НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ
ІНСТИТУТ ПРОБЛЕМ МОДЕЛЮВАННЯ В ЕНЕРГЕТИЦІ
ІМ. Г.Є. ПУХОВА

Кваліфікаційна наукова праця
на правах рукопису

ШКАРУПИЛО ВАДИМ ВІКТОРОВИЧ

УДК 004.052.42

ДИСЕРТАЦІЯ

МЕТОДИ І ЗАСОБИ КОНТРОЛЮ АРТЕФАКТІВ ПРОЦЕСУ ПРОЄКТУВАННЯ ПРОГРАМНО-АЛГОРИТМІЧНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМ КРИТИЧНОГО ПРИЗНАЧЕННЯ

05.13.05 – комп’ютерні системи та компоненти
(технічні науки)

Подається на здобуття наукового ступеня доктора технічних наук.

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

_____ В.В. Шкарупило

Науковий консультант: Чемерис Олександр Анатолійович,
доктор технічних наук, професор

Київ – 2024

АНОТАЦІЯ

Шкарупило В.В. Методи і засоби контролю артефактів процесу проєктування програмно-алгоритмічного забезпечення систем критичного призначення. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора технічних наук за спеціальністю 05.13.05 – комп'ютерні системи та компоненти. – Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України, Київ, 2024.

Дисертаційну роботу присвячено вирішенню актуальної науково-технічної проблеми забезпечення контролю артефактів процесу проєктування програмно-алгоритмічної складової (ПАС) систем критичного призначення (СКП) стосовно їх несуперечності – за заданими показниками функціональних (ФХ) і нефункціональних характеристик (НФХ).

Для вирішення зазначеної проблеми за мету ставиться підвищення ефективності контролю артефактів у процесі розроблення програмно-алгоритмічної складової систем критичного призначення на етапі проєктування для забезпечення несуперечності артефактів та зниження супутніх витрат, за рахунок розроблення, дослідження і застосування формальних методів, розвитку методу, моделей, супутніх засобів, у тому числі засобів автоматизації.

Для досягнення сформульованої мети поставлено відповідні задачі, вирішення яких викладено у межах шести розділів. Результатами вирішення є, у тому числі, наступні наукові здобутки – методи, розвиток методу, відповідні моделі, призначені до комплексного застосування при проєктуванні ПАС.

При викладенні отриманих результатів застосовано відповідний понятійний апарат, серед центральних елементів якого – поняття «артефакт» – сутність, що характеризується архітектурою (структурою та зв'язками) і змістом

– як результат виконання певного кроку етапу проєктування ПАС, поданий у формі файлу операційної системи.

У роботі проведено класифікацію залучених до дисертаційного дослідження артефактів – виокремлено відповідні типи. Серед опрацьованих артефактів – графічні подання ПАС (блок-схеми алгоритмів, UML-діаграми дій); відповідні формалізовані подання – формальні специфікації (ФС); програмні реалізації як засоби автоматизації; комп'ютерні моделі як засоби контролю значення досліджуваного показника НФХ. У якості такого показника залучено часові витрати, супутні реалізації кроків ПАС. При цьому несуперечність ПАС опрацьовано у якості досліджуваного показника ФХ.

У якості засобу сполучення ключових отриманих наукових здобутків розроблено і застосовано відповідний комплексний підхід, призначений до реалізації при проєктуванні ПАС. Комплексність підходу полягає у поєднанні методів і засобів контролю показників і ФХ, і НФХ.

У першому розділі дисертацій викладено отримані результати проведеного аналізу аспектів застосування формальних методів і засобів, зокрема методів перевірки на моделі, у процесі розроблення систем критичного призначення. Поставлено акцент на важливості залучення зазначених засобів у якості інструментів реалізації контролю показників і ФХ, і НФХ при проєктуванні ПАС. Обґрунтовано доцільність опрацювання саме ПАС комп'ютерної системи. На підставі отриманих результатів сформульовано засади для розроблення комплексного рішення – у формі підходу, яке є засобом сполучення винесених до захисту наукових здобутків, призначених до застосування при проєктуванні ПАС.

У другому розділі викладено запропонований підхід до комплексного застосування розроблених методів, розвитку методу, моделей – у якості засобів контролю несуперечності ПАС як досліджуваного показника ФХ і супутніх часових витрат – як показника НФХ.

Представлено розроблену модель подання ПАС у формі ФС, призначену слугувати засобом уніфікації формалізованих подань. Уніфікацію при цьому опрацьовано у якості чинника уможливлення автоматизації процесу постачання ФС – вихідних конструкцій, по відношенню до яких застосовується метод перевірки на моделі як метод формальної верифікації (ФВ). Розроблений розвиток зазначеного методу викладено у четвертому розділі.

Розроблена модель базується на виокремленні двох рівнів опрацювання ФС, одержуваних на її основі, – аналітичного рівня і рівня реалізації. Аналітичний рівень – для мисленнєвого охоплення артефактів розробником. Рівень реалізації – для уможливлення автоматизації процесу ФВ.

Складовими розробленої моделі є конструкції на основі виразних засобів структури Кріпке, формалізмів CSP, PlusCal, TLA+.

Залучення аксіоми композиції Гоара дозволило скоротити кількість рядків коду у складі результуючих ФС. Проведено кількісне оцінювання одержуваного у результаті цього корисного ефекту. Отримані значення охарактеризовано як вагомі.

У третьому розділі викладено два розроблені методи – метод синтезу ФС на основі графічних подань ПАС – блок-схем алгоритмів, UML-діаграм дій; метод контролю відповідності результуючих ФС первинним артефактам – графічним поданням ПАС, призначений до застосування у якості допоміжного засобу – на заключному кроці розробленого методу синтезу ФС. Залучення такого засобу дає підстави поширювати висновки за результатами формальної верифікації ФС на первинні графічні подання ПАС.

Розроблений метод синтезу ФС базується на оперуванні запропонованими правилами перетворення конструкцій у складі артефактів відмінних типів, із дотриманням засад, сформованих у межах розробленої моделі подання ПАС, викладеної у другому розділі.

У четвертому розділі викладено розроблений розвиток поширеного методу перевірки на моделі TLC (TLA Checker), призначений до застосування за ітеративного підходу до організації процесу ФВ.

Проведений розвиток полягає у сполученні методів обходу у глибину і у ширину теорії графів при здійсненні обходів простору станів системи переходів, що будується і перевіряється у відповідності до формалізованого подання ПАС у процесі ФВ. Викладено і проаналізовано отримані результати проведених досліджень.

Дослідження проведено і для граничних, і для предметно-орієнтованих сценаріїв застосування розробленого розвитку методу. Граничні випадки при цьому опрацьовано для визначення меж одержуваного корисного ефекту від проведеного розвитку, у залежності як від архітектурної складової ФС, такі від кількості залучених змінних. Дослідження проведено у розрізі опрацювання і обчислювальних, і просторових витрат, супутніх процесу ФВ.

Предметно-орієнтовані сценарії залучено з галузі енергетики, аерокосмічної галузі. При цьому, у розрізі актуальних особливостей обчислювальних систем, було також опрацьовано одержуваний корисний ефект від введення мультипоточності до складу програмної реалізації.

У п’ятому розділі викладено розроблену модель – стратифіковану архітектуру, де ієрархічний підхід застосовано у якості засобу досягнення архітектурної відповідності результуючої складеної комп’ютерної моделі вихідній ФС, несуперечність якої вже було підтверджено шляхом формальної верифікації – методом перевірки на моделі. Розроблена модель включає, у тому числі, засоби подання НФХ. Модель побудовано на основі засобів математичного апарату DEVS (Discrete Event System Specification) Бернарда Зейглера (Bernard Phillip Zeigler). Розроблена модель базується на оперуванні конструкціями «атомарної» і «складеної» моделей формалізму DEVS.

Залучення розробленої моделі у якості складової запропонованого комплексу засобів, призначених до застосування при проєктуванні ПАС згідно розробленого підходу, викладеного у другому розділі, дозволяє слідувати принципам модульності і безшовності при опрацюванні формалізованих подань.

У якості досліджуваного показника НФХ опрацьовано часові витрати, супутні реалізації ФХ згідно ПАС.

У шостому розділі викладено розроблений метод контролю досліджуваного показника НФХ при проєктуванні ПАС. Метод полягає у проведенні дискретно-подійного імітаційного моделювання – на основі комп'ютерних моделей, побудованих у відповідності до моделі – стратифікованої архітектури, викладеної у п'ятому розділі. Методом передбачено опрацювання як оціночних значень складових досліджуваного показника, так і фактичних. На основі отриманих результатів проведених експериментальних досліджень розробленого методу, на прикладі композитного вебсервісу, показано, що оперування оціночними значеннями супроводжується меншими часовими витратами, супутніми застосуванню методу.

Винесені на захист результати дисертаційної роботи пройшли відповідну апробацію. Практичне значення отриманих результатів підтверджено актом впровадження, листами підтвердження впровадження, листами підтримки. Копії документів, досліджувані артефакти, фрагменти артефактів, фрагменти програмних реалізацій засобів автоматизації подано у додатках.

Ключові слова: артефакт, верифікація, перевірка на моделі, програмно-алгоритмічна складова, система критичного призначення, специфікація.

ABSTRACT

Shkarupylo V.V. Methods and means for safety-critical systems' software and algorithmic constituent design process artifacts control. – Manuscript.

Thesis for Doctor of Technical Sciences Degree in 05.13.05 specialty – Computer Systems and Components. – G.E. Pukhov Institute for Modelling in Energy Engineering of the National Academy of Sciences of Ukraine, Kyiv, 2024.

Dissertation is devoted to the following actual scientific and technical problem solving: provide the control of the artifacts taking place at the design stage of software-algorithmic constituent (SAC) of the software critical system (SCS) with respect to its consistency – basing on specified indexes of both functional (FPs) and non-functional (NFPs) properties.

To solve named problem, the goal is to increase the effectiveness of artifacts control during the process of safety critical systems' software and algorithmic constituent engineering at the design stage, to ensure the consistency of artifacts and reduce the related costs, due to the development, research and application of formal methods, further development of the existing method, models and related tools, including the automation-centric ones.

When presenting the obtained results, an appropriate conceptual apparatus has been used. Among the central elements of named apparatus is the concept of an "artifact" – an entity with architecture (structure and couplings) and content – as an outcome of certain step of the SAC design stage accomplishment, presented in the form of an operating system file.

In the work, the artifacts involved in the dissertation research have been classified – the corresponding types have been distinguished. Among the encompassed artifacts are graphic representations of the SAC (algorithm block diagrams, UML activity diagrams); corresponding formalized representations –

formal specifications (FS); software implementations as the means of automation; computer models as the means of controlling the values of the NFP index under consideration, e.g., time costs related to the SAC steps accomplishment. At the same time, the consistency of the SAC has been considered as the FP index under study.

As an instrument for combining the key obtained scientific results, an appropriate comprehensive approach, intended to be applied at the design of the SAC, has been developed and implemented. The complexity of the approach grounds on the combination of methods and means of controlling the indexes of both the FPs and the NFPs.

In the first section, the results of the analysis of formal methods and tools, model checking methods in particular, application aspects during the process of safety critical systems engineering have been provided. An emphasis has been put on the importance of named means involvement to control the indexes of both the FPs and the NFPs at the design of the SAC. The expediency of the SAC research has been substantiated. Basing on the obtained results, the principles for the development of a complex solution have been formulated – as an approach to combining the obtained scientific achievements at the design of the SAC.

In the second section, the proposed approach to created methods, method amendmend and models complex application as the means of controlling the consistency of the SAC as the FP index and the related time costs as the NFP index has been introduced.

The proposed model of the SAC representation in the form of the FS has been described. Named model is intended to serve as an instrument for the FSs unification. At the same time, unification has been considered as a factor enabling the automation of the FSs supply process – as the input constructs for the model checking formal verification (FV) method. The proposed modification of such method has been introduced in the fourth section.

Created model is based on differentiation between two layers of the FS processing – the analytical layer and the implementation-related one. Analytical layer – as a plane of artifact perception by the developer. Implementation-related layer – to brign to life the automation of FV process.

The constituents of the proposed model are the constructs based on the expressive means of the Kripke structure, CSP, PlusCal, and TLA+ formalisms.

Usage of Hoare's composition rule has made it possible to reduce the number of code lines in the resulting FSs. Quantitative evaluation of the outcome has been conducted. Obtained values have been characterized as significant.

In the third section, two proposed methods have been described – the FSs synthesis method based on graphical representations of the SAC – algorithm block diagrams, UML activity diagrams; the method of controlling the compliance of the resulting FSs with the initial artifacts – graphical representation of the SAC, intended to be applied as an auxiliary tool – at the final stage of the proposed method of the FSs synthesis. Usage of this method provides the grounding for spreading the conclusions regarding the results of formal verification of the FSs on the initial artifacts – graphical representations of the SAC.

Proposed method of the FSs synthesis is based on the operation with the created rules for constructs transformation of the artifacts of different types, in compliance with the principles formed within the proposed model of the SAC representation, introduced previously – in the second section.

In the fourth section, proposed modification of widespread TLC (TLA Checker) model checker has been described: it is intended to be applied within iterative approach to the FV process organization.

Modification that has been conducted is about combining the depth- and breadth-first search methods of the graph-theory while traversing the transition system's state space. Named transition system is being constructed and checked with

respect to formal representation of the SAC during the FV process. Obtained results of the conducted research have been presented and analyzed.

Study has been carried out for both boundary and domain-specific scenarios of the proposed method modification application. At the same time, boundary cases have been encompassed to determine the limits of positive outcome from modification conducted – depending on the architectural constituent of the FS and the number of variables utilized. Study has been carried out with respect to processing both computational and spatial costs related to the FV process.

Domain-specific scenarios have been involved from the energy and aerospace domains. At the same time, to encompass also the multithreading as the distinctive feature of modern computing systems, related positive effect from multithreading implementation has also been estimated.

In the fifth section, the created model as a stratified architecture has been introduced. In the proposed model, the hierarchical approach has been utilized as an instrument for achieving the architectural compliance of the resulting computer model with the initial FS, which consistency has already been confirmed by formal verification – with the model checker. Named model also includes the means of the NFPs representation.

Model is built on the basis of the DEVS mathematical apparatus (by Bernard Phillip Zeigler), and is grounded on the operation with the constructs of “atomic” and “coupled” DEVS-models. Usage of the proposed model as a constituent of the introduced framework to be applied in accordance with the created comprehensive approach that is described in the second section allows to stick to the principles of modularity and seamlessness when processing the formal representations of the SAC at design.

Time costs related with the implementation of the FP with respect to the SAC are approached as the NFP index under consideration.

In the sixth section, it has been described the proposed method to control the specified NFP index during the the SAC design. The method is about carrying out the discrete-event simulation basing on the computer models created with respect to the model as a stratified architecture introduced in the fifth section. Method makes it possible to operate with both estimated and actual values of the NFP index under study. Grounding on the obtained results of the proposed method experimental study, with the composite web service as the implementation related scenario, it has been demonstrated that manipulation with the estimated values is accompanied with lower time costs related with method application.

Obtained results have been proven. Practical value of results has been proven documentary – with implementation act, support letters, letters of implementation confirmation. Copies of the specified documents, as well as the artifacts, artifact fragments, fragments of automation utilities software implementations, have been provided in the appendices.

Keywords: artifact, verification, model checking, software and algorithmic constituent, safety-critical system, specification.

СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Наукові праці, у яких опубліковано основні наукові результати.

Монографії (6 праць), з яких 2 – опубліковані у закордонних виданнях, серед яких 1 – з індексацією у міжнародній наукометричній базі Scopus:

1. Shkarupylo V.V., Timenko A.V. On the interoperability and consistency aspects with respect to the Internet of Things domain. Engineering sciences: development prospects in countries of Europe at the beginning of the third millennium: Collective monograph. Vol. 2. Riga: Izdevnieciba “Baltija Publishing”, 2018. P. 466–485. ISBN 978-9934-571-63-3 (**розділ колективної монографії**)

2. Блінов І.В., Парус Є.В., Шкарупило В.В. Структура та моделі інформаційної взаємодії учасників ринку електричної енергії: монографія. Вінниця: ГО «Європейська наукова платформа», 2021. 114 с. ISBN 978-617-8037-31-4. DOI: https://doi.org/10.36074/stmivyree-monograph.2021 (**колективна монографія**)

3. Шкарупило В.В., Блінов І.В. Сценарії, методи та засоби формальної верифікації артефактів процесу проєктування систем критичного призначення: монографія. Вінниця : ГО «Європейська наукова платформа», 2021. 104 с. ISBN 978-617-8037-55-0. DOI: https://doi.org/10.36074/smtzfvappskp-monograph.2021 (**колективна монографія**)

4. Shkarupylo V.V., Blinov I.V., Chemeris A.A., Dusheba V.V., Alsayaydeh J.A.J. On Applicability of Model Checking Technique in Power Systems and Electric Power Industry. In: Zaporozhets A. (eds) Systems, Decision and Control in Energy III. Studies in Systems, Decision and Control, 2022, Vol. 399. Springer, Cham. ISBN 978-3-030-87675-3. DOI: https://doi.org/10.1007/978-3-030-87675-3_1 (**Scopus:** https://www.scopus.com/record/display.uri?eid=2-s2.0-85120868312&origin=resultslist&sort=plf-f ; **розділ колективної монографії**)

5. Борукаєв З.Х., Блінов І.В., Остапченко К.Б., Чемерис О.А., Шкарупило В.В. Моделі та засоби автоматизації систем організаційного управління енергоринком: монографія / за заг. ред. З.Х. Борукаєва. — Вінниця: ГО «Європейська наукова платформа», 2022. 122 с. ISBN 978-617-8037-82-6 DOI: https://doi.org/10.36074/mtzasoye-monograph.2022 (**колективна монографія**)

6. Шкарупило В., Блінов І., Кучанський В., Давидюк А., Дімітрієва Д. Методи і засоби контролю артефактів процесу проєктування програмно-алгоритмічної складової систем критичного призначення: монографія / за заг. ред. В.В. Шкарупила. Publishing House «European Scientific Platform», 2023. 120 с. ISBN: 978-617-8126-22-3. DOI: https://doi.org/10.36074/mzkapppasskp-monograph.2023 (**колективна монографія**)

Статті у фахових періодичних виданнях (22 праці), серед яких 7 – у виданнях, що індексуються у міжнародних наукометричних базах Scopus та Web of Science Core Collection, 3 – у фахових виданнях категорії А:

7. Shkarupylo V.V., Tomičić I., Kasian K.M. The investigation of TLC model checker properties. *Journal of Information and Organizational Sciences*, 2016. Vol. 40, No. 1. P. 145–152. ISSN: 1846-9418 (Online), 1846-3312 (Print). DOI: https://doi.org/10.31341/jios.40.1.7 (**Web of Science Core Collection:** https://www.webofscience.com/wos/woscc/full-record/WOS:000409240900008 ; **Scopus, Q4:** https://www.scopus.com/record/display.uri?eid=2-s2.0-84975057117&origin=resultslist)

8. Shkarupylo V., Skrupsky S., Oliinyk A., Kolpakova T. Development of stratified approach to software defined networks simulation. *Eastern-European Journal of Enterprise Technologies. Information and controlling systems*, 2017. Vol. 5, No. 9 (89). P. 67–73. ISSN: 1729-3774 (Print), 1729-4061 (Online). DOI: https://doi.org/10.15587/1729-4061.2017.110142 (**Scopus, Q3**:

https://www.scopus.com/record/display.uri?eid=2-s2.0-85031750626&origin=resultslist; **фахове видання категорії А**)

9. Alsayaydeh J.A.J., Shkarupylo V., Hamid M. S. B., Skrupsky S., Oliinyk A. Stratified model of the Internet of Things infrastructure. *Journal of Engineering and Applied Sciences*, 2018. Vol. 13, No. 20. P. 8634–8638. ISSN: 1816-949x (Print), 1818-7803 (Online). DOI: https://medwelljournals.com/abstract/?doi=jeasci.2018.8634.8638 (**Scopus, Q3**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85056326734&origin=resultslist)

10. Timenko A.V., Shkarupylo V.V., Oliinyk A.O., Hrushko S.S. Formal Model for Checking the Interoperability Between the Components of the IoT system. *Problemele Energeticii Regionale*, 2019. Vol. 40, No. 1-1. P. 69–78. ISSN: 1857-0070. DOI: https://zenodo.org/record/3239196 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000472596400007)

11. Shkarupylo V., Alsayaydeh J.A.J, Tomičić I., Chemeris A., Dusheba V. A technique for checking the adequacy of formal model. *ARPN Journal of Engineering and Applied Sciences*, August 2021. Vol. 16, No. 16. P. 1707–1719. ISSN: 1819-6608. URL: http://www.arpnjournals.org/jeas/research_papers/rp_2021/jeas_0821_8670.pdf (дата звернення: 06.08.2023) (**Scopus, Q3**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85118181893&origin=resultslist)

12. Polska O.V., Kudermetov R.K., Shkarupylo V.V. An approach web service selection by quality criteria based on sensitivity analysis of MCDM methods. *Radio Electronics, Computer Science, Control*, 2021. No. 2. P. 133–143. ISSN: 1607-3274 (Online), 2313-688X (Print). DOI: https://doi.org/10.15588/1607-3274-2021-2-14 (**Web of Science Core Collection:**

https://www.webofscience.com/wos/woscc/full-record/WOS:000673377700014 **; фахове видання категорії А**)

13. Shkarupylo V., Blinov I., Dusheba V., Alsayaydeh J. A. J. Case Driven TLC Model Checker Analysis in Energy Scenario. *CEUR Workshop Proceedings*, 2023. Vol. 3392. P. 65–75. ISSN 1613-0073. DOI: https://doi.org/10.32782/cmis/3392-6 (**Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85160296577&origin=resultslist&sort=plf-f)

14. Шкарупило В.В., Кудерметов Р.К., Польська О.В. DEVS-орієнтована методика валідації композитних веб-сервісів. *Радіоелектроніка, інформатика, управління*, 2015. № 4. С. 79–86. ISSN: 1607-3274 (Print), 2313-688X (Online). DOI: 10.15588/1607-3274-2015-4-12 URL: http://ric.zntu.edu.ua/article/view/60404 (дата звернення: 06.08.2023) (**фахове видання категорії А**)

15. Polska O.V., Kudermetov R.K., Shkarupylo.V.V. Discovery and selection of web-services. *Electrotechnic and computer systems*, 2015. No. 19(95). P. 169–173. ISSN: 2221-3937 (Print), 2221-3805 (Online). URL: http://nbuv.gov.ua/UJRN/etks_2015_19_39 (дата звернення: 06.08.2023) (**фахове видання**)

16. Kudermetov R., Polska O., Shkarupylo V., Shcherbak N. Quality of services in scientific workflows. *Electrotechnic and Computer Systems*, 2018. Vol. 28, No. 104. P. 170–177. ISSN: 2221-3805 (Print), 2221-3937 (Online). DOI: 10.15276/eltecs.28.104.2018.20 URL: https://eltecs.op.edu.ua/index.php/journal/article/view/155/42 (дата звернення: 06.08.2023) (**фахове видання**)

17. Shkarupylo V.V., Tomičić I., Kasian K.M., Alsayaydeh J.A.J. An Approach to increase the Effectiveness of TLC Verification with Respect to the Concurrent Structure of TLA+ Specification. *International Journal of Software*

Engineering and Computer Systems, 2018. Vol. 4, No. 1. P. 48–60. ISSN: 2289-8522. DOI: https://doi.org/10.15282/ijsecs.4.1.2018.4.0037 (**фахове видання**)

18. Shkarupylo V.V., Kudermetov R.K., Polska O.V. On the approaches to cyber-physical systems simulation. *Advances in Cyber-Physical Systems (ACPS)*, 2018. Vol. 3, No. 1. P. 51–54. ISSN: 2524-0382 (Print), 2707-0069 (Online). DOI: https://doi.org/10.23939/acps2018.01.051 (**фахове видання**)

19. Шкарупило В.В., Євдокимов В.Ф., Душеба В.В. Застосування формальних методів для перевірки систем критичного призначення. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2019. Том 30 (69), Ч. 1, № 6. С. 188–193. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI https://doi.org/10.32838/2663-5941/2019.6-1/34 (**фахове видання**)

20. Тіменко А.В., Шкарупило В.В., Скрупський С.Ю., Смолій В.В. Дослідження шляхів підвищення пропускної спроможності підсистеми пам'яті сучасної обчислювальної системи. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2020. Том 31 (70), Ч. 1, № 2. С. 208–212. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/2663-5941/2020.2-1/32 (**фахове видання**)

21. Шкарупило В.В., Чемерис О.А., Душеба В.В., Кудерметов Р.К., Польська О.В. Метод синтезу формальних специфікацій на основі трійок Хоара. *Наукові праці ДонНТУ, Серія "Інформатика, кібернетика та обчислювальна техніка"*, 2020. № 1(30). С. 49–57. ISSN: 1996-1588. DOI: 10.31474/1996-1588-2020-1-30-49-57 URL: https://iktv.donntu.edu.ua/1-30-2020/ (дата звернення: 06.08.2023) (**фахове видання**)

22. Шкарупило В.В., Чемерис О.А., Душеба В.В. Оцінювання просторової складності задачі формальної верифікації, вирішуваної методом перевірки на моделі. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2020. Том 31 (70), № 5. С. 147–

151. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/2663-5941/2020.5/24 (дата звернення: 06.08.2023) (**фахове видання**)

23. Шкарупило В.В., Чемерис О.А., Душеба В.В., Кудерметов Р.К. Дослідження мультипоточної реалізації методу перевірки на моделі для темпоральної логіки дій. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2020. Том 31 (70), № 6, Ч. 1. С. 173–177. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/TNU-2663-5941/2020.6-1/28 (**фахове видання**)

24. Polska O.V., Kudermetov R.K., Shkarupylo V.V. The approach for QoS based web service selection with user's preferences. *Наукові праці Донецького національного технічного університету, серія: «Проблеми моделювання та автоматизації проектування»*, 2020. №2 (16). С. 19–27. ISSN: 2074-7888. DOI: 10.31474/2074-7888-2020-2-19-27 URL: http://pmap.donntu.edu.ua/sites/upload/articles/pmap_2020_19-27.pdf (дата звернення: 06.08.2023) (**фахове видання**)

25. Polska O.V., Kudermetov R.K., Zolotukhina O.A., Shkarupylo V.V. A UML profile for quality-based web service selection using logic scoring of preference method. *Telecommunication and information technologies*, 2021. No. 1 (2021). P. 65–78. ISSN: 2412-4338. DOI: https://doi.org/10.31673/2412-4338.2021.016578 (**фахове видання**)

26. Шкарупило В.В., Чемерис О.А., Душеба В.В., Кудерметов Р.К., Польська О.В. Модельно-орієнтований підхід до контролю показників нефункціональних характеристик під час проєктування. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2021. Том 32 (71), Ч. 1, № 1. С. 166–171. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/2663-5941/2021.1-1/27 (**фахове видання**)

27. Шкарупило В.В., Душеба В.В., Скрупський С.Ю., Блінов І.В. Стратифікована модель подання нефункціональних характеристик системи критичного призначення при проєктуванні. *Електронне моделювання,* 2022. Т. 44, № 2. С. 90–106. ISSN: 0204-3572. DOI: https://doi.org/10.15407/emodel.44.02.090 (**фахове видання**)

28. Куликовська Н.А., Руденко В.В., Тіменко А.В., Шкарупило В.В. Дослідження часу збирання додатків, побудованих на основі сучасних стратегій розроблення. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2023. Том 34 (73), № 4. С. 65–70. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32782/2663-5941/2023.4/11 (**фахове видання**)

Праці апробаційного характеру (40 праць), серед яких 7 – з індексацією у міжнародних наукометричних базах Scopus та Web of Science Core Collection:

29. Shkarupylo V. A Technique of DEVS-Driven Validation. *Proc. XIIIth Int. Conf. on Modern Problems of Radio Engineering, Telecommunications, and Computer Science, TCSET'2016* (Lviv-Slavske, Ukraine, February 23–26, 2016). P. 495–497. DOI: https://doi.org/10.1109/TCSET.2016.7452097 (**Web of Science Core Collection:** https://www.webofscience.com/wos/woscc/full-record/WOS:000381804300127 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-84969263650&origin=resultslist)

30. Shkarupylo V. A Simulation-driven Approach for Composite Web Services Validation. *Proc. 27th Int. Central European Conference on Information and Intelligent Systems, CECIIS 2016* (Varazdin, Croatia, September 21–23, 2016). P. 227–231. URL: http://archive.ceciis.foi.hr/app/public/conferences/1/ceciis2016/papers/QoS-1.pdf

(дата звернення: 06.08.2023) (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000595003500030)

31. Shkarupylo V., Polska O. The Approach to SDN Network Topology Verification on a Basis of Temporal Logic of Actions. *Proc. 14th Int. Conf. on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET'2018* (Lviv-Slavske, Ukraine, February 20–24, 2018). P. 183–186. DOI: https://doi.org/10.1109/TCSET.2018.8336182 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000465121700033 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85047524592&origin=resultslist)

32. Shkarupylo V., Kudermetov R., Golub T., Polska O., Tiahunova M. Towards Model Checking of the Internet of Things Solutions Interoperability. *Problems of Infocommunications. Science and Technology: proc. 2018 IEEE International Scientific and Practical Conference, PIC S&T-2018* (Kharkiv, Ukraine, October 9–12, 2018). P. 465–468. DOI: https://doi.org/10.1109/INFOCOMMST.2018.8632037 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000458659100087 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85062879597&origin=resultslist)

33. Shkarupylo V., Kudermetov R., Timenko A., Polska O. On the Aspects of IoT Protocols Specification and Verification. *Problems of Infocommunications. Science and Technology: 2019 International Scientific-Practical Conference, PIC S&T'2019* (Kyiv, Ukraine, October 8–11, 2019). P. 93–96. DOI: https://doi.org/10.1109/PICST47496.2019.9061406 (**Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85083637232&origin=resultslist)

34. Shkarupylo V., Chemerys O., Dusheba V., Kudermetov R., Oliinyk A. On Hoare triples applicability to dependable system specification synthesis. *Dependable Systems, Services and Technologies, DESSERT'2020: The 11th International Conference* (Kyiv, Ukraine, May 14–18, 2020). Kyiv, 2020. P. 371–375. DOI: https://doi.org/10.1109/DESSERT50317.2020.9125074 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000619228000064 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85087906543&origin=resultslist)

35. Shkarupylo V., Blinov I., Chemeris A., Dusheba V., Alsayaydeh J., Oliinyk A. Iterative Approach to TLC Model Checker Application. *Proc. 2021 IEEE KhPI Week on Advanced Technology* (Kharkiv, Ukraine, September 13–17, 2021). P. 283–287. DOI: https://doi.org/10.1109/KhPIWeek53812.2021.9570055 (**Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85118943601&origin=resultslist)

36. Шкарупило В.В. Концепція формальної верифікації UML-діаграм методами Model Checking. *Моделювання: XXXIV науково-технічна конференція*, 13–14 січня 2015 р.: тези доп. К.: ІПМЕ ім. Г. Є. Пухова НАН України, 2015. С. 13.

37. Шкарупило В.В. Особливості використання методу формальної верифікації TLC. *Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*: тези доп., м. Київ, 12 січня 2016 р. С. 31. DOI: http://dx.doi.org/10.5281/zenodo.2545399

38. Shkarupylo V.V. An in-depth look at TLC model checker. *Тиждень науки-2016*: зб. тез доп. науково-практ. конф., 18–22 квітня 2016 р. Запоріжжя: ЗНТУ, 2016. С. 523–524. URL:

https://zp.edu.ua/uploads/conference/2016/TN2016_T1.pdf (дата звернення: 06.08.2023)

39. Shkarupylo V.V., Tomičić I., Arapin D.V. The concurrency representation in TLA+ specification. Proc. VIIIth Int. scientific-practical conf. on Modern problems and achievements of radio engineering (electronics), telecommunications and information technology (Zaporizhzhya, Ukraine, September 21–23, 2016). P. 118–119. URL: http://rtt.zntu.edu.ua/data/Tezy_ZNTU_2016.pdf (дата звернення: 06.08.2023)

40. Shkarupylo V. TLC model checking and the concurrency in specification. *Proc. Tenth International Scientific-Practical Conference "INTERNET-EDUCATION-SCIENCE-2016", IES-2016* (Vinnytsia, Ukraine, October 11–14, 2016). P. 89–91. URL: http://ir.lib.vntu.edu.ua/handle/123456789/13390 (дата звернення: 06.08.2023)

41. Shkarupylo V.V. An Approach to DEVS-driven Simulation of Software-defined Networks. *Тиждень науки-2017*: науково-практ. конф., 18–21 квітня 2017 р.: тези доп. Запоріжжя: ЗНТУ, 2017. С. 652. URL: https://zp.edu.ua/uploads/dept_s&r/2017/conf/1/TN2017.pdf (дата звернення: 06.08.2023)

42. Shkarupylo V.V., Timenko A.V. An approach to the Internet of things simulation on a basis of discrete event system specification. *Proc. Int. research and practice conference on Modern methods, innovations, and experience of practical application in the field of technical sciences* (Radom, Republic of Poland, Dec. 27–28, 2017). P. 32–34.

