum_{k=1}^{\rho} g_{ik}^{(\lambda)} q_{kj}. \quad (2.3)$$

У матричній формі вирази (2.2) та (2.3) можуть бути записані як

$$G^{(\lambda)} Q = r_{\lambda} G^{(\lambda)}; \quad Q G^{(\lambda)} = r_{\lambda} G^{(\lambda)}.$$

Таким чином, стовпці матриці $G^{(\lambda)}$ є правими власними векторами матриці Q і визначаються при $r = r_\lambda$ ненульовими рішеннями $x_i^{(\lambda)}$ системи рівнянь.

$$\sum_{k=1}^{\rho} q_{ik} x_k - r \cdot x_i = 0; \quad i = \overline{1, \rho},$$

а рядки - лівими власними векторами, які визначаються ненульовими розв'язками $y_j^{(\lambda)}$ системи

$$\sum_{k=1}^{\rho} y_k q_{kj} - r \cdot y_j = 0; \quad j = \overline{1, \rho}.$$

Тоді з точністю до постійного множника $C^{(\lambda)}$

$$g_{ij}^{(\lambda)} = C^{(\lambda)} x_i^{(\lambda)} y_j^{(\lambda)},$$

а безпосереднє значення $C^{(\lambda)}$ може бути знайдено виходячи з властивості біортонормованості лівих і правих власних векторів, згідно з якого

$$C^{(\lambda)} \cdot \sum_{k=1}^{\rho} x_k^{(\lambda)} y_k^{(\lambda)} = 1.$$

Таким чином, розглянутим методом для марковських моделей можуть бути визначені теоретичні значення ймовірностей $P_{ij}^{(n)}$.

За даними спостережуваної траси станів реальної системи обчислюють частоти переходу системи з i -го в j -й стан за $k = \overline{1, N}$ кроків, які представляють матрицею $V_i = \{V_{ij}^{(k)}\}$, при цьому

$$\sum_{(j)} v_{ij}^{(k)} = n_i^{(k)}.$$

За умови, що $n_i^{(k)} \geq 50$ в якості статистики, що характеризує відхилення експериментальних частот від відповідних теоретичних значень, згідно з

критерієм згоди Пірсона [69, 111] може бути прийнята величина

$$\chi_n^2(V_i) = \sum_{(j)} \frac{[v_{ij}^{(k)} - n_i^{(k)} \cdot P_{ij}^{(k)}]^2}{n_i^{(k)} \cdot P_{ij}^{(k)}} = \sum_{(j)} \frac{[v_{ij}^{(k)}]^2}{n_i^{(k)} \cdot P_{ij}^{(k)}} - n_i^{(k)}.$$

Задаючись рівнем значущості α , марковську гіпотезу про поведінку системи з числом станів L відкидають у разі перевищення величини $\chi_n^2(V_i)$, розподіленої за законом χ^2 -квадрат з $L-1$ ступенем свободи, відповідного табличного значення $\chi_{1-\alpha, L-1}^2$.

Окрім критерію згоди Пірсона, який, як відомо, дає надійні результати тільки в разі приблизної рівності елементів рядка матриці V_j [111] і може застосовуватися тільки коли теоретичні $P_{ij}^{(k)} \neq 0$ для всіх k , для перевірки відповідності експериментальних частот теоретичним за більших значень L можна використовувати інформаційний критерій [117]:

$$J_C = \frac{\hat{H}_i^{(k)} - M(H_i^{(k)})}{\sqrt{D(H_i^{(k)})}},$$

де $\hat{H}_i^{(k)} = -\sum_{j=1}^L \frac{v_{ij}^{(k)}}{n_i^{(k)}} \cdot \ln\left(\frac{v_{ij}^{(k)}}{n_i^{(k)}}\right)$ – статистична оцінка ентропії емпіричного розподілу;

$M(H_i^{(k)}) = h_i^{(k)} - (L-1) / n_i^{(k)}$ – математичне сподівання;

$D(H_i^{(k)}) = \frac{1}{n_i^{(k)}} \left(\sum_{j=1}^L P_{ij}^{(k)} \cdot \ln^2(P_{ij}^{(k)}) - [h_i^{(k)}]^2 \right)$ – дисперсія ентропії

теоретичного розподілу;

$$h_i^{(k)} = -\sum_{j=1}^L P_{ij}^{(k)} \cdot \ln(P_{ij}^{(k)}).$$

У разі збігу емпіричного розподілу частот з передбачуваним (крім рівномірного закону розподілу) статистика J_C розподілена нормально з нульовим математичним очікуванням і одиничною дисперсією, і для заданого

рівня значущості α і квантиля нормованого гаусового розподілу $u_{1-\alpha}$ виконується рівність

$$|J_C| \leq u_{1-\alpha}.$$

Крім того, слід враховувати, що $D(H_i^{(k)}) = 0$ вказує на відсутність стохастичності в поведінці системи, що може цілком узгоджуватися з обраною моделлю.

Як показано в [117], інформаційний критерій за потужністю практично не поступається критерію Пірсона, тоді як ймовірність відкинути правильну гіпотезу для деяких видів розподілів значно нижча.

Перевагою розглянутого методу є його застосовність не тільки під час побудови мережних моделей комплексів програм, а й під час оцінювання адекватності будь-яких моделей, що описуються марковськими ланцюгами без станів, що поглинають.

Недоліком розглянутого методу є певна складність одержання виразів для теоретичних імовірностей $P_{ij}^{(n)}$, особливо якщо число станів системи досить велике.

У цьому разі як альтернативу аналітичному підходу значення теоретичних імовірностей можна одержати методом Монте-Карло, тобто під час імітаційного моделювання поведінки відповідного марковського ланцюга.

При цьому завдання перевірки гіпотези про відповідність експериментальних і модельних ймовірнісних розподілів можна розв'язати на основі критерію виду

$$\chi_n^2(V_i) = n_i^{(k)} m_i^{(k)} \sum_{(j)} \frac{[v_{ij}^{(k)} / n_i^{(k)} - \omega_{ij}^{(k)} / m_i^{(k)}]^2}{v_{ij}^{(k)} + \omega_{ij}^{(k)}},$$

де $\omega_{ij}^{(k)}$ – частота переходу моделі з i -го та j -й стан за $k = \overline{1, N}$ кроків,

$$\sum_{(j)} \omega_{ij}^{(k)} = m_i^{(k)}.$$

Таким чином, розроблений метод, що ґрунтується на аналізі процесів, які описуються моделями на основі мереж Петрі, дає змогу оцінити їхню адекватність із точністю до введених припущень, виявити причинно-наслідкові зв'язки між станами системи, врахувати передісторію процесів, які відбуваються, та обґрунтувати вибір її довжини.

Використання різних критеріїв при перевірці гіпотез про відповідність теоретичних експериментальним ймовірнісним розподілам ймовірностей підвищує якість ухваленого рішення, знижуючи можливість нераціонального збільшення розмірності синтезованої моделі.

Запропонований метод застосовний для оцінювання адекватності широкого класу моделей, в основі яких лежить опис процесу зміни станів системи марковським ланцюгом із поглинаючими станами.

2.4 Агрегативний опис функціональних моделей процесів у кластерах граничного шару

З метою визначення умов і можливостей застосування запропонованого в п. 2.2 методу побудови мережних моделей, проаналізуємо створювану СДЧМП з позицій теорії мереж Петрі, котра описує функціональну модель кластера граничного шару, вузлами котрого є територіально нерознесені пристрої IoT з обмеженими обчислювальними можливостями, наприклад, одноплатні комп'ютери.

В основі методу лежить виділення N типів аналізованих подій $T = \{t_0, t_1, \dots, t_N\}$, що відповідають переходам мережі. Вільною мовою мережі Петрі називається множина L усіх можливих послідовностей спрацьовування переходів [119], тобто множина слів, складених із символів подій алфавіту T . Траса подій, зареєстрованих в обчислювальній системі за допомогою вимірювальних засобів, розглядається як кінцева підмножина $L' \subset L$ слів, що породжуються шуканою мережею Петрі. Позначимо через Λ клас мов мереж будь-якого типу, утворених за допомогою визначених на всіх переходах

символів деякого алфавіту. Доведено [119], що якщо $L' \subset L$, то проблеми еквівалентності та включення для такої мови є нерозв'язними. Це означає принципову неможливість визначити, чи зміниться мова мережі, якщо в неї додати (або видалити) деякий перехід. Відповідно для будь-якого методу, на основі якого будується мережа, не можна довести правильність включення або виключення деяких переходів. Таким чином, у загальному випадку коректність представлення траси подій мережею Петрі не може бути строго доведена.

Однак перелічені проблеми розв'язні для класу L_A - регулярних мов, породжуваних, зокрема, скінченними автоматами. Це означає, що для застосування методу синтезу СДЧМП слід аналізувати трасувальні дані процесів, що складаються з подій, які задовольняють умовам послідовності й альтернативності (але не паралельності). У цьому випадку є обґрунтованою взаємнооднозначна відповідність між виміряною трасою і трасою подій, породжуваною мережею Петрі [84].

У [58] введено поняття розгортки мережі Петрі в ациклічну мережу-процес і досліджено способи такого перетворення. Метод синтезу СДЧМП, застосовний для послідовно-альтернативних процесів, дає змогу розв'язати зворотну задачу, яку раніше не розглядали. Таке перетворення пропонується назвати згортанням мережі-процесу в мережу Петрі.

З позицій структурного підходу найбільш загальною формалізованою схемою подання складної системи є її опис у вигляді агрегативної системи [58], утвореної шляхом з'єднання кількох агрегатів, що взаємодіють між собою. Під агрегатом у нашому випадку розумітимемо згортку i -го процесу виду

$$NP^i = \{P^i, T^i, F^i, GV^i, GS^i, M_o^i\},$$

де P^i - множина позицій умов виконання подій i -го процесу;

T^i - множина переходів, що відповідає множині подій i -го процесу;

$F^i \subseteq P^i \times T^i \cup T^i \times P^i$ - відношення інцидентності між множинами

позицій та переходів;

GV^i – функція відповідності переходам випадкових величин часу виконання подій i -го процесу;

GS^i – функція визначення параметрів групових переходів, що описують недетермінованість розвитку процесу;

M_0^i – початкове маркування.

Для довільного елемента $a \in A$ визначимо через $X(a)$ множину його вхідних, а через $Y(a)$ – множину його вихідних елементів. Тоді структура NP^i задовольняє умовам:

- 1) $P^i \cap T^i = \emptyset$ та $P^i \cup T^i \neq \emptyset$;
- 2) $\forall a \in P^i \cup T^i, \exists b \in P^i \cup T^i: b \in X(a) \vee b \in Y(a)$, тобто будь-який елемент мережі інцидентний хоча б одному елементу іншого типу;
- 3) $\forall t \in T^i: X(t) \neq \emptyset \wedge Y(t) \neq \emptyset$, тобто будь-який перехід має хоча б одне вхідне й одне вихідне місце;
- 4) $H^i = \{p | X(p) = \emptyset\}$ – єдина позиція, яка не має вхідних переходів, звана головною;

$$5) \quad M_0^i(p) = \begin{cases} 1, & p \in N^i \\ 0, & p \notin N^i \end{cases}, \text{ де } N^i - \text{єдина позиція, яку називають}$$

початковою.

СДЧМП розглянутого виду має всі ознаки агрегату [60]: у довільний момент часу перебуває в одному з безлічі можливих станів, обумовлених передісторією і вхідними (керівними) сигналами, і видає за кінцевий час кінцеву кількість вихідних сигналів-подій.

Для введення відносин слідування і паралелізму між процесами такої структури пропонується введення додаткових дуг, що зв'язують внутрішні переходи однієї мережі з головною позицією іншої, що можна побачити на схемі рис. 2.8.

Таке агрегативне об'єднання СДЧМП є розширенням поняття паралельних ациклічних мереж дій, або A -мереж [54], які використовують для уявлень паралельно-альтернативних процесів. Появу мітки в головній позиції

розглядають як надходження вимоги на виконання процесу, що організовується за умови його готовності, тобто за наявності мітки в початковій позиції.

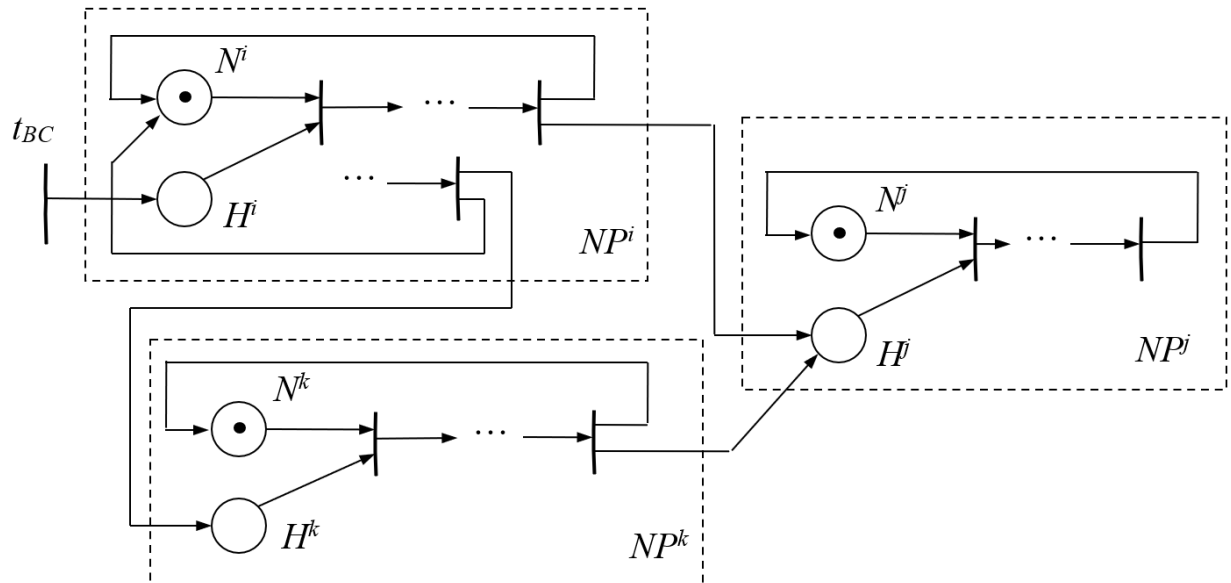


Рисунок 2.8 – Агрегативна модель взаємодії процесів, представлених часовими мережами Петрі

Запропоновану функціональну модель системи можна легко доповнити моделлю робочого навантаження, у цьому найпростішому випадку описуваного переходом t_{BC} , що генерує зовнішні заявки на обслуговування.

Отримана агрегативна мережа, з погляду теорії мереж Петрі, є скінченною, правильною (тобто такою, що не має глухої розмітки), живою (тобто такою, що не має мертвих переходів), K -щільною [54], такою, що містить циклічні компоненти, що особливо характерно для функціонування компонентів реального часу граничного шару комп'ютерних системах підтримки мережі IoT.

Розглянутий підхід до побудови мережі базується на концептуальній моделі функціонування обчислювальних засобів як сукупності реалізацій паралельних взаємодіючих процесів [51, 69]. Це дає змогу досліджувати насамперед динаміку поведінки програмних компонентів системи та їхні

алгоритмічні властивості. Важливо, що при цьому несуттєво, чи призначена програмна система для локального виконання, чи передбачено розподілене опрацювання інформації в мережі.

З метою розширення моделювальних можливостей агрегативної мережі, а саме врахування механізмів синхронізації процесів та їхньої конкуренції за спільні ресурси, можливе введення в мережу додаткових позицій $p_C \in P_C$, які називаються ресурсними і задовольняють умовам:

- 1) $\forall p_C \in P_C: M_0^C(P) \geq 1$, що означає, що всі позиції-ресурси мають розмітку, що відмінна від нульової;
- 2) $\forall p_C \in P_C: |X(p_C)| = |Y(p_C)| > 1$, що означає, що з кожної позиції-ресурсу виходить стільки ж дуг, скільки і входить, причому мінімум дві;
- 3) $Y(p_C) \in Y(N^i)$, тобто вихідні дуги, що означають захоплення ресурсу, так само як і дуги з головних позицій мережі-процесу, дозволяють його виконання;
- 4) $X(p_C) \subseteq T^i$, тобто процес під час свого виконання зобов'язаний звільнити ресурс.

У загальному випадку процес передбачає кілька шляхів свого розвитку, кожний із яких за умовою 4 має передбачити звільнення ресурсу, що суперечить умові 2. Однак припущення про циклічність процесів і можливість виокремлення початкових позицій зумовлює можливість введення в структуру СДЧМП додаткового стану, що відповідає закінченню процесу, і простого (не часового) переходу з цього стану в початковий, який не впливає на часові характеристики процесу.

Визначення ресурсних позицій дає змогу певним чином описувати конкуренцію процесів за апаратні засоби обчислювальної системи. Наприклад, виконання трьох процесів, що здійснюють приймання-передавання даних, два з яких реалізують обмін інформацією із зовнішнім накопичувачем на жорсткому диску, у системі з двома портами введення-виведення може бути реалізовано агрегативною функціональною моделлю, представленою на рис. 2.9.

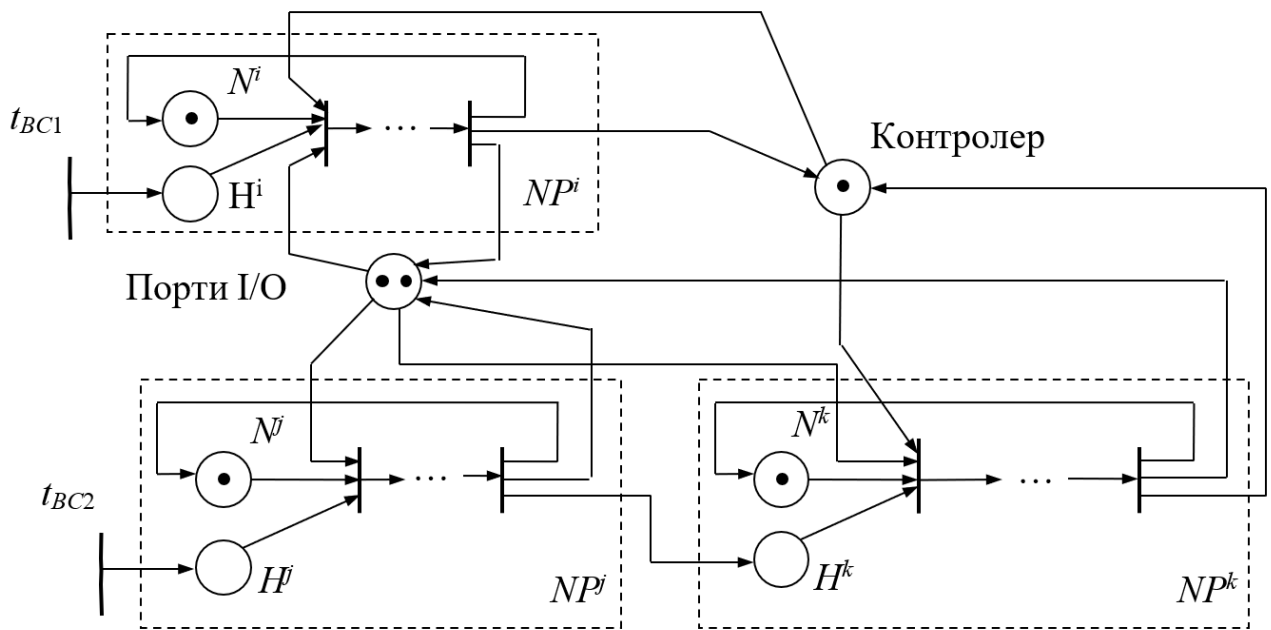


Рисунок 2.9 – Варіант моделювання процесів, які конкурують за ресурси

Так само стає можливим опис механізмів синхронізації процесів, що ґрунтуються в сучасних багатозадачних системах на використанні м'ютексів і семафорів - спеціальних об'єктів, які забезпечують неможливість одночасного доступу до даних різним процесам [68].

Слід зазначити, що, на відміну від методу побудови мережних моделей (п. 2.2), введення додаткових позицій і дуг призводить до потенційної небезпеки виникнення в мережі Петрі безвихідних ситуацій. Цю проблему можна розв'язати шляхом побудови й аналізу графа можливих маркувань мережі [95].

Таким чином, розглянуті способи й умови побудови функціональних моделей окремих шарів інфраструктури підтримки Інтернету речей на основі часових мереж Петрі, що описують як програмні, так і апаратні компоненти обчислювальних засобів вузлів відповідного шару, дають змогу розширити сукупність аналізованих чинників, які впливають на перебіг обчислювального процесу.

Впевненість в адекватності моделі обґрунтована використанням вимірювальних трас виконання реальних процесів, що містять інформацію про більшість подій, які відбуваються в системі. Агрегативний опис взаємодіючих

паралельних процесів відповідає сучасним концепціям проєктування та реалізації систем реального часу у розподілених обчислювальних середовищах, зокрема у граничному шарі інфраструктури підтримки Інтернету речей.

Поняття процесів як мінімальних незалежних об'єктів, яким виділяють ресурси, способи їхнього представлення та організація є спільними для більшості сучасних обчислювальних систем, що забезпечує універсальність запропонованого підходу до їхнього моделювання та дослідження.

Висновки за розділом 2

1. Побудова, уточнення й аналіз функціональних моделей кластерів граничного шару комп'ютерної системи підтримки Інтернету речей повинні проводитися паралельно з їх проєктуванням, розробкою та супроводом, що є основою для створення якісних кластерів, які задовольняють всім висунутим техніко-експлуатаційним вимогам. Такі функціональні моделі з обґрунтованим рівнем адекватності й точності виступають у ролі специфікацій КСРЧ і стають обґрунтуванням для реалізації конкретних проєктних рішень.

2. Запропонований метод побудови функціональної моделі кластера граничного шару комп'ютерної системи підтримки Інтернету речей базується на динамічній моделі поведінки асинхронних паралельних процесів і забезпечує можливість використання єдиного формально-логічного апарату – часових мереж Петрі, для опису як моделі самої системи, так і моделі зовнішнього середовища. Підвищений рівень довіри до побудованої моделі досягається шляхом обґрунтованого врахування виявлених взаємозалежностей між подіями, що полягають у переходах системи за обраними станами, і визначеними на основі трасувальних даних.

3. Метод побудови функціональних моделей кластера граничного шару комп'ютерної системи підтримки Інтернету речей ґрунтується на принципах багаторівневого ієрархічного моделювання, що дає змогу

проводити уточнення властивостей системи в міру зростання її розуміння. Різні етапи обробки інформації можливо розглядати з різним рівнем деталізації. При цьому для завдання часу спрацювання окремих переходів мережної моделі допускається використання випадкових величин із законами розподілу загального вигляду, що дає змогу врахувати особливості виконання застосунків у системах підтримки мережі IoT.

4. Можливість урахування передісторії розвитку модельованих процесів дає змогу розглядати відомі стохастичні та трасові моделі застосунків як окремі випадки пропонованої мережної моделі. Запропонований метод оцінювання адекватності мережних моделей застосовний не тільки для перевірки відповідності поведінки реальної системи і моделі, а й для обґрунтування її розмірності. У його основі лежить застосування критерію згоди Пірсона та інформаційного критерію, які доповнюють один одного й які використовують під час порівняння теоретичних ймовірностей та експериментальних частот досягнення системою певних станів за кілька кроків. Універсальність розробленого методу полягає в можливості його використання для оцінювання адекватності не тільки моделей, побудованих на основі мереж Петрі, а й інших моделей, процес зміни станів яких можна описати ланцюгом Маркова без поглинаючих станів.

5. Використання агрегативного підходу під час опису функціонування застосунків на вузлах кластера граничного шару, що розглядається як сукупність декількох процесів, дає змогу описати всі основні особливості їхньої взаємодії: послідовність, паралелізм, синхронізація, недетермінізм, поділ ресурсів. Використання такого рівня абстрагування зумовлює високий ступінь гнучкості функціональної моделі кластера, а також можливість її подальшого розширення з метою аналізу впливу на організацію обчислювального процесу різноманітних додаткових факторів.