43. Shkarupylo V.V. On the applicability of model checking techniques in the Internet of Things domain. *Тиждень науки-2018*: науково-практ. конф., 16–20 квітня 2018 р.: тези доп. Запоріжжя: ЗНТУ, 2018. С. 967–968. URL: https://zp.edu.ua/uploads/dept_s&r/2018/conf/1/TN2018.pdf (дата звернення: 06.08.2023)

44. Shkarupylo V.V., Timenko A.V. On the expediency of stratification to foster the reconfigurability of formal specifications. *Тенденції та вектор розвитку науки в сучасному світі: VI Міжнародна науково-практична інтернет-конференція*: тези доповідей, Дніпро, 30 квітня 2018 р. Ч. 1. Дніпро: НБК, 2018. С. 46–49. URL: https://ispic.ngo-seb.com/assets/files/6_conf_30.04.18_P.1.pdf (дата звернення: 06.08.2023)

45. Shkarupylo V., Kudermetov R. On the aspects of cyber-physical systems modeling with UPPAAL. *Simulation-2018: 6th Int. conference*, September 12–14, 2018: theses. Kyiv: Pukhov Institute for Modelling in Energy Engineering, 2018. P. 267–269. URL: https://ipme.kiev.ua/en/conference/simulation-2018/ (дата звернення: 06.08.2023)

46. Shkarupylo V., Polska O., Shcherbak N. On the classification of model checking methods for the Internet of Things. *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: IX Міжнародна науково-практична конференція*, 3–5 жовтня 2018 р.: тези доп. Запоріжжя: ЗНТУ, 2018. С. 77–78.

47. Шкарупило В.В., Кудерметов Р.К., Польська О.В., Тіменко А.В. Щодо доцільності перевірки протоколів взаємодії компонентів систем інтернету речей. *Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2019*: матеріали VII Міжнародної науково-практичної конференції, 15–16 травня 2019 р. Київ: НУБіП України, 2019. С. 63–65. URL: https://lib.lntu.edu.ua/sites/default/files/2021-03/Київ_Конференція_НУБІП_2019_UA.pdf (дата звернення: 06.08.2023)

48. Шкарупило В.В., Євдокимов В.Ф., Душеба В.В. Аспекти застосування методів перевірки на моделі при проектуванні систем критичного призначення. *Безпека енергетики в епоху цифрової трансформації: науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України* : програма та матеріали, 20

грудня 2019 р. Київ : ІПМЕ ім. Г.Є.Пухова НАН України, 2019. С. 94–96. URL: https://ipme.kiev.ua/wp-content/uploads/2019/12/Програма-КБЕЕЦ-2019.pdf (дата звернення: 06.08.2023)

49. Шкарупило В.В. Про застосування правила композиції при синтезі формальних специфікацій. *Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*, 15 травня 2020 р. Київ : ІПМЕ ім. Г.Є. Пухова НАН України, 2020. С. 21–22. URL: https://zenodo.org/record/3813710 (дата звернення: 06.08.2023)

50. Шкарупило В.В. Дослідження методу перевірки на моделі TLC. *Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2020 : VIII Міжнародна науково-практична Інтернет-конференція* (м. Київ, Україна, 14–15 травня, 2020). 2020. Київ: НУБіП України. С. 84–86. URL: http://econference.nubip.edu.ua/index.php/grpi/grpi20/paper/view/2306/317 (дата звернення: 06.08.2023)

51. Шкарупило В.В., Скрупський С.Ю. Комбінований підхід до застосування методу перевірки на моделі TLC. *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: X Ювілейна міжнародна науково-практична конференція, присвячена 120-річчю з дня заснування Національного університету «Запорізька політехніка»* (Запоріжжя, Україна, 7–9 жовтня, 2020). Запоріжжя: НУ «ЗП». С. 95–97. URL: http://rtt.zntu.edu.ua/data/Tezy_NUZP_2020.pdf (дата звернення: 06.08.2023)

52. Шкарупило В.В., Кудерметов Р.К., Польська О.В. Дослідження просторової складності алгоритмів в основі методу верифікації TLC. *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: X Ювілейна міжнародна науково-практична конференція, присвячена 120-річчю з дня заснування Національного університету «Запорізька*

політехніка» (Запоріжжя, Україна, 7–9 жовтня, 2020). Запоріжжя: НУ «ЗП». С. 93–95. URL: http://rtt.zntu.edu.ua/data/Tezy_NUZP_2020.pdf (дата звернення: 06.08.2023)

53. Шкарупило В.В., Чемерис О.А., Душеба В.В. Дослідження впливу мультипоточності на швидкодію методу перевірки на моделі. *Безпека енергетики в епоху цифрової трансформації*: Друга науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України (Київ, Україна, 28–29 грудня, 2020). Київ : ІПМЕ ім. Г.Є. Пухова НАН України, 2020. С. 75–77. URL: https://ipme.kiev.ua/wp-content/uploads/2021/01/Програма-та-матеріали-КБЕЕЦ-2020.pdf (дата звернення: 06.08.2023)

54. Шкарупило В.В., Блінов І.В. Щодо застосування методу перевірки на моделі при проєктуванні інформаційно-технологічних систем суб'єктів ринку електроенергії. *XXXIX науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, присвячена 40-річчю Інституту, Дню науки в Україні та з нагоди відзначення 30-ї річниці незалежності України* (м. Київ, Україна, 12 травня, 2021). Київ: ІПМЕ ім. Г.Є. Пухова НАН України. С. 7–9. URL: https://drive.google.com/file/d/1QOydMJU3nHOjXZ92vcLF2zAclu8w1RcG/view?usp=share_link (дата звернення: 06.08.2023)

55. Шкарупило В.В., Блінов І.В. Модельно-орієнтований підхід до формалізації нефункціональних характеристик систем критичного призначення, зокрема у природокористуванні. *Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні 2021*: IX Міжнародна науково-практична Інтернет-конференція (м. Київ, Україна, 13–14 травня, 2021). Київ: НУБіП України. С. 55–57. URL: https://drive.google.com/file/d/1IPmtWaLu85W3c9CsrYXYqx2_iFYIq4q2/view?usp=sharing (дата звернення: 06.08.2023)

56. Шкарупило В.В., Блінов І.В., Душеба В.В., Тіменко А.В. Дуальний підхід до формалізації функціональних характеристик систем критичного призначення. *European scientific discussions : 9th International scientific and practical conference. Potere della ragione Editore* (м. Рим, Італія, 18–20 липня, 2021 р.). С. 143–149. URL: https://sci-conf.com.ua/wp-content/uploads/2021/07/EUROPEAN-SCIENTIFIC-DISCUSSIONS-18-20.07.2021.pdf (дата звернення: 06.08.2023)

57. Шкарупило В.В., Блінов І.В., Душеба В.В., Кучанський В.В. Щодо мультипоточного застосування формального методу перевірки на моделі TLC. *Topical issues of modern science, society and education. Proceedings of the 2nd International scientific and practical conference. SPC "Sci-conf.com.ua"*. Kharkiv, Ukraine. 2021. P. 231–236. URL: https://sci-conf.com.ua/wp-content/uploads/2021/09/TOPICAL-ISSUES-OF-MODERN-SCIENCE-SOCIETY-AND-EDUCATION-5-7.09.21.pdf (дата звернення: 06.08.2023)

58. Дімітрієва Д.О., Шкарупило В.В. Огляд інструментів використання формальних методів та засобів при проєктуванні систем критичного призначення. *ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ: ЕКОНОМІКА, ТЕХНІКА, ОСВІТА '2021*: Збірник матеріалів XI Міжнародної науково-практичної конференції молодих вчених, 11–12 листопада 2021 року, НУБіП України, Київ. С. 164–165. URL: https://drive.google.com/file/d/10iRiRUwpXqTY510LzL1j0BeDt-Kpx4Ab/view?usp=sharing (дата звернення: 06.08.2023)

59. Шкарупило В.В., Душеба В.В. Підхід до синтезу формалізованих подань нефункціональних характеристик на етапі проєктування. *Безпека енергетики в епоху цифрової трансформації* : III науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України : матеріали, 22 грудня 2021 р. Київ : ІПМЕ ім. Г.Є.Пухова НАН України, 2021. С. 128–130. URL: https://ipme.kiev.ua/wp-

content/uploads/2021/12/Матеріали-КБЕЕЦ-2021-1.pdf (дата звернення: 06.08.2023)

60. Шкарупило В.В., Душеба В.В. Спадковість артефактів у контексті багатовимірної верифікації. *Тиждень науки-2022*: науково-практ. конф., 18–22 квітня 2022 р.: тези доп. Запоріжжя: НУ "Запорізька політехніка", 2022. С. 789–791. URL: https://zp.edu.ua/uploads/dept_s&r/2022/conf/4.1/TN_2022.pdf (дата звернення: 06.08.2023)

61. Шкарупило В.В., Душеба В.В. Модельно-орієнтований підхід до синтезу формалізованих подань. *XL Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*, 11 травня 2022 р.: тези доп. Київ: ІПМЕ ім. Г.Є. Пухова НАН України, 2022. С. 20–22. URL: https://bit.ly/3cPGyFk (дата звернення: 06.08.2023)

62. Дімітрієва Д.О., Шкарупило В.В. Огляд нефункціональних характеристик систем критичного призначення. *XL Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*, 11 травня 2022 р.: тези доп. Київ: ІПМЕ ім. Г.Є. Пухова НАН України, 2022. С. 78–79. URL: https://bit.ly/3cPGyFk (дата звернення: 06.08.2023)

63. Шкарупило В.В., Тіменко А.В. Складові методу контролю показників нефункціональних характеристик розроблюваної комп'ютерної системи при проєктуванні. *XLIX Міжнародна науково-практична інтернет-конференція «Проблеми та перспективи розвитку сучасної науки в країнах Європи та Азії»*, 31 серпня 2022 р.: тези доп., Переяслав, 2022. С. 69–71.

64. Шкарупило В.В., Душеба В.В. Щодо аспектів контролю несуперечності програмно-алгоритмічної складової систем критичного призначення. *Продовольча та екологічна безпека в умовах війни та повоєнної відбудови, присвячена 125-річчю заснування Національного університету*

біоресурсів і природокористування України: виклики для України та світу: мат. Міжн. наук.-практ. конф., секція 5: Інженерія, енергетика та інформаційні технології в умовах війни та післявоєнній відбудові країни (м. Київ, 25 трав. 2023 р.): тези доп. Київ: НУБіП України, 2023. С. 170–172. URL: https://nubip.edu.ua/sites/default/files/u381/sekciya_5.pdf (дата звернення: 06.08.2023)

65. Шкарупило В.В., Блінов І.В., Душеба В.В. Дослідження методу верифікації TLC при вирішенні задач енергетики. *XLI Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України* (м. Київ, Україна, 17 травня, 2023 р.). С. 21–22. URL: https://ipme.kiev.ua/wp-content/uploads/2023/05/Матеріали-конференції-2023.pdf (дата звернення: 06.08.2023)

66. Шкарупило В.В., Душеба В.В., Тіменко А.В., Казакова Н.О. Аспекти досягнення функційної безпечності при розробленні систем критичного призначення. *Актуальні проблеми та перспективи розвитку фундаментальних, прикладних, загальнотехнічних та безпекових наук*: ІІІ Всеукраїнської науково-практичної конференції пам’яті академіка Академії наук вищої освіти, професора Анатолія Володимировича Касперського (м. Київ, Україна, 21 червня 2023 р.): тези доп. Київ: УДУ ім. Михайла Драгоманова, 2023. С. 394–395. URL: https://drive.google.com/file/d/1nS1d9cf9EUNUZDnY0ZWLkQTaBlb0xoIN/view (дата звернення: 06.08.2023)

67. Шкарупило В.В., Душеба В.В., Тіменко А.В. Огляд рівнів забезпечення резилієнтності у галузі енергетики. *Survivability & Resilience – 2023: collection of materials of the international scientific and practical conference*, Kyiv, October 19, 2023, PIMEE of NAS of Ukraine. 2023. P. 33–34. URL: https://ipme.kiev.ua/konferencii/zhivuchist-ta-rezilyentnist-2023/ (дата звернення: 21.10.2023)

68. Шкарупило В.В., Душеба В.В. Аспекти введення мультипоточності до реалізації методу формальної верифікації TLC. *Безпека енергетики в епоху цифрової трансформації*: П’ята науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України, Київ, Україна, 22 листопада, 2023 р. Київ : ІПМЕ ім. Г.Є. Пухова НАН України. С. 121–122. URL: https://ipme.kiev.ua/konferencii/naukovo-praktichna-konferenciya-bevect-2023/ (дата звернення: 23.11.2023)

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

BFS – Breadth-first Search;

BMC – Bounded Model Checking;

CSP – Communicating Sequential Processes;

CTL – Computation Tree Logic;

CTMC – Continuous-time Markov Chain;

DEVS – Discrete Event System Specification;

DFS – Depth-first Search;

DTMC – Discrete-time Markov Chain;

ECSS – European Cooperation for Space Standardization;

JRE – Java Runtime Environment;

LTL – Linear Temporal Logic;

MBSA – Model-based Safety Analysis;

MC – Model Checking (Model Checker);

PMC – Probabilistic / Statistical Model Checking;

RV – Runtime Verification;

SAT – Boolean Satisfiability Problem;

SIL – Safety Integrity Level;

SMT – Satisfiability Modulo Theories;

SSIL – Software Safety Integrity Level;

SMC – Symbolic Model Checking;

STMC – Stateless Model Checking;

TLA – Temporal Logic of Actions;

TLC – TLA Checker;

UML – Unified Modeling Language;

VCC – Verifying Concurrent C;

V&V – Verification and Validation;

АМ – атомарна модель;

БК – бортовий комп’ютер;

БУК – блок управління конфігурацією;

БЦОК – бортовий цифровий обчислювальний комплекс;

ВА – вихідний артефакт;

КА – космічний апарат;

КПА – контрольно-перевірочна апаратура;

ЛОВ – локальний орган видачі (кодів реєстру);

МФС – модель ФС;

НФХ – нефункціональна характеристика;

ОП – оперативна пам’ять;

ПА – первинний артефакт;

ПАС – програмно-алгоритмічна складова;

ПВВ – пристрій введення/виведення;

ПВКО – підсистема визначення і керування орієнтацією;

ПСЕП – підсистема електропостачання;

РА – результуючий артефакт;

СКП – система критичного призначення;

СК – система керування;

СМ – складена модель;

СП – система переходів;

ФБ – функційна безпечність;

ФВ – формальна верифікація;

ФС – формальна специфікація;

ФХ – функціональна характеристика;

ЦОВ – центральний орган видачі (кодів реєстру).

ВСТУП

Дисертаційну роботу присвячено вирішенню актуальної науково-технічної проблеми забезпечення контролю артефактів процесу проєктування програмно-алгоритмічної складової (ПАС) систем критичного призначення (СКП) стосовно їх несуперечності.

Процес проєктування опрацьовано у якості складового етапу процесу розроблення. При цьому процес розроблення подано послідовністю наступних етапів: аналіз вимог, проєктування, реалізація, валідація.

Актуальність теми. Сучасне суспільство у чисельних сферах своєї діяльності істотним чином покладається на функціонування СКП (Safety-critical System), що також відомі як критичні системи або системи критичного застосування – такі системи, до функційної безпечності (ФБ) яких висуваються підвищені вимоги, системи, збої і відмови в роботі яких можуть призвести до критичних наслідків – призвести до значних небажаних наслідків критичного характеру [1].

Показовими прикладами названих систем можуть слугувати, у тому числі, системи керування орієнтацією космічного апарату (КА) [2], системи керування, що функціонують у сфері атомної енергетики, залучені для вирішення задач оборонної, хімічної промисловостей, авіації, медицини тощо [3]. З урахуванням поточного рівня розвитку науки і техніки, у наш час зазначені системи можна розглядати як комп’ютерні – такі, що включають і програмну, і апаратну складові. При цьому у межах представленої дисертації опрацьовано саме програмний рівень, який адресовано як програмно-алгоритмічну складову (ПАС). Остання відіграє ключову роль, наприклад, у бортових комплексах керування КА [4]. Більше того, у відповідних тематичних науково-технічних публікаціях зазначається, що на сьогодні 80% функцій таких

систем реалізуються програмно [5]. Разом із цим наголошується, що поточний рівень складності названих систем актуалізує потребу застосування відповідних методів і засобів контролю ПАС на етапах процесу розроблення.

Згідно до вищезазначеного, представлена дисертація базується на концептуальній складовій міжнародного стандарту IEC 61508, у якому регламентовано аспекти забезпечення ФБ. Вони полягають у своєчасному виявленні та усуненні потенційно небезпечних умов у роботі електронних систем із програмною складовою, що можуть призвести до негативних наслідків значного масштабу [6].

Окрім наведеного вище, мають місце також і галузеві стандарти, серед яких – ISO 26262:2018, де для забезпечення заданого рівня ФБ при розробленні вбудованих систем рекомендується залучати формальні методи і засоби [7].

Більше того, у Наказі Державної інспекції ядерного регулювання України від 22.07.2015 № 140, із змінами, внесеними згідно з Наказом Державної інспекції ядерного регулювання № 508 від 25.11.2019, дається наступне визначення поняттю ФБ: «функційна безпечність – властивість системи (компонента) атомної станції, що полягає у здатності виконувати всі потрібні функції, важливі для безпеки, зберігати потрібні властивості та відповідати заданим характеристикам в усіх передбачених проєктом режимах й умовах експлуатації» [8].

Інший показовий приклад галузевого документу – стандарт Європейського комітету з електротехнічної стандартизації CENELEC – EN 50128, застосовуваний до програмної складової комп’ютерних систем, що функціонують у залізничній галузі. У межах стандарту пропагується системний підхід до розроблення СКП. При цьому у якості ключового вводиться в обіг і залучається поняття SSIL (Software Safety Integrity Level) – як показник рівня ФБ [9]. При цьому для досягнення заданого рівня ФБ у межах стандарту рекомендується застосовувати формальні методи і засоби.

Ще один приклад галузевого документу, де регламентовано аспекти забезпечення ФБ – стандарт DO-178C, валідний по відношенню до програмної складової вбудованих систем авіаційної галузі, де також рекомендовано у якості засобів контролю застосовувати формальні методи і засоби [10].

Як узагальнення до вищезазначеного, з урахуванням складності / комплексності ПАС, актуальності набуває залучення у процесі розроблення таких формальних методів і засобів, що є придатними до їх автоматизованого застосування. Даному критерію на сьогодні відповідають представники сімейства методів перевірки на моделі.

Зауваження:

– у межах дисертаційної роботи використовується саме конструкція ПАС, замість конструкції «програмна складова», аби поставити наголос на досліджуваному показнику функціональних характеристик (ФХ) розроблюваної системи, у якості якого опрацьовано несуперечність ПАС;

– у свою чергу, у якості досліджуваного показника супутніх нефункціональних характеристик (НФХ), опрацьовано часові витрати, обумовлені виконанням кроків згідно ПАС;

– у якості об’єкту дослідження обрано саме етап проєктування у складі етапів процесу розроблення ПАС, керуючись позицією своєчасності виявлення і усунення потреби доопрацювання подань прийнятих проєктних рішень. Такі подання у межах дисертації опрацьовано згідно контекстного навантаження поняття «артефакт». Виконаний крок узгоджується із загальною позицією в основі охоплених вище стандартів – залучати формальні методи і засоби на кожному із етапів процесу розроблення. Дана рекомендація також фігурує у працях доктора технічних наук, професора, заслуженого винахідника України, завідувача кафедри комп’ютерних систем і мереж Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут» – В’ячеслава Сергійовича Харченка, доктора технічних наук,

професора, професора кафедри програмного забезпечення автоматизованих систем Національного аерокосмічного університету ім. М.Є. Жуковського «Харківський авіаційний інститут» – Бориса Михайловича Конорева [2, 11]. У якості предметної області стосовно сценаріїв залучення СКП при цьому фігурує аерокосмічна галузь – у розрізі інваріантно-орієнтованого оцінювання якості програмного забезпечення відповідних СКП, із застосуванням відповідних метрик оцінювання ризиків виникнення небажаних з позиції ФБ подій.

Серед робіт інших визначних вітчизняних діячів за окресленою тематикою доречним вбачається виокремити, у тому числі, праці доктора технічних наук, професора, професора FH JOANNEUM (University of Applied Sciences, Austria) Межуєва Віталія Івановича, чиї результати науково-дослідної діяльності у сфері формальних методів і засобів було успішно застосовано для контролю несуперечності артефактів розроблюваної операційної системи реального часу OpenComRTOS, призначеної до використання в аерокосмічній галузі.

Значних схвальних відгуків міжнародної науково-дослідницької спільноти, серед якої – і колектив компанії Motorola, здобули праці доктора фізико-математичних наук, старшого наукового співробітника Інституту кібернетики ім. В.М. Глушкова НАН України Олександра Олександровича Летичевського, присвячені дослідженню, розробленню і розвитку методів і засобів символьної формальної верифікації [12].

Серед численних праць закордонних діячів за окресленою тематикою варто виокремити здобутки Едмунда Кларка (Edmund Clarke), Алана Емерсона (Ernest Allen Emerson), Джозефа Сіфакіса (Joseph Sifakis) – лауреатів премії Тюрінга 2007 р. за вагомий внесок у розвинення техніки перевірки на моделі (Model Checking) до рівня ефективної технології формальної верифікації (ФВ), яка у наш час є широко застосовуваною у процесі розроблення програмних, апаратних, комп’ютерних систем, у тому числі СКП.

Не менш вагомим є науковий внесок лауреата премії Тюрінга 2013 р., першого лауреата премії Дейкстри – Леслі Лемпорта (Leslie Lamport), чиї праці здобули світове визнання. Відомі формальні методи і засоби: темпоральна логіка дій TLA (Temporal Logic of Actions), відповідний формальний метод перевірки на моделі TLC (TLA Checker), виразні засоби LaTeX, PlusCal, TLA+ тощо.

Статусу фундаментальних набули і праці Дорона Пеледа (Doron A. Peled), Орни Грумберг (Orna Grumberg).

У межах представленої дисертації метод TLC, а також засоби TLA, TLA+ і PlusCal, залучено у якості інструментів здійснення контролю несуперечності ПАС як досліджуваного показника ФХ розроблюваної ПАС. При цьому у якості показника НФХ охоплено часові витрати, супутні реалізації ПАС. Такий підхід застосовано до вирішення науково-технічної проблеми забезпечення контролю артефактів процесу розроблення ПАС систем критичного призначення стосовно їх несуперечності. Проблему опрацьовано у розрізі сприяння ФБ розроблюваної ПАС.

Роботу представлено із використанням поняття «артефакт». Згідно визначення компанії IBM, артефактом є результат виконання певного кроку процесу розроблення, поданий у формі файлу, що зберігається у пам’яті обчислювальної системи. Доповненням виступає також визначення професора Мюнхенського технічного університету Манфреда Броя (Manfred Broy): артефакт – сутність, яка характеризується структурою і змістом [13, с. 3]. Останню інтерпретацію у межах дисертації було розширено: артефакт – сутність, що характеризується архітектурою (структурою та зв’язками) і змістом. При цьому у якості досліджуваних артефактів опрацьовано графічні і формалізовані подання ПАС, що фігурують на етапі проєктування ПАС. Серед них: UML-діаграми дій, станів, формальні специфікації (ФС) і програмні реалізації як засоби автоматизації.

З урахуванням вищезазначеного, обрану тему дисертаційного дослідження можна вважати актуальною.

Зв'язок роботи з науковими програмами, планами, темами. Дисертаційні дослідження проведено протягом 2015−2023 рр. Роботу виконано, у тому числі, у відповідності до планів і поставлених до вирішення задач наступних науково-дослідних робіт: НДДКР № 0120U102683 «Розроблення спеціалізованих комп'ютерних технологій моделювання та опрацювання оперативної інформації в задачах енергетики» (2020−2024 рр.; науковий керівник); НДДКР № 0121U110615 «Розроблення методів та засобів верифікації артефактів процесу проєктування систем критичного призначення» (2021−2022 рр.; науковий керівник), виконуваних / виконаних в Інституті проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України. При цьому НДДКР № 0121U110615 було профінансовано у межах гранту НАН України дослідницьким лабораторіям/групам молодих вчених НАН України для проведення досліджень за пріоритетними напрямами розвитку науки і техніки (2021−2022 рр.). Окрім зазначеного, отримані результати проведених досліджень також було залучено при вирішенні задач міжнародного проєкту Erasmus+ Internet of Things: Emerging Curriculum for Industry and Human Applications ALIOT Project (reference number: 573818-EPP-1-2016-1-UK-EPPKA2-CBHE-JP (2016−2019 рр.; виконавець).

Мета і задачі дослідження. Метою дисертаційного дослідження є підвищення ефективності контролю артефактів у процесі розроблення ПАС систем критичного призначення на етапі проєктування для забезпечення їх несуперечності та зниження супутніх витрат.

Для досягнення сформульованої мети у дисертаційній роботі ставляться і вирішуються наступні задачі:

1. Проведення аналізу аспектів прикладного застосування методів і засобів контролю показників ФХ і НФХ розроблюваної ПАС системи

критичного призначення, з використанням методів формальної верифікації, у тому числі методів перевірки на моделі, моделей, підходів, інструментів і виразних засобів темпоральних логік. За результатами проведеного аналізу – розроблення комплексного підходу до застосування формальних методів і супутніх засобів контролю показників ФХ і НФХ розроблюваної ПАС на етапі проєктування.

2. Розроблення моделі подання ПАС у формі формальної специфікації (ФС). Модель призначена слугувати прототипом – засобом уніфікації – ФС, які, у свою чергу, залучаються у якості вихідних конструкцій для застосування по відношенню до них формального методу перевірки на моделі як засобу контролю несуперечності ПАС в автоматизованому режимі.

3. Розроблення методу синтезу ФС для ПАС, що дасть змогу одержувати в автоматизованому режимі відповідні первинним артефактам – графічним поданням – результуючі формалізовані подання. Це уможливить застосування по відношенню до останніх методу перевірки на моделі – засобу автоматизованого контролю досліджуваних артефактів у розрізі несуперечності.

4. Розроблення методу контролю відповідності одержуваних результуючих ФС первинним графічним поданням ПАС, що дозволить поширювати висновки, сформульовані за результатами проведення формальної верифікації ФС, на відповідні графічні подання ПАС.

5. Дослідження шляхів підвищення ефективності поширеного методу формальної верифікації – TLC – стосовно зниження супутніх його застосуванню обчислювальних витрат при проведенні контролю несуперечності розроблюваної ПАС на основі відповідної ФС.

6. Розроблення моделі, призначеної слугувати засобом регламентування формалізованого подання ПАС, де охоплюватимуться, у тому числі, засоби подання НФХ-складової, що уможливить проведення контролю

значень заданих показників зазначеної складової вже на етапі проєктування процесу розроблення ПАС.

7. Розроблення методу автоматизованого контролю значення заданого показника НФХ розроблюваної ПАС на етапі проєктування. Метод має забезпечувати механізм накопичення значення показника шляхом проведення дискретно-подійного імітаційного моделювання.

Об'єкт дослідження – процес розроблення програмно-алгоритмічної складової комп'ютерних систем критичного призначення.

Предмет дослідження – формальні методи і супутні засоби, зокрема моделі, засоби автоматизації, призначені до застосування на етапі проєктування програмно-алгоритмічної складової систем критичного призначення – по відношенню до артефактів, одержуваних у процесі розроблення. У якості досліджуваних артефактів залучаються, у тому числі, графічні і формалізовані подання програмно-алгоритмічної складової. При цьому у якості досліджуваного показника функціональних характеристик опрацьовано несуперечність програмно-алгоритмічної складової, а у якості показника нефункціональних характеристик – супутні реалізації кроків зазначеної складової часові витрати.

Методи дослідження. Засоби, прийоми теоретико-множинного підходу – для формалізації аналітичної складової. Формальні методи перевірки на моделі – для вирішення задачі ФВ в автоматизованому режимі. Методи теорії графів, а саме – метод обходу вершин графу в ширину (Breadth-first Search, BFS) і метод обходу в глибину (Depth-first Search, DFS) – для дослідження і розвитку поширеного методу перевірки на моделі TLC, у тому числі – оцінювання обчислювальних і просторових витрат, супутніх вирішенню задачі ФВ. Математичний апарат структури Кріпке – для аналітичного подання системи переходів (СП), що будується при вирішенні задачі ФВ методом перевірки на моделі. Метод імітаційного дискретно-подійного моделювання – для реалізації

процесу ФВ в автоматизованому режимі, а також для агрегування значення заданого показника НФХ. Математичний апарат числення послідовних процесів, що взаємодіють Чарльза Гоара (C.A.R. Hoare) – для формалізації послідовностей станів, у яких перебуває СП у процесі ФВ; залучено також трійки і аксіоми Гоара, а саме – правила композиції, виведення (імплікації) і умовного оператора, у тому числі – для зменшення обсягу ФС. Математичний апарат в основі темпоральної логіки дій TLA – для подання ПАС у формі ФС. Математичний апарат модульного ієрархічного формалізму DEVS (Discrete Event System Specification) Бернарда Зейглера (Bernard P. Zeigler) – для формалізованого подання і накопичення результуючого значення заданого показника НФХ. Методи і прийоми математичної статистики – для опрацювання і узагальнення одержуваних експериментальних даних. Методи теорії паралельних обчислень – для дослідження впливу залучення мультипоточності на результуючі часові витрати, супутні застосуванню формального методу перевірки на моделі TLC. Методи теорії алгоритмів, у тому числі – теорії обчислювальної складності – для подання досліджуваних артефактів процесу проєктування ПАС, оцінювання просторових витрат, граничних випадків одержуваного корисного ефекту від проведеного розвитку методу TLC.

Наукова новизна отриманих результатів полягає у розроблених методах, моделях, розвитку вже існуючого методу, застосовуваних згідно запропонованого комплексного підходу до їх залучення при проєктуванні ПАС. Комплексність досягається за рахунок охоплення показників ФХ і НФХ.

Наукові результати, винесені на захист і отримані автором особисто:

1. Розроблено модель подання ПАС СКП у формі ФС, де правило композиції Гоара вперше застосовано з позиції стратифікованого підходу до подання ПАС у формі ФС, що, на відміну від альтернативних рішень, дозволяє скоротити кількість рядків псевдокоду результуючої ФС. Названа модель

призначена слугувати засобом уніфікації ФС як форм подання ПАС – для проведення автоматизованого контролю несуперечності останніх шляхом формальної верифікації методом перевірки на моделі. Модель побудовано шляхом виокремлення двох рівнів опрацювання розробником складових формалізованих подань як артефактів – аналітичного рівня і рівня реалізації. При цьому у якості виразних засобів було залучено, у тому числі, засоби структури Кріпке, числення CSP, алгоритмічну мову PlusCal, а також формалізм TLA+. Перші два перелічені виразні засоби призначені до застосування на рівні аналітичному, останні два засоби – на рівні реалізації.

2. Розроблено метод синтезу ФС на основі графічного подання ПАС СКП, де вперше комплексно охоплено і аналітичний рівень подання ФС, і рівень реалізації. Такий крок, на відміну від альтернативних рішень, забезпечує прозорий механізм подання складових ФС та зв’язків між ними як на аналітичному рівні, так і на рівні програмної реалізації. Для цього було залучено математичний апарат темпоральної логіки дій TLA, відповідний формалізм TLA+ та алгоритмічну мову PlusCal. Застосування виразних засобів PlusCal дозволяє задати прототип результуючої ФС, із наголосом на архітектурній складовій. Цей прототип залучається у якості артефакту, на основі якого одержується результуюча ФС, побудована на основі виразних засобів TLA+. Останні, у свою чергу, уможливлюють здійснення процесу формальної верифікації ФС в автоматизованому режимі.

3. Розроблено метод контролю відповідності одержуваної ФС первинному артефакту – графічному поданню ПАС, де вперше запропоновано уніфікований механізм співставлення сутностей аналітичного рівня подання досліджуваного артефакту із відповідними сутностями рівня реалізації – елементами у складі результуючої ФС. Це дозволило узагальнити процес контролю одержуваних похідних артефактів – ФС – на рівні їх архітектурної складової (структури та зав’язків): шляхом співставлення СП для аналітичного

рівня і рівня реалізації – за показниками кількостей станів СП і глибин обходу просторів станів СП.

4. Розроблено модель – стратифіковану архітектуру, де ієрархічний підхід вперше застосовано у якості інструменту досягнення архітектурної відповідності результуючої складеної імітаційної моделі вихідній ФС, несуперечність якої вже було підтверджено шляхом формальної верифікації ФС – методом перевірки на моделі. Ієрархічний підхід реалізовано на основі засобів математичного апарату DEVS: шляхом оперування конструкціями «атомарної» і «складеної» DEVS-моделей, які включають засоби подання досліджуваного показника НФХ.

5. Розроблено метод контролю значення досліджуваного показника НФХ, де вперше враховано можливість оперування і оціночними, і фактичними значеннями складових названого показника – вже на етапі проєктування ПАС. Метод реалізовано шляхом проведення дискретно-подійного імітаційного моделювання на основі засобів математичного апарату DEVS. Накопичення значення показника у процесі моделювання реалізовано на основі механізму обміну повідомленнями між компонентами результуючої складеної ієрархічної DEVS-моделі, що, у випадку оперування оціночними значеннями, дозволяє скоротити часові витрати, супутні застосуванню методу. У якості досліджуваного показника НФХ опрацьовано часові витрати, супутні реалізації ФХ згідно ПАС.

6. Набув подальшого розвитку поширений метод перевірки на моделі TLC. На відміну від базового методу, проведений розвиток базується на комбінуванні методів обходу в ширину (BFS) і в глибину (DFS) теорії графів при здійсненні обходу простору станів СП за ітеративного підходу до організації процесу ФВ: на початковій ітерації застосовується метод BFS, що дає змогу визначити глибину обходу простору станів СП; на наступних ітераціях застосовується метод DFS, що дозволяє скоротити результуючі часові

витрати, супутні ітеративному процесу ФВ, – за рахунок зниження обчислювального навантаження за DFS-обходів для заданої кількості змінних станів. Одержуваний при цьому корисний ефект залежить від кількості змінних станів, архітектурної складової ФС, кількості ітерацій процесу ФВ.

Практичне значення отриманих результатів. Розроблені моделі, методи, розвиток методу, сполучені згідно запропонованого підходу, забезпечують механізм здійснення комплексного контролю артефактів процесу розроблення ПАС системи критичного призначення вже на етапі проєктування, у тому числі:

1. Розроблений підхід, викладений у першому розділі, слугує засобом сполучення винесених на захист наукових результатів у формі комплексного рішення, що уможливлює здійснення контролю показників і ФХ, і НФХ вже на етапі проєктування процесу розроблення ПАС системи критичного призначення.

2. Розроблена модель подання ПАС системи критичного призначення у формі ФС є засобом уніфікації ФС, до яких в автоматизованому режимі застосовується ФВ методом перевірки на моделі. Залучення моделі дозволяє зменшити кількість рядків псевдокоду результуючої ФС.

3. Розроблений метод синтезу ФС, що базується на зазначеній вище моделі подання ПАС, є засобом автоматизації процесу постачання вихідних даних для методу ФВ – методу перевірки на моделі TLC, а також розробленого розвитку цього методу. Залучення розробленого методу на етапі проєктування процесу розроблення ПАС дозволить знизити вплив людського фактору у процесі одержання формалізованих подань досліджуваних артефактів.

4. Розроблений метод контролю відповідності ФС, одержуваних у результаті застосування зазначеного вище розробленого методу синтезу ФС, є засобом контролю архітектурної відповідності результуючих ФС первинним графічним поданням ПАС. Використання названого методу призначене сприяти

підвищенню рівня довіри розробників до похідних артефактів – ФС, побудованих на основі виразних засобів формалізму TLA+, які, у свою чергу, є конструкціями, до яких застосовується метод TLC, а також розроблений розвиток зазначеного методу.

5. Розроблений розвиток поширеного методу перевірки на моделі TLC дозволить знизити результуючі часові витрати, супутні процесу ФВ, за ітеративного підходу до організації процесу проєктування ПАС. Одержуваний при цьому корисний ефект залежить, у тому числі, від архітектурної складової ФС, кількості змінних, кількості ітерацій процесу ФВ. Названий ефект, зокрема, зростає із зменшенням кількості умовних переходів у ФС.

6. Розроблена модель як стратифікована архітектура призначена слугувати засобом уніфікації комп'ютерних моделей, одержуваних у відповідності до ФС, несуперечність яких вже було підтверджено формальним методом перевірки на моделі, згідно розробленого підходу. Модель включає, у тому числі, засоби подання НФХ-складової. За рахунок залучення математичного апарату DEVS, модель характеризується властивостями модульності та ієрархічності – чинниками, що сприяють структурованості результуючої програмної реалізації, а також спрощенню процесу внесення доопрацювань.

7. Розроблений метод контролю значення заданого показника НФХ є засобом уможливлення зазначеного контролю вже на етапі проєктування процесу розроблення ПАС – шляхом проведення імітаційного дискретно-подійного моделювання на основі комп'ютерних моделей, побудованих у відповідності до зазначеної вище стратифікованої моделі, із можливістю оперування і оціночними, і фактичними значеннями названого показника. При цьому корисний ефект від залучення саме оціночних значень проявляється у скороченні результуючих часових витрат, супутніх застосуванню методу.

Практичне значення отриманих і винесених на захист результатів проведених досліджень підтверджено документально – актом впровадження, листами підтримки від організацій та установ, листами підтвердження впровадження, серед яких: акт впровадження у робочий процес ТОВ «НВП «ХАРТРОН-ЮКОМ», що співпрацює з КБ «Південне»; лист підтримки від Громадської спілки «Міжнародна рада з великих електроенергетичних систем СІГРЕ в Україні»; лист підтримки від Державного підприємства «Державний науково-технічний центр з ядерної та радіаційної безпеки»; лист підтримки від Державного центру кіберзахисту Державної служби спеціального зв'язку та захисту інформації України (ДЦКЗ Держспецзв'язку); лист підтвердження впровадження у навчальний процес Навчально-наукового інституту енергозбереження та енергоменеджменту Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського»; лист підтвердження впровадження у навчальний процес кафедри комп'ютерних систем, мереж та кібербезпеки факультету інформаційних технологій Національного університету біоресурсів і природокористування України (НУБіП України).

Особистий внесок здобувача. Усі представлені та винесені на захист наукові та науково-технічні результати отримано автором самостійно. У працях, опублікованих у співавторстві, здобувачеві належать: [1] – розроблена модель подання ПАС у формі ФС; [2] – розроблений метод синтезу ФС, розроблена модель як стратифікованої архітектура, розроблений метод контролю значення досліджуваного показника НФХ; [3] – результати проведеного аналізу предметної області, розроблена модель подання ПАС у формі ФС, розроблений метод синтезу ФС, включаючи розроблений метод контролю відповідності у якості допоміжного засобу; [4] – результати проведеної класифікації формальних методів і засобів, елементи розробленого підходу у частині контролю досліджуваного показника ФХ, розроблена модель подання ПАС у

формі ФС, елементи розробленого методу синтезу ФС, результати експериментальних досліджень базового методу TLC; [5] – розроблена модель як стратифікована архітектура, розроблений метод контролю значення досліджуваного показника НФХ, результати аналізу предметно-орієнтованого сценарію енергетики; [6] – представлення і результати проведеного розвитку базового методу перевірки на моделі TLC; [7] – результати дослідження альтернативних реалізацій базового методу TLC для граничного випадку послідовного сценарію; [8] – результати експериментального дослідження розроблених моделі як стратифікованої архітектури і методу контролю значення досліджуваного показника НФХ; [9] – елементи розробленої моделі подання ПАС у формі ФС, на прикладі розподіленої комп'ютерної системи; [10] – елементи розробленої моделі подання ПАС у формі ФС, на прикладі сценарію контролю відповідності програмних складових компонентів розподілених комп'ютерних систем; [11] – запропонований підхід до контролю адекватності розробленої моделі подання ПАС у формі ФС, розроблений метод контролю відповідності результуючих ФС, предметно-орієнтовані результати досліджень; [12] – опрацювання часових витрат у якості досліджуваного показника НФХ; [13] – результати дослідження розробленого розвитку методу TLC на прикладі предметно-орієнтованого сценарію енергетики; [14] – запропонований підхід до організації процесу проєктування, елементи розробленої моделі як стратифікованої архітектури, результати проведених досліджень; [15] – опрацювання часових витрат як показника доцільності залучення компонентів розподілених систем; [16] – опрацювання часових витрат як показника НФХ стосовно сценаріїв потоків робіт; [17] – результати дослідження базового методу TLC для граничного випадку подання паралелізму згідно моделі чергування; [18] – елементи розробленого методу контролю досліджуваного показника НФХ, опрацювання у якості такого показника часових витрат; [19] – результати проведеного аналізу предметної області, у тому числі обґрунтування

доцільності застосування формальних методів і засобів на етапі проєктування; [20] – результати дослідження впливу конфігурації апаратної складової комп'ютерної системи на показники швидкодії обчислень, у тому числі у частині впливу від введення мультипоточності; [21] – елементи і результати дослідження розробленої моделі подання ПАС у формі ФС, розроблений метод синтезу ФС згідно зазначеної моделі; [22] – результати дослідження базового методу TLC за показниками просторових витрат; [23] – результати дослідження впливу від введення мультипоточності до реалізації методу TLC на часові витрати, супутні процесу формальної верифікації, опрацювання предметно-орієнтованого сценарію аерокосмічної галузі; [24] – дослідження часових витрат як показника НФХ вебсервісів; [25] – опрацювання часових витрат як показника НФХ стосовно UML-подань як артефактів; [26] – розроблений модельно-орієнтований підхід до контролю показників НФХ, елементи розроблених моделі як стратифікованої архітектури, розробленого методу контролю досліджуваного показника НФХ; [27] – розроблена модель як стратифікована архітектура; [28] – обґрунтування важливості проведення контролю показників і ФХ, і НФХ у процесі розроблення; [31] – залучення стратифікації стосовно формалізації архітектурної складової артефактів процесу розроблення програмно-конфігурованих мереж, елементи розробленої моделі подання ПАС у формі ФС; [32] – представлення і результати застосування розробленої моделі подання ПАС у формі ФС стосовно предметно-орієнтованого сценарію; [33] – застосування розроблених моделі подання ПАС, а також відповідного розробленого методу синтезу ФС по відношенню до артефактів як подань протоколів взаємодії компонентів розподілених комп'ютерних систем; [34] – розроблена модель подання ПАС у формі ФС; [35] – розроблений розвиток методу TLC, результати дослідження застосування проведеного розвитку методу для критичного сценарію аерокосмічної галузі; [39] – застосований підхід до подання паралелізму у ФС, результати оцінювання

просторових характеристик СП, що конструюються у процесі формальної верифікації; [42] – елементи розробленої моделі як стратифікованої архітектури; [44] – обґрунтування доцільності проведення стратифікації складових ФС; [45] – елементи застосованого підходу в основі розробленого методу контролю досліджуваного показника НФХ; [46] – результати проведеного аналізу методів та засобів формальної верифікації; [47] – обґрунтування важливості опрацювання несуперечності ПАС у якості досліджуваного показника ФХ на прикладі сценаріїв систем Інтернету речей; [48] – обґрунтування важливості застосування методів перевірки на моделі при проєктуванні СКП; [51] – запропонований підхід до застосування методу перевірки на моделі TLC, отримані результати проведених досліджень у частині обчислювальних витрат, супутніх процесу ФВ; [52] – отримані результати оцінювання просторової складності вирішуваної методом перевірки на моделі задачі ФВ; [53] – отримані результати оцінювання впливу від залучення мультипоточності на часові витрати, супутні реалізації процесу ФВ; [54] – обґрунтування важливості застосування методу перевірки на моделі при проєкутванні інформаційно-технологічних систем суб'єктів ринку електроенергії, отримані результати проведених експериментальних досліджень; [55] – модельно-орієнтований підхід до формалізації досліджуваного показника НФХ, покладений в основу розробленого методу контролю значення названого показника при проєктуванні ПАС, елементи розробленого комплексного підходу до контролю показників і ФХ, і НФХ; [56] – дуальний підхід, покладений в основу розроблених моделі подання ПАС у формі ФС та методу синтезу ФС; [57] – результати дослідження мультипоточної реалізації методу TLC; [58] – узагальнення засобів ФВ у частині відповідних програмних реалізацій; [59] – підхід, покладений в основу розробленого методу контролю показника НФХ; [60] – обґрунтування важливості проведення контролю показників і ФХ, і НФХ на етапі проєктування у складі етапів процесу розроблення; [61] – запропонований модельно-

орієнтований підхід як засіб сполучення артефактів виокремлених типів; [62] – узагальнення стосовно шляхів проведення контролю НФХ у контексті поняття ФБ; [63] – складові розробленого методу контролю значення показника НФХ; [64] – узагальнення стосовно аспектів контролю несуперечності ПАС у якості досліджуваного показника ФХ; [65] – висновки, узагальнення за результатами застосування методу TLC при вирішенні задач енергетики; [66] – обґрунтування і узагальнення стосовно вважливості опрацювання при проєктуванні ПАС у якості показників ФХ і НФХ, відповідно, несуперечності ПАС і супутніх реалізації кроків ПАС часових витрат; [67] – висвітлення предметної області енергетики з позиції поняття резилієнтності; [68] – узагальнення стосовно одержуваного корисного ефекту від мультипоточної реалізації методу TLC.

Відомості про апробацію результатів дисертації. Отримані результати проведених дисертаційних досліджень пройшли апробацію на наступних науково-технічних та науково-практичних конференціях і семінарах:

1. Науково-технічний семінар «Критичні комп’ютерні технології та системи» – КриКТехС-2024/1/186 (м. Харків, 2024 р.).
2. П’ята науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України «Безпека енергетики в епоху цифрової трансформації» (м. Київ, 2023 р.).
3. Міжнародна науково-практична конференція «живучість та резильєнтність – 2023» (Survivability & Resilience – 2023), (м. Київ, 2023 р.).
4. III Всеукраїнська науково-практична конференція пам’яті академіка Академії наук вищої освіти, професора Анатолія Володимировича Касперського «Актуальні проблеми та перспективи розвитку фундаментальних, прикладних, загальнотехнічних та безпекових наук» (м. Київ, 2023 р.).
5. XLI Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України (м. Київ, 2023 р.).

6. Sixth International Workshop on Computer Modeling and Intelligent Systems, CMIS-2023 (Zaporizhzhia, 2023).

7. Міжнародна науково-практична конференція «Продовольча та екологічна безпека в умовах війни та повоєнної відбудови: виклики для України і світу», присвячена 125-річчю заснування Національного університету біоресурсів і природокористування України (м. Київ, 2023 р.).

8. XLIX Міжнародна науково-практична інтернет-конференція «Проблеми та перспективи розвитку сучасної науки в країнах Європи та Азії» (м. Переяслав, 2022 р.).

9. XL Науково-технічна конференцію молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України (м. Київ, 2022 р.).

10. Науково-практична конференція «Тиждень науки-2022» (м. Запоріжжя, 2022 р.).

11. Третя науково-практична конференція «Безпека енергетики в епоху цифрової трансформації» (м. Київ, 2021 р.).

12. XII Міжнародна науково-практична конференція молодих вчених «Інформаційні технології: економіка, техніка, освіта ‘2021» (м. Київ, 2021 р.).

13. 2021 IEEE KhPI Week on Advanced Technology (м. Харків, 2021 р.).

14. 2nd International scientific and practical conference on Topical issues of modern science, society and education – SPC «Sci-conf.com.ua» (м. Харків, 2021 р.).

15. IX Міжнародна науково-практична конференція «European scientific discussions» (м. Рим, Італія, 2021 р.).

16. IX Міжнародна науково-практична Інтернет конференція «Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні 2021» (м. Київ, 2021 р.).

17. XXXIX Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, присвячена 40-річчю Інституту, Дню науки в Україні та з нагоди відзначення 30-ї річниці незалежності України (м. Київ, 2021 р.).

18. Друга науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України «Безпека енергетики в епоху цифрової трансформації» (м. Київ, 2020 р.).

19. X Ювілейна міжнародна науково-практична конференція «Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій», присвячена 120-річчю з дня заснування Національного університету «Запорізька політехніка» (м. Запоріжжя, 2020 р.).

20. VIII Міжнародна науково-практична Інтернет-конференція «Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2020» (м. Київ, 2020 р.).

21. 11th International Conference on Dependable Systems, Services and Technologies, DESSERT'2020 (м. Київ, 2020 р.; доповідь відзначено сертифікатом за кращу доповідь).

22. Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України (м. Київ, 2020 р.).

23. 2019 International Scientific-Practical Conference on Problems of Infocommunications. Science and Technology, PIC S&T`2019 (м. Київ, 2019 р.).

24. Науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України «Безпека енергетики в епоху цифрової трансформації» (м. Київ, 2019 р.).

25. VII Міжнародна науково-практична конференція «Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2019» (м. Київ, 2019 р.).

26. 2018 IEEE International Scientific and Practical Conference on Problems of Infocommunications. Science and Technology, PIC S&T`2018 (м. Харків, 2018 р.).

27. IX Міжнародна науково-практична конференція «Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій» (м. Запоріжжя, 2018 р.).

28. VI Міжнародна наукова конференція «Моделювання-2018», приурочена до 100-річчя від дня утворення Національної академії наук України та річниці з Дня народження академіка НАН України Пухова Георгія Євгеновича (м. Київ, 2018 р.).

29. VI Міжнародна науково-практична інтернет-конференція «Тенденції та вектор розвитку науки в сучасному світі» (м. Дніпро, 2018 р.).

30. Науково-практична конференція «Тиждень науки-2018» (м. Запоріжжя, 2018 р.).

31. 14th Int. Conf. on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET'2018 (с. Славське, 2018 р.).

32. Int. research and practice conference on Modern methods, innovations, and experience of practical application in the field of technical sciences (Radom, Republic of Poland, 2017).

33. Науково-практична конференція «Тиждень науки 2017» (м. Запоріжжя, 2017 р.).

34. Tenth International Scientific-Practical Conference «Internet-education-science-2016», IES-2016 (м. Вінниця, 2016 р.).

35. VIIIth Int. scientific-practical conf. on Modern problems and achievements of radio engineering (electronics), telecommunications and information technology (м. Запоріжжя, 2016 р.).

36. 27th Int. Central European Conference on Information and Intelligent Systems, CECIIS 2016 (Varazdin, Croatia, 2016).

37. Науково-практична конференція «Тиждень науки 2016» (м. Запоріжжя, 2016 р.).

38. XIIIth Int. Conf. on Modern Problems of Radio Engineering, Telecommunications, and Computer Science, TCSET'2016 (с. Славське, 2016 р.).

39. Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України (м. Київ, 2016 р.).

40. XXXIV науково-технічна конференція «Моделювання» (м. Київ, 2015 р.).

Публікації. За темою дисертаційної роботи опубліковано 68 наукових праць, з яких: 4 – колективні монографії, 2 – розділи колективних монографій у закордонних виданнях, у тому числі 1 індексується у міжнародній наукометричній базі Scopus; 22 – статті у періодичних фахових виданнях, з яких 7 – у виданнях, що індексуються у міжнародних наукометричних базах Scopus та Web of Science Core Collection, 3 – у фахових виданнях категорії А; 40 – матеріали доповідей на наукових конференціях, з яких 7 – індексуються у міжнародних наукометричних базах Scopus та Web of Science Core Collection.

Структура та обсяг роботи. Дисертаційна робота складається з анотації, вступу, шести розділів, висновків, списку використаних джерел, додатків.

У розділі 1 зведено отримані результати проведеного аналізу предметної області, за результатами якого сформульовано засади для розроблення комплексного підходу – інструменту сполучення засобів контролю ФХ і НФХ при проєктуванні ПАС, викладені у наступному розділі.

У розділі 2 подано розроблений підхід до контролю несуперечності ПАС, а також розроблену модель подання ПАС у формі ФС. Підхід при цьому

опрацьовано у якості засобу сполучення винесених на захист наукових здобутків.

У розділі 3 викладено розроблений метод синтезу ФС на основі первинних артефактів – подань ПАС у формі блок-схем алгоритмів, UML-діаграм дій. Метод побудовано на основі викладеної у попередньому розділі моделі.

Розділ 4 містить опис проведеного розвитку поширеного формального методу перевірки на моделі TLC у напрямі зниження супутніх його застосуванню часових витрат за ітераційного підходу до організації процесу формальної верифікації ФС при проєктуванні ПАС.

У розділі 5 викладено розроблену модель представлення ПАС у формі ФС як засіб стратифікованого подання архітектурної складової, призначену слугувати засобом уможливлення контролю значень досліджуваного показника НФХ при проєктуванні ПАС.

Розділ 6 присвячено опису розробленого методу контролю значень досліджуваного показника НФХ при проєктуванні ПАС. Метод побудовано на основі викладеної у попередньому розділі розробленої моделі.

У додатках зведено фрагменти програмних реалізацій залучених засобів автоматизації, досліджувані артефакти, копії документів, що підтверджують практичне значення та впровадження отриманих результатів, а також список опублікованих за темою дисертації праць.

Загальний обсяг дисертаційної роботи становить 369 сторінок, з яких: 255 сторінок основного тексту, список із 189 використаних джерел на 33 сторінках та 9 додатків на 48 сторінках. Робота містить 27 таблиць та 38 рисунків.

РОЗДІЛ 1
АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ КОНТРОЛЮ АРТЕФАКТІВ ПРОЦЕСУ ПРОЄКТУВАННЯ

У розділі подаються результати проведеного аналізу предметної області – методів і засобів контролю артефактів, застосовуваних у процесі розроблення ПАС СКП. Опрацьовується при цьому етап проєктування у складі етапів процесу розроблення ПАС. Здійснюється обґрунтування доцільності проведення контролю одержуваних при порєктуванні артефактів – подань ПАС у формі блок-схем алгоритмів, UML-діаграм дій, станів – вже на етапі проєктування процесу розроблення. Наголошується на важливості залучення у якості засобів контролю формальних методів і засобів, у тому числі – представників сімейства методів перевірки на моделі. У якості їх ключової переваги у порівнянні із альтернативними методами і засобами, акцент ставиться на придатності до автоматизованого застосування.

Формулюються засади стосовно доцільності розроблення комплексного підходу як інструменту сполучення методів і засобів контролю артефактів процесу проєктування ПАС. Комплексність полягає в охопленні показників ФХ і НФХ. У свою чергу, у якості досліджуваного показника ФХ опрацьовується несуперечність ПАС, а у якості показника НФХ – часові витрати, супутні реалізації ФХ згідно ПАС. При цьому варто зауважити, що у якості альтернативного показника НФХ можуть залучатися, наприклад, вартісні / матеріальні витрати, супутні реалізації ФХ згідно ПАС.

1.1 Обгрунтування актуальності контролю артефактів проєктування

Дисертаційне дослідження базується, зокрема, на працях видатного винахідника України, доктора технічних наук, професора В’ячеслава Сергійовича Харченка, доктора технічних наук, професора Бориса Михайловича Конорева [1, 11]. У цих працях наголошується на зростанні актуальності проблеми зниження ризиків відмов і аварій для систем, критичних до безпеки – систем критичного призначення (СКП). Показовими прикладами є системи, призначені до застосування у космічній галузі. Для вирішення цієї проблеми авторами, зокрема, пропонується інваріантно-орієнтована модель оцінювання якості програмного забезпечення у складі СКП, що будується на застосуванні методів перевірки на моделі [2].

СКП знаходять прикладне застосування у різноманітних предметних областях. Окрім космічної галузі, це і системи управління рухом залізничного транспорту, і системи керування пристроями, призначеними для вирішення завдань енергетики тощо. Загалом, СКП являють собою комплекси програмно-апаратних засобів, призначених для вирішення спеціалізованих задач із жорсткими обмеженнями за визначеними показниками: час, вірогідність виникнення помилки/відмови, нештатної ситуації тощо. Більше того, такі системи є представниками сімейства реагуючих систем – функціонування яких напрямлене на підтримку взаємодії із навколишнім середовищем [14]. Характерною особливістю названих систем є те, що, у процесі їх функціонування, число станів, у яких може знаходитися система, прямує до безкінечності. Це створює додаткові складності при дослідженні реагуючих систем.

Названі системи характеризуються високим рівнем складності. Це зумовлює потребу створення і використання ефективних підходів до

проєктування зокрема і розроблення у цілому таких систем – підходів, використання яких сприяло би зменшенню кількості та критичності помилок проєктних рішень – помилок, що потенційно можуть призвести до критичних наслідків. Відповіддю на це питання слугує, зокрема, використання модельно-орієнтованого підходу (MBSA, Model-based Safety Analysis) до аналізу ПАС СКП [15]. Шляхом реалізації такого підходу є використання формальних методів та засобів при проєктуванні. Це пояснюється зростанням складності названих систем і потребою виявлення помилок в проєктних рішеннях вже на етапі проєктування, а не на заключному етапі тестування процесу розроблення, коли вартість усунення виявлених помилок буде істотно вищою – як з позиції супутніх матеріальних витрат, так і з позиції часових витрат. При цьому зазначається, що безпечність інформаційно-керуючих СКП суттєвим чином залежить від якості програмного забезпечення, за допомогою, якого реалізуються критичні функції.

Стверджується, що процес усунення помилок прийнятих проєктних рішень на пізніх етапах життєвого циклу СКП – на етапах валідації, експлуатації системи – супроводжується істотно більшими матеріальними і часовими витратами [16, с. 15]. Більше того, тестування як шлях проведення валідації не гарантує відсутності помилок, а лише виявляє деякі з них – за умови використання належним чином підібраних тестових послідовностей. У контексті СКП така специфіка не є задовільною, оскільки невиявлені шляхом тестування помилки можуть потенційно призвести до критичних наслідків при експлуатації системи.

Ґрунтуючись на накопиченій статистиці, стверджується, що кожна п’ята аварія у космічній галузі пов’язана з відмовою комп’ютерних систем управління і їх компонентів [11]. У цьому контексті критичні з позиції ФБ помилки, проявом яких є порушення ФХ за показником несуперечності ПАС, доцільно виявляти і усувати вже на ранніх етапах процесу розроблення ПАС, у тому

числі – на етапі проєктування. Окрім зазначеного, проведення тестування не є гарантією виявлення чи підтвердження відсутності помилок. Натомість воно є інструментом виявлення певного класу помилок, успішність якого визначається, у тому числі, досвідом розробників, а також застосовуваним підходом. Більше того, у контексті ФБ названий підхід є таким, що надає недостатні підтвердження стосовно несуперечності ПАС. Компенсувати цю недостатність, узгоджуючись із положеннями охоплених вище міжнародних стандартів дозволить залучення формальних методів і засобів – у якості доповнюючих засобів.

У складі формальних методів і засобів виокремлюють, у тому числі, наступні сімейства: сімейство методів і засобів дедуктивної верифікації; сімейство методів і засобів перевірки на моделі (Model Checking, MC) [17]. Істотною перевагою при цьому характеризується саме сімейство методів перевірки на моделі – з позиції їх придатності до автоматизованого застосування. Відмінною рисою відповідних представників є їх високий ступінь придатності до автоматизованого застосування, який, з урахуванням експоненційного характеру зростання простору станів СП, можна вважати одним з визначальних при виборі того чи іншого засобу. При цьому має місце і інша характерна риса: судження стосовно несуперечності ПАС виноситься не на основі відповідного первинного артефакту-подання – блок-схеми алгоритму, UML-діаграми дій / станів, – а на основі похідного від нього артефакту – формалізованого подання (ФС) [18], що, власне, уможливлює автоматизацію процесу контролю, здійснюваного шляхом ФВ. У свою чергу, окреслена ситуація обумовлює потребу розроблення і залучення додаткових методів і засобів, які б дозволили отримувати відповіді на питання стосовно відповідності похідних артефактів – формалізованих подань – первинним артефактам. Позитивна відповідь у даному контексті дозволить поширювати зроблені висновки стосовно одержуваних результатів проведення формальної

верифікації ФС також і по відношенню до відповідних первинних артефактів. Це, у свою чергу, створить підстави для стверджувальних відповідей стосовно достовірності результатів ФВ, одержуваних на основі залученого методу перевірки на моделі.

Зауваження: у третьому розділі дисертації у якості допоміжного засобу викладено розроблений метод контролю відповідності результуючих ФС.

Змістове навантаження залученого понятійного апарату зведено на рис. 1.1.

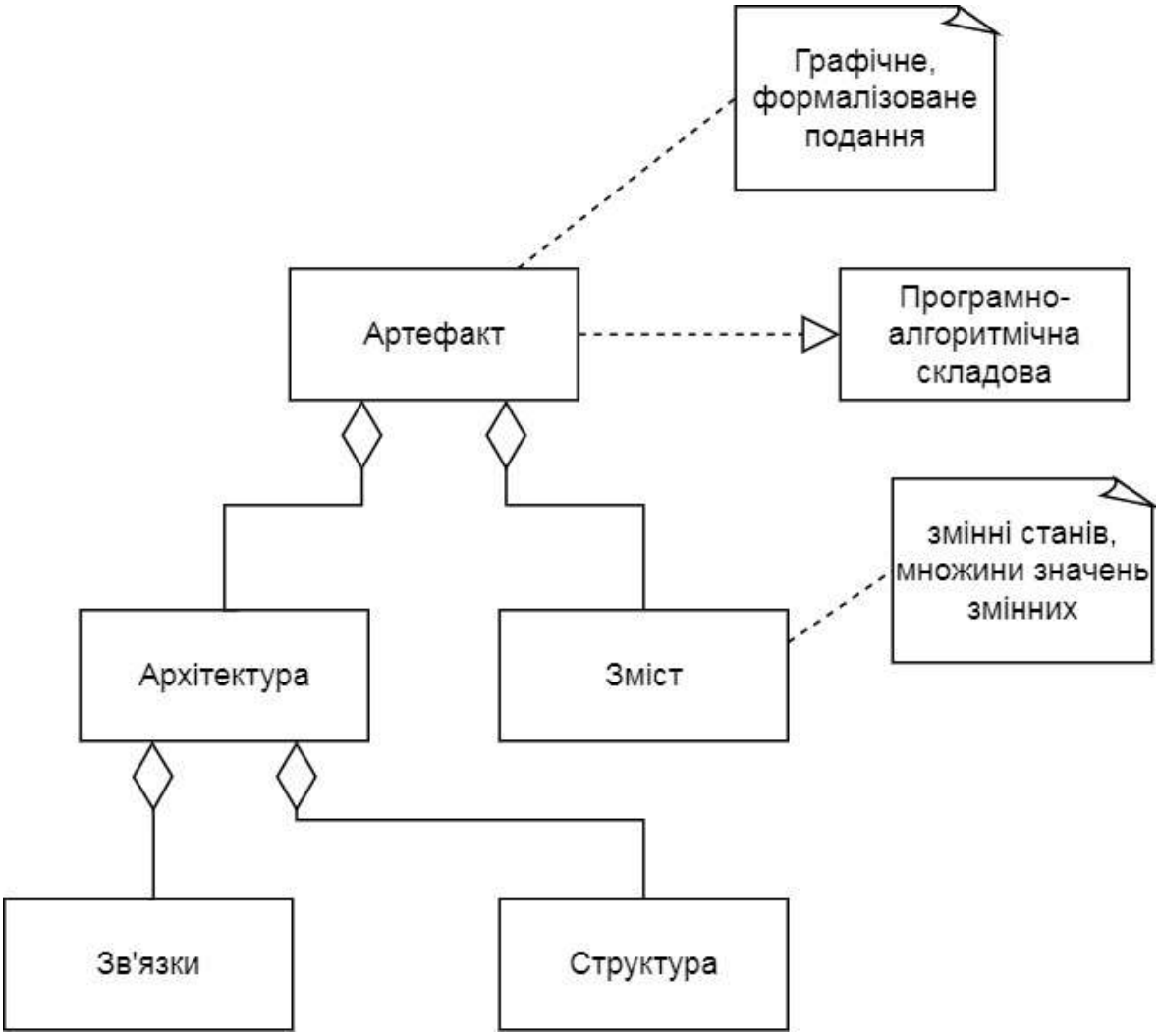

Рисунок 1.1 – Подання у формі UML-діаграми змістового навантаження залученого понятійного апарату

На рис. 1.1 залучено саме відношення агрегування, аби акцентувати увагу на допустимості оперування відповідними складовими як опосередковано, так і у складі альтернативних конструкцій.

Повертаючись до формальних методів і засобів, доречно зауважити, що методом перевірки на моделі (методом перевірки моделі), за визначенням лауреата премії Тюрінга – Едмунда Кларка (E. M. Clarke), є «автоматичний метод верифікацій систем з кінцевою кількістю станів» [19]. Названі системи або вимоги до них при цьому подаються у формі ФС і моделюються як СП. Задачею ФВ при цьому є встановлення істинності темпоральної формули, заданої у ФС, для кожного із станів СП. Процес ФВ при цьому полягає, у тому числі, в обході простору станів СП. У свою чергу, застосування саме МС-методу у якості засобу реалізації MBSA-підходу супроводжується вагомою перевагою – істотно вищим рівнем розвитку супутніх засобів автоматизації, результатом чого стало широке застосування МС-методів згідно критичних сценаріїв прикладного спрямування [18].

Демонстративними прикладами успішного застосування методів перевірки на моделі при проєктуванні ПАС у складі СКП є система TAS Control Platform, призначена слугувати основою програмної системи керування рухом залізничного транспорту [20]. За рахунок застосування названих методів авторам вдалося досягти значення показника ФБ системи на рівні SIL 4 (Safety Integrity Level) [6], що полягає у забезпеченні інтенсивності відмов на рівні $10^{-8} - 10^{-9}$ 1/год.

У сфері атомної енергетики виокремлюється досвід Фінляндії: засвідчено, що у період з 2008 по 2020 рр. шляхом застосування формальних методів і засобів було виявлено 66 підтверджених помилок у артефактах проєктування – при вирішенні задач I&C (Instrumentation and Control) [21]. У свою чергу, по причині нестачі висвітлення аспектів такого застосування, серед відкритих до подальшого розвитку лишається, у тому числі, позиція досягнення балансу між

одержуваним корисним ефектом від верифікації, супутніми цьому процесу часовими витратами і доступними обчислювальними ресурсами. У даному контексті попередні експерименти показали негативний вплив надмірної деталізації ФС, де паралелізм було подано згідно моделі чергування [22]. Результатом стала нестача доступної обчислювальній системі оперативної пам’яті (ОП) і, як результат, передчасне припинення процесу автоматизованої ФВ.

Показовим прикладом результативності залучення методів сімейства МС є, у тому числі, системи керування, застосовувані в аерокосмічній галузі [19]. Стверджується також стосовно успішності використання методу TLC і виразного засобу TLA+ у якості засобів контролю несуперечності ПАС, поданої у формі артефактів процесу розроблення системи керування автономними програмно-апаратними комплексами [23].