РОЗДІЛ 3

РОЗРОБКА МЕТОДІВ ЗМЕНШЕННЯ ЧАСУ ОБРОБКИ ОПЕРАТИВНИХ ТРАНЗАКЦІЙ ІНТЕРНЕТУ РЕЧЕЙ

3.1 Розробка методу формування кластера граничного шару комп'ютерної системи підтримки Інтернету речей

Типовим підходом до аналітичного дослідження моделей, побудованих на часових мережах Петрі, є генерація множини досяжності та на її основі оцінювання стаціонарних імовірностей станів деякого випадкового часового процесу $X(t)$, що визначається як зміна розміток шуканої мережі. Для кінцевого простору всіх припустимих розміток-станів $E = \{e_0, e_1, \dots, e_n\}$ перехід з імовірністю q_{ij} із розмітки e_i в розмітку e_j визначає марковський ланцюг, вкладений у процес, що розглядається. За умови кінцевості множини досяжності стохастичної мережі Петрі стаціонарні ймовірності π_j станів e_j марковського ланцюга, вкладеного в процес $X(t)$, визначаються шляхом розв'язання системи однорідних рівнянь

$$\pi_j = \sum_{e_i \in E} \pi_i q_{ij}, \quad \sum_{e_j \in E} \pi_j = 1.$$

За припущенням, що за скінченний проміжок часу відбувається кінцева кількість переходів, стаціонарні ймовірності p_j знаходження мережі Петрі в обраній розмітці не залежать від виду розподілу часу, а залежать тільки від математичного сподівання $M(t_j)$ цього часу і стаціонарних імовірностей марковського ланцюга:

$$p_j = \frac{\pi_j M(t_j)}{\sum_{e_i \in E} \pi_i M(t_i)}.$$

У [116] даний підхід до визначення імовірнісних характеристик розширено для випадку як обмежених, так і необмежених мережевих моделей

із безкінечною множиною досяжності, але за умови регулярності її структури, так що її можна звести до кінцевого числа еквівалентних безкінечних множин. Це дає змогу розраховувати стаціонарні ймовірності станів як елементи геометричної прогресії з однаковим показником.

Методика розрахунку граничного розподілу ймовірностей досяжних розміток для неекспоненціальних часових стохастичних мереж Петрі ґрунтується на понятті стану випадкового процесу, утвореного парою «розмітка - вектор, що складається з функцій розподілу остаточного часу спрацьовування всіх переходів», що дає змогу описувати випадкові процеси в мережі як напівмарковські. У [122] розглядається область застосування цієї методики - узагальнені часові стохастичні мережі Петрі (УЧСМП), і необхідна умова застосування - обмеженість передісторії мережі.

В УЧСМП використовуються як часові переходи з довільним розподілом часу спрацьовування, так і миттєві переходи, які перерозподіляють мітки в мережі з метою здійснення процедур логічного характеру, що значно розширює можливості моделювання. При цьому правила спрацьовування часових переходів задовольняють таким умовам:

- а) відлік часу спрацьовування збудженого переходу починається з наявності в позиції-попередниці необхідної кількості міток;
- б) стан збудження не переривається при появі в позиціях-попередниках надлишкових міток;
- в) розв'язання конфлікту переходів (коли вилучення міток одним із переходів знімає стан збудження для інших) розв'язується випадковим чином, виходячи з найменшого часу спрацьовування «найшвидшого» переходу.

Зазначимо, що мережеві моделі комп'ютерних систем реального часу (КСРЧ), розглянуті в розділі 2, належать до класу УЧСМП.

Обмеженість передісторії мережі полягає в тому, що в будь-якому циклі на графі розміток існує принаймні одна розмітка, у якій або збуджено лише переходи з експоненціальним розподілом часу їхнього спрацьовування, або в усіх збуджених неекспоненціальних переходів відлік часу спрацьовування

починається заново. Інакше кажучи, у такій розмітці - точці регенерації - відбувається «забування» історії процесу, водночас кількість можливих станів процесу скінченна. Це дає змогу, використовуючи техніку кратних інтегралів, визначити функції розподілу часу «сидіння» процесу в кожній розмітці, розрахувати значення перехідних імовірностей і граничних розподілів імовірностей досяжних розміток.

Під час моделювання часових процесів функціонування складних систем, крім вищенаведених характеристик, практичний інтерес представляє визначення часу переходу процесу з одних визначених станів в інші. Наприклад, для трасових мережевих моделей КСРЧ важливим показником, що характеризує загальну продуктивність системи, є час виконання алгоритму, для оцінки якого бажано визначити значення його математичного сподівання, дисперсії, а також час виконання за найкоротшим і максимальним маршрутами.

Проблеми знаходження найдовших і найкоротших шляхів виконання транзакції IoT, що описується простою часовою мережею Петрі (з детермінованими часом спрацьовування переходів), розглядаються в [110]. Аналіз часових інтервалів між подіями в часовій мережі з довільними часом спрацьовування переходів у [119] пропонується проводити з використанням методу Монте-Карло, тобто імітаційним способом. Однак такий підхід, у разі необхідності отримання потрібної точності результатів, може потребувати надмірно великих обсягів випробувань.

Наведена вище умова обмеженості передісторії мережі Петрі дає змогу розглянути загальний випадок часових мереж Петрі та отримати аналітичні вирази для математичного сподівання і дисперсії часу досягнення процесом аналізованих станів. Стосовно функціональних моделей КСРЧ це дасть змогу оцінити показники продуктивності їхніх окремих компонентів.

Для графа досяжності мережі Петрі, що задовольняє умові обмеженості передісторії, позначимо через $F_i(x)$ функцію розподілу часу перебування процесу в розмітці e_i . Приймемо припущення, що в кожній досяжній розмітці

залишкові часи спрацьовування одночасно збуджених переходів є незалежними, а процес зміни розміток можна описати ланцюгом Маркова з єдиним позитивним зворотнім класом. При цьому умовна ймовірність того, що, пробувши в розмітці-стані e_i час, що дорівнює x , процес перейде в стан e_j , визначається як

$$Q_{ij}(x) = P\{\lambda_n = e_j, \vartheta_i^{(n-1)} \leq x \mid \lambda_{n-1} = e_i\} =$$

$$= \int_0^x P\{\lambda_n = e_j \mid \vartheta_i^{(n-1)} = t, \lambda_{n-1} = e_i\} dF_i(t) = \int_0^x q_{ij}(t) dF_i(t),$$

де λ_n - стан напівмарковського процесу (НМП) на n -му кроці;

$\vartheta_i^{(n-1)}$ - час перебування НМП в e_i -му стані на $n-1$ кроці еволюції.

Якщо в стані e_i збуджено k переходів, яким відповідають функції розподілу $\Psi_1^{(i)}(x)$, ..., $\Psi_k^{(i)}(x)$ залишкового часу їхнього спрацьовування, то функція розподілу для часу перебування НМП у стані e_i має вигляд:

$$F_i(x) = 1 - \prod_{r=1}^k [1 - \Psi_r^{(i)}(x)].$$

Математичне сподівання і дисперсія цього часу відповідно дорівнюють

$$M[\vartheta_i^{(n-1)}] = \int_0^\infty x dF_i(x), \quad D[\vartheta_i^{(n-1)}] = \int_0^\infty x^2 dF_i(x).$$

Ймовірність переходу q_{ij} з розмітки e_i в розмітку e_j може бути обчислена відповідно до рівності

$$q_{ij} = \lim_{x \rightarrow \infty} Q_{ij}(x) = \int_0^\infty p_{ij}(x) dF_i(x).$$

Знання функцій розподілу $\Psi_1^{(i)}(x)$, ..., $\Psi_k^{(i)}(x)$ залишкового часу

спрацьовування k збуджених переходів робить завдання знаходження $p_{ij}(x)$ розв'язним. Слід зазначити, що в низці практичних випадків для визначення ймовірності спрацьовування s -го переходу, що переводить НМП зі стану e_i в стан e_j , виявляється зручніше скористатися співвідношенням

$$p_{ij}^{(s)} = \int_0^\infty \left[\prod_{r \neq s} \Psi_r^{(i)}(x) \right] d\Psi_s^{(i)}(x); \quad p_{ij} = \sum_s p_{ij}^{(s)}.$$

З цього, зокрема, випливає, що

$$p_{ij}^{(s)}(x) = \frac{[\Psi_s^{(i)}(x)]' \prod_{r \neq s} \Psi_r^{(i)}(x)}{\sum_{l=1}^k [\Psi_l^{(i)}(x)]' \prod_{r \neq l} \Psi_r^{(i)}(x)}; \quad p_{ij}(x) = \sum_s p_{ij}^{(s)}(x). \quad (3.1)$$

Знайдемо закони розподілу остаточного часу спрацьовування збуджених переходів, що зберегли збудження після потрапляння НМП зі стану e_i в стан e_j . При цьому для визначення, наприклад, функції розподілу $\Psi_1^{(i)}(x)$ остаточного часу спрацьовування переходу, що зберіг збудження, вводиться в розгляд випадкова величина $\tilde{\Psi}_{n-1} = \min \{ \xi_r \mid r = 2, \dots, k \}$. Її щільність розподілу має вигляд

$$\tilde{\Psi}'_i(x) = \prod_{m=2}^k [1 - \Psi_m^{(i)}(x)] \sum_{r=2}^k [\Psi_r^{(i)}(x)]' [1 - \Psi_r^{(i)}(x)].$$

З огляду на те, що умовна ймовірність того, що ПМП потрапляє в стан e_s , після того, як побув в стані e_i час, що дорівнює x (у разі спрацьовування не першого переходу), дорівнює

$$\tilde{q}_{is}(x) = P\{\lambda_n = e_s, \tilde{\Psi}_{n-1} = x \mid \lambda_{n-1} = e_i\}, \quad \sum_s \tilde{q}_{is}(x) = 1.$$

Тоді умовна функція розподілу випадкового часу $\tilde{\Theta}_{is}$ перебування НМП у стані e_i за умови переходу в стан e_s (без урахування першого переходу)

$$\tilde{Q}_{ij}(x) = \int_0^x \tilde{q}_{ij}(t) \tilde{\Psi}'_i(t) dt.$$

Використовуючи формулу для визначення щільності розподілу різниці випадкових величин ξ_1 (часу спрацьовування першого переходу, що задається функцією щільності $f_1(x)$) та $\tilde{\mathfrak{G}}_{is}$, отримаємо

$$\left[\Psi_1^{(i)}(x) \right]' = \frac{1}{\mu} \int_0^\infty f_1(x+y) \tilde{Q}'_{ij}(y) dy,$$

де

$$\mu = \int_0^\infty \int_0^\infty f_1(x+y) \tilde{Q}'_{ij}(y) dy dx.$$

Нехай D - підмножина множини всіх допустимих розміток-станів E , що визначає деякі досяжні стани. Відповідно \bar{D} - множина всіх інших станів, тобто $D \cup \bar{D} = E$ і $D \cap \bar{D} = \emptyset$.

Визначимо через τ_i час досягнення НМП стану $e_k \in D$ за умови, що початковим станом процесу є e_i , а через $T_i(t)$ - функцію розподілу випадкової величини τ_i . Будемо позначати також

$$m_i(r) = \int_0^\infty t^r dF_i(t).$$

Тоді математичні сподівання для моментів r -го порядку часу досягнення будь-якого стану з області D визначається системою рівнянь:

$$M(\tau_i^r) = a(i, r) + \sum_{e_j \in \bar{D}} p_{ij} M(\tau_j^r),$$

де

$$a(i, r) = \sum_{e_j \in D} p_{ij} m_i(r) + \sum_{e_j \in \bar{D}} p_{ij} \left(\sum_{k=1}^r C_r^k m_i(k) M(\tau_j^{r-k}) \right).$$

При цьому необхідно враховувати, що $p_{ij}M(\tau_j^r) = 0$ та $p_{ij}m_i(r) = 0$, якщо $p_{ij} = 0$ та $M(\tau_j^0) = 1$.

Після нескладних перетворень для оцінки математичного сподівання часу досягнення процесом області D отримуємо систему рівнянь

$$M(\tau_i) = M(\vartheta_i) + \sum_{e_j \in \bar{D}} p_{ij} M(\tau_j), \quad (3.2)$$

а для моментів другого порядку

$$M(\tau_i^2) = D(\vartheta_i) + [M(\vartheta_i)]^2 + \sum_{e_j \in \bar{D}} p_{ij} (M(\tau_j^2) + 2M(\tau_j)M(\vartheta_i)), \quad (3.3)$$

звідки дисперсія часу досягнення процесом області D з урахуванням (3.1)

$$\begin{aligned} D(\tau_i) &= M(\tau_i^2) - [M(\tau_i)]^2 = D(\vartheta_i) + \sum_{e_j \in \bar{D}} p_{ij} M(\tau_j^2) - [M(\tau_i) - M(\vartheta_i)]^2 = \\ &= D(\vartheta_i) + \sum_{e_j \in \bar{D}} p_{ij} (D(\tau_j) + [M(\tau_j)]^2) - [M(\tau_i) - M(\vartheta_i)]^2. \end{aligned} \quad (3.4)$$

Іншим способом вирази (3.1) і (3.3) можуть бути отримані, виходячи з визначення функції розподілу $T_i(t)$ через систему інтегральних рівнянь:

$$T_i(t) = \sum_{e_j \in \bar{D}} p_{ij} \int_0^t T_j(t-u) dP_{ij}(u) + \sum_{e_j \in D} P_{ij}(t),$$

де $P_{ij}(t) = p_{ij}F_i(t)$ - ймовірність переходу зі стану e_i у стан e_j за час, не більший ніж t , при цьому

$$F_i(t) = \sum_{e_j \in E} P_{ij}(t).$$

Переходячи до перетворень Лапласа-Стілтєса, отримуємо

$$\tilde{T}_i(s) = \sum_{e_j \in \bar{D}} \tilde{p}_{ij}(s) \tilde{T}_j(s) + \sum_{e_j \in D} \tilde{p}_{ij}(s) = \tilde{F}_i(s) + \sum_{e_j \in \bar{D}} \tilde{p}_{ij}(s) \cdot (\tilde{T}_j(s) - 1).$$

Диференціювання цього виразу дає

$$\tilde{T}_i'(s) = \tilde{F}_i'(s) + \sum_{e_j \in \bar{D}} \tilde{p}_{ij}'(s) \cdot (\tilde{T}_j(s) - 1) + \sum_{e_j \in D} \tilde{p}_{ij}(s) \tilde{T}_j'(s), \quad (3.5)$$

і враховуючи, що $\tilde{T}_i(0) \div 1$, $\tilde{p}_{ij}(0) \div p_{ij}$, $\tilde{T}_i'(0) \div -M(\tau_i)$, $\tilde{F}_i'(0) \div -M(\vartheta_i)$ приходимо до системи рівнянь, аналогічної до виразу (3.1).

Вторинно диференціюючи (3.4), отримуємо:

$$\tilde{T}_i''(s) = \tilde{F}_i''(s) + \sum_{e_j \in \bar{D}} \tilde{p}_{ij}''(s) \cdot (\tilde{T}_j(s) - 1) + 2 \sum_{e_j \in \bar{D}} \tilde{p}_{ij}'(s) \tilde{T}_j'(s) + \sum_{e_j \in D} \tilde{p}_{ij}(s) \tilde{T}_j''(s).$$

Враховуючи, що $\tilde{T}_i''(0) \div M(\tau_i^2)$, $\tilde{F}_i''(0) \div M(\vartheta_i^2) = D(\vartheta_i) + [M(\vartheta_i)]^2$, $\tilde{p}_{ij}'(0) \div -p_{ij}M(\vartheta_i)$, отримуємо систему рівнянь, що описується виразом (3.2).

Для подальшого розрахунку середнього часу задіяння одного застосунку із граничного кластера обслуговування оперативних транзакцій Інтернету речей скористаємось специфічними особливостями граничного шару. Вважаючи на обмеженість обчислювальних ресурсів вузлів даного шару (зазвичай одноплатні комп'ютери) розподіляємо на кожний вузол тільки один застосунок. Нехай в якості елементів даного кластера можна задіяти m пристроїв Інтернету речей із множини $K = \{k_j, j = \overline{1, m}\}$, причому $m > n$. Тоді всі можливі варіанти розподілу складають таку множину розподілів:

$$\Gamma = \left\{ \gamma_k : e_i \rightarrow k_j \mid i = \overline{1, n}; \quad j = \overline{1, m}; \quad \varphi(e_i) \leq \varphi(k_j) \right\},$$

де $\varphi(\bullet)$ – функція продуктивності, а γ_k – один з можливих варіантів розподілу (рис. 3.1).

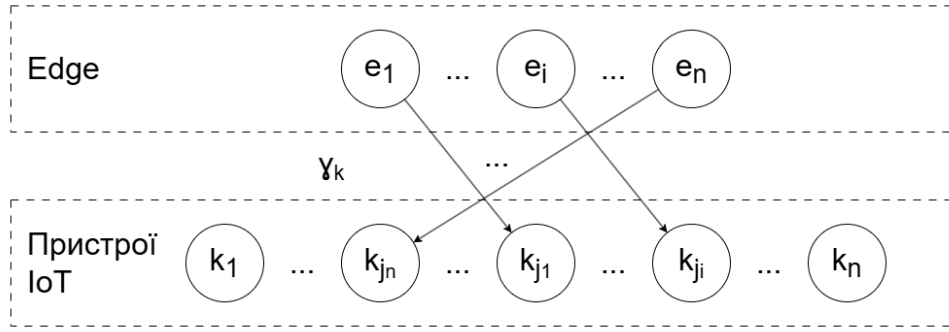


Рисунок 3.1 – Приклад елемента множини розподілів застосунків по вузлах граничного шару

Врахуємо, що в даному випадку кожен i -й застосунок буде відповідати i -му стану системи, тобто ймовірність його застосування на протязі конкретного часового інтервалу буде складати p_i , а математичне сподівання часу очікування запиту буде складати $M(\tau_i)$. Якщо середній час виконання i -го застосунку на виділеному вузлі граничного шару буде складати t_i , то можна для кожного розподілу γ_k розрахувати прогнозований середній час виконання одного застосунку, та обрати найкращий за даним критерієм розподіл:

$$t_{cep}(\gamma_k) = \sum_{i=1}^n (t_i + M(\tau_i)) \cdot p_i; \quad (3.6)$$

$$\gamma^* = \left(\gamma_\ell \mid t_{cep}(\gamma_\ell) \leq t_{cep}(\gamma_k) \quad \forall k \in \overline{1, m} \right).$$

Таким чином, отримані аналітичні вирази застосовні для дослідження доволі широкого класу мереж Петрі, а саме узагальнених часових стохастичних мереж, що оперують із довільними законами розподілу часу спрацьовування переходів.

Умова обмеженості передісторії мережі дає змогу одержати точкові оцінки основних характеристик досліджуваних процесів, причому для їхнього розрахунку в загальному випадку не потрібне знання виду законів розподілу часу перебування системи в різних станах, а тільки їхнього математичного сподівання і дисперсії. Крім того, запропонований метод дозволяє визначити оптимальний варіант розподілу застосунків по вузлах граничного шару за

критерієм мінімального середнього часу виконання одного застосунку.

3.2 Редукція кількості станів часових мереж Петрі

Розглянутий метод знаходження точкових оцінок часових процесів у мережах Петрі має недолік, пов'язаний із практичними труднощами розв'язування значної кількості рівнянь, що виникають під час розгляду моделей з великою кількістю позицій і переходів і, як наслідок, відповідним числом станів-розміток графа досяжності.

Розв'язанням зазначеної проблеми може бути вироблення правил виключення з розгляду певних позицій і переходів та введення замість них нових, унаслідок чого відбувається зниження (редукція) числа станів аналізованої мережі, за умови збереження її поведінкових і часових властивостей.

Для простих часових мереж Петрі (де переходам приписано детермінований час спрацьовування) такі правила «злиття переходів» запропоновано в [106]. Аналогічний підхід під час аналізу складніших часових мереж моделі Мерліна (моменти спрацьовування переходів описуються інтервалами) розглянуто в [84].

Для узагальнених часових мереж Петрі з переходами, час спрацьовування котрих описують закони, що описуються відомими аналітичними виразами, функція сумарного часу спрацьовування послідовності переходів знаходиться як згортання їхніх відповідних функцій розподілу, для чого може бути використано Фур'є-перетворення. Зауважимо, що подібний аналіз є схожим на методи дослідження ГЕРТ-мереж [102], де під час злиття паралельних і послідовних дуг попередньо знаходять приписані дугам похідні функції моментів, для яких, так само, як і для перетворень Фур'є, добуток визначає згортку функцій розподілів незалежних випадкових величин.

Застосування таких методів виявляється можливим тільки для досить

вузького класу видів розподілів і є неприйнятним, наприклад, для функцій розподілів загального вигляду, у тому числі описуваних таблично на інтервалі допустимих значень.

Такий опис особливо характерний при побудові моделей транзакцій на основі експериментальних даних.

Розподіли часу виконання реальних комп'ютерних компонентів не є ні нормальними, ні експоненціальними, і за своєю суттю досить асиметричні. Водночас вони мають суттєву особливість - обмежений інтервал можливих значень, на відміну від переважної більшості застосовуваних нині теоретичних розподілів із безкінечними областями значень випадкових величин.

З метою врахування таких часових особливостей, притаманних більшості процесів, що відбуваються в системах реального часу, а також розв'язання розглянутих вище проблем, запровадимо нове розширення часової мережі Петрі, що описується набором

$$NP = \{P, T, F, H, GV, GS, M_0\},$$

де P – множина позицій;

T – множина переходів;

$F: P \times T$ – функція передування,

$H: T \times P$ – функція проходження множини позицій P і переходів T ;

$GV: T \times f^{(i)}(X)$ – функція відповідності між множиною переходів T і множиною стохастичних величин, що таблично описують функцію густини розподілу часу спрацьовування переходу на кінцевому інтервалі можливих значень, тобто

$$f^{(i)}(X) = \{f_0^{(i)}(x_0^{(i)}), f_1^{(i)}(x_1^{(i)}), \dots, f_{N_i}^{(i)}(x_{N_i}^{(i)})\};$$

$GS: T \times PS$ – функція відповідності між множиною групових переходів T , що утворюють повну групу несумісних подій, і множиною ймовірностей їхнього спрацьовування PS ;

M_0 – початкове маркування.

Пропоноване розширення мереж Петрі, по-перше, використовує підвищені моделювальні можливості стохастично-детермінованих часових мереж Петрі, по-друге, дає змогу описувати з потрібним рівнем точності закони розподілу випадкових величин практично будь-якого виду, по-третє, зберігає інтервальні особливості часових мереж моделі Мерліна і можливість застосування для їхнього аналізу відомих математичних апаратів (зокрема, темпоральної логіки реального часу).

Крім того, за такого опису часових параметрів спрацьовування переходів виявляється можливим запропонувати метод редукції кількості станів часової мережі Петрі, який узагальнює відомі підходи до аналізу графових і мережевих моделей, та дає змогу одержати значення математичного сподівання й дисперсії часу досягнення аналізованим процесом обраних станів без побудови й розв'язання систем рівнянь, розглянутих у п. 3.1.

Нехай кожному переходу t_i мережі Петрі поставлено у відповідність, окрім заданої функції $f^{(i)}(X)$ густини ймовірності часу спрацьовування переходу, $M(t_i)$ і $D(t_i)$ - значення його математичного сподівання і дисперсії, а для групових переходів визначено ймовірності $P(t_i)$ спрацьовування переходу в групі.

Тоді під час злиття будь-яких послідовно виконуваних переходів (рис. 3.2) для нового переходу t_k :

$$M(t_k) = M(t_i) + M(t_j),$$

$$D(t_k) = D(t_i) + D(t_j),$$

$$P(t_k) = P(t_i) \cdot P(t_j).$$



Рисунок 3.2 – Злиття послідовних переходів

Для перерахунку функції густини ймовірності $f^{(k)}(X)$, що являє собою згортку функцій $f^{(i)}(X)$ та $f^{(j)}(X)$, використовуються формули

$$x_m^{(k)} = x_r^{(i)} + x_s^{(j)},$$

$$f_m^{(k)}(x_m^{(k)}) = f_r^{(i)}(x_r^{(i)}) \cdot f_s^{(j)}(x_s^{(j)})$$

для всіх можливих пар значень з індексами r і s .

Розташовуючи отримані величини $x_m^{(k)}$ в порядку зростання, рівні між собою значення об'єднуються, при цьому відповідні їм значення $f_m^{(k)}$ підсумовуються.

Розглянемо злиття паралельних альтернативних переходів (рис. 3.3).