1.2 Підходи, методи та засоби контролю

Для деталізації предмету дослідження адресуємо стандарт ДСТУ ISO 9000:2015 «Системи управління якістю: основні положення та словник термінів», де поняття «верифікація» визначається наступним чином: «Підтвердження наданням об’єктивних доказів, що встановлені вимоги виконано» [24]. У свою чергу, у контексті дисертаційної роботи оперуватимемо поданим вище поняттям ФВ. Застосовуваний при цьому підхід полягає у здійсненні ФВ шляхом залучення методу МС і супутніх засобів – моделей, засобів автоматизації.

Для позиціонування поняття ФВ в ієрархії суміжних понять залучимо стандарт IEEE 1012-2016 [25], де верифікація фігурує у якості складової

комплексного процесу V&V (Verification and Validation), де охоплено вищезазначені етапи процесу розроблення. Верифікація при цьому визначається як засіб встановлення відповідності одержуваних артефактів процесу розроблення, зокрема на етапі проєктування, специфікаціям вимог до системи. У свою чергу, шляхом валідації встановлюється придатність вже реалізованої системи до призначеного застосування. Запропоноване ієрархічне подання названих і супутніх понять, із зазначенням зв'язків між ними, наведено на рис. 1.2.

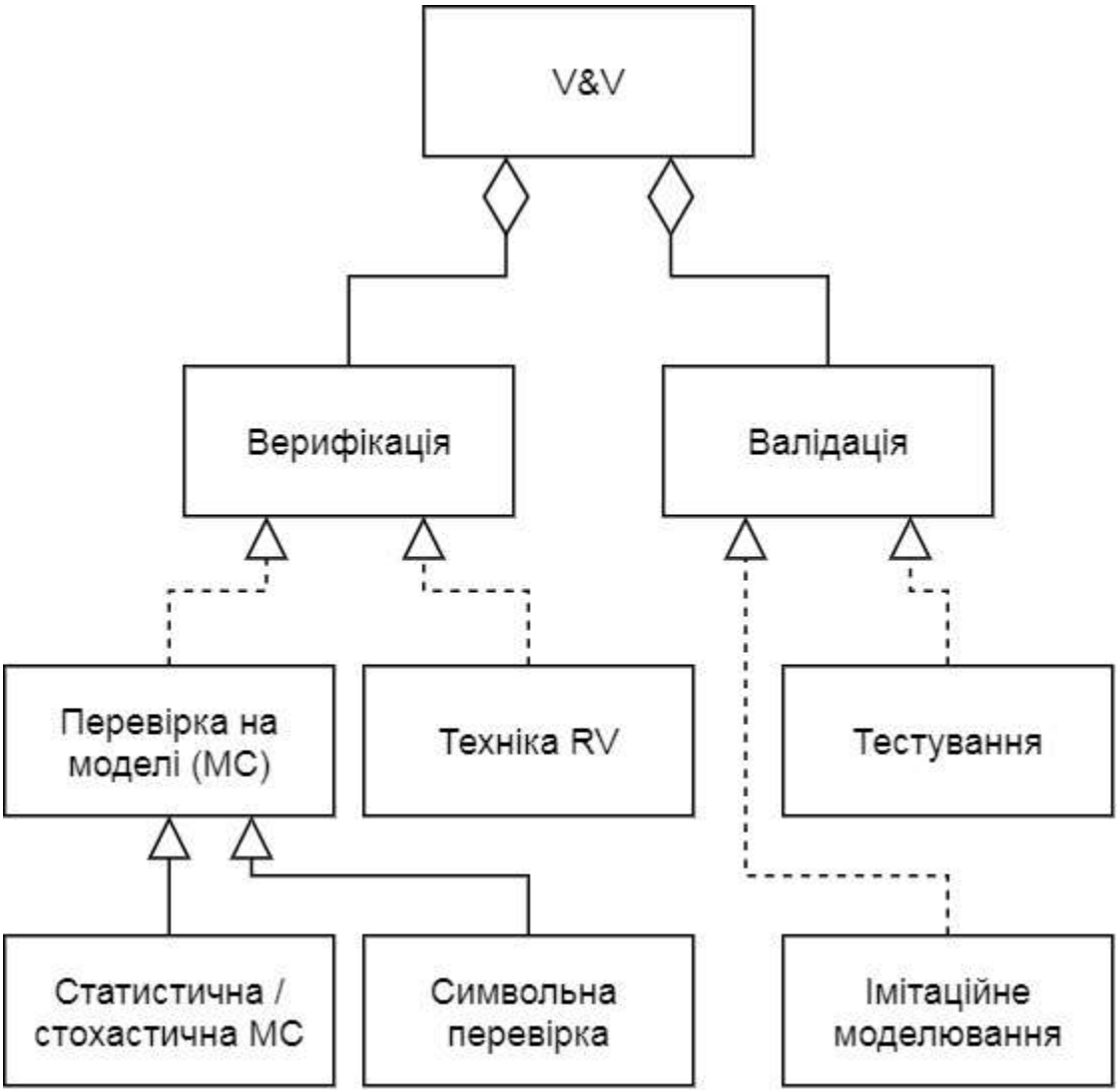

Рисунок 1.2 – Ієрархічне подання понятійної складової процесу V&V

На рис. 1.2 ромбовидною стрілкою зазначається відношення агрегування, трикутною пунктирною – відношення реалізації, трикутною суцільною – відношення розширення:

– поняття верифікації і валідації є основоположними формуючими сутностями охопленого понятійного апарату у межах процесу V&V;

– у свою чергу, верифікація може бути реалізована, у тому числі, як шляхом залучення формальних методів і засобів (МС), так і згідно техніки RV (Runtime Verification) – контроль заданого показника здійснюється у режимі виконання – відповідна досліджувана система вже має бути реалізованою програмно [26]. Істотна перевага техніки RV, у порівнянні із залученням методу МС, – не постає проблема експоненційного зростання простору станів СП. У даному контексті, однак, зазначається важливість досягнення балансу між обсягом формалізованих описів і супутніми процесу RV обчислювальними витратами [27, с. 143]. З урахуванням названого, техніка RV застосовується, у тому числі, для вирішення проблеми «архітектурної ерозії» (Architectural Erosion), що проявляється у порушенні прийнятих архітектурних рішень у процесі поступального внесення доопрацювань [28]. Для цього залучається, зокрема, виразний засіб FACTum [29]. Аналогічним чином вирішується суміжна проблема – архітектурної невідповідності (Architectural Mismatch), що адресується з позиції сумісності систем, підсистем/компонентів [30]. Це, у свою чергу, доповнює озвучену вище концепцію Бертрана Мейєра на рівні архітектурної складової;

– згідно стандарту IEEE 1012-2016, валідація може бути проведена, зокрема, шляхами імітаційного моделювання і тестування, які, у свою чергу, можна розглядати як взаємодоповнюючі.

Як доповнення до вищезазначеного, варто зауважити, що у контексті ФБ розроблюваної системи оперують, у тому числі, поняттям «помилки», яке

розглядається у якості чинника, що може призвести до збоїв і відмов у роботі системи – чинника, проявом якого буде поведінка системи, що не відповідає вимогам, зазначеним у специфікації [31, с. 9]. Такою інтерпретацією, однак, не ставиться під сумнів відсутність помилок у специфікації безпосередньо. Зазначений аспект, у свою чергу, опрацьовується у межах представленої дисертаційної роботи. Більше того, одержуваний рівень ФБ результуючої системи можна вважати наслідком усунення помилок, що мали місце на етапі проєктування процесу розроблення [31, с. 15].

1.2.1 Специфіка методів перевірки на моделі

На відміну від RV, перевірка на моделі виконується шляхом побудови СП, і обходу відповідних станів. Вона полягає у перевірці істинності наступного твердження [18]:

$$M, b \models \psi, \tag{1.1}$$

де M – структура Кріпке – математична модель як засіб подання СП із кінцевим числом станів; b – поведінка системи – послідовність станів, формалізована засобами структури M; ψ – темпоральна формула, істинність якої перевіряється для кожного із елементів b; $\models$ – оператор виконуваності (формули ψ для кожного із елементів b).

У свою чергу, призначення ФС – подання ψ на основі виразних засобів обраного формалізму. Це уможливить автоматизацію процесу ФВ ФС. При цьому ФС адресується у межах дисертаційної роботи як похідний артефакт – від первинних артефактів – графічних подань – блок-схем алгоритмів, UML-діаграм дій.

Для формалізації СП залучаємо структуру Кріпке над кінцевою множиною атомарних висловлювань AP [18]:

$$M = \langle S, S_0, R, L \rangle, \tag{1.2}$$

де S – кінцева множина станів; $S_0 \subseteq S$ – непуста множина початкових станів СП; $R \subseteq S^2$ – тотальна множина переходів: $\forall s \in S \; \exists s' \in S : (s, s') \in R$; $L : S \to 2^{AP}$ – функція розмітки станів СП елементами множини AP, що приймають істинні значення у відповідних станах.

У якості засобів подання ФС було розглянуто, у тому числі, нижченаведені темпоральні логіки:

– логіка лінійного часу (LTL, Linear Temporal Logic), коли для деякого поточного стану $s \in S$ має місце лише один наступний стан $s' = R(s) \in S$ [32];

– логіка дерев обчислень (CTL, Computation Tree Logic), де альтернативні стани s' допустимі [33].

Поширеним засобом автоматизації процесу ФВ відповідних ФС є інструментарій PRISM, що включає реалізації методів ФВ, з підтримкою дискретного, неперервного часу, з підтримкою недетермінізму і без [34]. Названий засіб застосовано у різних сферах, зокрема при здійсненні контролю несуперечності протоколів взаємодії компонентів бездротових мереж – згідно стандарту IEEE 802.15.4 [35]. Реалізовано, зокрема, підтримку виразних засобів темпоральної логіки PCTL* (Probabilistic CTL*), що дозволяє формалізувати обчислювальні процеси стохастичної природи [36]. Це уможливлено шляхом залучання математичних апаратів ланцюгів Маркова дискретного (Discrete-time Markov Chain, DTMC) і неперервного часу (Continuous-time Markov Chain, CTMC) при вирішенні задачі ФВ. У якості вагомого недоліку названого інструментарію зазначається неможливість одержання контр-прикладів –

ланцюгів станів СП, що призводять до виникнення збоїв та відмов. Як засіб усунення цього недоліку пропонується підхід, що базується на пошуку сильно зв'язних компонент СП [37].

Підхід до ФВ, згідно якого реалізовано інструментарій PRISM, носить назву PMC (Probabilistic / Stochastic Model Checking). Альтернативним засобом реалізації названого підходу є інструментарій Storm [38]. Наявні опубліковані результати дослідження ефективності останнього (за критерієм швидкодії) свідчать про перевагу Storm над PRISM у близько 2,75 разів [39].

Підхід PMC полягає у вирішенні задачі ФВ шляхом проведення імітаційного моделювання – на основі математичних апаратів DTMC, CTMC. Це певним чином знижує ефект експоненційного зростання простору станів СП від числа змінних станів, що показово проявляється за класичного підходу – перебору усіх станів СП. Вагомим недоліком PMC, однак, є варіативність закону розподілу випадкових величин, що фігурують у ФС [40, с. 31]. Це, у свою чергу, підвищує вагу аспекту достовірності даних-результатів ФВ ФС шляхом PMC. Можливим шляхом опрацювання зазначеного аспекту може слугувати ітеративний підхід до здійснення ФВ, із варіюванням законів розподілу випадкових величин та узагальненням одержуваних результатів.

У якості альтернативного шляху зниження ефекту експоненційного зростання простору станів СП від числа змінних станів можна зазначити також і залучення поширеного методу Event-B, що полягає у проведення декомпозиції досліджуваної ФС на складові ФС, і проведенні ФВ складових ФС. Наслідком цього є потреба, у тому числі, встановлювати і опрацьовувати зв'язки між складовими ФС – як на рівні змінних станів, так і на рівні відповідних подій модифікування значень змінних [41]. Стосовно критичних сценаріїв застосування, названий метод було успішно залучено, зокрема, у залізничній галузі – при проведенні ФВ ФС для ПАС системи оповіщення залізничного

транспорту. Окрім цього, метод Event-B успішно застосовується на індустріальному рівні вже більше 20 років [42].

Дієвим підходом є, у тому числі, застосування бінарних діаграм рішень (ROBDDs, Reduced Ordered Binary Decision Diagrams) – реалізується за «символьної» перевірки на моделі (SMC, Symbolic Model Checking). При цьому оперують не безпосередньо станами СП, а логічними виразами над змінними станів СП, на основі яких формуються множини станів СП. Це дозволяє проводити ФВ ФС, для яких будуються СП з понад 10^{20} станами [43]. Відповідним засобом, що автоматизує зазначений процес, є інструментарій NuSMV, що було застосовано, зокрема, у атомній енергетиці Фінляндії [21]. При цьому авторами вказується на актуальність створення і розвитку засобів автоматизації процесу синтезу ФС. У якості вихідних конструкцій, у свою чергу, було використано описи на мові VHDL (Very high speed integrated circuits Hardware Description Language). Авторами при цьому зазначено, що одержання ФС вручну – без залучення засобів автоматизації – потребує як значної обізнаності експерта-розробника у заданій предметній області, так і супроводжується впливом людського фактору, що, у свою чергу, ставить під сумнів достовірність даних-результатів ФВ ФС. Як результат, відсутність розвинутих засобів уніфікації формалізованих подань, а також супутніх засобів автоматизації, у тому числі процесу синтезу таких подань, обумовлює потребу створення ad-hoc-рішень для кожного окремого випадку побудови ФС – з позиції контролю відповідності результуючих ФС первинним поданням ПАС. Натомість, розвиток засобів автоматизації процесу синтезу ФС дозволить уніфікувати процес контролю адекватності одержуваних рішень і, таким чином, знизити вплив людського фактору.

Подібною до концепції в основі методу Event-B є застосування логічного прийому «абдукції», згідно якої кожне наступне припущення стосовно несуперечності заданої складової ФС будується на основі вже підтверджених

припущень стосовно несуперечності інших складових ФС, і результуючого висновку стосовно несуперечності ФС у цілому [44]. Подібно до цього, авторами роботи [45] застосовується індуктивний підхід, згідно якого саме декомпозиція задачі ФВ на складові задачі визначається у якості ключового фактору, що сприяє зниженню ефекту експоненційного зростання простору станів СП. Названий підхід полягає, у тому числі, у поступальному збільшенні рівня деталізації ФС. Це, у свою чергу, породжує суміжну проблему, винесену, однак, за межі представленої роботи, – встановлення такого рівня деталізації ФС, за якого проведення ФВ зазначеної ФС супроводжуватиметься для розробника ПАС вагомим корисним ефектом, що може проявлятися, у тому числі, наступним чином:

– у підтвердженні несуперечності ПАС, що, зокрема, сприятиме як упевненості розробника у несуперечності досліджуваних артефактів, так і уніфікації сприйняття артефактів колективом розробників;

– у виявленні суперечностей у ПАС на основі відповідної ФС, що проявляє потребу доопрацювання ПАС.

Для обох вищезазначених аспектів постає суміжна задача – встановлення відповідності результуючої ФС первинному артефакту – графічному поданню ПАС. Ця задача у межах дисертаційної роботи вирішується – шляхом розроблення і дослідження методу контролю відповідності ФС. Окрема увага приділяється співставленню конструкцій аналітичного рівня і рівня реалізації ФС.

Як узагальнення розглянутих вище підходів, методів, засобів вирішення задачі ФВ ФС, їх згруповано наступним чином:

– вирішення задачі ФВ зводиться до вирішення задачі виконуваності булевої формули (SAT, Boolean Satisfiability Problem). Ця задача є NP-повною, що було доведено лауреатом премії Тюрінга – Стівеном Артуром Куком [46]. Опубліковані результати застосування такого підходу полягають у засвідченні

вагомості одержуваного корисного ефекту шляхом варіювання рівня деталізації ФС, разом із ітеративним внесенням доопрацювань [47, с. 166];

– вирішення задачі ФВ зводиться до вирішення задач виконуваності формул у теоріях – цілих, дійсних чисел, списків тощо (SMT, Satisfiability Modulo Theories), де у якості змінних фігурують висловлювання логіки першого порядку. Показовими у даному контексті є праці д.ф-м.н. О. О. Летичевського, де, згідно висвітленої технології інсерційного моделювання, здійснювати ФВ артефактів процесу розроблення ПАС пропонується на кожному із етапів процесу розроблення. При цьому у якості засобу формалізації залучено виразний засіб Live UCM (Use Case Maps), математичний апарат якого подібний до ієрархічних мереж Петрі [12, с. 79, 80]. При цьому виокремлюються, зокрема, методи «чорної», «білої» (коли вже є програмна реалізація) скриньок, інтеграційної, регресійної символьних перевірок.

Ключова відмінність між SAT- і SMT-підходами – у першому випадку вирази у складі ФС приймають виключно булеві значення.

У представленій дисертаційній роботі задача ФВ ФС вирішується згідно SAT-підходу.

Варто зазначити, що широко застосовуваним засобом реалізації SMT-підходу є інструментарій *Z*3 від Microsoft, за якого ФВ здійснюється дедуктивним шляхом [48, 49].

У контексті згаданої вище технології інсерційного моделювання, з позиції залучення формальних методів і засобів на кожному із етапів процесу розроблення, уваги заслуговує також інструментарій VCC (Verified Concurrent C), призначений, згідно концепції RV, до залучення на етапі реалізації процесу розроблення – шляхом доповнення програмного коду на мові C/C++, що, у свою чергу, є поширеним засобом реалізації операційних систем, відповідними анотаціями – засобами перевірки перед- і пост-умов [50]. Інструментарій VCC при цьому реалізовано згідно логіки Гоара [51].

Результативним виявилось також сполучення інструментаріїв VCC і Z3: темпоральні формули одержано автоматизованим шляхом – на основі VCC-анотацій. Результати впроваджено у процес розробленя вбудованих систем шведською компанією Scania [52].

Серед перспективних напрямів розвитку SMT-засобів доречно зазначити також темпоральну логіку STL (Signal Temporal Logic), де охоплено також засоби подання дійсних значень змінних станів і модельного часу [53].

Отримані результати проведеного аналізу шляхів і засобів зниження ефекту експоненційного зростання простору станів СП зведено на рис. 1.3.

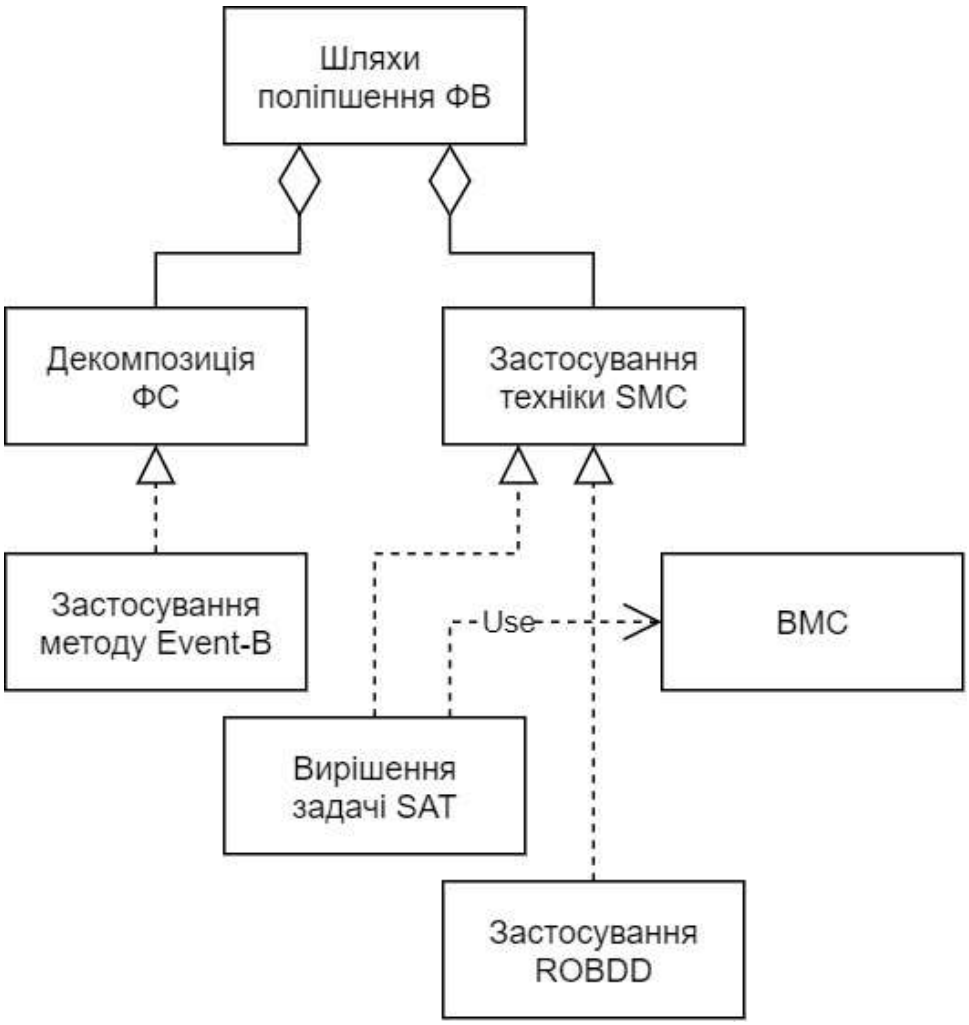

Рисунок 1.3 – Виокремлені напрями зниження ефекту експоненційного зростання простору станів СП

На рис. 1.3 вирішення задачі SAT та застосування ROBDD подано як альтернативні шляхи реалізації SMC. При цьому одержуваний корисний ефект від застосування засобів ROBDD залежить, у тому числі, від упорядкованості змінних станів СП. У свою чергу, застосування методів сімейства BMC (Bounded Model Checking), згідно яких глибина обходу простору станів СП обмежується заданим значенням, з урахуванням складності досліджуваного артефакту-подання ПАС, а також доступних розробникам обчислювальних можливостей наявного програмно-апаратного забезпечення для проведення ФВ, може супроводжуватися співставним одержуваним корисним ефектом, у порівнянні із ROBDD-засобами [54]. Допускається також сполучення BMC- та ROBDD-засобів.

Для врахування у ФС також і НФХ залучається, у тому числі, математичний апарат теорії часових автоматів (TA, Timed Automata) [55], що дозволяє у якості відповідного показника залучати часові витрати – як умову виконання переходу між станами СП. Це дає підстави розглядати зазначений математичний апарат також і у якості засобу валідації (рис. 1.2). Показовим і поширеним прикладом реалізації концепції TA є інструментарій UPPAAL, що дозволяє виконувати ФВ згідно підходу SMC [56]. Окремої уваги при цьому заслуговує можливість подання досліджуваної ФС у формі зваженого орієнтованого графу [57, 58], що сприяє поліпшенню сприйняття розробником архітектурної складової зазначеного артефакту. Разом із цим аспект уможливлення варіювання рівня деталізації ФС потребує подальшого доопрацювання.

З-поміж напрацювань у напрямі SMC доречним також вбачається виокремити технологію інсерційного моделювання, що будується згідно алгебраїчного підходу до формалізації процесів, поданих у формі артефактів.

Згідно даної технології залучення формальних методів і засобів має проводитись на кожному із етапів процесу розроблення [59, с. 61].

У якості поширеної альтернативи зазначеної вище структури (1.2) як засобу формалізованого аналітичного подання СП, що будується у процесі ФВ, доречно пригадати також математичний апарат мереж Петрі, що дозволяє подати досліджувану СП у формі орієнтованого дводольного мультиграфу, за якого вирішення задачі ФВ зводиться до вирішення задачі встановлення досяжності цільового стану СП [60], яка, у свою чергу, також характеризується експоненційним характером зростання обчислювальних і просторових витрат, у залежності від зростання кількості змінних, що фігурують у ФС.

Розглядаючи процес ФВ на основі засобів мереж Петрі як дискретно-подійне імітаційне моделювання, під «подією» розуміється виконання переходу, за якого мітка з «вхідної» локації переміщується до «вихідної». Особливістю такого математичного апарату є, подібно до зазначеного вище засобу UPPAAL, встановлення вагових коефіцієнтів для переходів між станами СП як умов переходів, а не для станів СП безпосередньо. Водночас проблема експоненційного зростання простору станів СП лишається, і вирішується, у тому числі, шляхом залучення методу структурної редукції та методів редукції часткових порядків, що дозволило одержати корисний ефект за показником зниження кількості станів СП, що потребують контролю, на величину до 60 % [61].

Підсумовуючи вищезазначене, доречно зауважити, що вибір методу ФВ обумовлюється, у тому числі, наявними / опрацьованими засобами формалізації і покладеними в їх основу концепціями. Наприклад, у контексті зазначеного вище поняття «події» у якості засобу формалізації може бути застосовано виразний засіб Live UCM – наступним чином [12, с. 80]: $\forall x\left(\alpha(x)\rightarrow\langle P(x)\rangle\beta(x)\right)$, де $\alpha(x)$ – ФС поточного стану СП, з якого система за рахунок процесу $\langle P(x)\rangle$

переходить до наступного стану, заданого як $\beta(x)$. При цьому можна зауважити, що дана формалізація подібна до такої, що застосовується у численні CSP, у тому числі – у трійках Гоара, де вирази вигляду $\alpha(x)$ і $\beta(x)$ опрацьовуються як перед- і пост-умови відповідно [62].

1.2.2 Сфери застосування методів перевірки на моделі

Відзначається, що, при розробленні електронних систем, перевірка артефактів процесу проєктування – контроль прийнятих проєктних рішень (Design Verification) – є ключовим аспектом, якому приділяється увага з позиції ФБ, і може займати до 80% усього часу процесу розроблення системи [63]. Більше того, формальні методи і супутні засоби застосовуються для контролю показників як ФХ (safety, liveness properties), так і НФХ зазначених систем на основі відповідних ФС. При цьому доречним вбачається зауважити, що ґрунтовна таксономія існуючих підходів до застосування формальних методів з метою контролю показників і ФХ, і НФХ (захищеності, надійності, часових витрат, енергетичних витрат) при проєктуванні була проведена у розрізі проєктування апаратної складової СКП [64]. За результатами проведеного порівняльного аналізу авторами відзначено важливість розроблення комплексних уніфікованих засобів ФВ, що дозволяло б виконувати перевірку при проєктуванні показників як ФХ, так і НФХ. Для виокремлення такої комплексності було залучено поняття «багатовимірної верифікації» (multidimensional verification).

З урахуванням вищезазначеного, висувається теза, що, по відношенню до програмної складової комп’ютерних систем, у тому числі – СКП, аспект опрацювання при проєктуванні показників і ФХ, і НФХ є не менш значимим, у порівнянні із проведенням контролю названих показників для апаратної складової – особливо з точки зору варіативності і комплексності ПАС. Більше

того, складність ПАС невпинно зростає з плином часу, через що у якості дієвого засобу стримування небажаного ефекту експоненційного зростання простору станів СП у залежності від кількості змінних станів вбачається варіювання рівня деталізації ФС. У межах представленої до захисту праці цей аспект адресується шляхом стратифікації формалізованих подань.

З позиції опрацювання ПАС за показником ФХ на етапі реалізації як складовому етапі процесу розроблення ПАС, що слідує за етапом проєктування, показовим прикладом може слугувати підхід, що ґрунтується на комплексному застосуванні методів перевірки на моделі Spin і Divine [65]. У якості предмету дослідження при цьому було опрацьовано інструментарій ADAPRO (ALICE Data Point Processing Framework), розроблений у Європейській організації з ядерних досліджень CERN, реалізований на мові програмування C++14 і призначений для створення мультипоточних програмних СКП [66]. Авторами було відзначено, що комплексне застосування засобів Spin і Divine супроводжувалося ефектом синергії – з позиції поглиблення розуміння розробником архітектурної складової системи. Для цього програмну реалізацію досліджуваної ПАС було переведено у ФС на основі виразних засобів формалізму Promela.

У якості зауваження до розглянутого вище підходу доречно зазначити, що для його застосування вже потрібно мати програмну реалізацію досліджуваної системи, а, отже, можна стверджувати стосовно надлишковості виконуваних кроків – за яких контроль архітектурної частини розроблюваної ПАС виконується не на етапі проєктування, а на етапі реалізації процесу розроблення. Як результат, маємо підхід, подібний до такого, що застосовується за зворотного розроблення (reverse engineering). У даному контексті супутніми небажаними факторами є додаткові часові і матеріальні витрати – у порівнянні з такими, що мали б місце, якби зазначений контроль проводився вже на етапі проєктування процесу розроблення. Із акцентом на даному аспекті у межах

представленої дисертаційної роботи адресується саме етап проєктування – як такий, за якого проводиться контроль показника / показників і ФХ, і НФХ. У якості досліджуваного показника ФХ при цьому опрацьовується несуперечність ПАС, а у якості показника НФХ – часові витрати, супутні реалізації ФХ згідно ПАС.

Варто, однак, зазначити, що припущення стосовно суперечності / несуперечності ПАС можуть бути як вірними, так і хибними. Це, у свою чергу, може бути спричинено як невідповідністю досліджуваної похідної ФС первинному артефакту – графічному поданню ПАС, так і суперечністю безпосередньо ПАС. Показовими прикладами у даному контексті можуть слугувати, у тому числі, модельно-орієнтовані системи виявлення і локалізації помилок (FDI, Fault Detection and Isolation) [67], де авторами пропонується кількісний стохастичний підхід до опрацювання небажаних подій, що можуть виникнути у процесі функціонування розроблюваної системи. У контексті СКП рівень критичності зазначених подій зростає.

Іншим демонстративним прикладом ПАС як СКП є «ядро» операційної системи Linux, де механізм синхронізації конкуруючих процесів (RCU, Read-copy Update) відіграє центральну роль у забезпеченні узгодженої взаємодії компонентів зазначеної системи. Через те несуперечність відповідної ПАС як показник ФХ набуває критичного значення. У якості засобу досягнення цього авторами праці [68] пропонується системний підхід до контролю несуперечності RCU шляхом застосування методу МС сімейства STMC (Stateless Model Checking), що дозволяє підвищити ефективність процесу ФВ за показником просторової складності – за рахунок того, що безпосередньо стани СП в ОП не зберігаються. При цьому зберігаються вирази на основі темпоральних засобів, що дають змогу вирахувати той чи інший стан СП. Варто, однак, зауважити, що методи сімейства STMC доречно застосовувати по відношенню до систем, що характеризуються недетермінізмом поведінки [69]. І

в даному контексті змістове навантаження відповідної ФВ вже стає подібним до такого, характерного для тестування – як форми реалізації валідації, коли є намагання сформувати тестові набори таким чином, щоб охопити найбільш імовірні, на думку розробника, сценарії функціонування системи. При цьому межі між ФВ і тестуванням, що виконуються, у тому числі, на різних етапах процесу розроблення ПАС, стають ще більш неявним (рис. 1.2).

З урахуванням вищезазначеного, при застосуванні методів сімейства STMC розробники оперують вже не станами СП безпосередньо, а «шляхами виконання» досліджуваної системи згідно ФС – коли стани СП вираховуються на основі складових ФС – темпоральних виразів. Відповідним прикладом є, у тому числі, метод RCMC, застосований для контролю несуперечності мультипоточних програм на мові програмування C++11, що, за твердженням авторів, істотним чином випереджає аналоги за показниками швидкодії [70].

У контексті випадку зворотного розроблення, згаданого вище, мають місце також обставини, за яких вичерпна інформація стосовно досліджуваної системи відсутня. За таких умов застосовуються підходи, у яких зазначена система розглядається як «чорний ящик» – шляхом опрацювання вихідних і результуючих даних, і автоматизованого синтезу відповідних моделей системи [71]. Зручним математичним апаратом для таких випадків є, у тому числі, ланцюги Маркова. Показовим прикладом відповідних СКП є, зокрема, системи реального часу, серед яких – система керування парогенератором [72]. Авторами праці при цьому відзначається потреба автоматизованого одержання формалізованих подань ПАС, призначених слугувати засобами уможливлення проведення ФВ.

Окремої уваги заслуговує аерокосмічна галузь, де відповідні СКП можуть розглядатися як системи систем, і формальні методи і засоби залучаються вже на початкових етапах процесу розроблення, починаючи від одержання формалізованих подань вимог до розроблюваної системи [73]. При цьому у

якості засобів формалізованого подання досліджуваних артефактів залучаються, тому числі, засоби SysML (The Systems Modeling Language). Наголошується на важливості розроблення і застосування засобів автоматизації процесу синтезу ФС. Для одержання останніх використовуються виразні засоби числення процесів CCS (Calculus of Communicating Systems) Робіна Мілнера [74].

Серед інших критичних сфер залучення формальних методів і засобів – оборонна, залізнична галузі, хімічна промисловість, авіація, медицина тощо [75].

Розглядаючи СКП як складні системи – системи систем, варто зауважити, що у процесі їх розроблення вагомі результати дає застосування, у тому числі, технік глибинного навчання, що базуються на математичному апараті нейронних мереж. Відповідним прикладом є, зокрема ПАС системи координування сукупності безпілотних літальних апаратів, призначення якої, серед іншого, виключення можливості зіткнень між зазначеними автономними програмно-апаратними комплексами [76]. Нейронні мережі при цьому залучаються у якості засобів досягнення цільової поведінки розроблюваної СКП. У свою чергу, якщо результати роботи нейронних мереж не є задовільними за показником достовірності, – маємо підґрунтя ставити під сумнів ФБ розроблюваної СКП. І, щоб підтвердити достовірність результатів роботи нейронних мереж, реалізованих у відповідності до технік глибинного навчання, використання формальних методів і засобів у якості засобів контролю достовірності таких результатів є дієвим [76]: відповідний метод ФВ – Reluplex – виявився представником згаданих вище SMT-засобів.