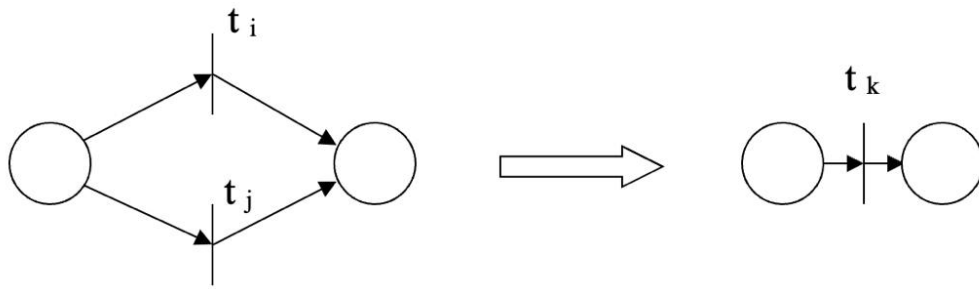


Рисунок 3.3 – Злиття паралельних альтернативних переходів

У разі для нового переходу математичне сподівання, дисперсія часу його спрацьовування та ймовірність спрацьовування визначаються виразами:

$$M(t_k) = \frac{P(t_i) \cdot M(t_i) + P(t_j) \cdot M(t_j)}{P(t_i) + P(t_j)},$$

$$D(t_k) = \frac{P(t_i)[D(t_i) + M^2(t_i)] + P(t_j)[D(t_j) + M^2(t_j)]}{P(t_i) + P(t_j)} - M^2(t_k),$$

$$P(t_k) = P(t_i) + P(t_j).$$

У разі, якщо паралельні альтернативні переходи не є груповими, ймовірності їхнього спрацьовування можна визначити з виразу:

$$P(t_i) = \sum_r f_r^{(i)}(x_r^{(i)}) \prod_{j \neq i} \sum_a f_a^{(j)}(x_a^{(j)})$$

за всіх таких a , що $x_a^{(j)} \geq x_r^{(i)}$.

Зокрема, під час розгляду двох конфліктних переходів, що конфліктують

$$\begin{aligned} P(t_i) &= \sum_r f_r^{(i)}(x_r^{(i)}) \sum_a f_a^{(j)}(x_a^{(j)}) = \\ &= 1 - P(t_j) = 1 - \sum_s f_s^{(j)}(x_s^{(j)}) \sum_b f_b^{(i)}(x_b^{(i)}) \end{aligned}$$

для всіх b , що задовольняють умову $x_b^{(i)} \geq x_s^{(j)}$.

Для визначення нової функції розподілу (у разі злиття паралельних переходів) процедуру впорядкування та об'єднання однакових значень застосовують для об'єднаної сукупності $x_r^{(i)}$ та $x_s^{(j)}$, водночас значення функції $f^{(k)}(X)$ знаходяться, зважаючи на приналежність $x_m^{(k)}$ області визначення тієї чи іншої функції:

$$f_m^{(k)}(x_m^{(k)}) = \begin{cases} f_r^{(i)}(x_r^{(i)}) \cdot \frac{P(t_i)}{P(t_i) + P(t_j)}, & \text{если } x_m^{(k)} = x_r^{(i)}; \\ f_s^{(j)}(x_s^{(j)}) \cdot \frac{P(t_j)}{P(t_i) + P(t_j)}, & \text{если } x_m^{(k)} = x_s^{(j)}. \end{cases}$$

Нарешті, у разі злиття паралельних альтернативних переходів мережі, один із яких входить до складу петлі, як наведено на рис. 3.4), застосовуються такі вирази:

$$\begin{aligned} M(t_k) &= M(t_j) + \frac{P(t_i) \cdot M(t_i)}{1 - P(t_i)}, \\ D(t_k) &= D(t_j) + \frac{P(t_i) \cdot D(t_i)}{1 - P(t_i)} + \frac{P(t_i) \cdot M^2(t_i)}{[1 - P(t_i)]^2}, \\ P(t_k) &= \frac{P(t_j)}{1 - P(t_i)}. \end{aligned}$$

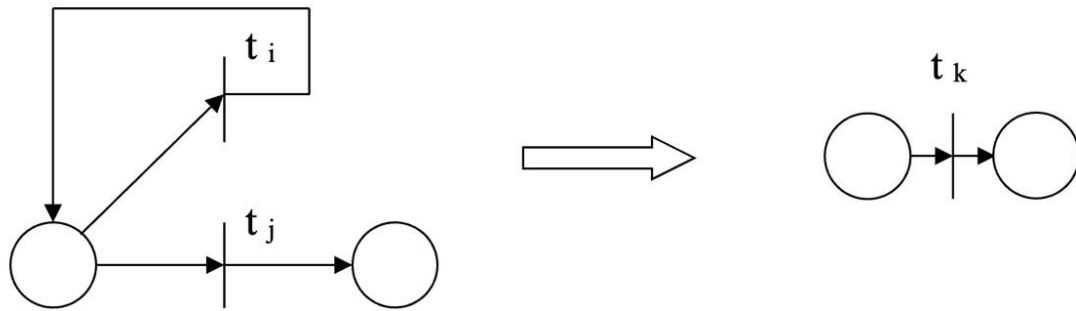


Рисунок 3.4 – Злиття переходів у складі петлі

Оскільки визначення всіх значень $f^{(k)}(X)$ у цьому випадку неможливе, тому що число значень аргументу визначається числом виконання петлі, яке може бути і нескінченним, обмежимо їхню кількість величиною n , виходячи з того, що:

$$P(t_k) = \sum_{r=1}^{\infty} P(t_j) \cdot P(t_i)^{r-1} = \sum_{r=1}^n P(t_j) \cdot P(t_i)^{r-1} + \frac{P(t_j)}{1 - P(t_i)} \cdot P(t_i)^n$$

Розрахунок значень і перевірку виконання цієї умови можна здійснювати рекурентно, на кожному кроці порівнюючи 2 можливі варіанти розвитку процесу. Для 1-го варіанта розрахунок значень ймовірності, математичного сподівання і дисперсії необхідно проводити за формулами послідовного випадку (виходу з петлі не відбувається, обчислені значення використовують на наступному кроці), для 2-го – за формулами паралельного (відбувається вихід з петлі).

Слід зазначити, що під час моделювання за допомогою запропонованого розширення мереж Петрі комп'ютерних систем реального часу, наявність петлі в структурі мережі (або її виявлення під час редукції) свідчить про потенційну небезпеку, пов'язану з можливістю невиконання часових обмежень, що накладаються на виконання певних транзакцій.

Розглянемо метод редукції числа станів часової мережі Петрі на найпростішому прикладі (рис. 3.5).

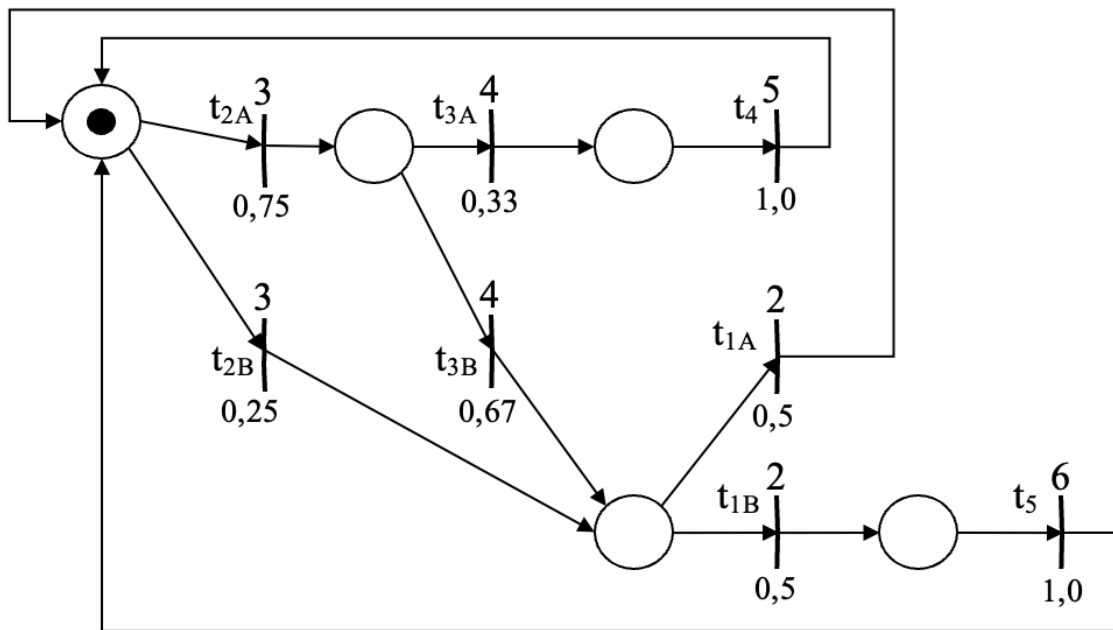
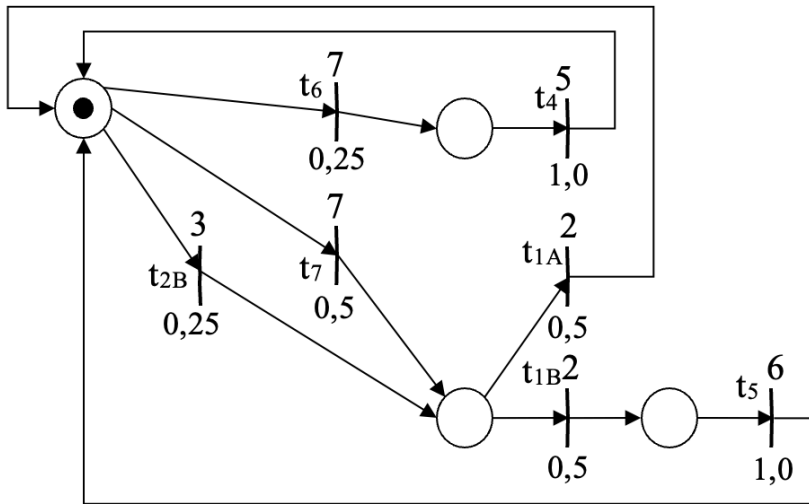


Рисунок 3.5 – Часова мережа Петрі, що моделює виконання циклічної транзакції

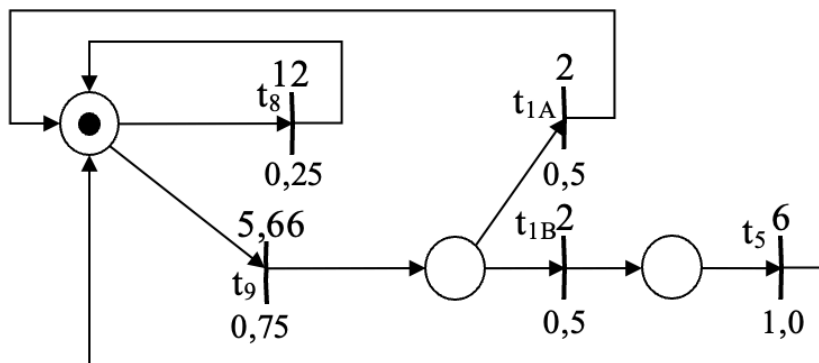
У якості вихідних даних будемо використовувати чисельні значення, наведені на прикладі рис. 2.6. Біля кожного переходу, крім його позначення, вгорі вказуватимемо математичне сподівання часу його спрацьовування, а внизу – ймовірність спрацьовування (групові переходи мають однаковий початковий індекс).

На першому етапі проводиться злиття переходів t_{2A} і t_{3A} , а також t_{2A} і t_{3B} , з використанням формул для послідовного випадку.

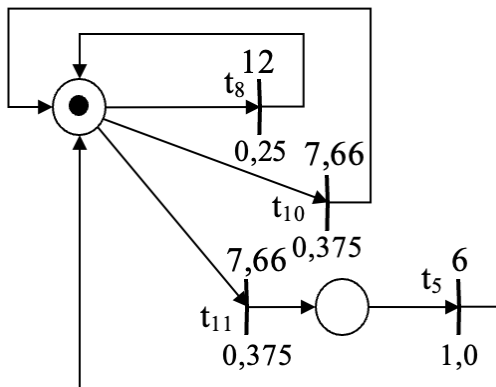
У результаті чого буде отримано мережу Петрі з новими переходами t_6 і t_7 , наведену на рис. 3.6, А.



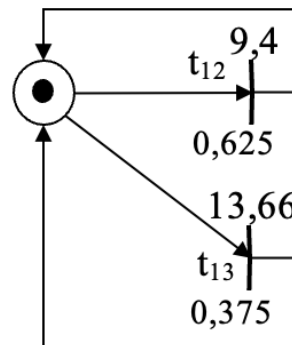
А. Перший етап злиття



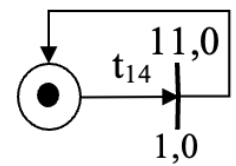
Б. Другий етап злиття



В. Третій етап



Г. Четвертий етап



Д. П'ятий етап

Рисунок 3.6 – Етапи послідовної редукції числа станів
часової мережі Петрі

На другому етапі злиття підлягають переходи t_6 і t_4 (послідовний випадок) і переходи t_{2B} і t_7 (паралельний випадок), з утворенням відповідних переходів t_8 і t_9 (рис. 3.6, Б).

На третьому етапі відбувається злиття послідовних пар переходів t_9, t_{1A} (що утворюють перехід t_{10}), t_9 та t_{1B} (що утворюють перехід t_{11}). На четвертому - паралельної пари t_8, t_{10} і послідовної t_{11} та t_5 .

Остаточне об'єднання переходів t_{12} і t_{13} (рис. 3.6 В, Г, Д) дає змогу одержати значення математичного сподівання часу виконання транзакції, що дорівнює 11.

Злиття переходів, як було зазначено вище, передбачає збільшення первісної розмірності N множин заданих значень $f^{(i)}(X)$: у найгіршому варіанті для паралельного випадку в 2 рази, для послідовного в N разів після першого ж злиття.

Така особливість може викликати певні обчислювальні складнощі під час редукції станів часової мережі Петрі великої розмірності. Для розв'язання цієї проблеми, з огляду на початкову похибку даних, що таблично характеризують функцію щільності ймовірності, пропонується її представлення значеннями в точках дискретизації, утворених не інтервалами Δx , а рівними інтервалами $\Delta f = 1 / N$.

Така дискретизація дає змогу:

- по-перше, не зберігати значення $f_j^{(i)}(x_j^{(i)})$, оскільки вони відомі;
- по-друге, не обчислювати їхні добутки під час згортки (вони всі рівні $1 / N^2$);
- по-третє, після кожного злиття двох переходів і утворення нового, знижувати утворену розмірність множини $f^k(x)$ до первісної шляхом апроксимації значень аргументу, за яких $f_m^k(x_m^k) = 1 / N$.

Таким чином, запропоноване розширення мереж Петрі дає змогу:

- об'єднати моделювальні можливості стохастично-детермінованих часових мереж Петрі та часових мереж моделі Мерліна, зберігаючи при цьому можливості якісного та кількісного аналізу мережевих моделей відомими методами;

- задавати практично будь-які закони розподілу випадкових величин, що характеризують періоди спрацьовування переходів;
- використовувати для аналізу мережі досить прості й ефективні з обчислювальної точки зору алгоритми, що забезпечують отримання як точкових, так і інтервальних оцінок досліджуваних часових процесів.

3.3 Розробка методу короткострокового прогнозу часових параметрів кластера граничного шару Інтернету речей

Розглянуті вище методи дають змогу сформувати оптимальний кластер граничного шару інфраструктури підтримки Інтернету речей. Однак не менш важливим завданням, що стоїть перед розробниками комп'ютерних систем реального часу, є така організація стійкого обчислювального процесу, адаптивного до різного роду відхилень від запланованого процесу, за якого забезпечується виконання застосунків за директивні терміни, або, інакше кажучи, витримуються в певних межах задані показники продуктивності.

Під адаптацією розуміють процес зміни структури, алгоритмів і параметрів системи на основі інформації, отримуваної в процесі її функціонування для досягнення оптимального за прийнятим критерієм стану системи (збереження особливо важливих показників) за наявності невизначеності та мінливості умов роботи у взаємодії із зовнішнім середовищем [112].

В основі адаптації лежить накопичення даних про поведінку системи і зовнішні умови та їх використання для поліпшення обраного показника якості.

Процес накопичення й обробки інформації пов'язаний із витратами часу, що зрештою призводить до певних затримок під час формування рішень, які компенсують непередбачені викиди. Це може істотно знижувати ефективність роботи в реальному масштабі часу. Тому для побудови адаптивних комп'ютерних систем завдання контролю і прогнозування стану системи та зовнішнього середовища є актуальним.

Для цього зазвичай у підпрограмі управління завантаженням комп'ютерної системи, основними функціями якої є визначення поточного рівня завантаження і вжиття заходів для усунення факту перевантаження за продуктивністю, розраховують час переведення комп'ютерної системи в режим підвищеної продуктивності, зокрема, для кластера граничного шару треба спрогнозувати настання моменту перевантаження кластера та задіяти для виконання деяких оперативних транзакцій альтернативні варіанти, наприклад, з використанням вузлів туманного шару.

Поточне завантаження комп'ютерної системи обчислюється за виразом

$$S = S_0 + \sum_i \alpha_i N_i,$$

де S_0 – константа, що визначає завантаження КС транзакціями і запитами, які не залежать від вхідної інформації;

N_i – кількість об'єктів Інтернету речей i -го типу, що підтримуються розглядаємим кластером;

α_i – ваговий коефіцієнт, що враховує вплив на завантаження вузлів кластера особливостей обробки інформації про різні категорії об'єктів Інтернету речей i -го типу.

Під час функціонування мережі Інтернету речей вибір режиму функціонування КС здійснюється на основі порівняння поточного завантаження із заданими фіксованими рівнями. Таким чином, завдання прогнозу продуктивності в КС взагалі не вирішується, а контроль завантаження кластера здійснюють опосередковано, з урахуванням тільки кількості та типів оброблюваних системою заявок.

Проте на часові параметри виконання оперативних транзакцій у середовищі Інтернету речей впливають:

– випадкові динамічні зміни кількості та типу оброблюваних транзакцій IoT, які визначаються зовнішнім середовищем, що може призводити до активації різних процесів, які споживають обчислювальні ресурси;

- наявністю в кожен конкретний момент часу різної кількості та обсягу обчислювальних ресурсів, доступних для використання активним процесом;
- невизначеністю необхідних при обробці навіть конкретної транзакції ресурсних витрат.

У таких умовах своєчасне виявлення критичного завантаження КС (у разі зростання до граничної кількості оброблюваних транзакцій; у разі появи або активації в багатозадачній системі додаткових процесів, що споживають ресурси тощо) для адаптивних систем реального часу має здійснюватися шляхом використання ефективних методів прогнозу параметрів обчислювального процесу, і насамперед його часових характеристик.

Контроль над виконанням паралельних обчислювальних процесів, що ґрунтується на динамічному уточненні статичних прогнозів часу виконання транзакцій, пропонується в [95]. При цьому час виконання кожного модуля розглядається як випадкова величина, розподілена за експоненціальним законом, а в основі рішення на адаптацію лежить просте порівняння вимірених реальних оцінок із розрахованим раніше критичним порогом.

Однак таке припущення, як показує практика, є досить сумнівним. Крім того, запропонований метод не враховує характерної особливості IoT - періодичності виконання окремих завдань і пов'язаної з нею можливості аналізу значень кількох вимірювань за деякий період спостережень з метою виявлення динаміки зміни часових параметрів оперативних транзакцій.

У [30, 68] розглянуто строгу постановку задачі виявлення зміни властивостей часових рядів, критерії та методи її розв'язання:

1. Методи послідовного (у темпі надходження інформації) аналізу:
 - алгоритм кумулятивних сум;
 - алгоритм на основі критерію Неймана-Пірсона;
 - адаптивні алгоритми (експоненціального згладжування, Брауна, Трігга-Ліча, Ч.Хольта, адаптивної авторегресії).
2. Апостеріорні методи:
 - алгоритм на основі критерію Байєса;

– алгоритм моделі авторегресії - ковзного середнього.

Ефективність застосування цих методів визначається наявністю апріорної інформації про вид і параметри розподілу спостережуваної випадкової величини, характером її зміни і значенням встановлюваного критичного значення для обраного параметра. Практично для всіх методів використовується еквідистантна модель вимірювань - через фіксований інтервал часу. Крім того, більшість розглянутих методів визначають «розладку» процесу (перевищення вимірних значень заданого порогу) постфактум і мають певний час запізнювання її виявлення [13].

З метою усунення розглянутих недоліків пропонується новий метод контролю часових параметрів КСРЧ підтримки граничного шару Інтернету речей, що дає змогу в реальному масштабі часу здійснити прогноз часових характеристик системи на основі виявлення тренду (тобто зміни в часі) середнього значення випадкового часу виконання певної транзакції, що періодично викликається.

Для цього оцінки часу виконання транзакції (які можуть бути отримані, наприклад, за допомогою запропонованого вище монітора) представимо у вигляді ряду

$$y_i = f(t_i) + n(t_i), \quad i = 1, 2, \dots, N,$$

де N - кількість проведених вимірювань;

$f(t_i)$ – детермінована функція, що визначає характер зміни часових оцінок (тренд);

$n(t_i)$ – стаціонарна випадкова функція, значення якої зумовлене стохастичною природою часу виконання транзакції, зовнішніми впливами, помилками вимірювання тощо.

Подальші викладки ґрунтуються на припущенні, що моменти вимірювань утворюють пуасонівський потік подій з інтенсивністю λ , а випадкові величини $n(t_i)$ є незалежними й однаково розподіленими з математичним сподіванням, що дорівнює нулю.

Апроксимуємо $f(t_i)$ лінійною функцією

$$f(t_i) = a + b \frac{\lambda t_i}{N} \quad (3.7)$$

при цьому параметри a і b мають однакову розмірність, що збігається з розмірністю процесу $y(t_i)$. Тоді завдання виділення тренду часового ряду

$$y(t_i) = a + b \frac{\lambda t_i}{N} + n(t_i)$$

зводиться до задачі отримання невідомих коефіцієнтів a і b .

Оскільки $t_i = \tau_1 + \tau_2 + \dots + \tau_i$, де τ - інтервал між проведеними вимірами, і для прийнятої моделі вимірювань $P(\tau) = \lambda e^{-\lambda \tau}$,

$$M(t_i) = iM(\tau) = \frac{i}{\lambda},$$

та
$$M(y_i) = a + b \frac{\lambda}{N} M(t_i) = a + b \frac{i}{N}. \quad (3.8)$$

Оцінки \hat{a} і \hat{b} можуть бути знайдені методом найменших квадратів (МНК) за критерієм:

$$\sum_{i=1}^N \left(y_i - \hat{a} - \hat{b} \frac{i}{N} \right)^2 \rightarrow \min_{a,b}.$$

Обчислюючи похідні за \hat{a} і \hat{b} та прирівнюючи їх до нуля, отримуємо наступну систему рівнянь:

$$\begin{cases} \sum_{i=1}^N \left(y_i - \hat{a} - \hat{b} \frac{i}{N} \right) = 0, \\ \sum_{i=1}^N \frac{i}{N} \left(y_i - \hat{a} - \hat{b} \frac{i}{N} \right) = 0, \end{cases}$$

Застосовуючи під час розв'язання цієї системи відомі співвідношення

$$\sum_{i=1}^N i = \frac{N(N+1)}{2}, \quad \sum_{i=1}^N i^2 = \frac{N(N+1)(2N+1)}{6},$$

і упускаючи низку проміжних перетворень, для оцінок \hat{a} і \hat{b} отримуємо такі остаточні вирази

$$\begin{aligned} \hat{a} &= \frac{2(2N+1)}{N(N-1)} L_0 - \frac{6}{N(N-1)} L_1, \\ \hat{b} &= -\frac{6}{N-1} L_0 + \frac{12}{N^2-1} L_1 \end{aligned} \quad (3.9)$$

де

$$L_0 = \sum_{i=1}^N y_i, \quad L_1 = \sum_{i=1}^N i \cdot y_i.$$

Оцінка \hat{b} , як видно з (3.9), визначає нахил прямої, що апроксимує наявні значення, і дає змогу зробити висновок про збільшення (зменшення) середнього значення часу виконання транзакції за N вимірювань. Для визначення довірчого інтервалу для оцінки \hat{b} необхідне знання її закону густини розподілу і значення дисперсії $D[n(t_i)]$. В якості незміщеної оцінки дисперсії може бути взята статистика

$$s^2 = \gamma_0 L_0^2 + \gamma_1 L_0 L_1 + \gamma_2 L_1^2 + \gamma_3 R_0,$$

де

$$R_0 = \sum_{i=1}^N y_i^2,$$

а коефіцієнти, що забезпечують незміщеність, визначаються виразами:

$$\gamma_0 = -\frac{2(10N^2 + 13N + 9)}{N(N-1)(N-2)(5N+1)},$$

$$\gamma_1 = \frac{60(N+1)}{N(N-1)(N-2)(5N+1)},$$

$$\gamma_2 = -\frac{60}{N(N-1)(N-2)(5N+1)},$$

$$\gamma_3 = -\frac{(5N^2 - 4N + 3)}{(N-1)(N-2)(5N+1)}.$$

За великих значень N величина $\frac{s^2(N-2)}{\sigma^2}$ має розподіл χ^2 з $N-2$ ступенями свободи, а закон розподілу оцінки \hat{b} асимптотично наближається до нормального, що дає змогу визначити значення довірчого інтервалу за формулою

$$b_{1,2} = \frac{1}{1 - \frac{6}{5} \frac{g_\alpha^2}{N}} \left(\hat{b} \mp \sqrt{\frac{6}{5} \frac{g_\alpha^2}{N} \hat{b}^2 + 12 \frac{s^2(N-2)}{\chi_{N-2,\alpha}^2} \frac{g_\alpha^2}{N} \left(1 - \frac{6}{5} \frac{g_\alpha^2}{N} \right)} \right), \quad (3.10)$$

де 2α - заданий рівень значущості;

g_α - відповідний квантиль нормального розподілу;

$\chi_{N-2,\alpha}^2$ - відповідний квантиль розподілу χ^2 .

Таким чином, під час практичної реалізації розглянутого методу виявлення тренду на базі МНК необхідна організація «ковзного вікна» вимірюваних значень y_1, \dots, y_N часу виконання періодично запускаємої оперативної транзакції IoT, розрахунок базових статистик L_0 , L_1 та R_0 , і на їхній основі - прогнозного значення середнього часу виконання оперативної транзакції Інтернету речей на майбутній $(N+i)$ -й момент вимірів, відповідно до виразів (3.6) і (3.7).

Розглянутий підхід до отримання прогнозних оцінок може бути розширено для випадку, коли апроксимація вимірюваних даних здійснюється за допомогою не тільки лінійної функції, але й поліномів вищого порядку. При цьому ця задача, розв'язувана методом найменших квадратів, у загальному випадку визначається наступним чином.

Для вихідних даних, поданих у вигляді $\{y_k\}$, $k=1, \dots, N$, потрібно побудувати функцію $f(m)$ виду

$$f(m) = a_0\phi_0(m) + a_1\phi_1(m) + \dots + a_r\phi_r(m)$$

так, щоб мінімізувати зважену середню квадратичну помилку

$$\sigma^2 = \sum_{k=1}^N \gamma_k [f(k) - y_k]^2,$$

де γ_k - задані позитивні ваги.

Якщо функції $\phi_i(m)$ попарно ортогональні між собою з вагами γ_k на заданій множині точок, тобто

$$\sum_{k=1}^N \gamma_k \phi_i(k) \phi_j(k) = 0 \quad \text{для всіх } i \neq j,$$

то коефіцієнти a_i визначаються за формулами

$$a_i = \frac{\sum_{k=1}^N \gamma_k \phi_i(k) y_k}{\sum_{k=1}^N \gamma_k \phi_i^2(k)}.$$

Важливою перевагою використання ортогональних многочленів є те, що поліпшення апроксимації шляхом додавання нового члена не змінює $a_{r+1}\phi_{r+1}(m)$ раніше обчислених коефіцієнтів a_0, a_1, \dots, a_r .

В якості таких багаточленів (за умови рівності всіх ваг $\gamma_k=1$) можуть бути взяті поліноми ступеня i виду

$$\phi_i(m) = p_i \left(\frac{2m-1-N}{N-1} \cdot \frac{N}{2}, N \right),$$

$$p_i(t, N) = \sum_{k=0}^i (-1)^{i+k} \frac{(i+k)^{[2k]} (N/2+t)^{[k]}}{(k!)^2 N^{[k]}},$$

$$z^{[k]} = z(z-1)(z-2)\dots(z-k+1), \quad z^{[0]} = 1, \quad 0^{[k]} = 0.$$

Виходячи з цього, ортогональні багаточлени до третього ступеня матимуть такий вигляд:

$$\phi_0(m) = 1;$$

$$\phi_1(m) = \frac{2}{N-1}m - \frac{N+1}{N-1};$$

$$\phi_2(m) = \frac{6}{(N-1)(N-2)}m^2 - \frac{6(N+1)}{(N-1)(N-2)}m + \frac{(N+1)(N+2)}{(N-1)(N-2)};$$

$$\begin{aligned} \phi_3(m) = & \frac{20}{(N-1)(N-2)(N-3)} \cdot m^3 - \frac{30(N+1)}{(N-1)(N-2)(N-3)} \cdot m^2 + \\ & + \frac{12N^2 + 30N + 22}{(N-1)(N-2)(N-3)} \cdot m - \frac{(N+1)(N+2)(N+3)}{(N-1)(N-2)(N-3)}. \end{aligned}$$

Тоді, враховуючи введені позначення L_0 та L_1 , і визначаючи

$$L_2 = \sum_{k=1}^N k^2 \cdot y_k, \quad L_3 = \sum_{k=1}^N k^3 \cdot y_k,$$

отримаємо:

$$a_0\phi_0(m) = \frac{L_0}{N};$$

$$a_1\phi_1(m) = 3 \cdot \left(\frac{2m - N - 1}{N - 1} \right) \cdot \frac{2L_1 - (N+1)L_0}{N(N+1)} =$$

$$= \left(-\frac{6}{N(N-1)} \cdot L_0 + \frac{12}{N(N^2-1)} \cdot L_1 \right) \cdot m + 3 \frac{(N+1) \cdot L_0 + 2 \cdot L_1}{N(N-1)}.$$

Апроксимуюча функція $f(m)$, складена тільки з цих двох членів, як легко переконатися, повністю аналогічна лінійній функції, вираз для якої впливає з (3.6) і (3.7).

Наступні члени визначаються як

$$\begin{aligned}
 a_2\phi_2(m) &= 5 \cdot \frac{6 \cdot L_2 - 6(N+1) \cdot L_1 + (N+1)(N+2) \cdot L_0}{(N-2)(N-1)N(N+1)(N+2)} \times \\
 &\quad \times (6 \cdot m^2 - 6 \cdot m(N+1) + (N+1)(N+2)); \\
 a_3\phi_3(m) &= 7 \cdot [20 \cdot L_3 - 30(N+1) \cdot L_2 + (12N^2 + 30N + 22) \cdot L_1 - \\
 &\quad - (N+1)(N+2)(N+3) \cdot L_0] \times \\
 &\quad \times \frac{20m^3 - 30m^2(N-1) + m(12N^2 + 30N + 22) - (N+1)(N+2)(N+3)}{(N-3)(N-2)(N-1)N(N+1)(N+2)(N+3)}.
 \end{aligned}$$

Важливо зазначити, що достовірність і точність розв'язання задачі прогнозування істотно залежать від кількості реалізацій N , які витрачено на отримання вихідних статистичних даних.

Внаслідок цього виникає проблема пошуку компромісу між необхідністю збільшення витрат часу і пам'яті на отримання і зберігання необхідної кількості оброблюваних даних, і необхідністю зменшення споживання обчислювальних ресурсів, виходячи з вимог обробки інформації в реальному масштабі часу.

Для вирішення цієї проблеми запропоновано використання рекурентних співвідношень, що базуються на апроксимаційній функції першого порядку і залежать тільки від вимірювання, що надійшло на k -му кроці. Зауважимо, що базові статистики L_0 , L_1 та R_0 можуть бути визначені через їхні попередні значення, отримані на $k-1$ кроці:

$$L_0^{(k)} = L_0^{(k-1)} - y_1 + y_k,$$

$$L_1^{(k)} = L_1^{(k-1)} - L_0^{(k-1)} + y_k N,$$

$$R_0^{(k)} = R_0^{(k-1)} - y_1^2 + y_k^2.$$

При цьому віднімання значення y_1 (та його квадрата), що належить до початку «ковзного вікна» вимірюваних даних для базового методу, замінимо відніманням оцінки математичного сподівання прогнозного ряду значень y_1^* , що визначається за (3.6) на основі поточних параметрів \hat{a} та \hat{b} з використанням додаткового коефіцієнта ϕ , тобто

$$\hat{M}(y_1^*) = \hat{a} + \frac{\hat{b}}{N} \phi = \frac{4N+2-6\phi}{N(N-1)} L_0 + \frac{12\phi-6(N+1)}{N(N^2-1)} L_1. \quad (3.11)$$

Коефіцієнт ϕ , близький до одиниці, визначає деяку точку на прямій, що апроксимує виміряні дані, в околицях початку «ковзного вікна». У разі заміни коефіцієнта ϕ на коефіцієнт β згідно з рівністю $\phi = 1 + \beta(N-1)$ вираз (3.9) набуває особливо простого вигляду:

$$\hat{M}(y_1^*) = \frac{4-6\beta}{N} L_0 + \frac{12\beta-6}{N(N+1)} L_1. \quad (3.12)$$

Таким чином, базові статистики пропонується розраховувати, враховуючи (3.10), за рекурентними виразами виду

$$\begin{aligned} L_0^{(k)} &= L_0^{(k-1)} - \hat{M}(y_1^*) + y_k, \\ L_1^{(k)} &= L_1^{(k-1)} - L_0^{(k-1)} + y_k N, \\ R_0^{(k)} &= R_0^{(k-1)} - \hat{M}^2(y_1^*) + y_k^2. \end{aligned} \quad (3.13)$$

З'ясуємо фізичний зміст коефіцієнта β . Враховуючи, що

$$L_0 = \sum_{i=1}^N y_i,$$

$$L_1 = \sum_{i=1}^N i \cdot y_i,$$

вираз (3.10) може бути переписано у вигляді

$$\hat{M}(y_1^*) = \sum_{i=1}^N K_i y_i,$$

де
$$K_i = \frac{(4 - 6\beta)(N + 1) + i \cdot (12\beta - 6)}{N(N + 1)}.$$

Зазначимо, що за будь-яких β

$$\sum_{i=1}^N K_i = 4 - 6\beta + \frac{12\beta - 6}{N(N + 1)} \sum_{i=1}^N i = 1.$$

Таким чином, коефіцієнт β , який надалі називатимемо коригувальним, визначає ваги вихідних значень часового ряду, що використовуються для отримання оцінки математичного сподівання прогнозного ряду значень y_1^* . Наприклад, за $\beta=0,5$ коефіцієнти $K_1 = K_2 = \dots = K_N = 1/N$.

Вибір конкретного чисельного значення коригувального коефіцієнта залежить від характеру даних прогнозованого часового ряду і на практиці може бути здійснено з використанням критерію мінімуму середньої суми квадратів помилок прогнозу, отриманої на контрольній вибірці.

За відсутності такої можливості, виходячи з формули для K_1 - вагового коефіцієнта при y_1 :

$$K_1 = \frac{4N - 2 - 6\beta(N - 1)}{N(N + 1)},$$

вираз для коригувального коефіцієнта визначається як

$$\beta = -\frac{K_1 N^2 + K_1 N - 4N + 2}{6(N - 1)}.$$

Задаючись $K_1=1$ (з метою максимального врахування істинного значення y_1 при визначенні оцінки математичного сподівання зворотного прогнозного значення y_1^*), отримаємо, що

$$\beta = -\frac{N-2}{6}.$$

При цьому, наприклад, у випадку $N=5$:

$$K_1=1; K_2=0,6; K_3=0,2; K_4=-0,2; K_5=-0,6.$$

Приклад вимірювань часу виконання реальної транзакції IoT, оцінка його середнього значення та розрахований довірчий інтервал для оцінок \hat{b} нахилу прямої за формулами (3.7), (3.8) наведено на рис. 3.6, 3.7, 3.8, а за формулами (3.10), (3.11) - на рис. 3.9, 3.10.

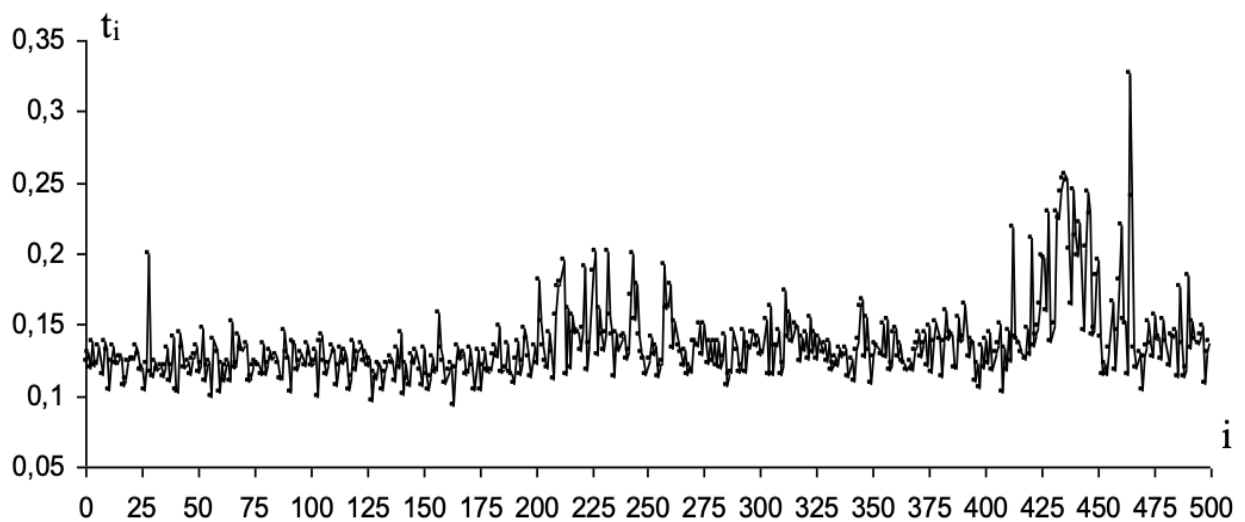


Рисунок 3.7 – Заміри часу виконання (сек) транзакції IoT,
що періодично викликається (500 вимірювань)

Як показує аналіз наведених графіків, застосування рекурентних формул дещо розширює межі довірчого інтервалу, що призводить до незначного погіршення точності одержуваних оцінок нахилу прямої, що апроксимує тренд. З іншого боку, заміна точних оцінок згладженими, збільшує стійкість алгоритму загалом до аномальних викидів вимірювань і призводить до зменшення середнього значення помилок прогнозу.

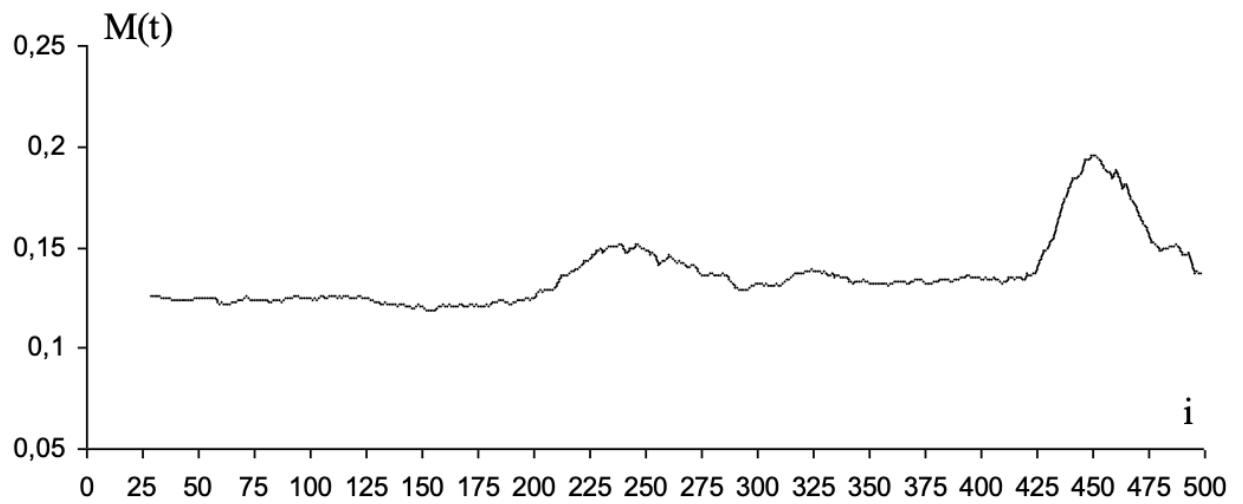


Рисунок 3.8 – Зміна середнього часу виконання транзакції IoT
(базовий метод на основі МНК, $N=30$)

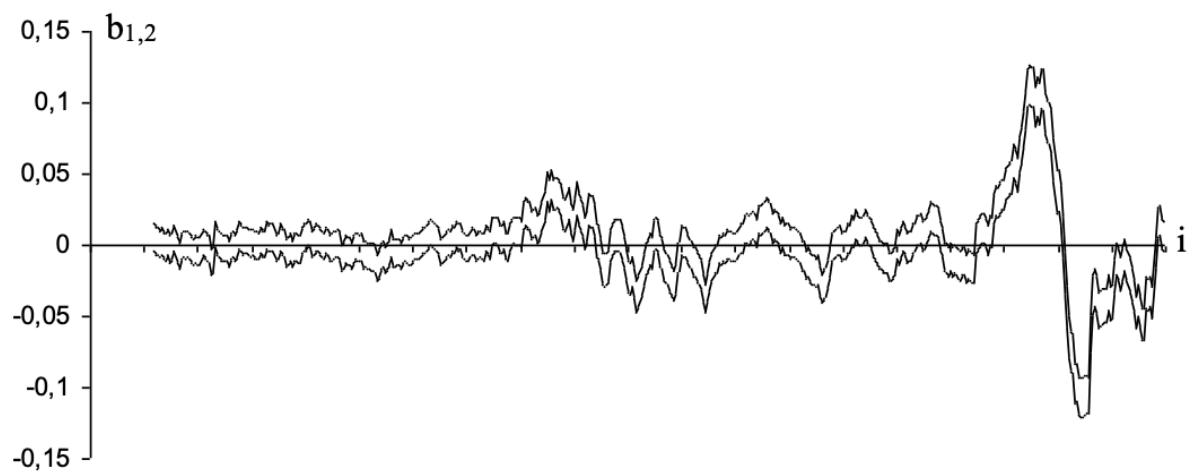


Рисунок 3.9 – Верхня і нижня межі довірчого інтервалу для оцінок \hat{b}
(базовий метод, рівень значущості $\alpha=0,05$)

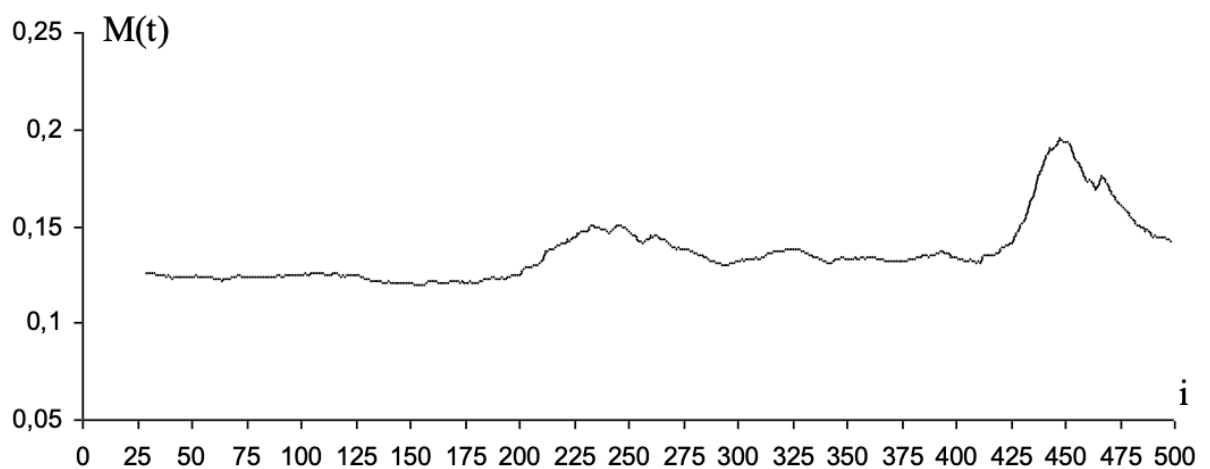


Рисунок 3.10 – Зміна середнього часу виконання транзакції IoT
(запропонований метод, $N=30$, $\beta=-0,5$)

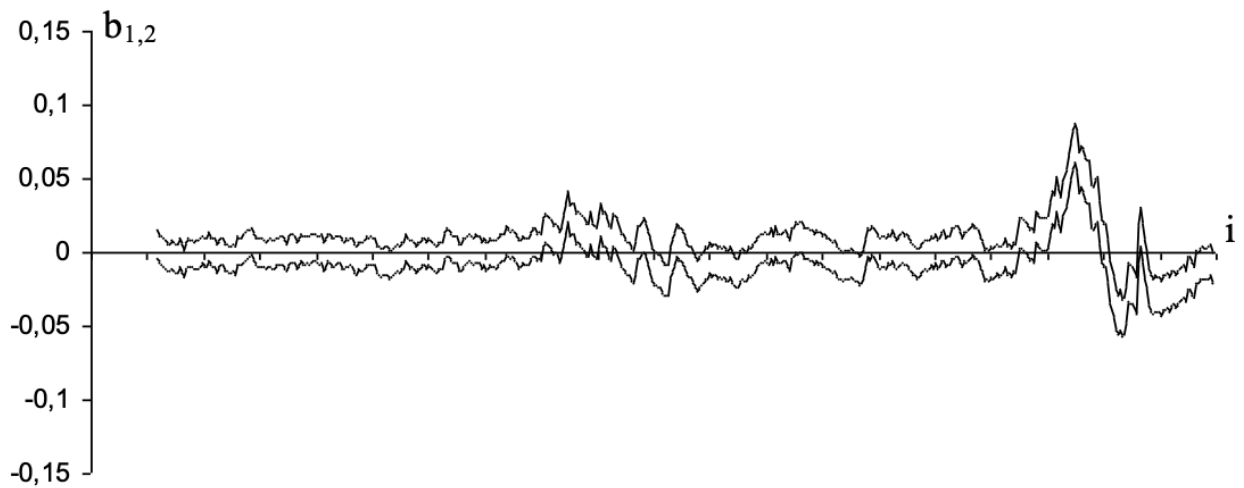


Рисунок 3.11 – Верхня і нижня межі довірчого інтервалу для оцінок \hat{b}
(запропонований метод, $N=30$, $\beta=-0,5$)

Таким чином, використання оцінок тренда, одержуваних за допомогою запропонованих рекурентних виразів, дає змогу здійснити прогнозування часу виконання окремих оперативних транзакцій у реальному масштабі часу за мінімального споживання обчислювальних ресурсів. Крім аналізу часових характеристик, розглянутий метод можна застосувати і для контролю інших параметрів обчислювального процесу, наприклад, розміру вільної оперативної пам'яті. На основі виконаного прогнозу стає можливим адаптивне управління ходом виконання застосунків кластера, що передбачає динамічний перерозподіл обчислювальних ресурсів.

Висновки за розділом 3

1. Виявлені особливості часових мереж Петрі, що використовуються при описі мережних моделей комп'ютерних систем підтримки граничного шару мережі IoT, дали змогу застосувати для їхнього кількісного аналізу метод, що базується на описі модельованих процесів як напівмарковських. У його основі лежить визначення функцій розподілу часу перебування напівмарковського процесу в окремих станах, що дає змогу розрахувати

значення перехідних ймовірностей і граничних розподілів ймовірностей досяжних станів мережі.

2. Запропонований метод редукції кількості станів часової мережі Петрі, який узагальнює відомі підходи до аналізу графових і мережних моделей, та дає змогу одержати значення характеристик обраних станів без побудови й розв'язання відповідних систем рівнянь. Метод дозволяє суттєво зменшити обчислювальну складність визначення функціональних характеристик кластера за рахунок зменшення кількості аналізуємих станів.