Проаналізовані вище підходи, відповідні методи і супутні засоби у контексті їх застосування у якості засобів контролю артефактів процесу розроблення СКП, у тому числі – ПАС, дають підстави вважати за доцільне залучати теоретико-множинний підхід до створення і опрацювання ФС як похідних артефактів на етапі проєктування ПАС у якості дієвого кроку [77, 78].

У контексті ФБ має місце припущення, що якість програмного забезпечення, що реалізує критичні ФХ, істотним чином впливає на надійність інформаційно-керуючих СКП; і для досягнення заданого рівня якості пропонується підхід, що ґрунтується на диверсному вимірюванні інваріантів (семантичних, інтервально-точносних, логічних) [79].

З урахуванням вищезазначеного, для окреслення спрямованості дисертаційного дослідження застосуємо системний підхід, подібний до такого, що фігурує, у тому числі, у стандартах аерокосмічної галузі – ECSS-E-00A [80] і актуальному ECSS-S-ST-00C [81] (рис. 1.4).

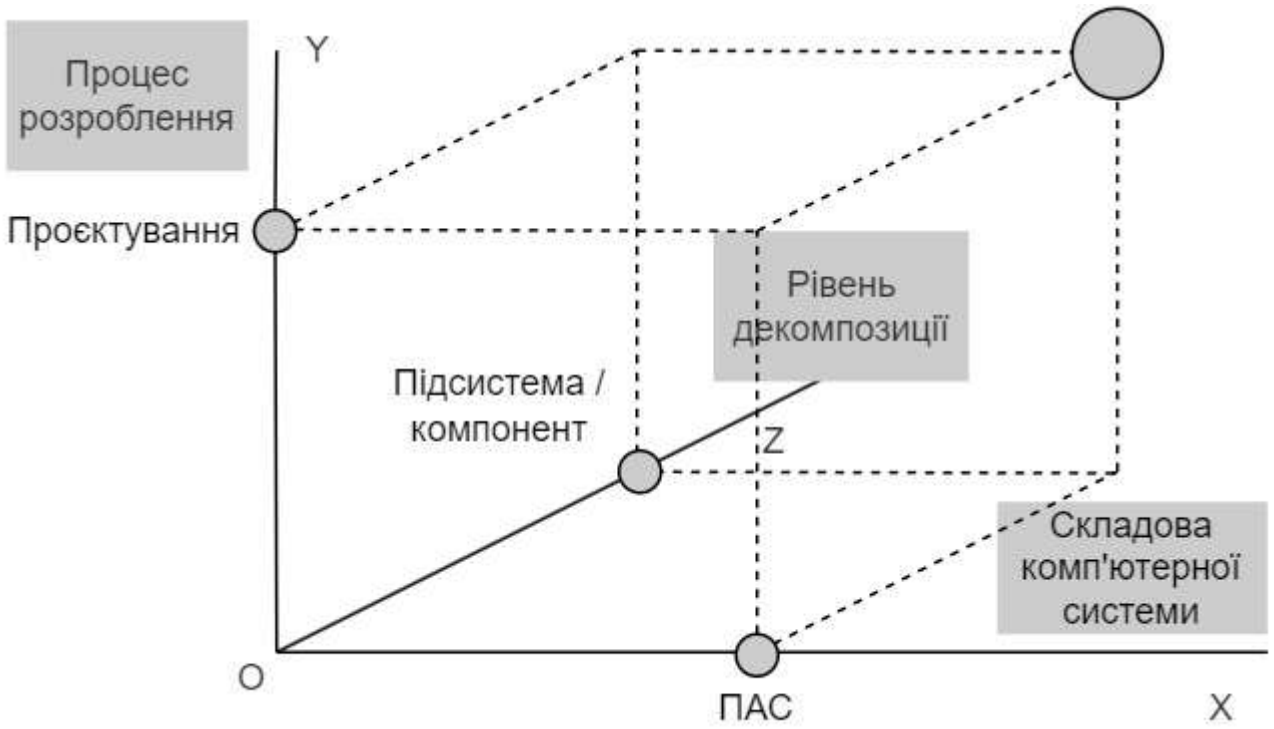

Рисунок 1.4 – Графічне подання спрямованості дисертаційного дослідження згідно системного підходу

У свою чергу, змістове навантаження осі Oy рис. 1.4 деталізовано на рис. 1.5.

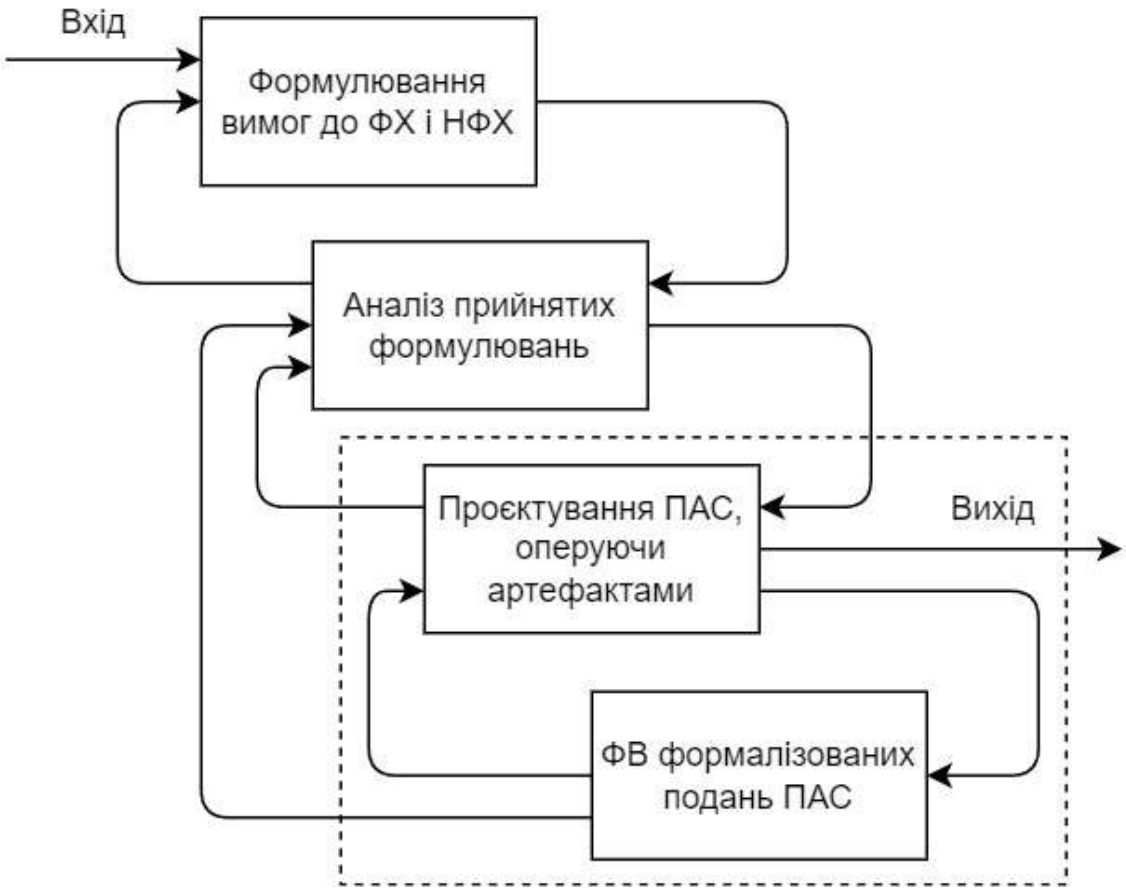

Рисунок 1.5 – Виокремлення специфіки дій, виконуваних у відповідності до розробленого підходу, викладеного у наступному розділі

На рис. 1.5 пунктиром окреслено кроки процесу розроблення ПАС, виконувані на етапі проєктування, згідно яких залучаються формальні методи і засоби у якості засобів контролю досліджуваних артефактів за показниками ФХ і НФХ.

При цьому у контексті досягнення ФБ особливої ваги набуває площина контролю адекватності похідних артефактів – формалізованих подань – ФС – первинним артефактам – поданням ПАС у формі блок-схем алгоритмів, UML-діаграм. Для сприяння зазначеному доцільним вбачається враховувати у ФС також предметно-орієнтовані аспекти досліджуваних артефактів, серед яких, – специфіка розподілу операцій, виконуваних згідно ПАС. Наприклад, для

випадку сценаріїв аерокосмічної галузі, за яких у якості досліджуваних СКП фігурують, у тому числі, ПАС систем керування космічним апаратом, має місце «суміш Шатл», що є поданням характеру розподілу відносних частин виконуваних обчислювальних операцій різних типів (табл. 1.1) [2, с. 70].

Таблиця 1.1 – Розподіл операцій для вирішення задач управління в аерокосмічній галузі

№ з/п	Операція	Частка, %
1	2	3
1	Завантаження операнду	26,1
2	Складання	15,5
3	Занесення до пам’яті	13,8
4	Множення	6,0
5	Ділення	1,9
6	Логічні операції	5,4
7	Зсув на 4 розряди	7,2
8	Перехід	22,3
9	Завантаження адреси	1,8
	Разом:	100,0

Аналізуючи вміст табл. 1.1, можна припустити, що маємо характер розподілу, подібний до нормального: з виокремлених 9 типів операції завантаження операнду, переходу, складання та занесення до пам’яті формують загальну відсоткову частку у близько 78 %. Це дає підстави вважати, що охоплення у ФС операцій зазначених типів може характеризуватися

розробником першорядним значенням – для випадку, якщо створюється система керування, призначена до застосування в аерокосмічній галузі.

Для узагальненого випадку у якості альтернативи до «суміші Шатл» застосовується «суміш Гібсона», для якої характер розподілу виокремлених типів операцій відрізняється: 40,3 % відводяться під операції складання (з фіксованою і плаваючою крапкою), 19 % – під операції індексної модифікації (робота з масивами), 17,5 % – під операції пересилання даних [2, с. 69; 82]. Загалом ці ти три типи формують відсоткову частку у близько 77 %.

Ґрунтуючись на вищезазначеному, висувається припущення, що, з урахуванням предметної галузі спрямування розроблюваної СКП, для сприяння адекватності похідних артефактів – ФС – первинним артефактам доцільним вбачається врахування, за можливості, саме предметно орієнтованого розподілу виконуваних операцій, що фігурують у розроблюваній ПАС. Це питання постає, у тому числі, у випадку, якщо виникає потреба змістити рівень деталізації похідної ФС – аби мати можливість опрацювати небажаний ефект експоненційного зростання простору станів СП за прийнятний час – з точки зору зниження супутніх вирішенню задачі ФВ обчислювальних і просторових витрат. Отримані результати проведеного оцінювання згаданого ефекту на рівні обчислювальних витрат для випадку застосування поширеного методу перевірки на моделі TLC зведено, у тому числі, у працях [83, 84]. При цьому мала місце ситуація, за якої доступної обчислювальній системі ОП виявилось недостатньо для проведення ФВ на основі зазначеного методу.

Варто зауважити, що для наведеного вище випадку актуальності набуває вирішення, у тому числі, проблеми стримування небажаного ефекту експоненційного зростання простору станів СП, яка може бути опрацьована як шляхом декомпозиції вирішуваної задачі ФВ – на рівні декомпозиції відповідної ФС, так і шляхом варіювання рівня абстракції / деталізації зазначеної ФС, з

урахуванням відповідної предметно орієнтованої «суміші» – якщо остання є доступною розробнику / колективу розробників.

Інший аспект – застосовуваний підхід до оцінювання одержуваного корисного ефекту від проведення ФВ. У межах представленої дисертації пропонується і застосовується дуальний підхід до оцінювання зазначеного ефекту:

– у випадку, якщо у результаті проведення ФВ для ФС із обраним рівнем деталізації було підтверджено несуперечність ПАС на основі зазначеної ФС, такий результат вбачається доречним розглядати як одержуваний корисний ефект – і з позиції поглиблення розуміння розробниками досліджуваної ПАС, і з позиції уніфікації цього розуміння представниками колективу розробників. При цьому постають суміжні питання – як з позиції достатності обраного рівня деталізації ФС, так і з позиції відповідності останньої первинному досліджуваному артефакту – щоб висновки, сформульовані за результатами проведеного контролю ФС, можна було поширювати також і на відповідний первинний артефакт;

– в іншому випадку – за якого має місце виявлення суперечності / суперечностей у ФС – корисний ефект полягає безпосередньо у виявленні зазначених суперечності / суперечностей, у результаті чого ФС і відповідний первинний артефакт підлягають доопрацюванню. При цьому відповідність похідної ФС первинному артефакту має бути підтверджена. Цей аспект, у свою чергу, актуалізує потребу розроблення методу контролю відповідності результуючої ФС.

З урахуванням вищезазначеного, успішність проведення ФВ визначається, у тому числі, залученими виразними засобами формування ФС. Ґрунтуючись на вичерпному наборі вагомих результатів успішного застосування відповідних засобів, у якості останніх опрацьовано темпоральну логіку дій TLA, а також супутні виразні засоби PlusCal і TLA+, деталізовані нижче.

1.2.3 Темпоральна логіка дій та відповідні засоби

Розглядаючи проєктування ПАС з позиції ітераційного підходу [85], важливою складовою вбачається, у тому числі, розроблення і застосування засобів автоматизації процесу синтезу ФС. У якості підходу до досягнення зазначеного застосуємо комбінування виразних засобів подання ПАС у формі ФС – для реалізації прозорого механізму співставлення конструкцій у складі артефактів, що опрацьовуються розробником аналітично, із відповідними конструкціями, що фігурують у складі похідних артефактів, поданих у формі ФС. Зазначені похідні артефакти при цьому адресуються у якості засобів уможливлення автоматизації процесу ФВ.

Серед таких виразних засобів – імперативна алгоритмічна мова PlusCal [86], залучення якої надає можливість попередньо сформувати архітектурну складову ФС на основі виразних засобів TLA+ [87]. Остання, у свою чергу, вже безпосередньо є артефактом, до якої в автоматизованому режимі застосовується метод перевірки на моделі TLC [88].

Аргументом на користь залучення саме засобів TLA+ є, у тому числі, їх математична строгість та модульність. Це надає можливість подати ПАС у ФС у формі єдиної результуючої темпоральної формули ψ (1.1), одержуваної індукційним шляхом [89].

Підтвердженням дієвості залучення саме засобів TLA+ є, у тому числі, сценарії одержання корисного ефекту корпоративним сектором економіки – при проведенні ФВ артефактів процесу розроблення композитних вебсервісів компанією Amazon, починаючи з 2011 р. [90, 91].

У свою чергу, у контексті сценаріїв СКП показовим є процес розроблення операційної системи реального часу OpenComRTOS, призначеної для космічного апарату «Rosetta», запущеного Європейським космічним агентством

у 2004 р. При проєктуванні оновленої версії зазначеної системи було успішно залучено засоби TLA+ [92]. При цьому було застосовано наступний підхід до організації процесу розроблення: окрема група розробників працювала над архітектурною складової створюваної системи, інша – над побудовою і перевіркою відповідних ФС на основі засобів TLA+. У результаті було як поліпшено архітектуру системи, так і близько десятикратно зменшено обсяг програмного коду [93], – за рахунок, у тому числі, як поглиблення розуміння створюваної ПАС окремим розробником, так і уніфікації зазначеного розуміння серед учасників колективу розробників. Серед інших проявів одержуваного корисного ефекту від залучення засобів TLA+ – підвищення швидкодії результуючої програмної реалізації, зниження вимог до обчислювальних ресурсів. Цього було досягнуто за рахунок абстрагування від аспектів програмної реалізації на рівні дослідження формалізованих подань.

З урахуванням вищезазначеного, доречним вбачається співставлення виразних можливостей засобів PlusCal і TLA+, що залучаються, у тому числі, на етапі проєктування процесу розроблення ПАС, із виразними можливостями безпосередньо мов програмування, що мають місце на етапі реалізації. У даному контексті засоби PlusCal і TLA+ можна віднести до таких, що подібні до псевдокоду: вища математична строгість, відсутність потреби опрацьовувати питання виділення / вивільнення ОП тощо. У свою чергу, залучення засобів PlusCal і TLA+ відповідає положенням, у тому числі, згаданого вище стандарту IEC 61508 [6].

Узгоджуючись із висловлюванням лауреата премії Тюрінга 2013 р. – Л. Лемпорта [89, с. 24], згідно якого вибір «правильного» рівня деталізації ФС є запорукою результативності ФВ, дієвим кроком вбачається застосування ітеративного підходу до створення формалізованих подань досліджуваних артефактів. Показовим прикладом є, зокрема, ітеративне залучення засобу TLA+ і методу TLC у процесі поступального контролю несуперечності ФС для

шаблонів проєктування, за якого на кожному наступному кроці рівень деталізації ФС підвищується [94].

Ще одним демонстративним випадком результативності залучення логіки TLA і супутніх засобів є критичні сценарії контролю несуперечності взаємодії центрального процесора із кеш-пам’яттю, опрацьовані R&D-підрозділом компанії Intel 2008 р.: як результат, було повідомлено про виявлення 45 суттєвих помилок у артефактах [95].

Іншим прикладом дієвості використання засобу TLA+, методу TLC, а також відповідного засобу автоматизації TLA+ Toolbox, є їх використання для контролю артефактів проєктування компонентів розподілених програмних систем – вебсервісів Microsoft Azure, Amazon AWS [96].

Іншим показовим прикладом дієвості засобів TLA і TLC є їх успішне використання для підтвердження несуперечності реалізації протоколу Zab (ZooKeeper Atomic Broadcast Protocol), що застосовується у складі поширеної розподіленої програмної системи забезпечення високоефективного обміну повідомленнями Apache Kafka [97].

Як узагальнення до охоплених вище сценаріїв корпоративного спрямування доцільним вбачається зауважити, що аспекти автоматизації процесу синтезу ФС у відповідних публікаціях висвітлено не було.

Серед предметних областей, де було заявлено стосовно одержання корисного ефекту від залучення зазначених методу і засобів, є, у тому числі, і сфера Інтернету речей, зокрема – програмно-конфігуровані мережі у їх складі [98–100].

З урахуванням обчислювальних можливостей актуальних комп’ютерних систем, застосовуваних у якості програмно-апаратних платформ, на основі яких реалізується проведення ФВ в автоматизованому режимі, серед чинників, що поєднують розрізнені предметно-орієнтовані сценарії застосування формальних методів і засобів є, у тому числі, аспект мультипоточності, який у межах

представленої дисертації опрацьовується у якості фактору зниження часових витрат, супутніх процесу ФВ.

У реалізації поширеного методу перевірки на моделі TLC передбачено залучення мультипоточності: на прикладі сценарію проведення мультипоточної ФВ алгоритму мультипоточної реалізації двонапрямленої черги, заснованої на виконанні операції подвійного порівняння і обміну (Double Compare and Swap, DCAS), лауреатом премії Тюрінга Л. Лемпортом було отримано результат скорочення часових витрат у більш як 40 разів – на 384-процесорній обчислювальній системі [101]. При цьому автором було зазначено, що серед стримуючих чинників одержуваного прискорення постають, у тому числі, і обмеження, що накладаються програмною мовою реалізації методу TLC: у даному випадку – мовою програмування Java. Разом із цим, у зазначеній праці не було висвітлено аспектів залежності одержуваного прискорення від застосовуваного методу обходу простору станів СП, а також від архітектурної складової ФС, для якої СП будується у процесі ФВ. У свою чергу, опрацювання зазначених аспектів у межах представленої дисертаційної роботи дозволить посприяти підвищенню ефективності процесу ФВ на основі методу TLC за показником зниження супутніх часових витрат. Дану площину висвітлено у четвертому розділі, присвяченому розробленому розвитку методу TLC.

Показовими є, у тому числі, алгоритми широкомовного розсилання, застосовувані у розподілених системах: метод TLC і засіб TLA+ було успішно залучено у якості засобів контролю несуперечності відповідних алгоритмів, реалізованих на мові функційного програмування Erlang [102].

Узгоджуючись із засадами озвученого вище дуального підходу до оцінювання корисного ефекту, одержуваного розробниками у результаті використання ними формальних методів і засобів, на прикладі створення ФС розподіленої бази даних Microsoft Azure Cosmos DB на основі засобів TLA+ було відзначено, що таке подання, разом із тим, що є більш строгим, також

характеризується ліпшою виразністю – порівняно із поданнями на основі альтернативних засобів [103]. Це дозволило розробникам і уніфікувати, і поглибити розуміння ними створюваної системи, що посприяло, у тому числі, підвищенню повноти охоплення поведінок зазначеної системи у документації, включаючи і виявлені на основі методу TLC і засобу TLA+ нетипові поведінки.

Резюмуючи охоплені вище праці, у яких засвідчено стосовно вагомості одержуваного розробниками корисного ефекту від застосування методу TLC і супутніх засобів, варто, однак, зауважити, що даний метод ґрунтується на екстенсивному переборі елементів простору станів СП, і на вирішенні для кожного із цих елементів задачі SAT. Така особливість, у свою чергу, сполучена із експоненційним характером зростання простору станів СП від кількості змінних, що фігурують у ФС. Кроком у напрямі зниження цього ефекту є застосування техніки символьної перевірки – шляхом залучення методів, реалізованих згідно SMT-підходу. Для випадку логіки TLA таким методом є APALACHE, який, за свідченням його розробників, перевершує метод TLC за швидкодією [104, 105].

Метод APALACHE, однак, досі не висвітлено у достатній мірі у наукових публікаціях з позиції свідчень на користь вагомості одержуваного корисного ефекту від його застосування у критичних сценаріях. Через цей аспект у якості базового методу при проведенні дисертаційних досліджень залучено саме перевірений часом метод TLC. Розроблений розвиток даного методу у напрямі зниження результуючих часових витрат, супутніх його використанню за ітеративного підходу до організації процесу ФВ, викладено у четвертому розділі.

1.2.4 Підходи до застосування формалізованих подань

Підходи опрацьовуються у контексті архітектурної відповідності первинних артефактів – графічних подань прийнятих проєктних рішень – і похідних від них формалізованих подань. Останні, у свою чергу, можуть бути залучені, у тому числі, у якості конструкцій, на основі яких одержуються похідні від них тестові послідовності [106]. При цьому серед показників успішності такого залучення – повнота покриття тестовими послідовностями, що залежить, у тому числі, від специфіки архітектурної складової ФС, залучених виразних засобів. Наприклад, якщо застосовується символьний підхід, зокрема на основі методу NuSMV, то серед факторів є, у тому числі, і семантична складова в основі використаного формалізму. На противагу – у праці [106] було показано, що застосування випадкових тестових послідовностей нерідко супроводжується ліпшими результатами. Шляхом усунення цього ефекту авторами вбачається ускладнення ФС. Це, у свою чергу, супроводжується зростанням простору станів СП, яке пропонується стримувати шляхом застосування діаграм BDD і символьного методу NuSMV. Дієвість такого кроку висвітлено, зокрема, на прикладі автоматизованої верифікації ПАС системи захисту атомного реактору [107]. При цьому було виявлено близько $7.3 \cdot 10^{31}$ станів СП.

Альтернативний підхід до залучення формальних методів і засобів – одержання ФС для програмної реалізації. При цьому також постає питання повноти покриття програмного коду у ФС – оперують, у тому числі критерієм MC/DC (Modified Condition and Decision Coverage) [108]. Зазначено, що маніпулювання архітектурною складової названої реалізації для поліпшення показника MC/DC не дало позитивних результатів. На противагу – позитивний ефект було отримано шляхом збільшення повноти покриття програмного коду у ФС. Засобом здійснення цього є, у тому числі, вищезазначений інструментарій

VCC, що було залучено, зокрема, для перевірки системи Microsoft Hyper-V [109]. Метод перевірки при цьому базувався на SMT-підході.

1.2.5 Аспекти залучення модельно-орієнтованих засобів

Модельно-орієнтовані засоби контролю ФХ типово застосовуються на початкових та проміжних етапах процесу розроблення, серед яких, у тому числі, – етапи проєктування та реалізації. Серед таких засобів – як вузько напрямлені, так і комплексні рішення. Прикладом комплексного рішення слугують виразний засіб Lustre та відповідний інструментарій S3 (Systerel Smart Solver), застосовуваний і на етапі проєктування, і на етапі реалізації [110]. У першому випадку несуперечність ПАС контролюється методом перевірки на моделі, у другому – вирішується згадана вище задача контролю відповідності результуючої програмної реалізації стосовно ФС – для критичного з позиції ФБ сценарію контролю несуперечності вбудованої програмної системи автоматичного захисту.

Прикладом вузько напрямленого комплексу засобів є, у тому числі, інструментарій SBIP (Stochastic Behavior-Interaction-Priority), що дозволяє формувати ФС згідно ієрархічного підходу – індукційним шляхом, залучаючи при цьому у якості математичного апарату ланцюги Маркова [111]. Процес ФВ при цьому реалізовано згідно підходу PMC (рис. 1.2). У свою чергу, у розрізі модельно-орієнтованого підходу до організації процесу розроблення, формальні методи сімейства PMC залучаються також і у якості засобів контролю показників НФХ, серед яких – надійність, енергоспоживання [112].

Математичний апарат ланцюгів Маркова, а також мережі Петрі, було залучено у якості засобів уможливлення контролю при проєктуванні у якості показника НФХ продуктивності СКП – систем керування, застосовуваних на атомних електростанціях [113].

Альтернативний підхід до контролю показників НФХ базується на залученні згаданого вище інструментарію UPPAAL, реалізованого згідно положень теорії часових автоматів. У якості досліджуваних показників при цьому фігурують, у тому числі, часові обмеження на переходи між станами СП; процес ФВ, у свою чергу, реалізовано шляхом SMC. Дієвість такого кроку продемонстровано на прикладі розроблення програмної системи реального часу [114].

У контексті модельно-орієнтованого підходу до організації процесу розроблення СКП, у відповідності до положень зазначеного вище стандарту ISO 26262, шляхом залучення комплексу виразних засобів UML/MARTE (Modeling and Analysis of Real-Time and Embedded Systems) пропагується опрацювання аспектів ФБ на різних рівнях абстракції подань артефактів на етапі проєктування [115]. У свою чергу, з позиції безшовності сполучення артефактів, одержуваних за модельно-орієнтованого підходу, виникає потреба у створенні і залученні прозорих і строго регламентованих механізмів перетворення артефактів. Прикладом є оперування формалізованими поданнями ПАС на основі засобів UML/EAST-ADL (Architecture Description Language), реалізованих згідно стандарту ISO 26262, із наступним перетворенням таких подань у комп'ютерні моделі на основі виразних засобів SystemC/SystemC-AMS, що уможливлюють проведення ФВ шляхом імітаційного моделювання [116]. При цьому у контексті ПАС для системи керування джерелом живлення у якості досліджуваного показника НФХ було опрацьовано часову затримку розповсюдження сигналів між компонентами системи.

Згідно концепції поступального перетворення первинних артефактів-подань архітектурної складової розроблюваної системи при проєктуванні у відповідні ФС, на основі яких реалізується процес ФВ методом перевірки на моделі, представлено також підхід, за якого у якості засобу формалізації первинних артефактів залучено мову AADL (Architecture Analysis and Design

Language) [117]. При цьому у якості засобу подання результуючої ФС залучено логіку LTL. Авторами при цьому зауважено стосовно обмеженості даних виразних засобів. У якості потенційної альтернативи відзначено засоби μ-числення.

З урахуванням вищезазначеного, у якості засобу уможливлення варіювання рівня деталізації артефактів при проєктуванні ПАС доцільним вбачається залучення засобів математичного апарату DEVS – по причині наявності у складі останнього основоположних конструкцій «атомарної» (AM, Atomic Model) і «складеної» (CM, Coupled Model) DEVS-моделей, що містять також і засоби формалізованого подання НФХ-складової [118, 119]. Це дозволить здійснення контролю досліджуваних показників НФХ при проєктуванні ПАС.

Підтвердженням дієвості такого кроку є результативність залучення засобів DEVS для проведення контролю значень часових витрат у якості показника НФХ, супутніх реалізації ФХ згідно ПАС розподіленої системи на прикладі композитного вебсервісу [120–122].

Узагальнення отриманих результатів проведеного аналізу методів і засобів контролю показників ФХ і НФХ у процесі розроблення ПАС зведено у табл. 1.2.

У табл. 1.2 опрацьовані формальні методи і засоби згруповано за їх призначенням: таким чином, щоб представники різних груп були взаємодоповнюючими у своїй сукупності – у розрізі результуючого комплексу засобів, призначеного до застосування на етапі проєктування процесу розроблення ПАС. При цьому сформульовано припущення, що результуючий комплекс методів і засобів має уможливлювати проведення контролю показників і ФХ, і НФХ, в автоматизованому режимі вже на етапі проєктування у складі етапів процесу розроблення ПАС.

Таблиця 1.2 – Узагальнення опрацьованих методів і засобів контролю показників ФХ і НФХ розроблюваної СКП

№ з/п	Призначення методу / засобу		
	Виразні засоби	Метод контролю	Засоби автоматизації
1	3	4	5
1	Виразний засіб Live UCM – наступним чином [12, с. 79, 80]. Засоби структури Кріпке [18], FACTum [29], LTL, CTL [32, 33], PCTL* [36], DTMC, CTMC [37], трійки Гоара, ROBDDs [43], STL (Signal Temporal Logic) [53], логіка Гоара [62], PlusCal [86], TLA; TLA+ [88, 89], Числення CSP [51], CCS [74], Lustre [110], ланцюги Маркова, SBIP [111], мережі Петрі, UML/MARTE [115], UML/EAST-ADL [116], AADL [117], засоби DEVS [118, 119].	Методи NuSMV [21], PRISM [34], Storm [38], Event-B [41], Z3 від Microsoft [48, 49], Spin, Divine [65], RCMC [70], Reluplex [76], TLC [94], APALACHE [104, 105], метод S3 [110], метод імітаційного моделювання.	UPPAAL [56], ADAPRO (ALICE Data Point Processing Framework) [66], TLA+ Toolbox [96], VCC від Microsoft [109], DEVS Suite [174].

Згідно табл. 1.2, а також сформульованого вище припущення, було прийнято рішення залучати у якості засобів формування аналітичних подань досліджуваних артефактів виразні засоби структури Кріпке, а також засоби числення CSP:

– засоби структури Кріпке як засоби формування первинних аналітичних подань для первинних артефактів, представлених, у тому числі, у формі блок-схем алгоритмів, UML-діаграм дій / станів;

– засоби числення CSP як засоби формування похідних аналітичних подань – проміжна ланка, призначення якої – наблизити концептуальну складову первинних аналітичних подань до такої, що подібна до зазначеної складової іншої виокремленої концептуальної площини опрацювання досліджуваних артефактів – площини рівня реалізації.

Озвучену концептуальну площину рівня реалізації залучено до розгляду з позиції уможливлення введення автоматизації у контексті процесу контролю досліджуваних артефактів шляхом використання формальних методів і засобів. Із цією площиною прийнято рішення асоціювати наступні виразні засоби:

– виразні засоби алгоритмічної мови PlusCal як засоби попереднього формалізованого подання архітектурної складової ПАС. Зазначений формалізм характеризується математичною строгістю, модульністю;

– виразні засоби формалізму TLA+ як засоби подання артефактів, до яких безпосередньо застосовується метод перевірки на моделі. У якості останнього розглянуто метод TLC, використання якого уможливить автоматизований контроль досліджуваної при проєктуванні ПАС за показником несуперечності ПАС – показником ФХ. У якості допоміжного засобу автоматизації розглянуто інструментарій TLA+ Toolbox.

Сформульовані засади, окреслені попередніми двома тезами, адресують площину контролю несуперечності ПАС – досліджуваного показника ФХ. У свою чергу, для проведення при проєктуванні ПАС також і контролю показника / показників НФХ прийнято рішення залучити, у тому числі, і наступні методи і засоби, зведені у табл. 1.2:

– виразні засоби математичного апарату DEVS у якості засобів формування формалізованих подань, що включають також і засоби подання НФХ-складової.

– інструментарій DEVS Suite у якості допоміжного засобу автоматизації процесу контролю досліджуваного показника НФХ шляхом проведення дискретно-подійного імітаційного моделювання при проєктуванні ПАС.

Отже, у межах даного заключного пункту проведеного аналітичного огляду було окреслено напрями застосування, а також склад допоміжних методів і засобів, призначених до залучення при проведенні дисертаційних досліджень.

ВИСНОВКИ ДО РОЗДІЛУ 1

Таким чином, у розділі викладено отримані результати проведеного аналізу аспектів застосування методів і засобів контролю показників ФХ і НФХ розроблюваної ПАС системи критичного призначення, серед яких – методи перевірки на моделі, супутні моделі, інструменти, підходи, виразні засоби.

Отримані результати полягають у наступному:

1. Обґрунтовано актуальність проведення контролю артефактів, одержуваних у процесів розроблення ПАС, вже на етапі проєктування. У якості відповідних засобів опрацьовано формальні методи і супутні засоби. Наголошено на важливості стримування небажаного ефекту експоненційного зростання простору станів СП, що будується у процесів ФВ. У даному контексті відзначено першорядність аспекту автоматизації застосування формальних методів і супутніх засобів, особливо за ітеративного підходу до організації процесу ФВ.

2. Проведено класифікацію методів і засобів контролю артефактів процесу проєктування. У результаті було виокремлено сімейство методів перевірки на моделі, і супутні засоби, у якості предмету дослідження. Серед передумов такого кроку – високий рівень їх придатності до автоматизованого застосування, широкий спектр успішних сценаріїв застосування у процесі розроблення СКП. Вагомість аспекту придатності до автоматизації обґрунтовано з позиції складності ПАС і пов’язаного із цим експоненційного характеру зростання простору станів СП від кількості змінних, що фігурують у ФС. Було зауважено, що, у випадку застосування методів перевірки на моделі і супутніх засобів, разом із задачею ФВ, постає до вирішення також і суміжна задача – задача контролю відповідності результуючих ФС, на основі яких здійснюється ФВ, первинним артефактам.

3. Проаналізовано шляхи зниження небажаного ефекту експоненційного зростання простору станів СП від кількості змінних ФС. Серед дієвих заходів виокремлено, у тому числі, декомпозицію ФС.

4. Проаналізовано залежність архітектурної складової досліджуваних артефактів від предметно орієнтованого сценарію застосування СКП, ПАС якої подано у формі артефактів. Опрацьовано суміші Гібсона (загальний випадок) і «Шатл» (для випадку аерокосмічної галузі). За результатами проведеного аналізу висунуто припущення, що архітектурна складова досліджуваних артефактів залежить від предметної області застосування СКП.

5. У якості досліджуваного показника ФХ прийнято рішення опрацьовувати несуперечність ПАС – вже на етапі проєктування процесу розроблення ПАС. У свою чергу, у якості відповідного показника НФХ адресовано часові витрати, супутні реалізації ФХ згідно ПАС.

6. Обґрунтовано перспективність застосування методу перевірки на моделі TLC, а також супутніх засобів – темпоральної логіки дій TLA і відповідних виразних засобів PlusCal і TLA+ – у якості засобів проведення контролю артефактів, одержуваних у процесі розроблення ПАС – вже на етапі проєктування – за показником несуперечності ПАС. Серед залучених аргументів – математична строгість засобів PlusCal і TLA+, вичерпний перелік критичних сценаріїв успішного застосування методу TLC і засобів PlusCal і TLA+, для яких вже було одержано вагомий корисний ефект.

7. Для розроблення методу і відповідного засобу проведення контролю досліджуваного показника НФХ при проєктуванні ПАС прийнято рішення залучити конструкції «атомарної» і «складеної» моделей математичного апарату DEVS, а також метод дискретно-подійного імітаційного моделювання. У свою чергу, оперування конструкціями «атомарної» і «складеної» моделей має на меті, у тому числі, сприяти модульності одержуваних на їх основі результуючих складених артефактів.

8. Сформульовано засади-обґрунтування доцільності розроблення підходу, призначеного слугувати комплексним засобом контролю показників і ФХ, і НФХ розроблюваної ПАС вже на етапі проєктування процесу розроблення. Вони полягають, у тому числі, у своєчасності виявлення і усунення потреби доопрацювання прийнятих проєктних рішень, поданих у формі артефактів. Зазначений підхід викладено у наступному – другому – розділі. Підхід призначений слугувати засобом сполучення розроблених методів і засобів, викладених у наступних розділах.

РОЗДІЛ 2
РОЗРОБЛЕННЯ МОДЕЛІ ПОДАННЯ ПРОГРАМНО-АЛГОРИТМІЧНОЇ СКЛАДОВОЇ

На основі отриманих результатів попереднього розділу викладено розроблений підхід до комплексного застосування методів і засобів контролю артефактів процесу розроблення ПАС вже на етапі проєктування – за показниками і ФХ, і НФХ.

У розділі викладено розроблену модель подання ПАС у формі ФС при проєктуванні. Дана модель призначена слугувати засобом уніфікації, згідно якого будуються ФС. Останні, у свою чергу, призначені бути засобами, по відношенню до яких застосовується метод перевірки на моделі – TLC, а також розроблений розвиток названого методу, викладений у четвертому розділі. У свою чергу, метод одержання ФС у відповідності до представленої моделі викладено у третьому розділі.

Застосування запропонованої моделі має на меті сприяти автоматизації процесу постачання артефактів – ФС – вихідних конструкцій для методу TLC або розробленого розвитку цього методу.

Модель базується, у тому числі, на оперуванні трійками і аксіомами Гоара, серед яких – аксіома композиції. При цьому залучення правила композиції дозволило скоротити результуючу кількість рядків коду ФС на основі засобів TLA+, що було підтверджено експериментальним шляхом для граничних випадків, опрацьованих у наступному – третьому – розділі.

У якості допоміжного засобу, залучення якого передує одержанню результуючої ФС на основі засобів TLA+, було застосовано алгоритмічну мову PlusCal, що дозволило сформувати архітектурну складову результуючої ФС.

2.1 Підхід до сполучення засобів контролю

Розроблена і викладена у межах розділу модель формалізованого подання ПАС призначена до застосування у якості складової відповідного комплексного підходу – засобу сполучення винесених на захист наукових здобутків. Названа модель при цьому є засобом, що залучається у розрізі контролю досліджуваного показника ФХ – несуперечності ПАС.

Базуючись на результатах проведеного аналізу, викладених у першому розділі, у межах розробленого підходу реалізується механізм поєднання у формі комплексного рішення розроблених і винесених на захист методів, розвитку методу і моделей – у якості засобів контролю показників і ФХ, і НФХ – вже на етапі проєктування процесу розроблення ПАС (рис. 2.1).

На рис. 2.1 розроблений комплексний підхід викладено у формі UML-діаграми. Складовим діаграми надано наступне змістове навантаження:

– у формі пронумерованих блоків подано наукові здобути, винесені на захист;

– у формі коментарів виокремлено відмінні ознаки, характерні для відповідних здобутків, у тому числі – засоби формалізації;

– застосовано саме відношення агрегування, а не композиції, – щоб підкреслити допустимість використання відповідних складових як безпосередньо – поза межами зазначеного підходу, так і у складі альтернативних конструкцій.

Згідно рис. 2.1, результати застосування розробленого методу синтезу ФС (блок 1), викладеного у третьому розділі, постачаються у якості вихідних конструкцій для здобутків, поданих пронумерованими блоками 2 і 4, – розробленого методу контролю значення заданого показника НФХ (блок 2), викладеного у заключному, шостому, розділі, і розробленого розвитку

поширеного методу перевірки на моделі TLC, викладеного у четвертому розділі (блок 4).

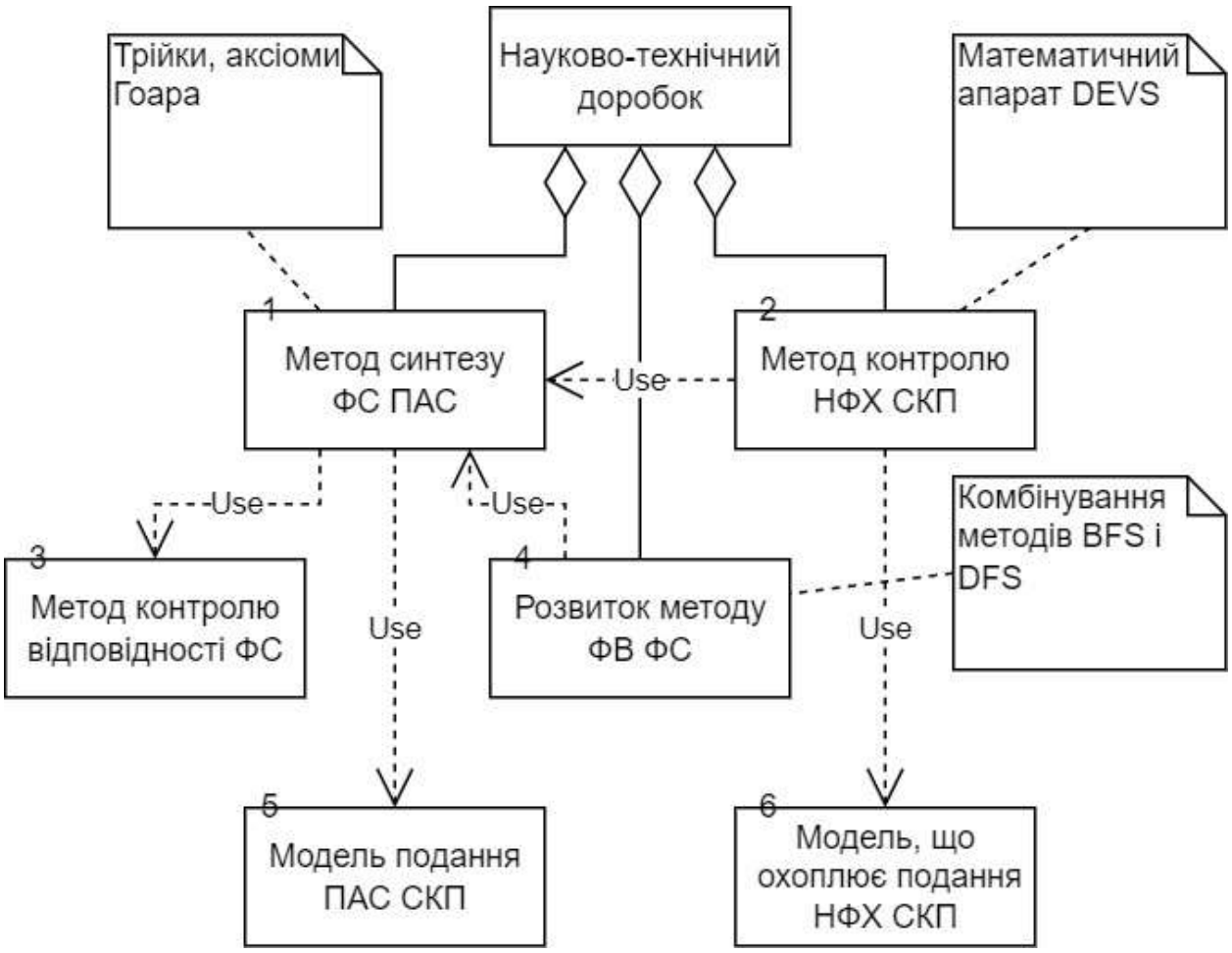

Рисунок 2.1 – Архітектурне подання розробленого комплексного підходу

На рис. 2.1 складові, подані блоками 3, 5 і 6, є допоміжними засобами, на основі яких базуються ключові здобутки – методи і розвиток методу, подані блоками 1, 2 і 4:

– блок 3. Розроблений метод контролю відповідності ФС (викладено у третьому розділі), одержуваних на основі розробленого методу синтезу, поданого блоком 1, – призначений слугувати засобом підтвердження архітектурної відповідності похідних ФС – первинним артефактам: блок-схемам

алгоритмів, UML-діаграмам дій. Застосовується як допоміжний засіб – у якості заключного кроку методу синтезу ФС;

– блок 5. Розроблена модель подання ПАС (викладена у другому розділі). Призначена слугувати засобом уніфікації ФС, одержуваних на основі розробленого методу синтезу ФС. Уніфікація при цьому розглядається у якості чинника, що уможливлює автоматизацію;

– блок 6. Розроблена модель на основі математичного апарату DEVS (викладена у п’ятому розділі), що включає, у тому числі, і засоби подання НФХ-складової. Побудована згідно ієрархічного підходу, призначена слугувати засобом уніфікації комп’ютерних моделей, по відношенню до яких застосовується розроблений метод, поданий блоком 2.

Розроблений підхід, викладений на рис. 2.1, побудовано у відповідності до системи положень об’єктно-орієнтованого підходу до програмування [123] – з проекцією на площину етапу проєктування процесу розроблення, а не етапу реалізації, що полягає у дотриманні єдності застосовуваних концепцій і нотацій: як у розрізі аналітичного сприйняття артефактів розробником, так і у розрізі їх програмної реалізації – з точки зору уможливлення автоматизації процесу ФВ.

Комплексність підходу полягає в охопленні на етапі проєктування ПАС засобів здійснення контролю і несуперечності ПАС як показника ФХ, і супутніх реалізації ФХ згідно ПАС витрат (часових, матеріальних) [124, с. 66].

Несуперечність ПАС як досліджуваний показник ФХ контролюється шляхом вирішення задачі ФВ (1.1) [125], із залученням розроблених методів, розвитку методу і моделі, поданих блоками 1, 3, 4 і 5 рис. 2.1 відповідно. Цим здобуткам присвячено перші чотири розділи дисертаційної роботи.

У свою чергу, у п’ятому і шостому розділах при проведенні дисертаційних досліджень згідно розроблених моделі і методу, поданих блоками 2 і 6 рис. 2.1, у якості досліджуваного показника НФХ залучено часові витрати, супутні реалізації кроків ПАС.

Надалі у межах поточного розділу викладається розроблена модель подання ПАС, зображена на рис. 2.1 блоком 5.

2.2 Постановка вирішуваної задачі

При описі розробленої моделі застосуємо теоретико-множинний підхід, згідно якого формулюватимемо основоположні конструкції і засоби їх одержання. Для постановки вирішуваної у розділі задачі за основу візьмемо структуру (1.2).

ФС розглянемо на двох рівнях – аналітичному рівні і рівні реалізації. Аналітичний рівень – площина сприйняття ФС розробником. Рівень реалізації – засіб уможливлення автоматизації процесу ФВ на основі ФС методом перевірки на моделі.

У якості засобів подання ФС на рівні реалізації використаємо алгоритмічну мову PlusCal та формалізм TLA+. Формалізм PlusCal призначений забезпечити подання на рівні реалізації архітектурної складової ФС, а саме – структури та зв’язків. Структура при цьому визначається складом блоків вихідних артефактів – блок-схеми алгоритму, UML-діаграми дій; зв’язки – сполученнями між названими блоками.

Зауваження:

– вирішувана у межах розділу задача охоплює також архітектурний аспект (структуру, доповнену зв’язками), що у визначенні поняття «артефакт» не фігурує. Цей аспект можна вважати розширенням базової інтерпретації зазначеного поняття у контексті представленої роботи;

– зміст ФС як похідного артефакту визначається складом змінних і конструкціями на їх основі.

У відповідності до вищезазначеного, у розділі ставиться і вирішується наступна задача:

– розробити модель формалізованого подання артефактів процесу проєктування ПАС СКП, що фігурують у графічній формі – у формі блок-схеми алгоритму, UML-діаграми дій. Зазначена модель має слугувати засобом уніфікації (шаблоном, прототипом) подання вихідних даних для формального методу перевірки на моделі TLC, а також для розробленого і викладеного у четвертому розділі дисертаційної роботи розвитку цього методу.

Розв’язання цієї задачі виконується у двох площинах – на аналітичному рівні і на рівні реалізації ФС. На аналітичному рівні фігурують конструкції, якими розробник оперує у процесі аналізу артефакту. У свою чергу, на рівні реалізації мають місце конструкції, що уможливлюють автоматизацію процесу ФВ ФС методом перевірки на моделі TLC і згаданим вище розвитком зазначеного методу.

2.3 Викладення підходу до вирішення задачі

2.3.1 Застосовувані концепти та припущення

Структуру Кріпке (1.2) розглядатимемо як засіб інтерпретації вихідних даних на аналітичному рівні. За відповідні дані візьмемо блок-схему алгоритму роботи СКП.

Застосуємо алгоритмічну мову PlusCal для створення прототипу цільової ФС на мові TLA+. Назвемо такий прототип моделлю ФС (МФС) – метамоделлю. Її призначення – подання архітектурної складової вихідних даних у формі псевдокоду як прообразу цільової ФС.

Такий підхід має на меті спростити процес сприйняття і аналізу архітектурної складової ФС розробником – за рахунок того, що у PlusCal-поданні ФС не враховуються наступні складові результуючої ФС на мові TLA+:

– типи даних змінних;

– формалізації подій, що обумовлюють оновлення значень змінних.

Отже, МФС, по суті, є псевдокодом.

Застосування МФС має на меті знизити вплив людського фактору на процес синтезу ФС на основі заданих вихідних даних. Структурно МФС містить наступні складові [85]: блок визначення змінних станів і блок опису алгоритму на основі цих змінних. Останній виокремлюється наступними ключовими словами: «begin» – на початку, «end algorithm» – у кінці.

Концептуальне подання застосовуваного підходу до синтезу ФС представлено на рис. 2.2.

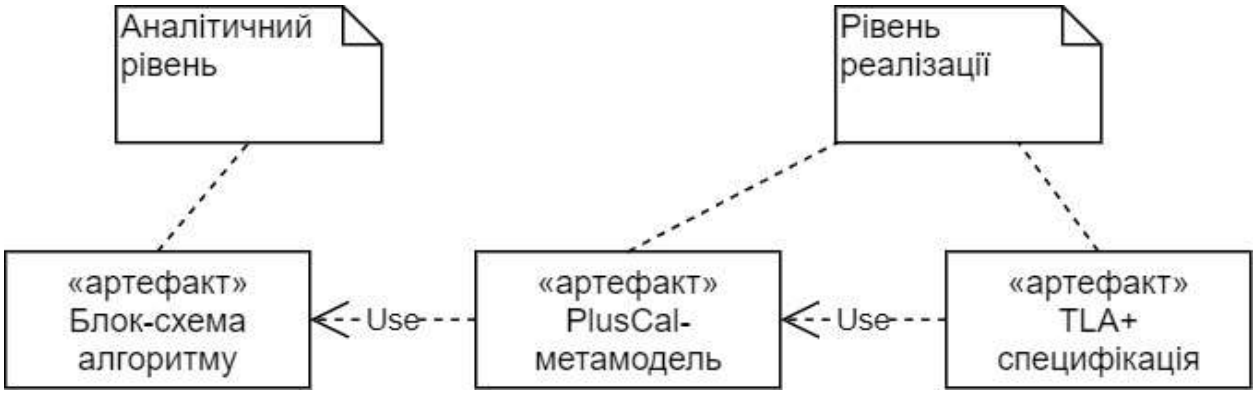

Рисунок 2.2 – Схема синтезу цільової ФС

На рис. 2.2 результуюча специфікація TLA+ позиціонується у якості вихідних даних для роботи методу перевірки на моделі.

Для зміщення з аналітичного рівня до рівня реалізації необхідно забезпечити прозорий і однозначний механізм синтезу конструкцій TLA на основі конструкцій СП, поданих засобами структури (1.2). Для цього у якості

допоміжного засобу застосуємо числення послідовних процесів, що взаємодіють (CSP, Communicating Sequential Processes) Ч. Гоара (рис. 2.3) [51].

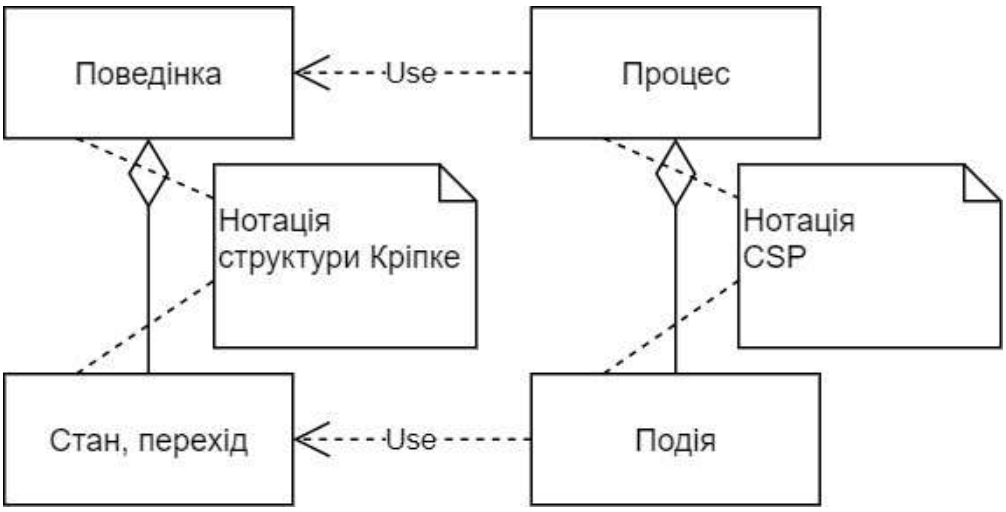

Рисунок 2.3 – Співвідношення між концептами структури Кріпке і числення CSP

Застосування CSP призначено надати виразні засоби подання складових ФС рівня реалізації, виходячи із засобів структури (1.2), якими оперуємо на аналітичному рівні. Більше того, основоположне поняття «подія» формалізму CSP призначене застосовуватися у якості прототипу складових подання ключового поняття «дії» логіки TLA.

На рис. 2.3 пустою ромбовидною стрілкою подано UML-відношення агрегування між концептами структури (1.2) і формалізму CSP. Застосування відношення агрегування, замість відношення композиції, означає, що відповідні базові елементи – стани, переходи, події – можуть фігурувати у якості складових різних поведінок, процесів.

Комплексне подання розробленого підходу до оперування озвученими засобами наведено на рис. 2.4 [126].

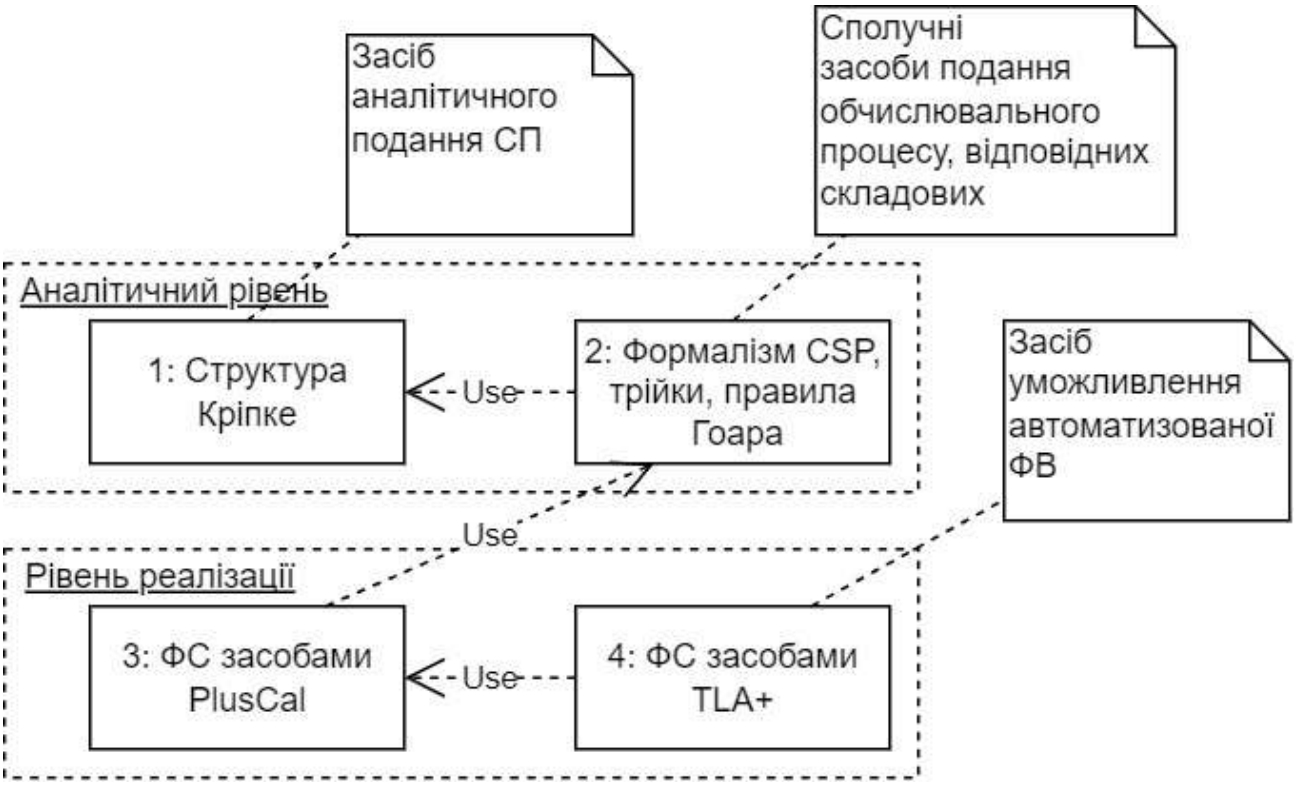

Рисунок 2.4 – Концептуальна інтерпретація застосовуваного підходу

На рис. 2.4 виокремлено два рівні сприйняття ФС – аналітичний рівень і рівень реалізації. Аналітичний рівень – площина подання концептів, якими оперує розробник при аналізі та проєктуванні ФС. Рівень реалізації – сфера оперування концептами, що застосовуються при реалізації ФС на основі формалізму (TLA+), придатному до автоматизованого застосування методу перевірки на моделі по відношенню до цієї реалізації. При цьому на кожному із рівнів фігурують по два типи артефактів, представлені відповідними прямокутниками:

– структура Кріпке;

– подання структури на основі формалізму CSP і трійок Гоара;

– МФС на основі формалізму PlusCal;

– результуюча ФС на основі формалізму TLA+.

Чотири зазначені типи артефактів (пронумеровані на рис. 2.4) обумовлюють потребу побудови трьох засобів поступального синтезу результуючого артефакту (ФС на мові TLA+) на основі вихідного артефакту, поданого на аналітичному рівні структурою (1.2). Відповідні засоби позначено пунктирними стрілками, дві з яких позначають перетворення у межах виокремлених рівнів, одна – між рівнями.

Отже, модель, представлена у межах даного розділу, має два рівні інтерпретації і чотири форми подання – блоки 1, 2,…, 4 (рис. 2.4). При цьому пунктирні стрілки, що з'єднують названі блоки, відображають послідовні кроки розробленого методу, викладеного у межах наступного розділу.

2.3.2 Формалізація архітектурної складової

Формалізуємо подання ФС для кожного із чотирьох виокремлених типів артефактів (рис. 2.4). Для поєднання зазначених формалізацій застосуємо теоретико-множинний підхід:

$$\langle A,T\rangle, \tag{2.1}$$

де $A=\{a_1,a_2,a_3,a_4\}$ – множина виокремлених типів артефактів, де кожен тип $a_i \in A(i=1,2,..,4)$ визначається відповідною формальною системою, яка при цьому залучається: $a_1 \in A$ – структурою (1.2), $a_2 \in A$ – засобами CSP, $a_3 \in A$ – засобами PlusCal, $a_4 \in A$ – засобами TLA+; $T \subset A^2$ – множина переходів між артефактами: $T=\{(a_1,a_2),(a_2,a_3),(a_3,a_4)\}$, а саме: $a_2=T(a_1)$, $a_3=T(a_2)=T(T(a_1))$, $a_4=T(a_3)=T(T(a_2))=T(T(T(a_1)))$. Для $i=3,4$ відповідний артефакт $a_i \in A$ формалізується як композиція функцій.

Відношення T, у свою чергу, є засобом подання кроків методу синтезу цільової ФС, викладеного у наступному розділі. При цьому вищенаведені вирази – $a_3 = T(a_2) = T(T(a_1))$ і $a_4 = T(a_3) = T(T(a_2)) = T(T(T(a_1)))$ із поданням артефактів як композицій функцій – демонструють опосередковані зв'язки між виокремленими типами артефактів (рис. 2.4).

Аргументом на користь поєднання артефактів на основі формалізмів CSP [51] і TLA+ як складових однієї послідовності (рис. 2.4) є твердження проф. Валентина Миколайовича Томашевського, що процес можна розглядати як послідовність взаємопов'язаних дій – за умови визначення початку і завершення дії [127, с. 32]. Відповідно до цього твердження, ключове поняття темпоральної логіки TLA – «дія» – зручно розглядати у якості виразного засобу реалізації ключового поняття формалізму CSP – «протокол процесу».

На рис. 2.4 формалізм CSP, трійки Гоара [62] і алгоритмічна мова PlusCal [86] застосовуються для зниження ефекту концептуального розмежування між формальною системою структури Кріпке (1.2) – аналітичною інтерпретацією ФС – і реалізацією ФС на основі формалізму TLA+. Останній при цьому розглядається у якості інструменту – засобу реалізації ФС у форматі, придатному до автоматизованої ФВ методом МС. Специфікація на основі алгоритмічної мови PlusCal застосовується у якості допоміжного засобу – проміжного кроку, що передує синтезу цільової TLA+ специфікації. Цей крок має на меті полегшити сприйняття і аналіз ФС розробником.

Ідея в основі підходу, викладеного на рис. 2.1 – рис. 2.4, базується на ієрархічному поданні системи [130]. Згідно цього, на аналітичному рівні рис. 2.4 нижній ієрархічний рівень представлений станами, переходами, подіями; верхній – поведінками і процесами [131–133].

Ієрархічність розглядається з позиції ступеню деталізації подання артефакту: чим вищим є ступінь деталізації, тим нижчим є ієрархічний рівень.

Підсумуємо вищесказане у формі наступних припущень згідно (2.1):

– структура Кріпке (1.2) розглядається як засіб подання ФС на рівні аналітичного сприйняття ФС розробником – артефакт $a_1 \in A$;

– CSP-подання є допоміжним засобом – засобом концептуального наближення СП, формалізованої складовими структури (1.2), до понятійного апарату, що має місце на рівні реалізації – артефакт $a_2 = T(a_1)$;

– МФС на основі PlusCal – прототип цільової ФС з точки зору архітектурної складової – артефакт $a_3 = T(a_2) = T(T(a_1))$;

– ФС на основі TLA+ – результуюча ФС, що уможливлює застосування методу МС в автоматизованому режимі – артефакт $a_4 = T(a_3) = T(T(a_2)) = T(T(T(a_1)))$.

Викладення змісту елементів структури (2.1) подано нижче.

2.4 Формалізація рівнів подання моделі

Викладення представленої моделі полягає у формалізації та роз'ясненні подань ФХ СКП, що фігурують як артефакти рис. 2.4, – елементів множини A (2.1). У свою чергу, елементи множини T демонструють кроки методу синтезу ФС ФХ СКП, викладеного у наступному розділі.

2.4.1 Аналітичне подання функціональних характеристик

2.4.1.1 Застосування засобів структури Кріпке

У процесі ФВ ФС будується СП, яку зручно аналітично подавати засобами структури Кріпке (1.2). У випадку, якщо ФС містить умовні переходи, матимемо множину шляхів, що беруть початок із деякого стану $s_0 \in S$, і

завершуються певним заключним станом. Таку множину подамо як множину відповідних «поведінок»:

$$B = \{b_i\},\ i = 1,2,\ldots,m \in N\ , \tag{2.2}$$

де $b_i \in B$ – i-та поведінка СП як скінченна послідовність станів – елементів множини S (1.2):

$$b_i = s_0, s_1, \ldots, s_f, \ldots, s_{l \in N}\ , \tag{2.3}$$

де $s_0 \in S$ – початковий стан СП, на основі якого, шляхом застосування відношення R (1.2), одержуємо наступні елементи послідовності: $s_1 = R(s_0)$, $s_2 = R(s_1) = R(R(s_0))$ і т. д., $0 \le f \le l = (n \cdot p) - 1$: n – число змінних станів СП, p – число допустимих значень змінних станів СП. Таким чином, покроково, досягається заключний елемент послідовності – $s_l \in S:\ R(s_l) = s_l$.

Отже, множина B (2.2), у межах якої об’єднано елементи, формалізовані згідно виразу (2.3), є аналітичним поданням виокремленого згідно розробленої моделі (2.1) типу артефактів $a_1 \in A$.

2.4.1.2 Застосування засобів числення процесів

Наступний крок полягає у тому, щоб розглядати сусідні елементи послідовностей $b_i \in B$ попарно. Це виконано для реалізації переходу $(a_1, a_2) \in T$ (2.1) – від використання виразних засобів структури Кріпке (1.2) – до оперування виразними засобами числення послідовних процесів, що взаємодіють – CSP.

Зазначений крок необхідний для одержання – у режимі «один-до-одного» – конструкцій аналітичного рівня, відповідних конструкціям, що будуть сформовані на рівні реалізації (рис. 2.4). Для цього виконуються наступні дії:

– елементу $(s,s')\in R$ структури (1.2) ставиться у відповідність поняття «подія» числення CSP;

– поведінці (2.3) як послідовності станів – поняття «протокол» (обчислювального процесу) як послідовність подій.

Формалізуємо процес перетворення елементів множини B (2.2) у відповідні їм елементи результуючої множини протоколів обчислювальних процесів у формі наступного відношення:

$$\delta_1 : B \to P, \tag{2.4}$$

де $P=\{p_i\}$ – множина протоколів процесів: $|B|=|P|$, $\delta_1(b_i)=p_i$, де $b_i \in B$, $p_i \in P$:

$$p_i = \langle e_1, e_2, ..., e_f, ..., e_{l\in N} \rangle \in P, \tag{2.5}$$

де $p_i \in P$ – протокол *i*-го обчислювального процесу, прообразом якого є $b_i \in B$ (2.3); e_f – f-та подія, передумовою виникнення якої є стан $s_{f-1} \in S\ (f=1,2,..,l)$ структури (1.2), пост-умовою – стан $R(s_{f-1})=s_f \in S$. При цьому відносний порядок виникнення подій задається шляхом застосування відношення R.

Отже, шляхом застосування відношення δ_1 (2.4), здійснюємо перехід, у межах аналітичного рівня розробленої моделі подання ПАС СКП (рис. 2.4), від оперування послідовностями станів (2.3) на основі засобів структури (1.2) – до

оперування послідовностями подій (2.5) на основі засобів формалізму CSP. У результаті одержуватимемо артефакти типу $a_2 \in A$ на основі артефактів типу $a_1 \in A$ – (2.1), рис. 2.4.