3. Отримав подальший розвиток метод формування кластера граничного шару комп'ютерної системи підтримки Інтернету речей шляхом використання функціональної моделі кластера та апарату полумарковських процесів з можливою редукцією кількості станів мережі Петрі, що дозволило зменшити середній час обробки оперативних транзакцій, що надходять до граничного шару Інтернету речей. Запропонований метод дозволяє визначити оптимальний варіант розподілу застосунків по вузлах граничного шару за критерієм мінімального середнього часу виконання одного застосунку.

4. Удосконалений метод короткострокового прогнозу часових параметрів кластера граничного шару Інтернету речей, який враховує обмежені обчислювальні можливості вузлів граничного шару Інтернету речей та базується на відповідній функціональній моделі, що дозволило підвищити точність прогнозу при плануванні оперативного перерозподілу ресурсів системи підтримки Інтернету речей. В основі методу лежить виявлення тренду середнього значення випадкового часу виконання оперативних транзакцій IoT. Визначення характеристик тренду здійснюється за допомогою розроблених на основі методу найменших квадратів рекурентних формул, що дають змогу одержувати в реальному масштабі часу прогнозні оцінки та їх довірчі межі, на основі яких здійснюється адаптивне до завантаження обчислювальної системи керування перебігом обчислень.

РОЗДІЛ 4

ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ ГРАНИЧНОГО ШАРУ КОМП'ЮТЕРНОЇ СИСТЕМИ ПІДТРИМКИ ІНТЕРНЕТУ РЕЧЕЙ

4.1 Опис вхідних даних для побудови функціональної моделі кластера граничного шару системи підтримки IoT

Для оцінки та порівняння можливостей запропонованих методів був сформований кластер граничного шару Інтернету речей на базі інтелектуального шлюзу, який пропонує потужні можливості для обчислень IoT, що можуть підвищити ефективність усього процесу передачі та обробки даних, зменшити затримку при реалізації транзакцій IoT. Розгортання граничних обчислювальних шлюзів може допомогти прийняти рішення швидше й автономніше, оскільки розуміння отримується в джерелі даних, що може зменшити затримку; в той же час, оскільки менше даних зберігається централізовано, це зменшує витрати на зберігання та управління централізованим управлінням даними; граничні обчислення також забезпечують покращену безпеку та конфіденційність, оскільки вони не зберігають і не передають дрібні дані.

Шлюзи даних IoT призначені для прийому різномірних даних від польових пристроїв (лічильників, контролерів, інтелектуальних датчиків, виконавчих механізмів та іншої побутової та промислової периферії) по кількох різних каналах зв'язку, їх обробки та подальшого надсилання у вищі шари інфраструктури підтримки або хмарні сервіси. Інтелектуальні шлюзи можуть виконувати функції перетворення протоколів та локального або віддаленого керування. Дані можуть накопичуватися, інтегруватися та архівуватися у локальних чи віддалених базах даних чи сховищах. Поняття граничного контролера свідчить про проміжне розташування такого шлюзу – між пристроями Інтернету речей і туманними чи хмарними сервісами верхніх рівнів, тобто "на границі". Головні їх завдання полягають у зменшенні потоку

інформації з рівня пристроїв Інтернету речей нагору за рахунок первинної обробки даних та інтегрування та оперативній обробці інформації транзакцій IoT, котрі повинні виконуватися у режимі реального часу або у режимі, який наближений до реального часу.

На сьогодні існує багато різноманітних інтелектуальних комунікаційних шлюзів IoT. Вони класифікуються:

- за способом передачі даних: провідні, бездровідні, комбіновані;
- за рівнем інтелектуальності: базові (передають дані без обробки); інтелектуальні (мають потужні процесори, підтримують Edge Computing, машинне навчання, аналіз даних);
- за сферою застосування: промислові (працюють у важких умовах), Smart City (збирають дані з датчиків трафіку, освітлення, камер відеоспостереження), транспортні (використовуються для моніторингу транспорту, GPS, підключення CAN-шини), енергетичні (контролюють розподіл енергії, інтегруються в SCADA);
- за підтримкою протоколів.

У якості базового шлюзу для кластера був обраний інтелектуальний комунікаційний шлюз ECU-1051TL на базі RISC-процесора (рис. 4.1) тайваньської компанії Advantech, яка на сьогодні є світовим лідером в галузі рішень для промислового Інтернету речей. На вибір вплинуло те, що на офіційному сайті компанії міститься детальна інформація про даний шлюз та результати його тестувальних випробувань для різних конфігурацій пристроїв IoT. ECU-1051TL має відкриту архітектуру платформи з процесором Cortex A8, оснащений двома портами



Рисунок 4.1 – ECU-1051TL

10/100 Ethernet, двома RS-232/485, одним портом USB 2.0 та повнорозмірним слотом MiniPCIE для можливого розширення. Операційна система Linux і відкритий Edgeline SDK дозволяють системним інтеграторам розробляти застосунки, які точно відповідають їх потребам. Пристрій поставляється з встановленим пакетом EdgeLink на 2000 тегів, який дає можливість без навичок програмування налаштувати прийом, обробку, архівування та відправлення даних (підтримуються понад 200 протоколів) та в найкоротші терміни перетворити ECU-1051TL на комунікаційний вузол необхідної конфігурації IoT. Основні характеристики даного шлюзу наведені у табл. 4.1.

Таблиця 4.1 – Характеристики шлюзу Advantech ECU-1051TL

Характеристика	Опис
Процесор	ARM Cortex-A8, 800 МГц
Оперативна пам'ять (RAM)	256 МБ DDR3
Пам'ять зберігання	512 МБ NAND Flash + слот microSD
Операційна система	Вбудований Linux
Ethernet	2 × 10/100 Mbps
Послідовні порти	2 × RS-232/422/485
Цифрові входи/виходи (DI/DO)	2 DI / 2 DO
USB	1 × USB 2.0
Бездротовий зв'язок	Підтримка 4G LTE через Mini-PCIE (опціонально)
Підтримувані протоколи	Modbus RTU/TCP, MQTT, OPC UA, IEC 60870-5-104
Живлення	9–36V DC
Діапазон робочих температур	-40°C ~ 70°C
Корпус та захист	Металевий, захист IP30
Монтаж	DIN-рейка або настінний монтаж
Віддалене керування	Підтримка OTA-оновлень, VPN, SSL/TLS

Шлюз обробляє інформацію, що надходить з 255 пристроїв IoT, з яких 20 пристроїв обладнані одноплатними комп'ютерами та можуть бути задіяні для формування кластера граничного шару. Серед множини транзакцій, які надходять до даного шлюзу, 23 транзакції повинні виконуватися у режимі

реального часу або у режимі, наближеному до реального часу, тобто ті транзакції шлюз повинен надсилати для обробки до відповідного кластеру граничного шару. Для обробки даних оперативних транзакцій використовуються 12 застосунків. Кожен застосунок виконується на окремому вузлі кластера, тобто треба сформувати кластер із 12 вузлів. Деякі транзакції потребують послідовного виконання кількох застосунків.

Надалі застосунки будемо позначати буквами A, B, C, D, E, F, G, H, I, J, K та L. Будемо вважати, що кожна з оперативних транзакцій має можливість негайного доступу через шлюз до кожного з 12 вузлів кластеру (при цьому кожен задіяний вузол однозначно відповідає як поточному стану системи, так і поточному активованому застосунку).

Подальше опрацювання транзакції залежить від її логіки виконання, тобто може бути здійснений перехід або до іншого вузла, або до шлюзу у разі закінчення опрацювання транзакції.

4.2 Синтез функціональної моделі кластера граничного шару

Синтез функціональної моделі проводиться на базі трасувальних даних, отриманих за допомогою вимірювального монітору. Ці дані є послідовністю і часом викликів застосунків, задіяних при виконанні оперативних транзакцій, що надходять на обробку, що були отримані при 50 тестових прогонах фрагмента мережі IoT, який підтримується обраним шлюзом. Загальна кількість викликів застосунків за даний часовий період становила 7652.

Функціональна модель кластеру, що не враховує передісторію викликів застосунків, задається вектором кількості прогонів застосунків, наведеним в табл. 4.2 (отриманим в результаті тестового прогону) та подієвим графом, який повністю описується матрицею перехідних ймовірностей, отриманої в результаті аналізу даних трасування і наведених в табл. 4.3 ймовірностей переходів моделі між станами без урахування передісторії. При цьому

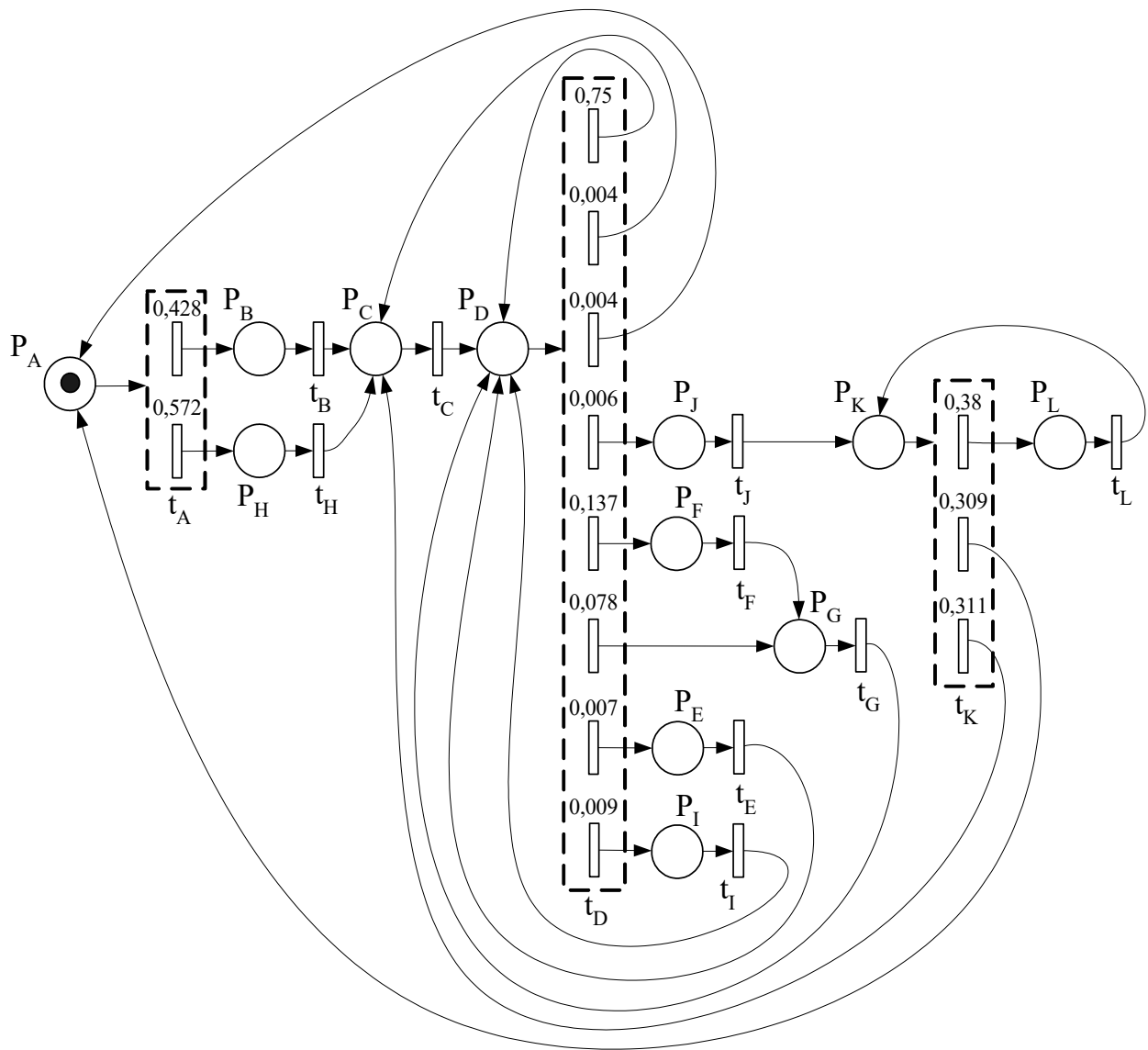


Рисунок 4.2 – Функціональна модель кластера граничного шару системи підтримки IoT у термінах апарату мереж Петрі

Для перевірки адекватності класичної моделі на основі СМО та запропонованої функціональної моделі з використанням апарату мереж Петрі, відповідно до ітеративної схеми, розглянутої в п. 2.2 дисертаційної роботи, було проведено імітаційне моделювання з числом експериментів, відповідним вихідним даним, в результаті чого були отримані ймовірнісні характеристики для кількості прогонів задіяних застосунків, котрі наведені в табл. 4.4 та 4.5 відповідно.

Ступінь відхилення результатів моделювання від експериментальних визначалася за допомогою χ^2 критерію Пірсона.

Таблиця 4.4 – Кількість прогонів застосунків (класична модель з використанням СМО)

Застосунок	A	B	C	D	E	F
Частота	2011	181	1002	1426	226	482
Ймовірність	0,264	0,0238	0,1315	0,1872	0,0297	0,0633
Відсоток	26,4	2,38	13,15	18,72	2,97	6,33
Відхилення (%)	0,008367	0,275672	0,54619	1,47797	0,371212	0,287915

Застосунок	G	H	I	J	K	L
Частота	841	117	79	212	820	221
Ймовірність	0,0978	0,0199	0,0041	0,009	0,094	0,0171
Відсоток	9,78	1,99	0,41	0,9	9,4	1,71
Відхилення (%)	0,143804	0,131494	0,381635	1,271367	0,171896	0,48831
χ^2 (%): 5,555832						

Таблиця 4.5 – Кількість прогонів застосунків (запропонована модель з використанням апарату мереж Петрі)

Застосунок	A	B	C	D	E	F
Частота	2001	215	1321	1799	181	277
Ймовірність	0,2609	0,028	0,1722	0,2346	0,0236	0,0361
Відсоток	26,09	2,8	17,22	23,46	2,36	3,61
	0,000981	0,054321	0,112201	0,011526	0,082034	0,519917

Застосунок	G	H	I	J	K	L
Частота	801	170	46	63	684	112
Ймовірність	0,1044	0,0222	0,006	0,0082	0,0892	0,0146
Відсоток	10,44	2,22	0,6	0,82	8,92	1,46
	0,041724	0,023829	0,060167	0,007805	0,02583	0,042808
χ^2 (%): 0,983143						

Як бачимо, застосування запропонованого методу синтезу функціональної моделі з використанням апарату мереж Петрі дозволило суттєво підвищити адекватність моделей граничного шару Інтернету речей за рахунок використання значень перехідних ймовірностей між станами мережі,

що моделюється. Зокрема, у розглянутому прикладі розходження за ймовірностями станів склало менше 1% при використанні запропонованого методу, тоді, як розходження при класичному методі перевищило 5%.

4.3 Вибір оптимального варіанту формування кластера граничного шару

Для визначення оптимального варіанту формування кластера граничного шару скористаємося результатами функціонального моделювання, зокрема отриманими оцінками щодо ймовірностей станів для даного кластера. Рішення можна отримати за допомогою оптимізаційної задачі (3.6), але навіть при невеликих значеннях параметрів m та n вона має дуже велику обчислювальну складність порядку $m!/(m! \cdot (m - n)!)$. Тому пропонується швидкий метод формування кластера, наближеного до оптимального.

Врахуємо, що на локальному фрагменті граничного шару більшість інформації через шлюз надсилається відразу для опрацювання до туманного або хмарного шарів, не задіяючи при цьому пристрої граничного шару, котрі в основному орієнтуються на виконання оперативних транзакцій. Тому затримки за рахунок передачі даних та очікування доступу до граничних вузлів будуть значно меншими, ніж час обробки запитів транзакцій на вузлах граничного шару, тим паче швидкісні характеристики обчислювальних пристроїв даних вузлів є доволі незначними. Отже, при розподілі застосунків по вузлах кластеру перевагу будуть мати ті застосунки, котрі виконуються частіше.

Для формування кластера можна задіяти 20 пристроїв IoT із множини $K = \{k_j, j = \overline{1,20}\}$, причому продуктивність кожного вузла визначається функцією $\varphi(k_j)$, заданою в умовних одиницях продуктивності. Упорядкуємо значення продуктивності за убuwанням, сформувавши відповідні вектори $\Phi = (j_1, j_2, \dots, j_{20})$ та $PR = (\varphi(k_{j_1}), \varphi(k_{j_2}), \dots, \varphi(k_{j_{20}}))$:

$$\Phi = (7, 11, 19, 20, 3, 5, 4, 8, 1, 18, 17, 2, 16, 7, 9, 15, 12, 14, 6, 10);$$

$PR = (92, 92, 74, 70, 55, 55, 55, 50, 45, 45, 32, 32, 30, 30, 30, 18, 18, 18, 12, 8)$.

Визначимо нижню границю продуктивності, необхідної для виконання пакету застосунків: $\varphi_{low} = 25$ (умовних одиниць). Виходячи з цієї оцінки, задіяти для формування кластеру можна лише 15 пристроїв IoT.

Упорядкуємо список задіяних застосунків за убутанням ймовірності вибору згідно з результатами моделювання з використанням апарату мереж Петрі, наведеними у табл. 4.5. Результат упорядкування наведений у табл. 4.6.

Таблиця 4.6 – Упорядкований список застосунків

Засто- сунок	A	D	C	G	K	F	B	E	H	L	J	I
Відсоток заван- тажень	26,09	23,46	17,22	10,44	8,92	3,61	2,8	2,36	2,22	1,46	0,82	0,6

Далі першим 12 пристроям IoT ставимо у відповідність транзакції із табл. 4.6. Схема результуючого кластера наведена на рис. 4.3.

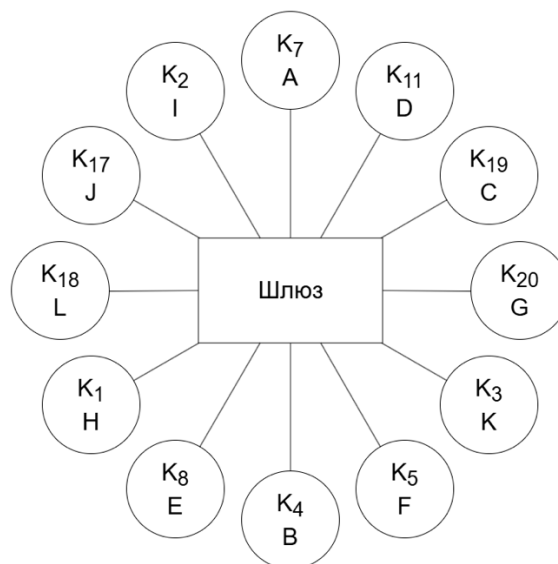


Рисунок 4.3 – Схема розподілу застосунків по вузлах кластера

Зауважимо, що у випадку нестачі пристроїв IoT, котрі можуть виконувати необхідні застосунки, застосунки з найнижчим прогнозованим відсотком завантажень розподіляються на вузли туманного шару.

Для порівняння класичного та запропонованого методів формування кластера граничного шару системи підтримки IoT за критерієм середнього часу виконання одного застосунку $t_{сер}(\gamma_\ell)$ було проведено моделювання з однаковими вхідними даними та різною кількістю застосунків, задіяних для виконання оперативних транзакцій. Узагальнені результати моделювання після 50 прогонів моделі наведені у табл. 4.7.

Таблиця 4.7 – Узагальнені результати моделювання

Кількість застосунків	4	8	12	16	20	24	28	32	36	40
Класичний метод	1,2	1,9	2,7	3,9	5,8	7,3	9,4	10,8	14,3	17,6
Запропонований метод	1,1	1,6	2,2	2,9	4,0	5,1	6,2	7,4	8,7	9,9

Результати моделювання показали, що із зростанням кількості застосунків, задіяних у кластері для виконання оперативних транзакцій, переваги запропонованого методу формування кластера граничного шару системи підтримки IoT за часовими характеристиками збільшуються, тобто його ефективність підвищується. Але зі збільшенням кількості застосунків суттєво збільшується обчислювальна складність запропонованого методу.

Для зменшення обчислювальної складності пропонується зменшити кількість станів моделі, використовуючи метод редукції кількості станів, запропонований у підрозділі 3.2. При цьому, враховуючи особливості пристроїв граничного шару та організації зв'язків, у кластері поетапно використовувались операції злиття послідовних переходів та паралельних альтернативних переходів, а також злиття переходів у складі петлі. Узагальнені результати моделювання після 50 прогонів моделі наведені у табл. 4.8.

Таблиця 4.8 – Узагальнені результати моделювання після проведення редукції числа станів мережі Петрі

Кількість застосунків	4	8	12	16	20	24	28	32	36	40
Аналізуємий метод	1,1	1,6	2,3	3,1	4,4	5,5	6,8	8,3	10,6	13,5

На рис. 4.4 графічно відображені узагальнені результати проведених моделювань.



Рисунок 4.4 – Залежність середнього часу виконання застосунку від кількості задіяних застосунків при різних методах формування кластера граничного шару

Отримані результати підтверджують ефективність запропонованого методу, особливо при зростанні розміру кластера. Крім того, бачимо, що для зменшення обчислювальної складності при формуванні кластера можна використовувати метод редукції кількості станів, при цьому ефективність обробки оперативних транзакцій теж збільшується.

4.4 Оцінка ефективності короткострокового прогнозу часових параметрів кластера граничного шару Інтернету речей

Аналіз розробленого в дисертації методу прогнозу часових параметрів кластера, в основі якого лежить прогнозування даних часового ряду, будемо проводити, починаючи з вивчення реакції класичного та запропонованого методів короткострокового прогнозування на стандартні збурення – імпульс, ступінчаста зміну рівня та на вхідний потік із лінійним зростанням (рис. 4.5 – 4.7). Для порівняння використовувалися методи прогнозування, умовні позначення яких на рисунках відповідають:

A – базовий класичний метод прогнозування за методом найменших квадратів (МНК) з використанням поліноміальної моделі першого порядку з кількістю вимірювань $N = 8$;

B – кількість вимірювань $N = 8$, коригувальний коефіцієнт $\beta = 0$;

C – кількість вимірювань $N = 8$, коригувальний коефіцієнт $\beta = 0,2$;

D – кількість вимірювань $N = 8$, коригувальний коефіцієнт $\beta = -0,2$.

На графіках реальний трафік позначений як R.

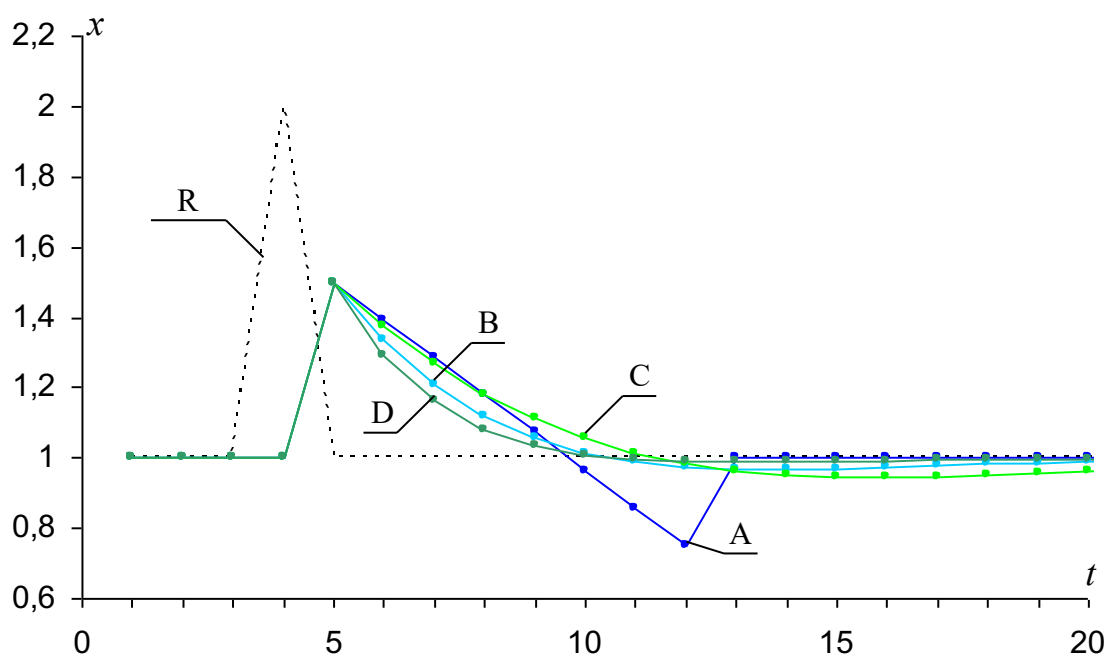


Рисунок 4.5 – Реакції прогнозних моделей на імпульс

З рис. 4.5 видно, що реакція запропонованого методу на імпульсний вплив краща, ніж у базовому класичному методі, швидкість повернення моделі у вихідний стан дещо вища, а середня абсолютна помилка прогнозу у розробленого рекурентного методу виявляється суттєво нижчою.