Для виконання переходу $(a_1, a_2) \in T$ (2.1) у якості першого кроку розробленого методу синтезу ФС, викладеного більш розгорнуто у третьому розділі, запропоновано правила одержання складових результуючого CSP-подання (типу $a_2 \in A$) на основі відповідних складових вихідного подання (типу $a_1 \in A$) на основі засобів структури (1.2). Правила зведено у табличній формі (табл. 2.1).

Таблиця 2.1 – Співвідношення між складовими артефактів аналітичного рівня (рис. 2.4)

Страти	Артефакти та правила синтезу		
	Структура Кріпке ($a_1 \in A$)	Перехід $(a_1, a_2) \in T$ (2.1)	Формалізм CSP ($a_2 \in A$)
1	2	3	4
1	Поведінка (2.3).	δ_1 (2.4).	Протокол процесу (2.5).
2	Перехід СП $(s_{f-1}, s_f) \in R$ (1.2).	$\delta_2 : (s_{f-1}, s_f) \mapsto e_f$.	Подія e_f.

У табл. 2.1 кожен рядок подає відповідний ієрархічний рівень опрацювання артефактів залучених типів $a_1, a_2 \in A$ (2.1): перший – верхню страту; другий – нижню страту.

На першому рівні у якості засобу перетворення залучається вираз (2.4), на другому – функція δ_2, згідно якої парі суміжних станів СП ставиться у відповідність подія, що фігурує у межах конструкцій CSP.

У табл. 2.1 другий стовпець містить вихідні конструкції суміжних ієрархічних рівнів для артефактів типу $a_1 \in A$; третій стовпець – засоби перетворення конструкцій типу $a_1 \in A$ у відповідні конструкції типу $a_2 \in A$; четвертий стовпець – результати застосування озвучених засобів.

У свою чергу, одержуване CSP-подання як результат застосування запропонованих правил, зведених у табл. 2.1, адресується у якості граничної ланки (рис. 2.4) між зазначеними типами артефактів виокремлених аналітичного рівня і рівня реалізації.

У свою чергу, виокремлені два ієрархічних рівні у межах типів артефактів призначені для охоплення кожної із складових поданого вище визначення поняття «артефакт» – архітектури і змісту: верхня страта охоплює архітектурну складову – шляхом залучення виразу (2.4); нижня – зміст.

Одержання артефактів типу $a_2 \in A$ на основі засобів CSP надає можливість оперувати на аналітичному рівні у режимі «один-до-одного» конструкціями-аналогами таких, що застосовуються на рівні реалізації. Такий крок призначений сприяти адекватності відтворення аналітичних подань у формі відповідних ФС, до яких в автоматизованому режимі застосовується метод перевірки на моделі.

Варто, однак, зазначити, що механізм сполучення подій у табл. 2.1 не охоплено. Для урахування цього аспекту залучаються «трійки Гоара» [62]:

$$\{\varphi\} e_f \{\xi\}, \tag{2.6}$$

де φ – передумова виникнення події e_f як складова результуючої ФС – кон'юнкція на основі елементів множини $L(s_{f-1}) \subset AP$ (1.2) [134]; ξ – пост-умова – кон'юнкція на основі елементів множини $L(R(s_{f-1})) = L(s_f) \subset AP$.

Вираз (2.6) означає, що, у випадку істинності передумови φ, у результаті виникнення події e_f, пост-умова ξ також прийматиме істинне значення.

У свою чергу, передумову у складі виразу (2.6) подамо наступним чином:

$$\varphi = ap \wedge \varphi', \tag{2.7}$$

де $ap \in L(s_{f-1}) \subset AP$ – атомарне висловлювання, сформоване із залученням певної змінної стану СП, значення якої модифікується у результаті виникнення події e_f; φ' – складова виразу-передумови, що лишається незмінюваною у результаті виникнення події e_f.

Подання передумови у складі виразу (2.6) у формі виразу (2.7) має на меті виокремити однакові і відмінні складові перед- і пост-умов.

Підхід-декомпозицію, застосований для формування виразу (2.7), використаємо, за аналогією, і при поданні пост-умови: $\xi = ap' \wedge \varphi'$, де $ap' \neq ap$, а саме – $ap' \in L(s_f)$, де $s_f = R(s_{f-1})$: $L(s_{f-1}) \Delta L(s_f) = \{ap, ap'\}$. Застосування відношення «симетрична різниця» є засобом виокремлення підмножини атомарних висловлювань, що є поданням відмінності пост-умови від передумови.

Підсумовуючи вищезазначене, вираз (2.6) розширює дані табл. 2.1, а саме – нижній рядок (нижню страту) – у розрізі сполучення подій – шляхом залучення трійок Гоара – на основі перед- і пост-умов.

Адресуючи аспект просторових витрат стосовно збереження ФС в ОП обчислювальної системи, у межах розробленої моделі подання ПАС СКП залучено правило композиції логіки Гоара, де у чисельнику фігурують трійки Гоара, по відношенню до яких застосовується назване правило, а у знаменнику – результат застосування правила [135]:

$$\frac{\{\varphi_0\}e_1\{\varphi_1\},\{\varphi_1\}e_2\{\varphi_2\},\ldots,\{\varphi_{l-1}\}e_l\{\varphi_l\}}{\{\varphi_0\}e_1;e_2;\ldots;e_l\{\varphi_l\}}, \qquad (2.8)$$

де φ_0 – початкова умова, що є, у тому числі, передумовою виникнення початкової події e_1 (чисельник виразу), а також передумовою ланцюга подій – результату застосування правила композиції (знаменник виразу).

При цьому φ_0 є ФС початкового стану $s_0 \in S$, розміченого як $L(s_0)$, φ_1 – ФС наступного стану $R(s_0) = s_1 \in S$, розміченого як $L(R(s_0))$, і т. д. Більше того, $(l-1)$ умов $\varphi_f\,(f = 1,2,\ldots,l-1)$ мають подвійну інтерпретацію: φ_f є пост-умовою для події e_f і передумовою – для наступної події e_{f+1}, де f – порядковий номер події. У чисельнику виразу (2.8) маємо $(l+1)$ перед- і пост-умов, при цьому $(l-1)$ ФС є водночас і перед- і пост-умовами для «сусідніх» трійок.

Чисельник виразу (2.8) представляє ФС фрагментовано – до застосування правила композиції, знаменник – після. Ключова ідея – застосування правила композиції у якості засобу компактизації ФС. При цьому знаменник містить протокол процесу (2.5), доповнений передумовою виникнення початкової події e_1 і пост-умовою заключної події e_l.

У контексті чисельника засобом «зв'язування» подій виступають відповідні перед- і пост-умови, у контексті знаменника – безпосередньо події. У цьому є наукова новизна представленої моделі – застосовувати у якості передумови виникнення наступної події не пост-умову попередньої події, а безпосередньо саму попередню подію. За рахунок цього досягається зниження числа рядків (компактизація) результуючої ФС. Іншими словами – знаменник виразу (2.8) є поданням протоколу процесу (2.5), доповненого початковою

перед- і заключною пост-умовами. У свою чергу, правило композиції є засобом одержання такого подання.

Більше того, правило (2.8) є засобом стратифікації ФС ФХ СКП, із застосуванням якого синтезуються елементи верхньої страти табл. 2.1 – на основі елементів нижньої страти. У свою чергу, вищезгадані умови φ_0 і φ_l, що фігурують у знаменнику виразу (2.8), є засобами групування m протоколів процесів для випадку m поведінок (2.2).

Отже, знаменник виразу (2.8) є прототипом ФС ПАС, що буде реалізована засобами TLA+. Концептуально, конструкція (2.8) формалізує поняття «силогізму», згідно якого із істинності елементів чисельника слідує істинність виразу у знаменнику.

Вираз (2.8) слід розуміти наступним чином: якщо кожна із трійок чисельника приймає істинне значення, то і вираз у знаменнику також приймає істинне значення. При цьому запис $e_1; e_2; \ldots; e_l$ означає таке:

$$e_1 \prec e_2 \prec \ldots \prec e_l, \tag{2.9}$$

де $\prec$ – оператор передування. Це означає, що події e_{f+1} передує подія e_f, і т.д., тобто події відбуваються послідовно.

Вираз (2.9) можна подати альтернативним чином – із застосуванням темпорального оператору X (neXt): $e_1 \wedge Xe_2 \wedge X^2 e_3 \ldots \prec X^{l-1} e_l$, де верхній індекс оператору X ідентифікує кількість застосувань оператору – кількість послідовних зміщень модельного часу відносно моменту виникнення початкової події.

Згідно (2.8), якщо розглядати виникнення події e_f як передумову виникнення наступної події e_{f+1}, і кожну подію при цьому формалізувати як функцію, вираз (2.9) можна подати як композицію функцій:

$$e_l \circ e_{l-1} \circ \ldots \circ e_1, \tag{2.10}$$

де e_l – завершальний виклик функції.

Вираз (2.10), у свою чергу, застосуємо як прообраз подання ФС ФХ СКП засобами алгоритмічної мови PlusCal.

Отже, застосування правила (2.8) слугує наступним аспектам:

– компактизація результуючої ФС на мові TLA+ для спрощення її сприйняття, аналізу і, як результат, – зниження впливу людського фактору, що є інструментом сприяння достовірності даних, одержуваних у результаті ФВ методом перевірки на моделі [136, 137];

– стратифікація складових результуючої TLA+ специфікації, що створює зручний механізм як варіювання рівня деталізації такої ФС, так і внесення змін до останньої.

Реалізації вищеназваних аспектів слугують також і наступні правила, а саме – правило виведення і правило умовного оператору. Перше базується на застосуванні операції імплікації:

$$\frac{\varphi^* \to \varphi, \{\varphi\} e_f \{\xi\}, \xi \to \xi^*}{\{\varphi^*\} e_f \{\xi^*\}}, \tag{2.11}$$

де $\varphi^* \to \varphi$, $\xi \to \xi^*$ – імплікативні вирази.

Сполучивши вирази (2.8) і (2.11), маємо наступне складене імплікативно-композиційне правило:

$$\frac{\varphi_0 \to \varphi_k, \{\varphi_k\} e_{k+1} \{\varphi_{k+1}\}, \ldots, \{\varphi_{l-x-1}\} e_{l-z} \{\varphi_{l-z}\}, \varphi_{l-z} \to \varphi_l}{\{\varphi_0\} e_{k+1}; \ldots; e_{l-z} \{\varphi_l\}}, \tag{2.12}$$

де $k < (l - z - 1)$.

Конструкція (2.12) призначена для виокремлення варіативної складової заданого протоколу процесу $p_i \in P$ з-поміж $m-1$ альтернативних протоколів $p_h \in P(h \neq i)$, з метою уникнення дублювання однакових підпослідовностей подій у ФС.

У залежності від розмірності вихідних даних – блок-схеми алгоритму із заданим числом блоків і змінних, з урахуванням експоненційного характеру зростання простору станів СП від числа змінних станів, а також обчислювальних можливостей доступної програмно-апаратної платформи, виникає потреба варіювання рівня деталізації ФС. Разом із цим виникає потреба опрацювання механізмів інтерпретації вищезгаданої події e_j.

Вибір рівня деталізації ФС пропонується здійснювати, у тому числі, з урахуванням нижченаведених позицій:

– оціночного значення часових витрат, супутніх процедурі ФВ ФС, з урахуванням обчислювальних можливостей доступної програмно-апаратної платформи; для цього пропонується використовувати штучно синтезовані ФС, викладені у наступному розділі;

– очікуваного корисного ефекту від виявлення помилок у артефактах або підтвердження відсутності останніх.

Урахування вищеназваних позицій пропонується досягати шляхом встановлення «відсоткового порогу» для таблиці на кшталт табл. 1.1, перевищення якого є умовою потрапляння відповідної складової вихідних даних до складу змінних ФС.

Як приклад, у якості порогового можна обрати значення 10 %. За потреби підвищення рівня деталізації ФС це значення пропонується зменшувати, і навпаки.

Далі деталізуємо $p_i \in P$ (2.5) поелементно. Для цього сформуємо множину подій, на основі яких будуються протоколи процесів:

$$\alpha P = \bigcup_{i=1}^{m} \alpha p_i \,. \tag{2.13}$$

де $\alpha p_i \subseteq \alpha P$ – множина подій, що формують *i*-й процес.

Скористаємось даними табл. 1.1, виокремивши типи операцій із відсотковими частками ≥ 10 %. Ці типи подамо відповідною множиною:

$$T^* = \{t_1, t_2, t_3, t_4\} \subseteq T \,, \tag{2.14}$$

де $t_1 \in T^*$ – операція завантаження операнду; $t_2 \in T^*$ – операція складання; $t_3 \in T^*$ – операція занесення до пам'яті; $t_4 \in T^*$ – операція переходу (умовного/безумовного); T – множина усіх типів операцій, поданих у табл. 1.1. При цьому, для окреслення специфіки предметної області, доречно оперувати поняттям «операційного спектру», що, у контексті космічної галузі, представлений табл. 1.1.

Склад елементів підмножини $T^* \subseteq T$ (2.14) може бути, за потреби, розширений.

Виокремлення підмножини $T^* \subseteq T$ має на меті досягти наступного:

– спрощення процедури автоматизованого синтезу ФС – шляхом зниження рівня деталізації ФС;

– як результат попереднього кроку – зниження просторових та обчислювальних витрат, обумовлених верифікацією ФС, синтезованої згідно представленої моделі, методом МС.

Множину αP (2.13) сформуємо згідно виокремлених типів складових вихідних даних (2.14): змінні вихідних даних (блок-схеми алгоритму, UML-діаграми дій), що не потрапили до операційного спектру, не враховуємо у якості змінних станів СП.

2.4.2 Формалізація на рівні реалізації

Наступний крок – формалізація безпосередньо подій – елементів множини αP (2.13).

Для цього виконується наступне [138, 139]:

– формується множина змінних станів СП, представленої структурою Кріпке (1.2);

– формується множина значень змінних станів;

– формується множина атомарних висловлювань AP для структури (1.2);

– події подаються згідно виразних можливостей формалізму TLA+, що дасть змогу співставит один до одного названі сутності аналітичного рівня із відповідними елементарними конструкціями рівня реалізації (рис. 2.4).

Множину змінних станів подамо наступним чином:

$$V = \{v_j\},\ j = 1,2,\ldots,n \in N, \tag{2.15}$$

де $v_j \in V$ – змінна станів СП: $\chi(v) = t \in T^*$ (2.14), де χ – функція визначення типу операції (у загальному випадку – фрагменту вихідних даних – складової блок-схеми алгоритму), тобто функція χ є засобом визначення типу заданої змінної $v_j \in V$.

Множина допустимих значень елементів $v \in V$ має наступний вигляд:

$$D = \bigcup_{j=1}^{n} D_j = \{d_k\}, k = 1,2,\ldots, p \in N, \quad (2.16)$$

де $\forall v_j \in V \; \exists D_j \subseteq D: \; s(v_j) = d \in D_j$, де $s \in S$ (1.2).

На основі множин (2.15) і (2.16) сформуємо множину атомарних висловлювань AP структури (1.2) як декартовий добуток, елементи якої використовуватимемо у якості елементарних (атомарних) конструкцій для формування елементів нижнього ієрархічного рівня для артефакту $a_4 \in A$ – рис. 2.4, (2.1) [139]:

$$AP = V \times D, \quad (2.17)$$

де для кожної змінної $v_j \in V$ маємо $(\{v_j\} \times D_j) \subseteq AP$ атомарних висловлювань.

Елементи множини AP застосуємо у якості складових для побудови ФС розміток станів $L(s)$ для заданих $s \in S$ на рівні реалізації, а також для формалізації подій, що обумовлюють переходи $(s, s') \in R$ (1.2).

2.4.2.1 Формалізація подій

До кожного елементу послідовності (2.3) застосуємо функцію розмітки станів L (1.2). Матимемо подання поведінки як траєкторії [17]: $L(b_i) = L(s_0), L(s_1), \ldots, L(s_l)$.

Інтерпретацію вказаної траєкторії – для формалізації подій – виконуватимемо згідно парадигми структурного програмування [140]: виокремлюються структури керування виконанням програми трьох типів – послідовність, розгалуження, цикл. При цьому виконання програми

відбувається покроково – зверху вниз. У даному контексті траєкторію $L(b_i)$ розглядаємо як один із можливих сценаріїв виконання програми.

Розглянемо подію як функцію, і подамо її як відношення наступним чином – на основі елементів траєкторії:

$$e_f : L(s) \setminus L(s') \to L(s') \setminus L(s), \tag{2.18}$$

де $L(s) \setminus (s') = \{(v_j, d_k)\} = dom(e_f)$, $L(s') \setminus (s) = \{(v_j, d_h)\} = ran(e_f)$, де $d_h, d_k \in D, d_h \neq d_k$, $dom(e_f)$ – область визначення e_f, $ran(e_f)$ – область значень e_f. Іншими словами, маємо наступне: $e_f : (v_j, d_k) \mapsto (v_j, d_h)$.

Аналізуючи вираз (2.8), варто відзначити дуальний характер перед- і постумов:

– для $f = 2,3,\ldots,l$ справедливо наступне: $dom(e_f) = ran(e_{f-1})$.

Саме на врахуванні цього аспекту і будується метод, викладений у наступному розділі.

Ідея в тому, щоб застосовувати у якості передумови виникнення події e_f не ФС розмітки стану, а безпосередньо ФС події, що їй передує. За рахунок цього і планується компактизувати результуючу ФС.

Це твердження справедливе для $f = 2,3,\ldots,l$. Відповідний результат демонструється знаменником виразу (2.8). Виключення становить початкова подія e_1, що ініціює обчислювальний процес, і передумовою виникнення якої є розмітка початкового стану $L(s_0)$.

Подію формалізуємо як імплікацію, модифіковану темпоральним оператором X (Next) [141]:

$$e_j \equiv ((v_j, d_k) \to X(v_j, d_h)) \equiv (\neg(v_j, d_k) \vee X(v_j, d_h)), \tag{2.19}$$

де вже застосовуємо індекс j, (а не f), тобто асоціюємо подію не з постумовою, а із відповідною змінною станів $v_j \in V$: $(v_j, d_k), (v_j, d_h) \in (V \times D_j) \subseteq AP$, де $d_k, d_h \in D, d_k \neq d_h$. При цьому вираз $(|D_j| - 1)$ дозволяє отримати для кожної e_j кількість породжуваних на її основі проявів події (із числа l проявів), що фіксуються протоколом (2.5) вже із акцентом на відносний порядок їх виникнення f.

При цьому варто зазначити, що атомарне висловлювання $(v_j, d_k) \in L(s_{f-1}) \subset AP$, де $L(s_{f-1})$ – розмітка стану $s_{f-1} \in S$ СП, що передує виникненню події e_j, а саме – її прояву e_f, що фігурує, зокрема, у протоколі (2.5), знаменнику правила композиції (2.8). У свою чергу, $(v_j, d_h) \in L(s_f) \subset AP$, де $L(s_f) = L(R(s_{f-1}))$ – розмітка наступного стану $s_f = R(s_{f-1}) \in S$ СП: $L(s_f) \Delta L(s_{f-1}) = \{(v_j, d_k), (v_j, d_h)\}$.

У виразі (2.19) і далі оператор $\equiv$ – оператор еквівалентності: ліва і права частини виразу формалізують ту саму сутність, тобто запис є тавтологією.

Вираз (2.19) означає, що у деякий поточний момент модельного часу значення змінної $v_j \in V$ становить d_k, а у наступний – d_h, про що свідчить оператор X. При цьому дію темпорального оператору X можна пояснити наступним чином:

$$(M, s_{f-1} \models X(v_j, d_h)) \equiv (M, s_f \models (v_j, d_h)), \qquad (2.20)$$

де вираз є тавтологією: істинність темпоральної формули $X(v_j, d_h)$ по відношенню до стану $s_{f-1} \in S$ СП, заданої структурою M (1.2), еквівалентна

істинності атомарного висловлювання (v_j, d_h) по відношенню до наступного стану $R(s_{f-1}) = s_f \in S$ СП.

2.4.2.2 Формалізація станів

Якщо перейти до нотації структури M (1.2), що будується на основі елементів множини AP (2.17), останню можна подати наступним чином:

$$AP \supseteq \bigcup_{f=0}^{l} L(s_f), \tag{2.21}$$

тобто маємо $l+1$ розміток станів СП на основі n змінних станів (2.15). У даному контексті розмітку $L(s_f)$ варто розглядати у якості передумови здійснення переходу $R(s_f)$, а розмітку $L(R(s_f))$ – у якості пост-умови. Отже, перехід $(s_f, R(s_f)) \in R$ структури (1.2) є прообразом події (2.19), а відповідні розмітки $L(s_f)$ і $L(R(s_f))$ – вихідними даними для синтезу φ і ξ відповідно (2.6), де φ і ξ – кон'юнкції на основі елементів множин $L(s_f)$ і $L(R(s_f))$ відповідно.

У якості прикладу розглянемо випадок, коли вихідним артефактом для синтезу ФС є блок-схема алгоритму. Тоді множину AP можна подати наступним чином: $AP = V \times D^*$, де $D^* = \{0,1\} \subseteq D$, елементи якої є поданнями булевих значень – «false» і «true», де $(v_j, 0) \in AP$ означає, що дії j-го блоку блок-схеми ще не було виконано; $(v_j, 1) \in AP$ – вже було виконано.

При цьому множину AP (2.21) доречно розглядати з позиції дихотомії:

$$AP = AP' \cup AP'' : AP' \cap AP'' = \varnothing, \tag{2.22}$$

де $AP' = V \times \{0\}$, а $AP'' = V \times \{1\}$. У даному контексті доречно зауважити, що $AP' = L(s_0)$, тобто операції жодного із блоків блок-схеми ще не було виконано.

Для запуску процесу автоматизованої перевірки ФС методом МС, у якості початкової точки відліку подамо елементи множини $L(s_0)$ як кон'юнкцію:

$$\varphi_0 \equiv (v_1,0) \wedge (v_2,0) \wedge \ldots \wedge (v_n,0), \tag{2.23}$$

Передумови на основі елементів $L(s)$, $\forall s \in S$ формалізуються аналогічним чином.

Вираз (2.23) можна узагальнити наступним чином:

$$\varphi \equiv ap_1 \wedge ap_2 \wedge \ldots \wedge ap_n, \tag{2.24}$$

де $ap_j \in AP$, причому $|AP| = 2 \cdot n$, оскільки $|AP'| = |AP''| = n$ і має місце умова (2.22).

Через властивість транзитивності правила (2.8), застосуємо відповідний знаменник у якості прототипу для подання елементів верхньої страти (табл. 2.1) на рівні реалізації – на основі виразних засобів PlusCal і TLA+ (рис. 2.4). Його семантичне навантаження у повній мірі відображається композицією (2.10). Даний крок позбавляє потреби безпосередньо задавати розмітки станів $s \in S \setminus \{s_0\}$ у ФС, що має на меті знизити число рядків результуючої ФС. Це, у свою чергу, призначено полегшити сприйняття ФС розробником, що можна охарактеризувати як позитивний чинник з позиції зменшення впливу людського фактору на вихідні дані для ФВ ФС методом МС.

Кількісне оцінювання корисного ефекту від застосування правила композиції (2.8) на основі трійок Гоара (2.6) здійснюється комплексно – згідно послідовного [83] і паралельного [22] шаблонів синтезу ФС. При цьому паралелізм подано згідно моделі чергування. Параметри синтезованих ФС для кожного із названих випадків подано таблично – у наступному розділі – у залежності від числа змінних станів (табл. 3.1, табл. 3.2).

ВИСНОВКИ ДО РОЗДІЛУ 2

Таким чином, у розділі викладено розроблений підхід, призначений слугувати у якості комплексного засобу сполучення при проєктуванні ПАС розроблених і винесених на захист методів, розвитку методу і моделей – для контролю досліджуваних артефактів за показниками і ФХ, і НФХ.

У розділі також представлено запропоновану стратифіковану модель подання розроблюваної ПАС на етапі проєктування процесу розроблення.

Модель призначена слугувати засобом уніфікації ФС, одержуваних на основі виразних засобів TLA+. Зазначені ФС, у свою чергу – артефакти, по відношенню до яких безпосередньо застосовується метод перевірки на моделі TLC, а також розроблений і викладений у четвертому розділі розвиток вказаного методу.

Розроблену модель викладено згідно теоретико-множинного підходу. Разом із цим модель реалізовано згідно дуального підходу до охоплення аспектів подання ПАС у формі ФС: у поданні артефактів виокремлено аналітичний рівень і рівень реалізації. Перший зазначений рівень призначено до опрацювання розробником аналітично – із залученням виразних засобів структури Кріпке і числення процесів CSP. Другий рівень – для автоматизованого машинного опрацювання – із застосуванням засобів алгоритмічної мови PlusCal і формалізму TLA+. Застосування засобів PlusCal призначено для попереднього формування архітектурної складової ФС.

Серед особливостей розробленої моделі – залучення трійок і правил Гоара, у тому числі – правила композиції. Застосування останнього дозволило скоротити кількість рядків коду результуючих ФС на основі засобів TLA+, що було підтверджено експериментально для граничних випадків у наступному – третьому – розділі, де викладено розроблений метод синтезу ФС.

РОЗДІЛ 3
РОЗРОБЛЕННЯ МЕТОДУ СИНТЕЗУ ФОРМАЛЬНИХ СПЕЦИФІКАЦІЙ

У розділі подається розроблений метод синтезу ФС ПАС згідно моделі, викладеної у попередньому – другому – розділі. Метод призначений слугувати засобом автоматизації процесу постачання вихідних даних – формалізованих подань – для методу перевірки на моделі TLC, а також для розробленого розвитку методу TLC, викладеного у наступному – четвертому – розділі.

Розроблений метод базується на оперуванні і артефактами аналітичного рівня – аналітичними конструкціями-засобами дослідження ПАС, і артефактами рівня реалізації – ФС – засобами уможливлення автоматизації процесу ФВ ФС методом перевірки на моделі TLC, а також розробленим і викладеним у четвертому розділі розвитком методу TLC.

Коректність розробленого методу досліджується опосередковано – шляхом співставлення просторових характеристик графів-подань систем переходів – для артефактів, що постачаються методу у якості вихідних конструкцій, і артефактів-результатів застосування розробленого методу – ФС.

Артефакти-результати застосування розробленого методу залучено для дослідження корисного ефекту від використання розробленої і викладеної у попередньому розділі моделі формалізованого подання ПАС.

Запропоновано, обґрунтовано і експериментально перевірено аналітичні оцінки обчислювальних і просторових витрат на здійснення ФВ методом TLC для ФС, де паралелізм подано згідно моделі чергування. При цьому архітектурну складову зазначеної ФС побудовано за аналогією до сценарію здійснення розподілених обчислень, що має місце у Grid-середовищі.

3.1 Постановка вирішуваної задачі

У межах розділу ставиться і вирішується задача розроблення методу синтезу ФС ПАС, призначеного слугувати засобом автоматизації процесу постачання вихідних даних для методу перевірки на моделі TLC, а також для розробленого і викладеного у наступному – четвертому – розділі розвитку методу TLC. Іншими словами – метод призначений автоматизувати процес залучення розробленої моделі формалізованого подання ПАС СКП, викладеної у попередньому – другому – розділі.

Необхідність розроблення зазначеного методу полягає у наступному: потреба в інструменті одержання уніфікованих формалізованих подань ПАС СКП в автоматизованому режимі, а саме – забезпечити механізм одержання подання темпоральної формули ψ на основі виразних засобів формалізму TLA+, виходячи із представлення названої формули на основі складових структури (1.2). Формалізм TLA+, у свою чергу, адресується у якості засобу уможливлення процесу ФВ ФС в автоматизованому режимі.

Для викладення суті вирішуваної задачі скористаємося методом аналогії. Демонстративним прикладом, у даному випадку, є процес емулювання (відтворення) цільової системи доступними засобами наявної системи. Подібний сценарій має місце і у вирішуваній задачі:

– властивості досліджуваної системи, представлені аналітично структурою (1.2), відтворюються засобами формалізму TLA+.

Варто, однак, відзначити, що вирішення поставленої задачі включає також розв’язання наступної допоміжної задачі:

– виконати перевірку адекватності результуючої ФС, синтезованої згідно представленого методу. Такий крок напрямлений на перевірку успішності

вирішення основної задачі – з позиції коректності відтворення архітектурної складової вихідного графічного подання-артефакту у результуючій ФС.

У свою чергу, успішне розв'язання зазначеної допоміжної задачі вбачається доречним розглядати у якості опосередкованого підтвердження достовірності даних результатів-лістингів процесу автоматизованої ФВ ФС, здійснюваного методом TLC, а також розробленим розвитком цього методу, викладеним у наступному розділі.

3.2 Формулювання підходу, викладення допоміжного методу

Для вирішення поставленої задачі розглянемо вираз (1.1) як формалізоване висловлювання, що охоплює лише успішний сценарій застосування методу МС – з точки зору невиявлення помилок у ФС, тобто темпоральна формула ψ приймає істинне значення для $s \in S$, що входить до складу b (2.3). Разом із цим, назване висловлювання є неповним – з позиції успішності застосування методу: у випадку виявлення помилок у ФС у результаті застосування методу МС, відповідне застосування також доречно вважати успішним. У даному контексті вираз (1.1) доцільно розвинути з позиції дихотомії – у контексті дуального підходу, і подати формалізацію задачі ФВ, вирішуваної методом МС, як диз'юнкцію:

$$(M, b \models \psi) \vee (M, b \models \neq \psi), \tag{3.1}$$

де кожен із диз'юнктів адресує відповідний результат вирішення задачі ФВ, вирішуваної методом перевірки на моделі:

– диз'юнктом $(M,b|=\psi)$ подається сценарій підтвердження несуперечності ПАС, поданої у формі ФС, побудованої згідно розробленої моделі подання, викладеної вище – у другому розділі;

– у свою чергу, диз'юнктом $(M,b|\neq\psi)$ подається альтернативний (протилежний) сценарій, з якого було виявлено суперечність ПАС на основі відповідної ФС – шляхом виявлення «блокування» (блокувань) при здійсненні обходу простору станів СП у процесі автоматизованої ФВ ФС методом перевірки на моделі.

Отже, з урахуванням вищезазначеного, у контексті дихотомії, маємо наступні варіації:

– у випадку $(M,b|=\psi)\equiv 1$ і $(M,b|\neq\psi)\equiv 0$, результат ФВ ФС для ПАС вважатимемо успішним – з позиції того, що несуперечність ПАС було підтверджено;

– у свою чергу, для альтернативного випадку $(M,b|=\psi)\equiv 0$ і $(M,b|\neq\psi)\equiv 1$ стверджуватимемо, що у процесі ФВ ФС було виявлено блокування при обході простору станів СП. Такий результат також вважатимемо успішним – з позиції того, що було встановлено потребу доопрацювання ФС – таким чином, щоб у результатів ФВ ФС мав місце перший зазначений випадок – $((M,b|=\psi)\equiv 1)\wedge((M,b|\neq\psi)\equiv 0)$.

У свою чергу, у випадку, коли істинним є вираз $((M,b|=\psi)\equiv 0)\wedge((M,b|\neq\psi)\equiv 1)$, маємо наступні варіанти виникнення передумов цього:

– недолік, що спричинив виявлену суперечність, міститься лише у ФС, а не у графічному артефакті-поданні ПАС безпосередньо;

– зазначений недолік міститься і у артефакті-поданні ПАС, і у похідній від нього ФС, одержаній на основі представленого у межах розділу методу.

Для обох зазначених вище варіантів виникнення передумов постає потреба у розробленні і залученні методу контролю відповідності результуючої ФС первинному графічному поданню ПАС, досліджуваної з позиції несуперечності останньої – як показника ФХ.

Варто, однак, зазначити, що вираз (3.1) є тавтологією: $(M,b|=\psi)\vee(M,b|\neq\psi)\equiv 1$ – згідно закону «виключення третього» логіки висловлювань.

Згідно до вищезазначеного, висунемо наступне припущення-обґрунтування доцільності залучення методу контролю відповідності, призначеного бути засобом вирішення допоміжної задачі, сформульованої у підрозділі 3.1:

– результатом застосування допоміжного методу має бути виключення з розгляду першої з двох згаданих вище передумов виникнення суперечності (суперечностей) у ФС внаслідок здійснення процесу ФВ ФС методом перевірки на моделі.

Отже, для слідування сформульованому припущенню пропонується надавати стверджувальні відповіді допоміжні питання наступного змісту:

– стосовно несуперечності досліджуваної ФС на рівні реалізації відповідної темпоральної формули ψ на основі виразних засобів формалізму TLA+, у відповідності до розробленої і викладеної у попередньому розділі моделі подання ПАС СКП у формі ФС;

– стосовно адекватності результуючої ФС, синтезованої на основі розробленого і викладеного у цьому розділі методу, у відповідності до розробленої моделі, викладеної у попередньому – другому – розділі, первинному артефакту – графічному поданню ПАС СКП.

Для відповіді на перше питання доречним вбачається забезпечення механізму прозорого і несуперечного аналітичного подання ПАС як

темпоральної формули ψ, викладеної у формі ФС на основі виразних засобів математично строгого формалізму TLA+ темпоральної логіки дій TLA. У свою чергу, залучення названого формалізму є засобом уможливлення автоматизації процесу здійснення ФВ ФС на основі методу перевірки на моделі TLC, а також на основі розробленого розвитку зазначеного методу, викладеного у наступному – четвертому – розділі.