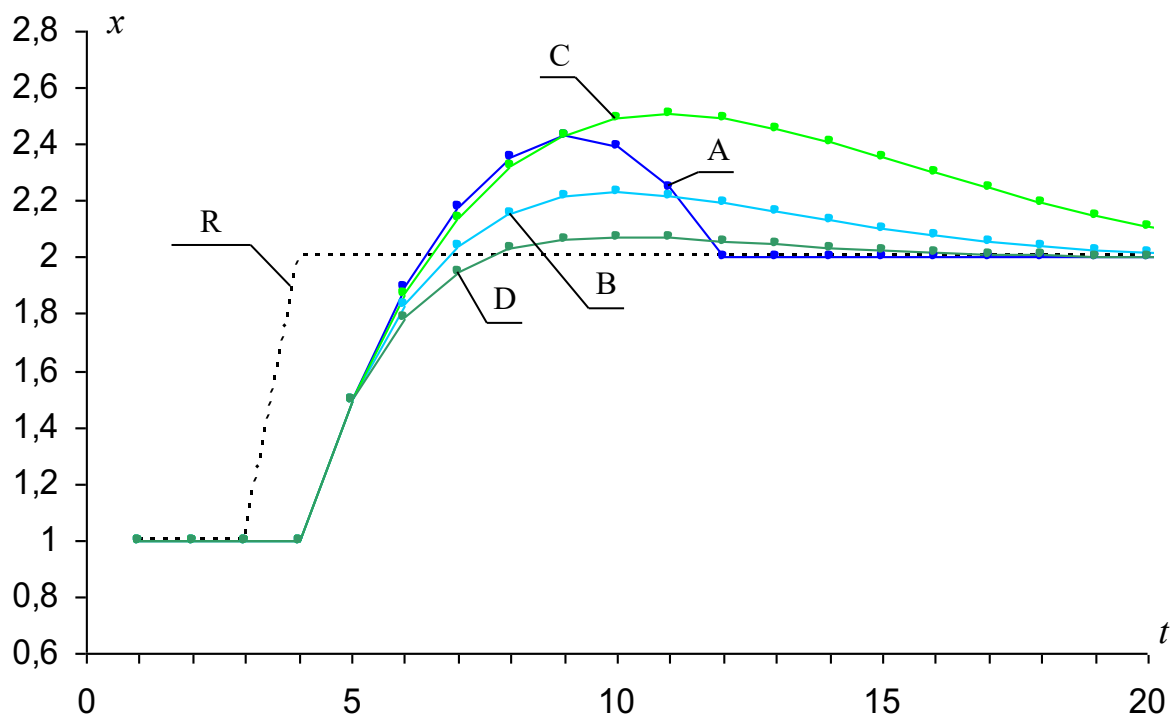


Рисунок 4.6 – Реакції прогнозних моделей на ступінчасту зміну рівня

Аналіз реакцій прогнозних моделей на ступінчасту зміну (рис. 4.6) показує, що прогнози, котрі отримані за розробленим рекурентним методом, спочатку перевищують рівень ступінчастого стрибка і потім наближаються до нього зверху, проте значно швидше. Це свідчить про високі адаптаційні можливості розробленого методу.

Графіки на рис. 4.7 дозволяють оцінити реакції прогнозних моделей на лінійно-наростаючий вхідний потік. Їх аналіз показує, що характер реакції пропонованого рекурентного методу істотно залежить від параметрів, що вибираються. При певних значеннях швидкість наближення прогнозів до дійсних даних виявляється приблизно такою, як у базовій класичній моделі на основі методу найменших квадратів.

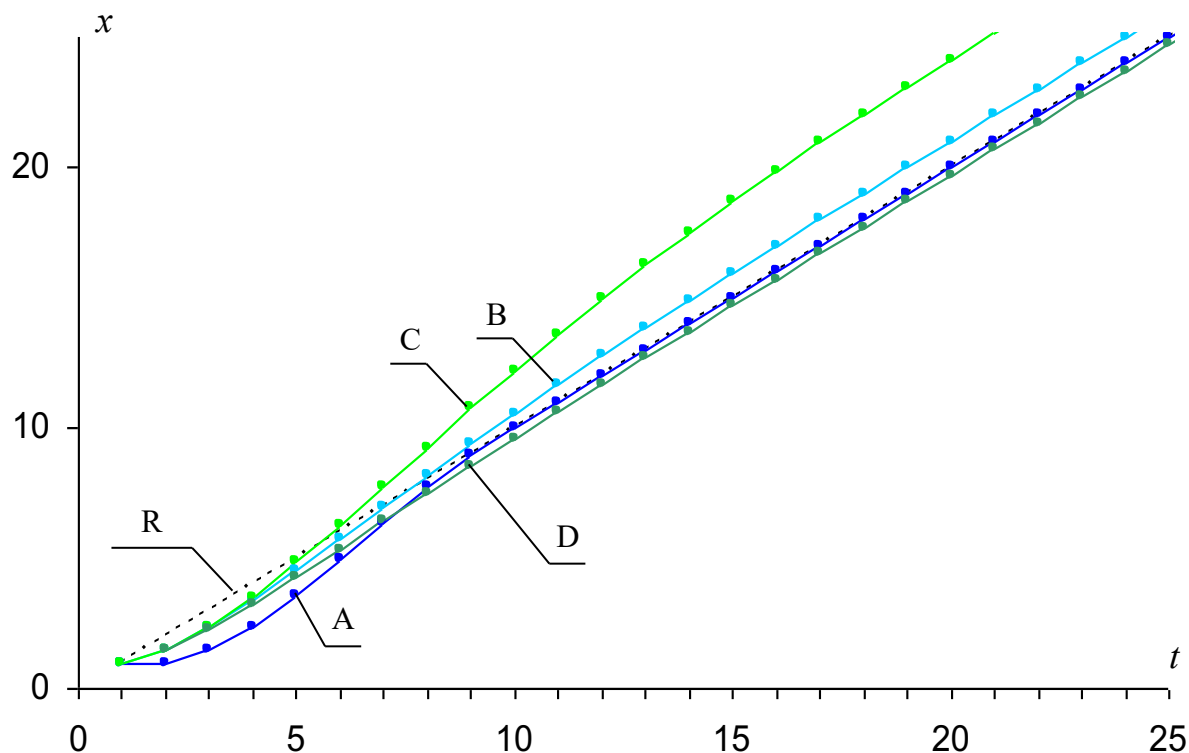


Рисунок 4.7 – Реакції прогнозних моделей
на лінійно-наростаючий вхідний потік

Розглянуті динамічні характеристики пропонованого рекурентного методу прогнозування дозволяють зробити припущення про його досить високі прогнозні та адаптаційні можливості, враховуючи наявність параметрів, що настраюються, і можливість їх варіювання, виходячи з властивостей конкретних часових рядів.

Проведені експерименти свідчать про можливість налаштування оптимальних параметрів розробленого методу при застосуванні у конкретних умовах на основі контрольної вибірки. При цьому слід враховувати залежність розміру довірчого інтервалу Δ оцінки нахилу прогнозуючої прямої від значення N – числа вимірювань, що використовуються для формування короткострокового прогнозу завантаженості кластера.

Графік такої залежності при різних постійних значеннях незміщеної оцінки дисперсії значень часового ряду представлено на рис. 4.8.

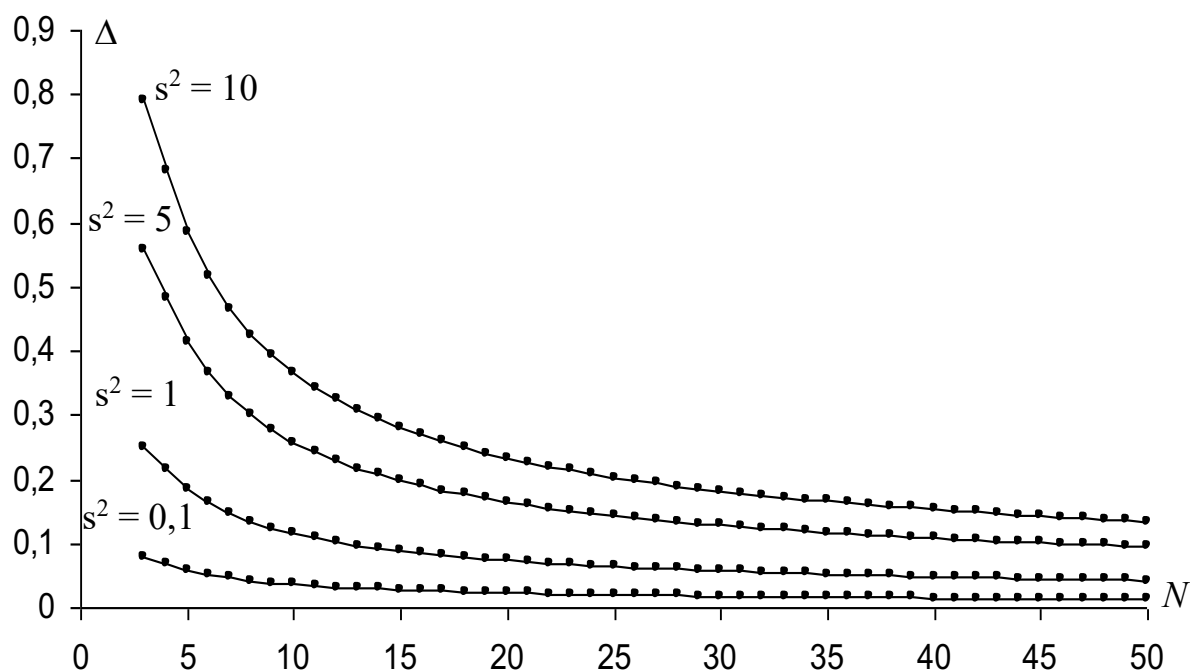


Рисунок 4.8 – Залежність розміру довірчого інтервалу від кількості вимірювань, що враховуються, при стаціонарній дисперсії для рівня значимості $\alpha = 0,05$

Як бачимо із наведених графіків, у більшості практичних випадків прогнозування часових рядів можна рекомендувати значення N , що не перевищує 30–35. Вибір більшого значення з метою зменшення розміру довірчого інтервалу одержуваних оцінок буде невиправданим, враховуючи основне завдання методу прогнозування – мінімізацію помилки прогнозу, оцінювану через середню суму квадратів помилок прогнозу (ССКПП).

Завдання вибору оптимальних параметрів розробленого методу у конкретних умовах ілюструється прикладом використання статистичних даних часу виконання реального застосунку (табл. 4.9). Залежність ССКПП характеризується графіками, представленими на рис. 4.9, 4.10. Аналіз наведених залежностей показує наявність єдиного мінімуму ССКПП, що розглядається як функція від параметрів розробленого методу прогнозування, і свідчить про можливість досить простої послідовної оптимізації.

Таким чином, виявлені властивості розробленого методу прогнозування свідчать про його високу ефективність при вирішенні завдань прогнозу

динаміки часових рядів різного виду, у тому числі й часових параметрів кластерів граничного шару системи підтримки Інтернету речей.

Таблиця 4.9 – Час виконання застосунку на вузлі граничного шару системи підтримки Інтернету речей (в сотнях мілісекунд)

№ ви- міру	Час	№ ви- міру	Час	№ ви- міру	Час	№ ви- міру	Час	№ ви- міру	Час
1	13,36261	21	11,62846	41	10,86812	61	17,74213	81	13,255
2	11,42076	22	13,34119	42	12,53658	62	17,91415	82	14,37226
3	10,3882	23	12,09231	43	13,8118	63	19,59144	83	20,12994
4	11,35893	24	10,42678	44	11,46787	64	11,53517	84	15,73159
5	12,75835	25	13,16333	45	12,34989	65	16,18677	85	14,26019
6	11,67567	26	10,45954	46	14,71083	66	11,98563	86	11,38146
7	11,81331	27	12,18932	47	13,9483	67	15,65313	87	14,50262
8	15,80492	28	13,24066	48	12,75473	68	14,39346	88	13,24918
9	13,35154	29	11,86087	49	11,32896	69	14,52254	89	14,14924
10	12,38518	30	11,64554	50	13,45193	70	14,36298	90	14,26018
11	10,90163	31	11,80906	51	12,30425	71	13,14092	91	12,62627
12	11,77862	32	12,94516	52	18,16724	72	14,69045	92	13,14276
13	12,23603	33	13,46817	53	15,13809	73	19,01502	93	17,09598
14	9,381216	34	12,79561	54	13,50488	74	11,75946	94	19,95713
15	12,03504	35	14,93394	55	12,38336	75	13,6998	95	15,40959
16	13,47579	36	11,69252	56	11,91786	76	18,75766	96	17,85948
17	12,63983	37	12,02886	57	14,43936	77	20,10625	97	14,33813
18	12,88324	38	13,65815	58	13,02173	78	12,82179	98	12,97212
19	13,01351	39	11,66589	59	11,25241	79	16,08633	99	12,52652
20	11,42358	40	11,49043	60	15,65313	80	14,243	100	11,43439

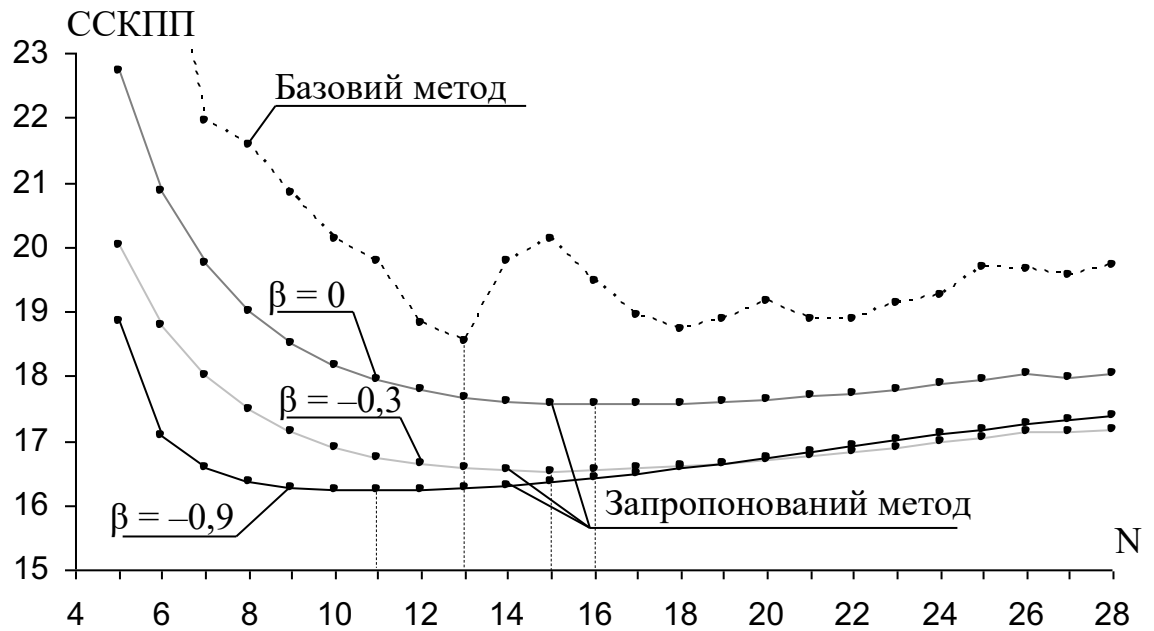


Рисунок 4.9 – Залежність ССКПП прогнозу часу виконання програми від числа вимірювань при різних значеннях коригуючого коефіцієнта β

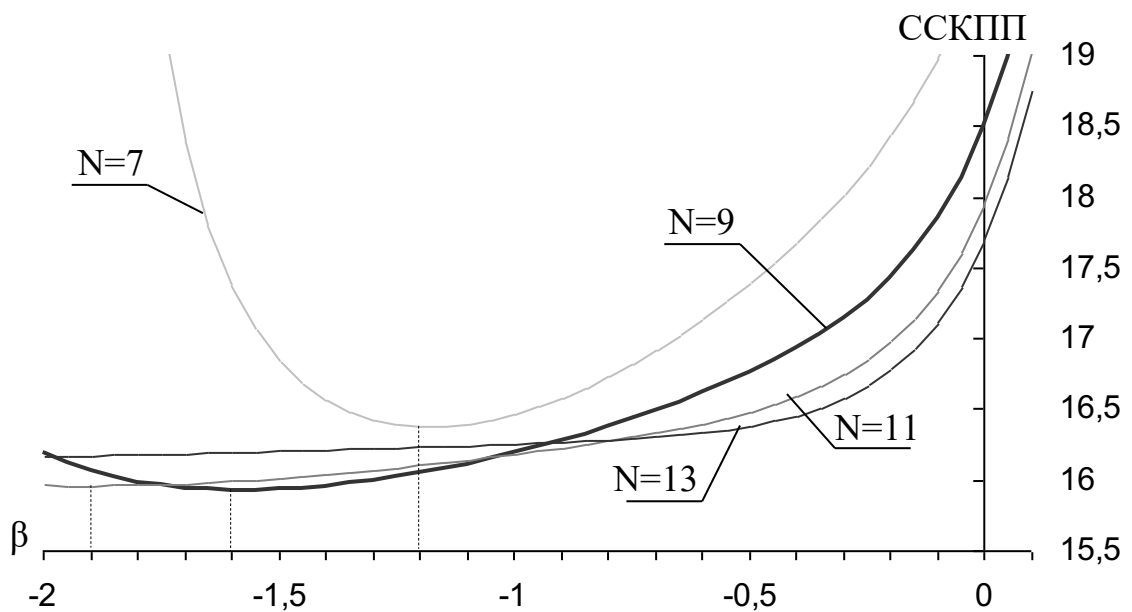


Рисунок 4.10 – Залежність ССКПП прогнозу часу виконання програми від коригуючого коефіцієнта β при різних значеннях числа вимірювань

Вибір оптимальних параметрів методу на основі контрольної вибірки дозволяє вирішити основну проблему всіх методів прогнозування –

досягнення компромісу між необхідністю згладжування випадкових відхилень та врахування важливості нових спостережень.

Висновки за розділом 4

1. Для проведення оцінки та порівняння можливостей запропонованого та класичного методів був обраний кластер граничного шару Інтернету речей на базі інтелектуального шлюзу, який пропонує потужні можливості для організації обчислень IoT, що можуть підвищити ефективність усього процесу передачі та обробки даних, зменшити затримку при реалізації транзакцій IoT. Кластер формується із територіально нерознесених пристроїв IoT та орієнтований на виконання оперативних транзакцій.

2. Синтез функціональної моделі кластера граничного шару був проведений на базі трасувальних даних, отриманих за допомогою вимірювального монітору при 50 тестових прогонах фрагмента мережі IoT, який підтримується обраним шлюзом. Для перевірки адекватності класичної моделі на основі СМО та запропонованої функціональної моделі з використанням апарату мереж Петрі було проведено імітаційне моделювання, причому ступінь відхилення результатів моделювання від експериментальних визначалася за допомогою χ^2 -критерію Пірсона. Застосування запропонованого методу дозволило суттєво підвищити адекватність моделі кластера граничного шару Інтернету речей, зокрема, у розглянутому прикладі розходження за ймовірностями станів склало менше 1% при використанні запропонованого методу, тоді, як при класичному методі перевищило 5%.

3. Для визначення оптимального варіанту формування кластера граничного шару використані результати функціонального моделювання, зокрема, отримані оцінки ймовірностей станів. Проведено порівняння класичного та запропонованого методів формування кластера граничного шару системи підтримки IoT за критерієм середнього часу виконання одного застосунку. Узагальнені результати моделювання показали, що із зростанням

кількості застосунків, задіяних у кластері для виконання оперативних транзакцій переваги запропонованого методу формування кластера граничного шару системи підтримки IoT за часовими характеристиками збільшуються, тобто його ефективність підвищується. Але із збільшенням кількості застосунків суттєво збільшується обчислювальна складність запропонованого методу. Для зменшення обчислювальної складності зменшена кількість станів моделі, за рахунок використання методу редукції кількості станів.

4. Проведена оцінка ефективності короткострокового прогнозу часових параметрів кластера граничного шару Інтернету речей. Оцінка проводилась для реакції класичного та запропонованого методів короткострокового прогнозування на стандартні збурення – імпульс, ступінчаста зміну рівня та на вхідний потік із лінійним зростанням. Результати оцінки показали перевагу запропонованого методу, причому при виборі оптимальних параметрів методу на основі контрольної вибірки, точність прогнозу можна збільшити до 15%.

ВИСНОВКИ

У дисертації наведено основні результати розв'язання актуальної науково-технічної задачі щодо підвищення ефективності використання обчислювальних ресурсів граничного шару при обробці оперативних транзакцій шляхом удосконалення методів синтезу окремих кластерів граничного шару Інтернету речей, що об'єднують низку територіально близьких пристроїв IoT.

Це дало змогу отримати такі науково-технічні і практичні результати:

1) проведений аналіз існуючих методів підвищення ефективності комп'ютерної системи підтримки Інтернету речей з метою виявлення проблем, що виникають при задіянні пристроїв граничного шару для виконання оперативних транзакцій, зокрема, визначені особливості концепції Інтернету речей та хмарної інфраструктури його підтримки, також виділені характеристики існуючих методів дослідження та моделей і методів оцінювання ефективності комп'ютерних систем підтримки Інтернету речей;

2) розроблений метод побудови функціональних моделей кластерів граничного шару комп'ютерної системи підтримки Інтернету речей, який відрізняється від відомих використанням апарату узагальнених часових мереж Петрі, що дозволило отримати значення перехідних ймовірностей між станами мережі та різні рівні деталізації при моделюванні процесу робочого навантаження, за рахунок чого підвищилась адекватність функціональної моделі реальному процесу обробки оперативних транзакцій Інтернету речей, зокрема, наведений опис методу побудови функціональних моделей, визначені переваги та недоліки запропонованого методу побудови функціональних моделей, розроблений метод оцінки адекватності мережних моделей фрагментів граничного шару та запропонований агрегативний опис функціональних моделей процесів у кластерах граничного шару.

3) розроблений метод формування кластера граничного шару комп'ютерної системи підтримки Інтернету речей шляхом використання

функціональної моделі кластера та апарату напівмарковських процесів з можливою редукцією кількості станів мережі Петрі, що дозволило зменшити середній час обробки оперативних транзакцій, що надходять до граничного шару Інтернету речей; даний метод дозволяє визначити оптимальний варіант розподілу застосунків по вузлах граничного шару за критерієм мінімального середнього часу виконання одного застосунку; також запропонований метод редукції кількості станів часової мережі Петрі, який узагальнює відомі підходи до аналізу графових і мережних моделей, дає змогу одержати значення характеристик обраних станів без побудови й розв'язання відповідних систем рівнянь та дозволяє суттєво зменшити обчислювальну складність визначення функціональних характеристик кластера за рахунок зменшення кількості аналізуємих станів;

4) розроблений метод короткострокового прогнозу часових параметрів кластера граничного шару комп'ютерної системи підтримки Інтернету речей, який враховує обмежені обчислювальні можливості вузлів граничного шару Інтернету речей та базується на відповідній функціональній моделі, що дозволило підвищити точність прогнозу при плануванні оперативного перерозподілу ресурсів системи підтримки Інтернету речей; в основі методу лежить виявлення тренду середнього значення випадкового часу виконання оперативних транзакцій IoT. Визначення характеристик тренду здійснюється за допомогою розроблених на основі методу найменших квадратів рекурентних формул, що дають змогу одержувати в реальному масштабі часу прогнозні оцінки та їх довірчі межі, на основі яких здійснюється адаптивне до завантаження обчислювальної системи керування перебігом обчислень;

5) проведена порівняльна оцінка розроблених та існуючих методів синтезу кластера граничного шару комп'ютерної системи підтримки Інтернету речей на базі інтелектуального шлюзу, який формується із територіально нерознесених пристроїв IoT та орієнтований на виконання оперативних транзакцій; застосування запропонованого методу дозволило суттєво підвищити адекватність моделі кластера граничного шару Інтернету речей,

зокрема, у розглянутому прикладі розходження за ймовірностями станів склало менше 1% при використанні запропонованого методу, тоді, як при класичному методі перевищило 5%; узагальнені результати моделювання показали, що із зростанням кількості застосунків, задіяних у кластері для виконання оперативних транзакцій переваги запропонованого методу формування кластера граничного шару системи підтримки IoT за часовими характеристиками збільшуються; проведена оцінка ефективності короткострокового прогнозу часових параметрів кластера граничного шару Інтернету речей показала, що точність прогнозу у порівнянні з класичним методом можна збільшити до 15%;

б) проведено впровадження розроблених методів синтезу кластера граничного шару комп'ютерної системи підтримки Інтернету речей, зокрема, практичні результати, які отримані, підтверджені актами впровадження та доводять коректність теоретичних положень дисертаційної роботи, високу ефективність розроблених методів; результати роботи впроваджено в ході дослідницьких робіт у підприємстві «EURO RAIL SP. Z O.O.» (м. Краків, Польща) та у навчальний процес кафедр “Комп'ютерна інженерія та програмування” та “Системи інформації ім. В.О. Кравця” Національного технічного університету “ХПІ” при викладанні дисциплін «Системи штучного інтелекту»; «Проектування серверних застосунків», «Паралельні та розподілені обчислення», «Об'єктно-орієнтоване програмування» відповідно.

Подальші дослідження підвищення ефективності використання ресурсів комп'ютерних систем підтримки Інтернету речей пов'язані із розробкою функціональних моделей туманного шару хмарної інфраструктури.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Гавриленко С. Ю., Кучук Н. Г., Луйкова-Чуйко Н. В., Собчук В. В. Перерозподіл інформаційних потоків в хмарній системі. *Сучасні інформаційні системи*. 2019. Т. 3, № 2. С. 116-122.
2. Гора М. В., Волк М. О. Моделі управління ресурсами для забезпечення функціональної стійкості процесу розподілених обчислень. *Вісник Херсонського національного технічного університету*. 2023. № 4. С. 244–251.
3. Гребенюк Д.С., Шиман А.П. Модель транзакційної системи для розгортання у хмарному середовищі. Проблеми інформатизації. Тези доповідей сьомої міжнародної науково-технічної конференції. Черкаси – Харків – Баку – Бельсько-Бяла, Т.1. 2019. С. 74.
4. Данкевич, В. Є., Данкевич А. Є. Інтернет речей та штучний інтелект як ключові елементи інноваційного розвитку підприємств в епоху цифрових викликів. *Актуальні проблеми економіки*. 2024. № 7. С. 165–173.
5. Досенко А. К. Хмарні технології: прикладні технології сучасних платформ. *Вчені записки ТНУ імені В. І. Вернадського*. 2022. Т. 33 (72), № 1, Ч. 3. С. 257-262. DOI: <https://doi.org/10.32838/2710-4656/2022.1-3/41>
6. Дявіл, А., Ноздріна, Л. (2020). Інтернет речей як складова Індустрії 4.0: проєктний підхід. *Socio-Economic Relations in the Digital Society*, 3 (39), 85–93. [https://doi.org/10.18371/2221-755X3\(39\)2020225589](https://doi.org/10.18371/2221-755X3(39)2020225589)
7. Заковоротний О. Ю., Орлова Т. О. Порівняльний аналіз хмарних та туманних середовищ Інтернету речей. Системи управління, навігації та зв'язку. 2023. Вип. 3. С. 152-154.
8. Зубик Л. В. Моделювання управління складними інформаційними системами у реальному часі. *Вісник Національного університету водного господарства та природокористування. Техн. науки*. 2019. Вип. 1. С. 211-217.
9. Коваленко А., Ляшенко О., Даниленко О. Поведінка черг під час використання ієрархічної моделі. *Системи управління, навігації та зв'язку*.

Збірник наукових праць. Полтава: ПНТУ, 2019. Т. 2 (54). С. 110-113. DOI: <https://doi.org/10.26906/SUNZ.2019.2.110>.

10. Коваленко А. А., Кучук Г. А. Методи синтезу інформаційної та технічної структур системи управління об'єктом критичного застосування. *Сучасні інформаційні системи*. 2018. Т. 2, № 1. С. 22–27. DOI: <https://doi.org/10.20998/2522-9052.2018.1.04>.

11. Кожевников Г. К., Черниш Д. С., Матяш О. Ю. Онтологічний підхід до перерозподілу навантаження Інтернету Речей. Системи управління, навігації та зв'язку. 2024. - Вип. 2. С. 111-114

12. Крайнюченко О. Ф., Єфісько Ю. Ю. Можливості та передумови використання Інтернету Речей на українських підприємствах. Науковий вісник Міжнародного гуманітарного університету. Серія : Економіка і менеджмент. 2019. Вип. 35. С. 160-164. URL: <http://jnas.nbuv.gov.ua/article/UJRN-0000989272>

13. Кучук Г. А., Коваленко А. А., Лукова-Чуйко Н. В. Метод мінімізації середньої затримки пакетів у віртуальних з'єднаннях мережі підтримки хмарного сервісу. *Системи управління, навігації та зв'язку*. Полтава . ПНТУ, 2017. Вип. 2(42). С. 117-120.

14. Кучук Н. Г. Мерлак В. Ю., Скороделов В. В. Метод зменшення часу доступу до слабкоструктурованих даних. *Сучасні інформаційні системи : щоквартальний науково-технічний журнал*. 2020. Т. 4, № 1. С. 97-102.

15. Кучук Н. Г. Синтез мережевої моделі комп'ютерної системи на гіперконвергентній платформі. *Системи управління, навігації та зв'язку*. Полтава : ПНТУ, 2020. Вип. 1(59). С. 86-93.

16. Кучук Н. Г. Метод вибору оптимального плану виконання транзакцій e-Learning. *Системи управління та зв'язку*. Полтава : ПНТУ, 2017. Вип. 5(45). С. 83-87.

17. Кучук Н. Г. Метод розрахунку максимальних інтенсивностей інформаційних потоків у гіперконвергентній системі. *Системи управління, навігації та зв'язку*. Полтава : ПНТУ, 2019. Вип. 4(56). С. 53-56.

18. Кучук Н.Г., Нечаусов С.М. Математична модель процесу оперативного перерозподілу обчислювальних ресурсів в гіперконвергентному середовищі. *Системи управління, навігації та зв'язку*. Полтава : ПНТУ, 2017. Вип. 3(43). С. 80-83.

19. Кучук Н.Г., Бульба С.С., Шиман А.П. Знаходження оптимального інформаційного навантаження компонентів СВКС за комплексним показником. Проблеми інформатизації. Тези доповідей дев'ятої міжн. науково-технічної конференції. Черкаси – Харків – Баку – Бельсько-Бяла, Т.1. 2021. С. 45.

20. Кучук Н.Г., Міхаль О.П., Шиман А.П., Науменко М.В. Динамічна відеофіксація транспортних засобів для системи моніторингу дорожнього руху у мегаполісі. Системи управління, навігації та зв'язку. Збірник наукових праць. Полтава: Національний університет «Полтавська політехніка імені Юрія Кондратюка». Вип. 2 (68). 2022. С. 55-58. DOI: 10.26906/SUNZ.2022.2.055.

21. Кучук Н.Г., Шиман А.П. Інтелектуальні транспортні системи. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління. Тези доповідей тринадцятої міжнародної науково-технічної конференції. Баку – Харків – Жиліна, Т. 1. 2023. С.55.

22. Кучук Н.Г., Шиман А.П., Гребенюк Д.С. Синтез інформаційної системи на гіперконвергентній платформі. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління. Тези доповідей десятої міжнародної науково-технічної конференції. Баку – Харків – Жиліна, Т.2. 2020. С.43.

23. Кучук Н.Г., Шиман А.П., Філоненко А.М., Бульба С.С. Розрахунок ефективності використання обчислювальних ресурсів самовідновлювальної комп'ютерної системи. Системи управління, навігації та зв'язку. Збірник наукових праць. Полтава: Національний університет «Полтавська політехніка імені Юрія Кондратюка». Вип. 3 (65). 2021. С. 92–95. DOI: 10.26906/SUNZ.2021.3.092

24. Петровська І. Ю., Кучук Н. Г., Панченко В. І., Філоненко А. М., Рівномірний розподіл ресурсів, що мають гіперконвергентну інфраструктуру. *Системи управління, навігації та зв'язку*. Полтава : ПНТУ, 2019. Вип. 2(54). С. 119-123.
25. Положий Д. С., Орехов О. О. Інтелектуальні системи автомобільної безпеки на основі хмарних архітектур. *Системи управління, навігації та зв'язку*. 2023. Вип. 4. С. 91-95.
26. Резанов Б. М., Кучук Г. А. Модель розподілу елементарних потоків даних у туманній платформі підтримки Інтернету речей. Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 3. С. 88-97.
27. Скоробогатько С. В., Фесенко Г. В. Перспективи використання літаючих хмарних, граничних та туманних обчислень компонентами системи моніторингу потенційно небезпечних об'єктів. *Системи управління, навігації та зв'язку*. 2022. Вип. 4. С. 145-152.
28. Субач І., Кучук Н., Чаузов О. Метод рішення задачі розподілу інформаційного ресурсу в АСУ спеціального призначення при варіативному розмірі інформаційних блоків. *Information Technology and Security*. Київ : НТУУ «КПІ», 2016. Т. 4, Вип. 2 (7). С. 269-276.
29. Субач І., Кучук Н., Чаузов О. Моделі розподілу інформаційного ресурсу в АСУ спеціального призначення. *Information Technology and Security*. Київ : НТУУ «КПІ», 2016. Т. 4., Вип. 1 (6). С. 75-83.
30. Петровська І. Ю., Кучук Г. А. Розподіл обчислювальних ресурсів у хмарних системах. *Системи управління, навігації та зв'язку*. Полтава: Національний університет «Полтавська політехніка імені Юрія Кондратюка», 2022. Вип. 2 (68). С. 75–78. DOI: 10.26906/SUNZ.2022.2.75.
31. Шелевер О. В., Лисак Г. О., Харлай Л. О. Технології інтернет речей в сучасній освіті: перспективи, особливості. *Інноваційна педагогіка*. 2022. № 50, т. 2. С. 210-213.
32. Шиман А.П., Кучук Н.Г., Бульба С.С., Давидов В.В. Інтелектуальні логістичні системи підтримки дорожнього руху. *Проблеми інформатизації*.

Тези доповідей десятої міжнародної науково-технічної конференції. Черкаси – Баку – Бельсько-Бяла – Харків, Т. 2. 2022. С. 115.

33. Abdelwahab, S., Hamdaoui, B., Guizani, M. and Znati, T. (2016). Replisom: Disciplined Tiny Memory Replication for Massive IoT Devices in LTE Edge Cloud. *IEEE Internet of Things Journal*, 3(3), 327–338, doi: <https://doi.org/10.1109/jiot.2015.2497263>

34. Ahmid, M. and Kazar, O. (2021), “A comprehensive review of the Internet of things security”, *Journal of Applied Security Research*, vol. 18(3), pp. 289–305, doi: <https://doi.org/10.1080/19361610.2021.1962677>

35. Aldossary M. A review of dynamic resource management in cloud computing environments. *Computer Systems Science and Engineering*. 2021. Vol. 36, no.3. P. 461–476.

36. Alwakeel, A. M. (2021), “An overview of fog computing and edge computing security and privacy issues”, *Sensors*, vol. 21(24), 8226, doi: <https://doi.org/10.3390/s21248226>

37. Atlam, H. F., Walters, R. J. and Wills, G. (2018), “Fog Computing and the Internet of Things: a review”, *Big Data and Cognitive Computing*, vol. 2(2), doi: <https://doi.org/10.3390/bdcc2020010>

38. Attar H., Khosravi M.R., Igorovich S.S., Georgievan K.N., Alhihi M. E-health communication system with multiservice data traffic evaluation based on a G/G/1 analysis method. *Current Signal Transduction Therapy*. 2021. Vol. 16(2). DOI: 10.2174/1574362415666200224094706

39. Belgacem Ali. Dynamic resource allocation in cloud computing: analysis and taxonomies. *Computing*. 2022. Vol. 104, is. 3. P. 681–710. DOI: <https://doi.org/10.1007/s00607-021-01045-2>

40. Breitbach, M., Schäfer, D., Edinger, J. and Becker, C. (2019), “Context-Aware Data and Task Placement in Edge Computing Environments”, *2019 IEEE Int. Conf. Pervasive Comput. Commun. PerCom*, doi: <https://doi.org/10.1109/percom.2019.8767386>

41. Bulba S. Composite application distribution methods. *Сучасні інформаційні системи*. 2018. Т. 2, № 3. Р. 128–131. DOI: 10.20998/2522-9052.2018.3.22
42. Capra, L. and Köhler-Bußmeier, M. (2024), “Modular rewritable Petri nets: An efficient model for dynamic distributed systems”, *Theoretical Computer Science*, vol. 990, 1184397, doi: <https://doi.org/10.1016/j.tcs.2024.114397>
43. Castellanos Contreras, J.U. and Rodríguez Urrego, L. (2023), “Technological Developments in Control Models Using Petri Nets for Smart Grids: A Review”, *Energies*, vol. 16, is. 8, 3541, doi: <https://doi.org/10.3390/en16083541>
44. Castillo, J.A.R. and Malinao, J.A. (2024), “Model Decomposition of Robustness Diagram with Loop and Time Controls to Petri Net with Considerations on Resets”, *Lecture Notes in Computer Science*, vol. 14799 LNCS, pp.193–201, doi: https://doi.org/10.1007/978-3-031-63031-6_17
45. Ch. Sanjeev Kumar Dash, Ajit Kumar Behera, Satchidananda Dehuri, Ashish Ghosh. An outliers detection and elimination framework in classification task of data mining. *Decision Analytics Journal*. Vol. 6, 2023, DOI: <https://doi.org/10.1016/j.dajour.2023.100164>
46. Chen J., Wang Y. A Hybrid Method for Short-Term Host Utilization Prediction in Cloud Computing. *Journal of Electrical and Computer Engineering*. 2019. P. 1-14. DOI: 10.1155/2019/2782349
47. Chen J., Wang Y., Liu T. A proactive resource allocation method based on adaptive prediction of resource requests in cloud computing. *J Wireless Com Network*. 2021. Vol. 24. DOI: <https://doi.org/10.1186/s13638-021-01912-8>
48. Chen, I., Guo, J. and Bao, F. (2016), “Trust Management for SOA-Based IoT and its application to service composition”, *IEEE Transactions on Services Computing*, vol. 9(3), pp. 482–495, doi: <https://doi.org/10.1109/tsc.2014.2365797>
49. De Donno, M., Tange, K. and Dragoni, N. (2019), “Foundations and evolution of modern computing paradigms: cloud, IoT, edge, and FOG”, *IEEE Access*, vol. 7, pp. 150.936–150.948, doi: <https://doi.org/10.1109/access.2019.2947652>

50. Farooq, M. U., Waseem, M., Khairi, A. and Mazhar, S. (2015), “A critical analysis on the security concerns of internet of things (IoT)”, *Int. Journal of Computer Appl.*, vol. 111(7), pp. 1–6, doi: <https://doi.org/10.5120/19547-1280>
51. Franti P. K-sets and k-swaps algorithms for clustering sets. *Pattern Recognition*. 2023. Vol. 139, No. 13. 109454. P. 1-29. DOI: 10.1186/s40537-018-0122-y
52. Geronimo, M. F., Martinez, E. G. H., Vazquez, E. D. F., Godoy, J. J. F. and Anaya, G. F. (2021), “A multiagent systems with Petri Net approach for simulation of urban traffic networks”, *Computers, Environment and Urban Systems*, vol. 89, 101662, doi: <https://doi.org/10.1016/j.compenvurbsys.2021.101662>
53. Gião, J., Nazarenko, A. A., Ferreira, F., Gonçalves, D. and Sarraipa, J. (2022), “A framework for Service-Oriented Architecture (SOA)-Based IoT application development”, *Processes*, vol. 10(9), 1782, doi: <https://doi.org/10.3390/pr10091782>
54. Hamdan, S., Ayyash, M. and Almajali, S. (2020), “Edge-Computing Architectures for Internet of Things Applications: A survey”, *Sensors*, vol. 20(22), 6441, doi: <https://doi.org/10.3390/s20226441>
55. Hamed Dinari. A Survey on Graph Queries Processing: Techniques and Methods. *Int. Journal of Computer Network and Inf. Security (IJCNIS)*. 2017. Vol. 9, No. 4. P. 48-56. URL : <http://dx.doi.org/10.5815/ijcnis.2017.04.06>
56. He, L., Liu, G. and Zhou, M. (2023), “Petri-Net-Based Model Checking for Privacy-Critical Multiagent Systems”, *IEEE Transactions on Computational Social Systems*, vol. 10, is. 2, pp. 563–576, doi: <https://doi.org/10.1109/TCSS.2022.3164052>
57. Hunko, M., Tkachov, V., Kovalenko, A. and Kuchuk, H. (2023), “Advantages of Fog Computing: A Comparative Analysis with Cloud Computing for Enhanced Edge Computing Capabilities”, *2023 IEEE 4th KhPI Week on Advanced Technology*, KhPI Week 2023 - Conference Proceedings, 02-06 October 2023, Code 194480, doi: <https://doi.org/10.1109/khpiweek61412.2023.10312948>

58. Jiang, W., Zhou, K.-Q., Sarkheyli-Hägele, A. and Zain, A.M. (2022), “Modeling, reasoning, and application of fuzzy Petri net model: a survey”, *Artificial Intelligence Review*, vol. 55(8), pp. 6567–6605, doi: <https://doi.org/10.1007/s10462-022-10161-0>
59. Ketykó, I., Kecskes, L. J., Nemes, C. and Farkas, L. (2016), “Multi-user computation offloading as Multiple Knapsack Problem for 5G Mobile Edge Computing”, *2016 Eur. Conf. Networks Commun.*, IEEE, doi: <https://doi.org/10.1109/eucnc.2016.7561037>
60. Khan, W. Z., Ahmed, E., Hakak, S., Yaqoob, I. and Ahmed, A. (2019), “Edge computing: A survey”, *Future Generation Computer Systems*, vol 97, pp. 219–235, doi: <https://doi.org/10.1016/j.future.2019.02.050>
61. Khatatneh, K., Nawafleh, O. and Al-Utaibi, D. (2020), “The Emergence of Edge Computing Technology over Cloud Computing”, *International Journal of P2P Network Trends and Technology*, vol. 10(2), pp. 1–5, doi: <https://doi.org/10.14445/22492615/ijptt-v10i2p401>
62. Khudov H., Tahyan K., Chepurnyi V., Khizhnyak I., Romanenko K., Nevodnichii A., Yakovenko O. Optimization of joint search and detection of objects in technical surveillance systems. *Сучасні інформаційні системи*. 2020. Т. 4, № 2. P. 156-162. DOI: 10.20998/2522-9052.2020.2.23
63. Kolodenko Vitalii, Shyman Anna, Kalinin Yevhen, Nina Kuchuk. Synthesis of The Theory of Motion of Solid Bodies Filled With Bulk Substances For Fault-Tolerant Identification of Their Parameters. IEEE 13th International Conference on Dependable Systems, Services and Technologies (DESSERT), 2023. Kharkiv. DOI: [10.1109/DESSERT61349.2023.10416449](https://doi.org/10.1109/DESSERT61349.2023.10416449)
64. Kosenko V. Mathematical model of optimal distribution of applied problems of safety-critical systems over the nodes of the information and telecommunication network. *Сучасні інформаційні системи*. 2017. Т. 1, № 2. С. 4-9. doi:<https://doi.org/10.20998/2522-9052.2017.2.01>.
65. Kovalenko A., Kuchuk H., Kuchuk N., Kostolny J. Horizontal scaling method for a hyperconverged network. *International Conference on Information*

and Digital Technologies 2021 (IDT-2021). 2021. P. 331–336. DOI: <https://doi.org/10.1109/IDT52577.2021.9497534>

66. Kuchuk G., Kovalenko A., Komari I.E., Svyrydov A., Kharchenko V. Improving big data centers energy efficiency: Traffic based model and method. *Studies in Systems, Decision and Control*, vol. 171, Kharchenko, V., Kondratenko, Y., Kacprzyk, J. (Eds.), Springer Nature Switzerland AG. 2019. P. 161-183, DOI: 10.1007/978-3-030-00253-4_8

67. Kuchuk G., Nechausov S., Kharchenko, V. Two-stage optimization of resource allocation for hybrid cloud data store. *International Conference on Information and Digital Technologies*. Zilina, 2015. P. 266-271. DOI: <http://dx.doi.org/10.1109/DT.2015.7222982>

68. Kuchuk H., Kovalenko A., Ibrahim B.F., Ruban I. Adaptive compression method for video information. *International Journal of Advanced Trends in Computer Science and Engineering*. 2019. Vol. 8(1). P. 66–69, DOI: <http://dx.doi.org/10.30534/ijatcse/2019/1181.22019>.

69. Kuchuk N. H. Comprehensive performance criterion for hyper-converged infrastructure. *Телекомунікаційні та інформаційні технології : науковий журнал*. Київ : ДУТ, 2019. № 3 (64). С. 55-63.

70. Kuchuk N., Bulba S. Mathematical model of distribution of resources between composite applications. *News of science and education*. Sheffield : Science and education LTD, 2017. No. 6 (54). P. 72-80.

71. Kuchuk N., Hani A., Alhihi M., Samour M., Shmatkov S. A Mathematical Model for Managing the Distribution of Information Flows for MPLS-TE Networks under CrC. *Communications and Network*. 2018. Vol. 10, No. 2. P. 31-42.

72. Kuchuk N., Hani Attar, Mohammad R. Khosravi, Shmatkov S. Review and performance evaluation of FIFO, PQ, CQ, FQ, and WFQ algorithms in multimedia wireless sensor networks. *Int. Journal of Distributed Sensor Networks*. 2020. Vol. 16, No. 2. P. 1-9.

73. Kuchuk N., Mohammed A. S., Shyshatskyi A., Nalapko O. The method

of improving the efficiency of routes selection in networks of connection with the possibility of self-organization. *International Journal of Advanced Trends in Computer Science and Engineering*. 2019. No. 8(1). P. 1-6.

74. Kuchuk N., Mozhaiev O., Semenov S. Brusakova O. Gnusov Y. (2023). Devising a method for balancing the load on a territorially distributed foggy environment. *Eastern-European Journal of Enterprise Technologies*. 2023. Vol. 1(4 (121)). DOI: 48–55. <https://doi.org/10.15587/1729-4061.2023.274177>

75. Kuchuk N., Shefer O., Cherneva G., Alnaeri F. A. Determining the capacity of the self-healing network segment. *Сучасні інформаційні системи*. 2021. Т. 5, № 2. P. 114–119. DOI: 10.20998/2522-9052.2021.2.16.5.

76. Kuchuk Nina, Davydov Viacheslav, Hrebeniuk Daryna, Shyman Anna. Mathematical Model of the Information System Synthesis Process. *Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління. Тези доповідей одинадцятої міжнародної науково-технічної конференції. Баку – Харків – Київ – Жиліна, Т.1. 2021. С. 21*

77. Kuchuk Nina, Shyman Anna, Shyman Maksym. Development of a Method for Synthesising Models of Real-Time Software Systems. *Проблеми інформатизації. Тези доповідей дванадцятої міжнародної науково-технічної конференції. Баку – Харків – Бельсько-Бяла, Т.2. 2024. С. 142.*

78. Kuchuk Nina, Shyman Anna. A Method for Distributing Transactions to Hybrid Cloud Data Storage. *Системи управління, навігації та зв'язку. Збірник наукових праць. Полтава: Національний університет «Полтавська політехніка імені Юрія Кондратюка». Вип. 4 (70). 2022. С. 100-103. DOI: 10.26906/SUNZ.2022.4.100.*

79. Kuchuk Nina, Shyman Anna. Artificial Intelligence for Traffic Data Analysis. *Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління. Тези доповідей чотирнадцятої міжнародної науково-технічної конференції. Баку – Харків – Жиліна, Т.1. 2024. С.59.*

80. Kuchuk Nina, Shyman Anna. Synthesis of a Computer Network for Controlling Moving Objects. *Системи управління, навігації та зв'язку. Збірник*

наукових праць. Полтава: Національний університет «Полтавська політехніка імені Юрія Кондратюка». Вип. 4 (74). 2023. С. 134-136. DOI: 10.26906/SUNZ.2023.4.134.

81. Kuchuk N., Mukhin V. Kosenko N., Artiukh R. Decomposition Method for Synthesizing the Computer System Architecture. *Advances in Intelligent Systems and Computing*. 2020. Vol 938. Springer, Cham. P. 289-300.

82. Kuchuk, N., Ruban, I., Zakovorotnyi, O., Kovalenko, A., Shyshatskyi, A. and Sheviakov, I. (2023), “Traffic Modeling for the Industrial Internet of NanoThings”, *2023 IEEE 4th KhPI Week on Advanced Technology*, KhPI Week 2023 – Conf. Proc., doi: <https://doi.org/10.1109/khpiweek61412.2023.10312856>

83. Landaluce, H., Arjona, L., Perallos, A., Falcone, F., Angulo, I. and Muralter, F. (2020), “A review of IoT sensing applications and challenges using RFID and wireless sensor networks”, *Sensors*, vol. 20(9), 2495, doi: <https://doi.org/10.3390/s20092495>

84. Li, J., Wang, Z., Sun, L. and Wang W. (2021), “Modeling and Analysis of Network Control System Based on Hierarchical Coloured Petri Net and Markov Chain”, *Discrete Dynamics in Nature and Society (DDNS)*, Article ID 9948855, doi: <https://doi.org/10.1155/2021/9948855>

85. Li, S., Da Xu, L. and Zhao, S. (2018), “5G Internet of Things: A survey”, *Journal of Industrial Information Integration*, vol. 10, pp. 1–9, doi: <https://doi.org/10.1016/j.jii.2018.01.005>

86. Li, S., Tryfonas, T. and Li, H. (2016), “The Internet of Things: a security point of view”, *Internet Research*, vol. 26(2), pp. 337–359, doi: <https://doi.org/10.1108/intr-07-2014-0173>

87. Li, Y., Qi, F., Wang, Z., Yu, X. and Shao, S. (2020), “Distributed edge Computing offloading algorithm based on deep reinforcement learning”, *IEEE Access*, vol. 8, pp. 204–215, doi: <https://doi.org/10.1109/access.2020.2991773>

88. Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H. and Zhao, W. (2017), “A survey on Internet of things: architecture, enabling technologies, security and privacy, and applications”, *IEEE Internet of Things Journal*, vol. 4(5), pp. 1125–

1142, doi: <https://doi.org/10.1109/jiot.2017.2683200>

89. Liu, J., Mao, Y., Zhang, J. and Letaief, K. B. (2016), “Delay-optimal computation task scheduling for mobile-edge computing systems”, *2016 IEEE Int. Symp. Inf. Theory*, IEEE, doi: <https://doi.org/10.1109/isit.2016.7541539>

90. Lung-Pin Chen, Fang-Yie Leu, Hsin-Ta Chiao, and Hung-Jr Shiu. Adaptive resource management for spot workers in cloud computing environment. *Int. J. Web Grid Serv.* 2022. Vol. 18, is. 4. P. 437–452. DOI: <https://doi.org/10.1504/ijwgs.2022.126124>

91. Maurya, S. and Mukherjee, K. (2019), “An Energy Efficient Architecture of IoT based on Service Oriented Architecture (SOA)”, *Informatica*, vol. 43(1), doi: <https://doi.org/10.31449/inf.v43i1.1790>

92. Nechausov A., Mamusu I., Kuchuk N. Synthesis of the air pollution level control system on the basis of hyperconvergent infrastructures. *Сучасні інформаційні системи*. 2017. Т. 1, № 2. P. 21–26. DOI: 10.20998/2522-9052.2017.2.04

93. Petrovska Inna, Kuchuk Heorhii. Adaptive resource allocation method for data processing and security in cloud environment. *Сучасні інформаційні системи*. Харків: HTУ «ХІІІ», 2023. Т. 7, № 3, С. 67–73. DOI: 10.20998/2522-9052.2023.3.10.

94. Petrovska Inna, Kuchuk Heorhii. Static allocation method in a cloud environment with a service model IAAS. *Сучасні інформаційні системи*. Харків: HTУ «ХІІІ», 2022. Т. 6, № 3, С. 99–105. DOI: 10.20998/2522-9052.2022.3.13.

95. PremSankar, G., Di Francesco, M. and Taleb, T. (2018), “Edge Computing for the Internet of Things: a case study”, *IEEE Internet of Things Journal*, vol/ 5(2), pp. 1275–1284, doi: <https://doi.org/10.1109/jiot.2018.2805263>

96. Qiu, T., Chi, J., Zhou, X., Ning, Z., Atiquzzaman, M. and Wu, D. (2020), “Edge Computing in Industrial Internet of Things: architecture, advances and challenges”, *IEEE Communications Surveys and Tutorials*, vol. 22(4), pp. 2462–2488, doi: <https://doi.org/10.1109/comst.2020.3009103>

97. Ray, P. P. (2018), “A survey on Internet of Things architectures”,

Journal of King Saud University - Computer and Information Sciences, vol. 30(3), pp. 291–319, doi: <https://doi.org/10.1016/j.jksuci.2016.10.003>

98. Rehman, M. H. U., Sun, C., Wah, T. Y., Iqbal, A. and Jayaraman, P. P. (2016), “Opportunistic Computation Offloading in Mobile Edge Cloud Computing Environments”, *2016 17th IEEE Int. Conf. Mob. Data Manag.*, IEEE, doi: <https://doi.org/10.1109/mdm.2016.40>

99. Rghioui, A., Sendra, S., Lloret, J. and Oumnad, A. (2016), “Internet of things for measuring human activities in ambient assisted living and e-Health”, *Network Prot. and Alg.*, vol. 8(3), p. 15, doi: <https://doi.org/10.5296/npa.v8i3.10146>

100. Rimal, B. P., Van, D. P. and Maier, M. (2017), “Cloudlet enhanced Fiber-Wireless access networks for Mobile-Edge Computing”, *IEEE Transactions on Wireless Communications*, vol. 16(6), pp. 3601–3618.

101. Ruban I., Kuchuk H., Kovalenko A. Redistribution of base stations load in mobile communication networks. *Innovative technologies and scientific solutions for industries*. 2017. No 1 (1). P. 75-81.

102. Semenov S., Cao Weilin. Testing process for penetration into computer systems mathematical model modification. *Сучасні інформаційні системи*. 2020. Т. 4, № 3. P. 133–138. 2020. DOI: 10.20998/2522-9052.2020.3.19

103. Semenov S., Mozhaiev O., Kuchuk N., Kuchuk H. (). Devising a procedure for defining the general criteria of abnormal behavior of a computer system based on the improved criterion of uniformity of input data samples. *Eastern-European Journal of Enterprise Technologies*. 2022. Vol. 6(4(120)). P. 40–49. DOI: <https://doi.org/10.15587/1729-4061.2022.269128>

104. Semenov S., Sira O., Gavrylenko S., Kuchuk N. Identification of the state of an object under conditions of fuzzy input data. *Eastern-European Journal of Enterprise Technologies*. 2019. Vol 1, No 4 (97). P. 22-30. DOI: 10.15587/1729-4061.2019.157085

105. Seng, K. P., Ang, L. and Ngharamike, E. (2022), “Artificial intelligence Internet of Things: A new paradigm of distributed sensor networks”, *International Journal of Distributed Sensor Networks*, 18(3), 155014772110628, doi:

<https://doi.org/10.1177/15501477211062835>

106. Serrano, S. and Scarpa, M. (2023), “A Petri Net Model for Cognitive Radio Internet of Things Networks Exploiting GSM Bands”, *Future Internet*, vol. 15, is. 3, number115, doi: <https://doi.org/10.3390/fi15030115>

107. Sethi, P. and Sarangi, S. R. (2017), “Internet of Things: architectures, protocols, and applications”, *Journal of Electrical and Computer Engineering*, pp. 1–25, doi: <https://doi.org/10.1155/2017/9324035>

108. Shyman Anna, Kuchuk Nina, Filatova Anna, Bellorin-Herrera Oleksandra. Development of a Method for Assessing the Adequacy of a Computer System Model Based on Petri Nets. *Сучасні інформаційні системи. Харків: НТУ «ХПІ»*, Т. 8, № 3. 2024. С. 46-52. DOI: 10.20998/2522-9052.2024.3.05

109. Shyman Anna, Kuchuk Nina, Hrebeniuk Daryna, Davydov Viacheslav. Analysis of the Level of Automation of Traffic Management Processes. *Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління. Тези доповідей дванадцятої міжнародної науково-технічної конференції. Баку – Харків – Жиліна. Т. 1. 2022. С.20*

110. Taghinezhad-Niar, A. (2024), “A Client-Centric Consistency Model for Distributed Data Stores using Colored Petri Nets”, 2024 10th Int. Conf. on Web Research, ICWR 2024, pp. 309–314, doi: <https://doi.org/10.1109/ICWR61162.2024.10533365>

111. Taha H. Operations Research. 10th edition. Upper Saddle River : Pearson, 2016. 848 p.

112. Uviase, O. and Kotonya, G. (2018), “IoT Architectural Framework: connection and integration framework for IoT systems”, arXiv (Cornell University), vol. 264, pp. 1–17, doi: <https://doi.org/10.4204/eptcs.264.1>

113. Vashi, S., Ram, J., Modi, J., Verma, S. and Prakash, C. (2017), “Internet of Things (IoT): A vision, architectural elements, and security issues”, 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), doi: <https://doi.org/10.1109/i-smac.2017.8058399>

114. Wang, F., Hu, L., Zhou, J. and Zhao, K. (2015), “A data processing

middleware based on SOA for the Internet of things”, *Journal of Sensors*, pp. 1–8, doi: <https://doi.org/10.1155/2015/827045>

115. Wang, F., Xu, J., Wang, X. and Cui, S. (2018), “Joint offloading and computing optimization in wireless powered Mobile-Edge computing systems”, *IEEE Transactions on Wireless Communications*, vol. 17(3), pp. 1784–1797, doi: <https://doi.org/10.1109/twc.2017.2785305>

116. Wang, R., Yan, J., Wang, D., Wang, H. and Yang, Q. (2018), “Knowledge-Centric edge computing based on virtualized D2D communication systems”, *IEEE Comm. Magazine*, vol. 56(5), pp. 32–38, doi: <https://doi.org/10.1109/mcom.2018.1700876>

117. Wang, Y., Sheng, M., Wang, X., Wang, L. and Li, J. (2016), “Mobile-Edge computing: Partial computation offloading using dynamic voltage scaling” *IEEE Trans. on Comm.*, 1, doi: <https://doi.org/10.1109/tcomm.2016.2599530>

118. Weyrich, M. and Ebert, C. (2016), “Reference architectures for the Internet of things”, *IEEE Software*, vol. 33(1), pp. 112–116, doi: <https://doi.org/10.1109/ms.2016.20>

119. Wiśniewski, R., Patalas-Maliszewska, J., Wojnakowski, M., Topczak, M. and Zhou, M. (2023), “Fast Verification of Petri Net-Based Model of Industrial Decision-Making Systems: A Case Study”, *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, pp. 3316–3322, doi: <https://doi.org/10.1109/SMC53992.2023.10394156>

120. Xiao, S., Li, T., Guo, B., Huang, Z. Retraction Note: Cloud platform wireless sensor network detection system based on data sharing. *Cluster Computing*. 2019., Vol. 22, no. 6. P. 14157-14168. DOI: 10.1007/s10586-018-2260-6

121. Xue, H., Huang, B., Qin, M., Zou, H. and Yang, H. (2020), “Edge Computing for Internet of Things: A Survey”, *2020 Int. Conf. Internet Things IEEE Green Comput. Commun. IEEE Cyber, Phys. Soc. Comput. IEEE Smart Data IEEE Congr. Cybermatics, IEEE*, doi: <https://doi.org/10.1109/ithings-greencom-cpscom-smartdata-cybermatics50389.2020.00130>

122. Yang, Y., Liu, X. and Lu, W. (2023), “A Cyber–Physical Systems-Based

Double-Layer Mapping Petri Net Model for Factory Process Flow Control”, *Applied Sciences* (Switzerland), vol. 13, is. 15, 8975, doi: <https://doi.org/10.3390/app13158975>

123. Yin, Y. and Deng, L. (2022), “A dynamic decentralized strategy of replica placement on edge computing”, *International Journal of Distributed Sensor Networks*, vol.18(8), doi: <https://doi.org/10.1177/15501329221115064>

124. Zhen Xiao, Weijia Song, Qu Chen. Dynamic Resource Allocation using Virtual Machines for Cloud Computing Environment. *IEEE transaction on parallel and distributed systems*. 2013. Vol. 24, Is. 6. P. 1107 –1117. DOI: 10.1109/TPDS.2012.283.v.

ДОДАТОК А

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА

Наукові праці, які відображають основні наукові результати дисертації.

Статті у періодичних наукових виданнях, що увійшли до переліку наукових фахових видань України:

1. Shyman A. Development of a Method for Assessing the Adequacy of a Computer System Model Based on Petri Nets / Shyman A., Kuchuk N., Filatova A., Bellorin-Herrera O. // *Advanced Information Systems*. Kharkiv, 2024. V. 8. № 3. P. 46–52. (Scopus, Ukraine, Б)

URL: <https://doi.org/10.20998/2522-9052.2024.3.05>

(Наказ МОН № 612 від 07.05.2019, Б, Scopus)

2. Шиман А.П. Розрахунок ефективності використання обчислювальних ресурсів самовідновлювальної комп'ютерної системи / Кучук Н.Г., Шиман А.П., Філоненко А.М., Бульба С.С. // *Системи управління, навігації та зв'язку*. Полтава, 2021. № 3 (65). С. 92–95. (Б)

URL: <https://doi.org/10.26906/SUNZ.2021.3.092>

(Наказ МОН № 409 від 17.03.2020, Б, зі змінами від 24.09.2020 №1188).

3. Шиман А.П. Динамічна відеофіксація транспортних засобів для системи моніторингу дорожнього руху у мегаполісі / Кучук Н.Г., Міхаль О.П., Шиман А.П., Науменко М.В. // *Системи управління, навігації та зв'язку*. Полтава, 2022. № 2 (68). С. 55–58. (Б)

URL: <https://doi.org/10.26906/SUNZ.2022.2.055>

(Наказ МОН № 409 від 17.03.2020, Б, зі змінами від 24.09.2020 №1188).

4. Shyman A. A Method for Distributing Transactions to Hybrid Cloud Data Storage / Kuchuk N., Shyman A. // *Control, Navigation and Communication Systems*. Poltava, 2022. № 4 (70). P. 100–103. (Б)

URL: <https://doi.org/10.26906/SUNZ.2022.4.100>

(Наказ МОН № 409 від 17.03.2020, Б, зі змінами від 24.09.2020 №1188).

5. Shyman A. Synthesis of a Computer Network for Controlling Moving Objects / Kuchuk N., Shyman A. // Control, Navigation and Communication Systems. Poltava, 2023. № 4 (74). P. 134–136. (B).

URL: <https://doi.org/10.26906/SUNZ.2023.4.134>

(Наказ МОН № 409 від 17.03.2020, Б, зі змінами від 24.09.2020 №1188).

Опубліковані праці апробаційного характеру:

6. Shyman A. Synthesis of The Theory of Motion of Solid Bodies Filled With Bulk Substances For Fault-Tolerant Identification of Their Parameters / Kolodenko V., Shyman A., Kalinin Y., Kuchuk N. // *IEEE 13th International Conference on Dependable Systems, Services and Technologies (DESSERT)* (Athens, Greece 13–15 October 2023). – Athens, 2023. – P. 1–6 (Scopus, Greece).

URL: <https://ieeexplore.ieee.org/document/10416449>

7. Shyman A. The Method of Dynamic Routing in Self-Healing Networks / Shyman A., Kuchuk N., Kovalenko A., Kashkevich S., Partyka S., Yankovskyi O. // *2024 IEEE 5th KhPI Week on Advanced Technology (KhPIWeek)* (Kharkiv, 07 – 11 October 2024). – Kharkiv, 2024. – P. 1–5 (Scopus, Ukraine).

URL: <https://ieeexplore.ieee.org/document/10877971>

8. Шиман А.П. Модель транзакційної системи для розгортання у хмарному середовищі / Гребенюк Д.С., Шиман А.П. // *Проблеми інформатизації: тези доповідей сьомої міжнародної науково-технічної конференції* (м. Харків, 13–15 листопада 2019 р.). Харків, 2019 – С. 74.

URL: <https://repository.kpi.kharkov.ua/handle/KhPI-Press/66589>

9. Шиман А.П. Синтез інформаційної системи на гіперконвергентній платформі / Кучук Н.Г., Шиман А.П., Гребенюк Д.С. // *Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей десятої міжнародної науково-технічної конференції* (м. Харків, 09 – 10 квітня 2020 р.). – Харків, 2020 – С.43.

URL: <http://repositsc.nuczu.edu.ua/bitstream/123456789/11448/1/%D0%9A>

[%D0%9E%D0%9D%D0%A4%D0%95%D0%A0%D0%A2%D0%9E%D0%BC_2_2020_%D0%BD%D0%BE%D0%B2.pdf](#)

10. Shyman A. Mathematical Model of the Information System Synthesis Process / Kuchuk N., Davydov V., Hrebeniuk D., Shyman A. // *Current Directions of Development of Information and Communication Technologies and Control Tools*: proceedings of 11-th International Scientific and Technical Conference (Kharkiv, 08 – 09 April 2021). – Kharkiv, 2021. – P. 21.

URL: <https://repository.kpi.kharkov.ua/handle/KhPI-Press/58528>

11. Шиман А.П. Знаходження оптимального інформаційного навантаження компонентів СВКС за комплексним показником / Кучук Н.Г., Бульба С.С, Шиман А.П. // *Проблеми інформатизації*: тези доповідей дев'ятої міжнародної науково-технічної конференції (м. Харків, 18 – 19 листопада, 2021 р.). – Харків, 2021. – С. 45.

URL: <https://repository.kpi.kharkov.ua/handle/KhPI-Press/58054>

12. Shyman A. Analysis of the Level of Automation of Traffic Management Processes / Shyman A., Kuchuk N., Hrebeniuk D., Davydov V. // *Current Directions of Development of Information and Communication Technologies and Control Tools*: proceedings of 12-th International Scientific and Technical Conference (Kharkiv, 27 – 28 April 2022). – Kharkiv, 2022. – P. 20.

URL: <https://repository.kpi.kharkov.ua/handle/KhPI-Press/57267>

13. Шиман А.П. Інтелектуальні логістичні системи підтримки дорожнього руху / Шиман А.П., Кучук Н.Г., Бульба С.С., Давидов В.В. // *Проблеми інформатизації*: тези доповідей десятої міжнародної науково-технічної конференції (м. Харків, 24 – 25 листопада 2022 р.). – Харків, 2022. – С. 115.

URL: <https://repository.kpi.kharkov.ua/handle/KhPI-Press/59913>

14. Шиман А.П. Інтелектуальні транспортні системи / Кучук Н.Г., Шиман А.П. // *Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління*: тези доповідей тринадцятої міжнародної

науково-технічної конференції (м. Харків, 26–27 квітня 2023 р.). – Харків, 2023 – С.55.

URL: <https://repository.kpi.kharkov.ua/handle/KhPI-Press/65156>

15. Shyman A. Artificial Intelligence for Traffic Data Analysis / Kuchuk N., Shyman A. // *Current Directions of Development of Information and Communication Technologies and Control Tools*: proceedings of 14-th International Scientific and Technical Conference (Kharkiv, 25 – 26 April 2024). – Kharkiv, 2024. – P. 59.

URL: <https://repository.kpi.kharkov.ua/handle/KhPI-Press/76845>

16. Shyman A. Development of a Method for Synthesising Models of Real-Time Software Systems / Kuchuk N., Shyman A., Shyman M. // *Problems of Informatization*: proceedings of 12-th International Scientific and Technical Conference (Kharkiv, 21–22 November 2024). – Kharkiv, 2024. – P. 142.

URL: https://nure.ua/wp-content/uploads/2024/pi_12_vol_2.pdf

ДОДАТОК Б
АКТИ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ НАУКОВИХ ДОСЛІДЖЕНЬ
ДИСЕРТАЦІЙНОЇ РОБОТИ



I APPROVE

CEO

OLEKSANDR
HORPYNCHENKO

«30» December 2024

IMPLEMENTATION ACT

of the results of scientific research of the dissertation work

Anna Shyman

This document certifies that the results of scientific research were applied in the design of the complex system of the company “EURO RAIL SP. Z O.O.”.

Anna Shyman's dissertation, "Synthesis of the Edge Layer Cluster of the Internet of Things Computer Support System," addresses a significant scientific and technical problem—enhancing the efficiency of processing operational transactions at the edge layer of the Internet of Things.

The practical significance of the obtained results lies in the improvement of the short-term forecasting method for the time parameters of the IoT edge layer cluster. This method takes into account the limited computational capabilities of IoT edge layer nodes and is based on an appropriate functional model.

The proposed approach has been integrated into the system architecture, optimizing resource allocation and improving real-time data processing. As a result, the efficiency of computational operations in the IoT edge environment has been significantly enhanced, contributing to the overall reliability and responsiveness of the system.



Oleksandr HORPYNCHENKO

EURO RAIL SP. Z O.O.
REGON: 388388101, NIP: 6772462040
ul. Józefa Ignacego Kraszewskiego 36/128, 30-110 Kraków, Poland



ЗАТВЕРДЖУЮ

 Проректор НТУ"ХПІ",
 за науково-педагогічної роботи
 професор

Руслан МИГУЩЕНКО

" 12 2024 р.

АКТ

про використання результатів дисертаційної роботи на здобуття наукового ступеня доктора філософії за спеціальністю 123 Комп'ютерна інженерія аспірантки кафедри «Комп'ютерна інженерія та програмування» Шиман Анни Павлівни

Матеріали дисертаційної роботи на здобуття наукового ступеня доктора філософії Шиман Анни Павлівни, в котрій отримав подальший розвиток метод формування кластера граничного шару інфраструктури підтримки Інтернету речей шляхом використання функціональної моделі кластера та апарату напівмарківських процесів з можливою редукцією кількості станів мережі Петрі, використовуються здобувачкою у навчальному процесі Національного технічного університету «Харківський політехнічний інститут» на кафедрі «Комп'ютерна інженерія та програмування» при підготовці бакалаврів за спеціальністю 123 Комп'ютерна інженерія.

При викладанні навчальної дисципліни «Системи штучного інтелекту» впроваджено результати дисертаційного дослідження, зокрема розроблені етапи обробки інформації, що дозволяють аналізувати процеси з різним рівнем деталізації. Особливу увагу приділено взаємодії програми з іншими компонентами програмної системи під час оцінки її продуктивності.

Результати дисертаційної роботи аспірантки кафедри «Комп'ютерна інженерія та програмування» НТУ «ХПІ» Шиман А.П. відображені у ряді наукових публікацій.

Завідувач кафедри "КІП",
 д.т.н., проф.

Олександр ЗАКОВОРОТНИЙ

Гарант ОНП «Комп'ютерна інженерія»
 д.т.н., проф.

Сергій ЛЕОНОВ



ЗАТВЕРДЖУЮ

Проректор НТУ "ХПІ",

з науково-педагогічної роботи

професор

Руслан МИГУЩЕНКО

"січня" 2025р.

АКТ

про використання результатів дисертаційної роботи на здобуття наукового ступеня доктора філософії за спеціальністю 123 Комп'ютерна інженерія асистентки кафедри «Системи інформації ім. В.О. Кравця» Шиман Анни Павлівни

Дисертаційна робота Шиман Анни Павлівни «Синтез кластера граничного шару комп'ютерної системи підтримки Інтернету речей» присвячена вирішенню важливої науково-технічної задачі щодо підвищення ефективності процесу обробки оперативних транзакцій.

Практичне значення отриманих результатів полягає в удосконаленні методу побудови функціональних моделей кластерів граничного шару інфраструктури підтримки Інтернету речей. Даний метод використовує апарат узагальнених часових мереж Петрі, що дозволяє отримувати значення перехідних ймовірностей між станами мережі на різних рівнях деталізації під час моделювання процесу робочого навантаження.

Практичні результати дисертаційної роботи використані у навчальних матеріалах дисциплін «Проектування серверних застосунків», «Паралельні та розподілені обчислення», «Об'єктно-орієнтоване програмування».

Завідувач кафедри «Системи інформації ім. В.О. Кравця»

д.т.н., проф. Павло ПУСТОВОЙТОВ