Розроблений підхід, у якому демонструється встановлене відношення між артефактами виокремлених аналітичного рівня і рівня реалізації, подано на рис. 3.1 у формі UML-діаграми класів.

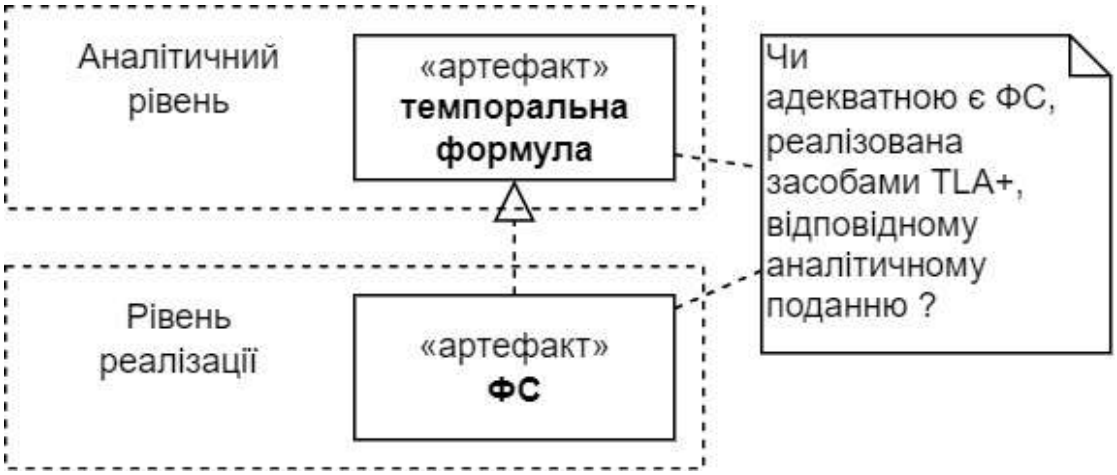

Рисунок 3.1 – UML-діаграма класів як засіб подання встановленого відношення між артефактами виокремлених рівнів

На рис. 3.1 залучено відношення «реалізація», призначене «підкреслити», що результуюча ФС, одержувана на основі розробленого методу, є реалізацією відповідного аналітичного подання у формі темпоральної формули ψ. Виразними засобами при цьому слугують засоби формалізму TLA+ темпоральної логіки дій TLA.

Варто зазначити, що перевірці адекватності результуючої ФС передує залучення програмного засобу – синтаксичного аналізатору, за допомогою якого перевіряється правильність використання засобів TLA+ при побудові ФС.

Згідно до рис. 3.1, перевірку адекватності результуючої ФС пропонується здійснювати у відповідності до положень концепції MSA (Model Structural Adequacy) – структурної адекватності моделей [128]. Для цього пропонується відповідний підхід, графічне подання якого викладено на рис. 3.2.

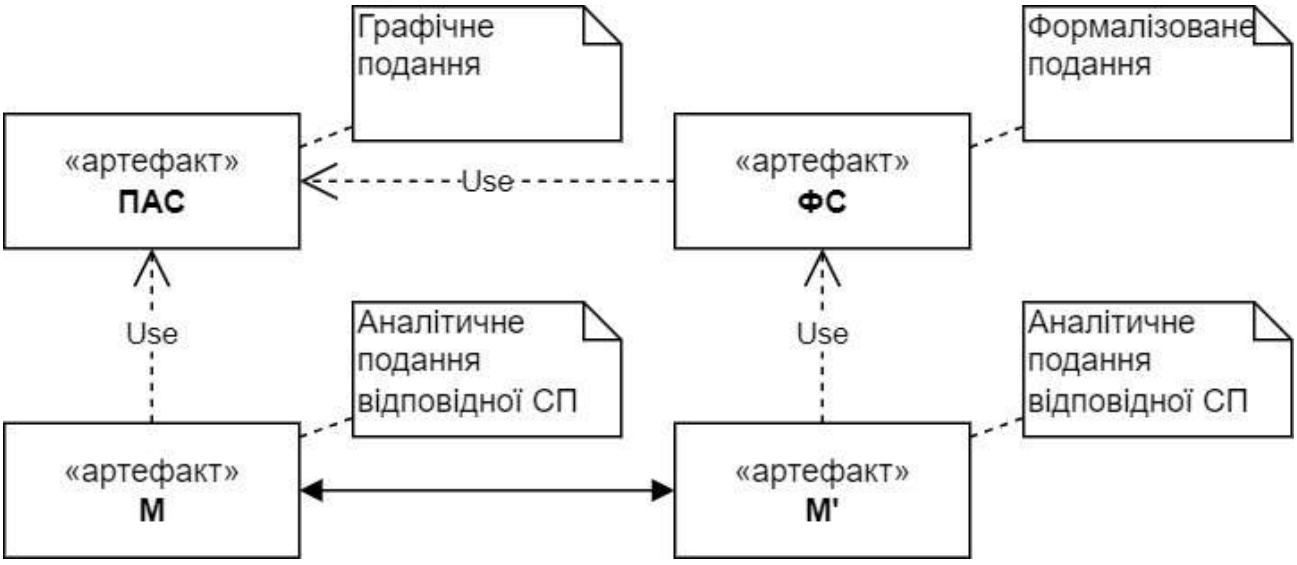

Рисунок 3.2 – Графічне подання розробленого підходу до перевірки адекватності результуючої ФС

На рис. 3.2 блок «ПАС» адресується як елемент виокремленого аналітичного рівня (рис. 3.1). При цьому структура Кріпке M залучається у якості засобу аналітичного подання відповідної СП.

Як засіб вирішення допоміжної поставленої у підрозділі 3.1 задачі, розроблений підхід до контролю адекватності результуючої ФС, одержуваної на основі представленого методу, полягає у наступному:

– здійснюється опосередковано – шляхом співставлення просторових показників СП для відповідних артефактів зазначених вище аналітичного рівня і

рівня реалізації на основі засобів структури Кріпке. Відповідні структури подано на рис. 3.2 у формі артефактів – структури M і M' – для аналітичного рівня і рівня реалізації відповідно.

Розроблений підхід полягає у поданні структур M і M' у формі графів $G=\langle S,R\rangle$ і $G'=\langle S',R'\rangle$ відповідно, де S і S' – множини вершин; R і R' – множини дуг.

Введемо також наступні позначення:

– $len(G)$ і $len(G')$ – глибини обходу просторів станів СП для M і M' відповідно, беручи відлік із початкових вершин $s_0 \in S$ і $s'_0 \in S'$ відповідно.

Суть розробленого підходу полягає у проведенні опосередкованого контролю адекватності результуючої ФС – на рівні архітектурної складової (структури та зв'язків) – первинному графічному / аналітичному поданню ПАС – шляхом співставлення просторових показників відповідних СП. «Опосередкованість» при цьому має місце через те, що оперуємо показниками не артефактів безпосередньо, а показниками відповідних СП, подання яких засобами структури Кріпке, у свою чергу, також є артефактами.

Згідно до представленого підходу реалізовано зазначений вище метод контролю відповідності результуючої ФС. Названий метод полягає у виконанні порівнянь значень наступних показників для структур M і M': $len(G)$ і $len(G')$; $|S|$ і $|S'|$. За результатами цих порівнянь виноситься судження стосовно архітектурної відповідності результуючої ФС, одержуваної на основі розробленого і викладеного у межах розділу методу синтезу ФС, первинному – графічному / аналітичному – поданню ПАС. При цьому значення показників $len(G)$ і $|S|$ для структури M одержуються аналітичним шляхом. У свою чергу, для випадку структури M', значення відповідних показників $len(G')$ і $|S'|$ одержуються в автоматизованому режимі – шляхом зчитування даних файлу-

лістингу результатів проведення ФВ ФС методом TLC або на основі розробленого розвитку цього методу, викладеного у наступному – четвертому – розділі. При цьому для наведеного нижче граничного випадку подання паралелізму у ФС згідно моделі чергування залучаються відповідні запропоновані оціночні функції.

Метод контролю відповідності результуючої ФС, розроблений для вирішення допоміжної задачі, викладеної у підрозділі 3.1, базується на наступному припущенні:

– вважатимемо, що результуюча ФС, одержувана на основі розробленого методу синтезу ФС, є відповідною первинному графічному / аналітичному поданню ПАС у випадку рівності значень показників $len(G)$ і $len(G')$, $|S|$ і $|S'|$ для структур M і M'.

Варто, проте, зауважити, що, у якості альтернативи зазначеному методу контролю відповідності, було розглянуто, у тому числі, методи, призначені для вирішення задачі встановлення ізоморфізму графів – по відношенню до графів G і G'. Відомо, однак, що названа задача є NP-складною [129]. Через цей аспект було прийнято рішення розробити відповідний допоміжний засіб – метод, що характеризується помірними вимогами до обчислювальних витрат стосовно вирішення допоміжної задачі контролю відповідності результуючої ФС.

Алгоритмічну складову розробленого допоміжного методу, із наголосом на аспектах його застосування у процесі ФВ ФС згідно виразу (3.1), подано на рис. 3.3. Зазначена складова полягає у виконанні нижченаведених кроків:

1. Крок 1 (блок 1). Згідно до рис. 3.3, початковою дією є проведення ФВ ФС методом перевірки на моделі TLC або на основі розробленого розвитку названого методу, викладеного у наступному – четвертому – розділі.

2. Крок 2 (блок 2). Базується на результатах здійснення початкового кроку. У випадку, якщо у результаті проведення ФВ ФС було встановлено

суперечність ФС, – виконується перехід до кроку 3. У протилежному випадку – підтверджено несуперечність ФС – перехід до кроку 4.

3. Крок 3 (блок 3). Провести доопрацювання ФС і перейти до кроку 1.

4. Крок 4 (блок 4). Здійснити контроль відповідності результуючої ФС згідно розробленого методу контролю відповідності. У випадку рівності значень показників $len(G)$ і $len(G')$, $|S|$ і $|S'|$ для структур M і M' – алгоритм завершує роботу. В іншому випадку – здійснюється перехід до кроку 5.

5. Крок 5 (блок 5). Доопрацювати первинний артефакт типу $a_1 \in A$ (2.1), синтезувати нову ФС – згідно моделі, викладеної у другому розділі. Перейти до кроку 1.

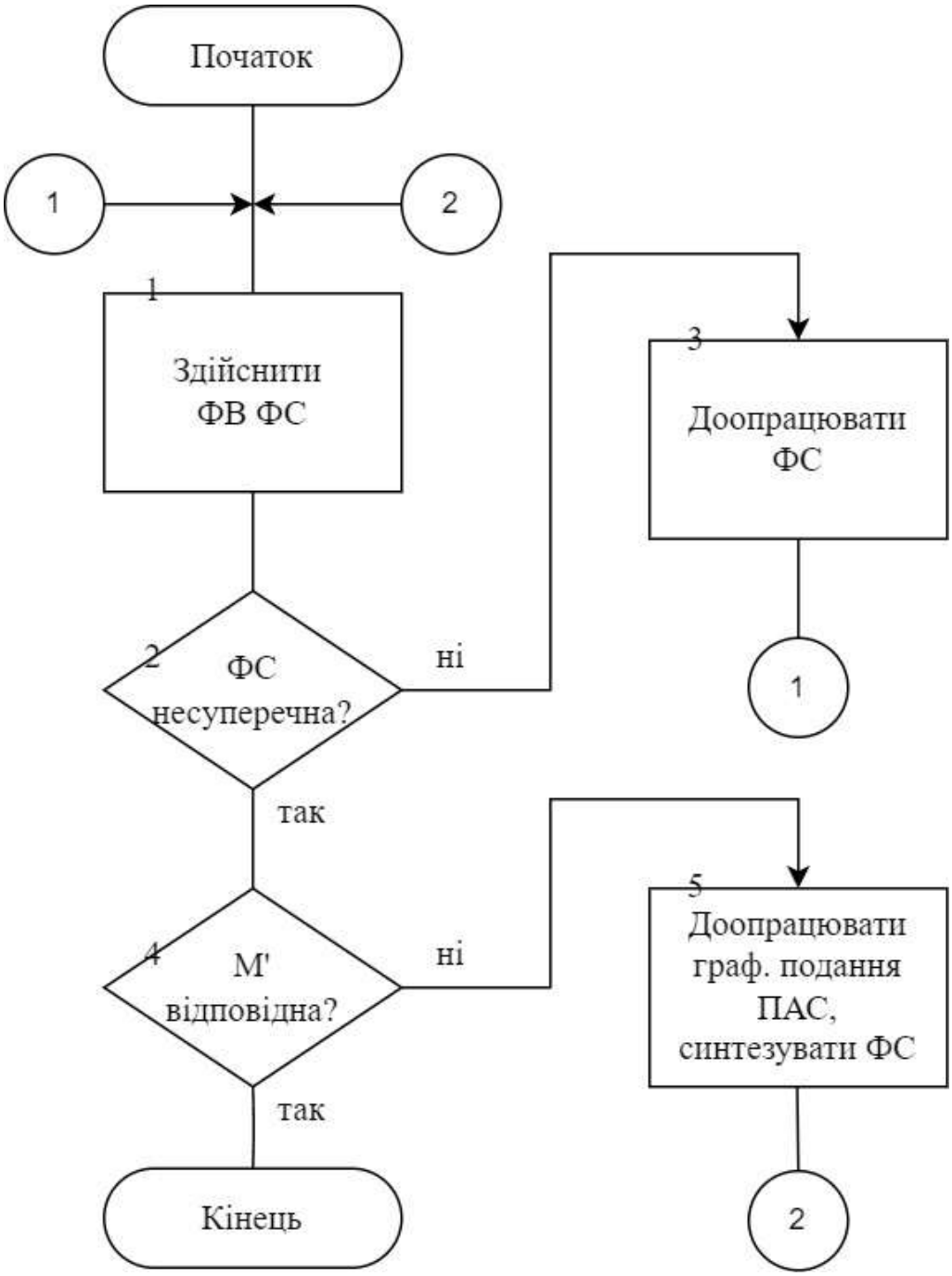

Рисунок 3.3 – Блок-схема алгоритму реалізації процесу ФВ ФС, із залученням розробленого методу контролю відповідності результуючої ФС

Згідно рис. 3.3, для застосування розробленого допоміжного методу контролю відповідності результуючої ФС має місце наступна передумова:

– попередньо має бути проведена ФВ одержаної на основі розробленого методу синтезу ФС – шляхом застосування методу перевірки на моделі, зокрема методу TLC або розробленого розвитку цього методу, викладеного у наступному – четвертому – розділі.

Отже, з урахуванням вищезазначеного – із залученням розробленого допоміжного методу контролю відповідності результуючої ФС як засобу вирішення допоміжної задачі, сформульованої у підрозділі 3.1, у межах розділу викладається розроблений метод одержання ФС ПАС – згідно до розробленої моделі, викладеної у другому розділі. При цьому доречним вбачається зауважити, що рівень деталізації подання складових досліджуваної ПАС у ФС визначається розробником, з урахуванням, у тому числі, наступних чинників:

– рівнем складності безпосередньо ПАС: кількість блоків графічного подання ПАС (формують структуру ПАС); кількість залучених змінних станів; множини допустимих значень змінних; специфіка архітектурної складової ПАС (структури та зв’язків) – визначається кількістю умовних переходів;

– заданими обмеженнями на часові (обчислювальні) витрати, супутні реалізації процесу ФВ відповідної ФС методом перевірки на моделі TLC або на основі розробленого розвитку цього методу, викладеного у четвертому розділі. Визначаються, у тому числі, продуктивністю наявної обчислювальної платформи;

– заданими обмеженнями на просторові витрати при здійсненні процесу ФВ ФС. Визначаються обсягом доступної обчислювальній системі оперативної пам’яті (ОП).

Зауваження:

– адресуючи вищенаведені чинники, варто зазначити, що кількість змінних станів, потужність множини допустимих значень змінних, а також специфіка архітектурної складової досліджуваної ПАС безпосередньо обумовлюють характер обчислювальних і просторових витрат, супутніх

процесу ФВ відповідної ФС, що буде продемонстровано нижче як аналітичним шляхом, так і за результатами проведених експериментальних досліджень.

Вбачається доцільним зазначити також і наступне:

– якщо у результаті проведення ФВ ФС було успішно підтверджено несуперечність ПАС на основі відповідної ФС, то таке підтвердження варто адресувати як правомірне у контексті заданого / обраного рівня деталізації ФС, який, у свою чергу, визначається заданими обмеженнями на часові витрати на здійснення ФВ ФС, продуктивністю обчислювальної платформи, обсягом доступної ОП;

– у свою чергу, якщо доступний ресурс часу розробника та / або обчислювальні можливості наявної у розпорядження програмно-апаратної обчислювальної платформи вдовольняють заданим обмеженням «із запасом», рекомендується підвищити рівень деталізації ФС і повторно виконати ФВ ФС.

Розроблений метод як результат вирішення основної задачі, сформульованої у підрозділі 3.1, викладається нижче покроково.

3.3 Кроки розробленого методу

Розроблений метод полягає у виконанні наступних кроків:

1. Крок 1. У відповідності до обраного розробником рівня деталізації цільової ФС, з урахуванням наведених вище зауважень, формуються множини V (2.15) і D (2.16). На основі зазначених множин, згідно до виразу (2.17), формується множина атомарних висловлювань AP, елементи якої є складовими, на основі яких задається обраний рівень деталізації ФС.

2. Крок 2. Із залученням елементів сформованої на попередньому кроці множини AP, будується структура Кріпке (1.2).

3. Кроки 3–5. Згідно (рис. 2.4), послідовно реалізуються переходи $(a_1, a_2) \in T$, $(a_2, a_3) \in T$ і $(a_3, a_4) \in T$ (2.1). У свою чергу, результатом виконання заключного переходу $(a_3, a_4) \in T$ є цільовий артефакт типу $a_4 \in A$ – ФС на основі виразних засобів формалізму TLA+.

4. Крок 6 – заключний крок – полягає у залученні розробленого і викладеного вище допоміжного засобу – методу контролю відповідності результуючого артефакту типу $a_4 \in A$ первинному артефакту типу $a_1 \in A$ – (2.1), рис. 2.4.

Зауваження до реалізації кроків методу.

Для виконання поданих вище кроків 3–5 розробленого методу синтезу ФС залучаються запропоновані правила послідовного одержання артефактів похідних типів – елементів множини $A \setminus \{a_1\}$ – на основі первинного артефакту типу $a_1 \in A$. Названі правила подано у табличній формі:

– у табл. 2.1 зведено правила виконання початкового переходу $(a_1, a_2) \in T$ – (2.1), (рис. 2.4);

– у наведених нижче табл. 3.4 і табл. 3.5 зведено правила здійснення наступних переходів – $(a_2, a_3) \in T$ і $(a_3, a_4) \in T$ відповідно.

3.4 Дослідження сценаріїв застосування методу

Розроблений метод синтезу ФС досліджено на основі синтетичних сценаріїв, адресованих у якості граничних випадків. Для цього було реалізовано і залучено допоміжне програмне забезпечення, призначене слугувати засобом автоматизації процесу одержання ФС згідно розробленого методу, а також засобом автоматизації процесу вимірювання часових і просторових витрат, супутніх процесу ФВ ФС на основі базового методу TLC і на основі розробленого розвитку зазначеного методу, викладеному у наступному – четвертому – розділі.

Зазначені синтетичні сценарії призначені визначати архітектурну складову ФС, до яких застосовуються базовий метод TLC, а також розроблений розвиток зазначеного методу:

– сценарій, за якого архітектурна складова первинного артефакту типу $a_1 \in A$ є виключно послідовного характеру – без блоків умовного переходу і циклічних конструкцій;

– сценарій, за якого конструкції-подання паралелізму ПАС реалізовано згідно моделі чергування. У якості концептуального прообразу при цьому адресовано сценарій реалізації розподілених Grid-обчислень.

Дослідження розробленого методу проведено на основі ФС, синтезованих із залученням названого методу, у відповідності до моделі подання, викладеної у попередньому розділі – шляхом ітераційного збільшення числа змінних станів, що фігурують у ФС, із збереженням при цьому характеру архітектурної складової ФС – у контексті відповідності заданому синтетичному сценарію.

Ключова ідея в основі наведеного вище підходу до організації процесу дослідження розробленого і викладеного у межах даного розділу методу, а

також моделі, викладеної у попередньому – другому – розділі, полягає у наступному:

– одержуваний корисний ефект від прикладного застосування розроблених моделі і методу прийнято рішення оцінювати кількісно, у залежності як від архітектурної складової первинного артефакту типу $a_1 \in A$, так і від числа залучених змінних станів. Зазначене оцінювання при цьому виконується на основі результуючого артефакту типу $a_4 \in A$ (рис. 2.4), одержуваного згідно названих моделі і методу.

3.4.1 Опрацювання послідовного сценарію

Для даного випадку механізм виникнення подій – зміни значень елементами $v \in V$ реалізовано наступним чином: події виникають послідовно – одна за одною. При цьому множину допустимих значень змінних сформовано наступним чином: $D = \{0,1,2\}$, базуючись на припущенні, що зазначена зміна відбувається концептуально подібно до механізму обміну повідомленнями різних типів, із підтвердженням отримання повідомлення зазначеного типу. У свою чергу, кожен окремий елемент множини V ідентифікує відповідний тип повідомлення. Показовим прикладом при цьому може слугувати заданий протокол взаємодії компонентів розподілених комп'ютерних систем, зокрема протокол MQTT – коли передумовою надсилання повідомлення певного типу є підтвердження компонентом-одержувачем надходження повідомлення іншого типу [125]. При цьому атомарні висловлювання-елементи множини $AP \subseteq V \times D$ є трьох типів – кількість типів визначається значенням $|D|$ – і мають наступну інтерпретацію:

– $(v,0) \in AP$ – «повідомлення, подане змінною $v \in V$, ще не було надіслано»;

– $(v,1)\in AP$ – «повідомлення, подане змінною $v\in V$, вже було надіслано, але ще не було доставлено»;

– $(v,2)\in AP$ – «повідомлення, подане змінною $v\in V$, вже було надіслано і доставлено».

Як доповнення-узагальнення до вищезазначеного варто зауважити наступне: деяка подія $\neg(v_j,0)\vee X(v_j,1)$ може відбутися лише за умови передування їй певної події $\neg(v_k,1)\vee X(v_k,2)$, $v_j,v_k\in V$, $j\neq k$. Отже, для випадку $|D|=3$, кожній $v\in V$ (2.15) ставимо у відповідність рівно дві події. У свою чергу, для випадку $|D|=2$, залишається одна подія.

Узагальнити вищезазначене можна наступним чином:

$$r(|D|)=|D|-1, \tag{3.2}$$

де r – кількість подій, які можна сформувати на основі змінної $v\in V$ для заданої множини допустимих значень D.

Для випадку досліджуваного граничного сценарію встановлено наступні обмеження, зокрема, на характер зміни значень елементами множини V:

– значення змінних модифікуються виключно у порядку збільшення на один;

– деякий поточний стан СП, що опрацьовується у процесі ФВ ФС, відрізняється від попереднього стану значенням лише однієї змінної $v\in V$;

– множина D є загальною для всіх елементів множини V.

Експериментальні дослідження розроблених і викладених у попередньому і даному розділах, відповідно, моделі і методу здійснено з метою кількісного оцінювання корисного ефекту, одержуваного у результаті залучення названих

моделі і методу у процесі синтезу ФС. Дослідження проведено в автоматизованому режимі – наступним чином:

– застосовано ітераційний підхід – на кожній наступній ітерації значення $n = |V|$ збільшувалося у геометричній прогресії: $n = 2^1, 2^2, \ldots, 2^8$;

– у якості кількісного показника корисного ефекту, одержуваного у результаті залучення названих моделі і методу, розглянуто відносне значення, обчислюване згідно наступного виразу:

$$\alpha(x, x') = (x - x') / x, \tag{3.3}$$

де x – результуюча кількість рядків псевдокоду ФС, отримана без застосування правила композиції (2.8), x' – із залученням названого правила.

Примітка:

– зазначене правило композиції адресовано у якості фактору, що обумовлює зниження результуючого числа рядків одержуваної ФС на основі виразних засобів TLA+.

Отримані результати проведених досліджень зведено у табл. 3.1.

Базуючись на даних табл. 3.1, доречно відзначити, що значення різниці $(x - x')$ можна оцінити наступним виразом: $(2 \cdot n - 1)$.

Згідно табл. 3.1, на залученому діапазоні вихідних даних ($n = 2^1, 2^2, \ldots, 2^8$) вдалося досягти корисного ефекту у діапазоні від близько 18 % – до близько 33 %, – за рахунок застосування правила композиції (2.8). Із табл. 3.1 видно, що, із зростанням числа змінних станів, корисний ефект також зростає.

Варто також зазначити, що, наприклад, для числа змінних $n = 2^8$ розмір файлу ФС склав 2119 КБ – для випадку без застосування правила (2.8).

Таблиця 3.1 – Зведена таблиця результатів кількісного оцінювання одержуваного корисного ефекту від застосування розроблених моделі і методу

№ з/п	n	Кількість рядків ФС		Різниця, $x - x'$	α
		x, рядків	x', рядків		
1	2	3	4	5	6
1	2^1	17	14	3	0,1765
2	2^2	29	22	7	0,2414
3	2^3	53	38	15	0,2830
4	2^4	101	70	31	0,3069
5	2^5	197	134	63	0,3198
6	2^6	389	262	127	0,3265
7	2^7	773	518	255	0,3299
8	2^8	1541	1030	511	0,3316

Значення, подані у стовпці 6 табл. 3.1, також доцільно розглядати у розрізі зниження просторових витрат на збереження результуючої ФС в ОП.

Для автоматизації процесу синтезу ФС згідно досліджуваного сценарію було реалізовано і залучено відповідний програмний засіб, створений на основі мови програмування C++. Приклади відповідних одержуваних ФС подано у додатку А. При цьому у додатку наведено випадки без застосування правила (2.8).

Адекватність результуючих ФС, одержуваних на основі названого програмного засобу, із залученням розроблених моделі і методу, викладених у другому і даному розділах відповідно, підтверджено шляхом співставлення кількостей станів СП і глибин обходів – для СП, побудованих аналітично – на основі засобів структури Кріпке (1.2) – і СП, одержуваних в автоматизованому режимі – у процесі ФВ ФС методом перевірки на моделі TLC, а також на основі

розробленого розвитку зазначеного методу, викладеного у четвертому розділі. У свою чергу, підтвердження адекватності результуючих ФС розглядається також у якості опосередкованого підтвердження адекватності розробленої моделі, викладеної у другому розділі, а також достовірності результатів застосування розробленого методу, викладеного у поточному розділі.

3.4.2 Опрацювання сценарію із поданням паралелізму

Подання паралелізму у ФС згідно моделі чергування розглянуто як граничний випадок:

– нехай маємо $n = |V| = 2^2, 2^3, 2^4$ змінних станів СП; при цьому множину D було «звужено», у порівнянні із дослідженим вище послідовним сценарієм: $D = \{0,1\}$;

– виокремлено два типи елементів множини AP, яким надано наступне змістове навантаження: $(v,0) \in AP$ – компонент розподіленої системи, поданий у ФС змінною $v \in V$, ще не завершив опрацювання призначеної йому частини обчислювального навантаження; $(v,1) \in AP$ – призначене компоненту, поданому змінною $v \in V$, обчислення завершено.

Окрім зазначеного, у ході проведення експериментальних досліджень, було прийнято рішення зупинити процес збільшення значення n на позначці $n = 2^4$ – через те, що було вичерпано доступний обчислювальній системі обсяг ОП. У свою чергу, причиною цьому став експоненційний характер зростання простору станів СП, із збільшенням значення n. Проявом цього є розмір результуючого файлу-подання ФС: для випадку $n = 2^2$ він склав 1 КБ, для випадку $n = 2^3$ – 7 КБ, для випадку $n = 2^4$ – 210727 КБ. Варто також зазначити, що для одержання відповідних ФС в автоматизованому режимі, аналогічно до попередньо розглянутого послідовного сценарію, було залучено допоміжний

реалізований для цього (на основі засобів мови програмування C++) програмний засіб.

Вибір наведених значень n обумовлено концептуальною складовою досліджуваного граничного сценарію, за якої зміна значень елементами множини V відбувається поетапно – каскадно. При цьому архітектурна складова відповідної ФС є такою, що, шляхом вилучення вершини $v_n \in V$, а також дуг, що сполучають зазначену вершину з рештою вершин відповідного графу-подання, одержуємо ідеальне бінарне дерево. Приклад такої конструкції для випадку $n = 2^3$ подано на рис. 3.4.

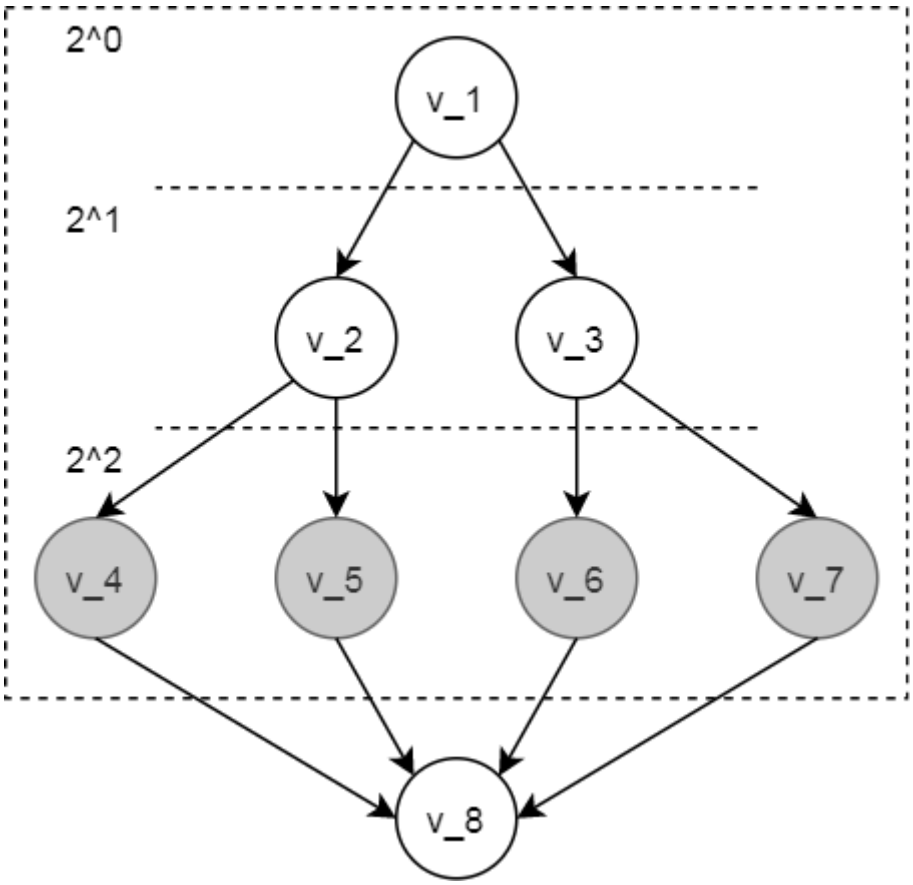

Рисунок 3.4 – Графічне представлення архітектурної складової ФС, із поданням паралелізму згідно моделі чергування

На рис. 3.4 прямокутною пунктирною областю окреслено ідеальне бінарне дерево, одержуване, якщо вилучити вершину $v_8 \in V$ і суміжні з вершиною дуги. При цьому сірим кольором зафарбовано термінальні вершини дерева. У свою чергу, у лівій верхній частині кожного із рівнів дерева зазначено кількість вершин, що фігурують на відповідному рівні.

Виокремлення вищеназваних рівнів означає наступне:

– спочатку модифікується значення змінної $v_1 \in V$, у результаті чого маємо наступне формалізоване подання відмінності наступного стану $s_1 = R(s_0) \in S$ від початкового стану $s_0 \in S_0 \subset S$: $L(s_0) \Delta L(s_1) = \{(v_1,0),(v_1,1)\}$, де, згідно (2.22), $(v_1,0) \in AP' \subset AP$, а $(v_1,1) \in AP'' \subset AP$: $AP' \cup AP'' = AP$;

– надалі модифікація значень змінними наступних рівнів (рис. 3.4) – з 2^1 і 2^2 елементами – відбувається каскадно – із залученням моделі чергування подій зміни значень елементів множини $V \setminus \{v_1, v_8\}$;

– завершальним кроком є модифікація значення змінної $v_8 \in V$.

Граф, наведений на рис. 3.4, є поданням архітектурної складової розподіленої обчислювальної системи, де $(n/2)-1$ вершин у складі дерева, окресленого пунктирною областю, є поданнями вузлів зазначеної системи, призначення яких – декомпозиція обчислювальної задачі. При цьому $n/2$ вершин, зафарбованих сірим кольором, є поданнями вузлів, що безпосередньо виконують обчислення. У свою чергу, вершина $v_8 \in V$ – подання вузла, на якому здійснюється накопичення результату.

Систему, подану на рис. 3.4, можна аналітично представити наступним графом: $G = \langle V, E \rangle$, де V – множина вершин – змінних станів, що фігурують у ФС: $|V| = n$; $E \subseteq V^2$ – множина дуг, що сполучають названі вершини.

Для оцінювання просторових характеристик ФС, одержуваних згідно досліджуваного сценарію, а також відповідних СП, що будуються у процесі ФВ ФС, запропоновано нижченаведені оціночні функції.

Функція оцінювання кількості вершин у складі підграфу-дерева, окресленого пунктирною областю (рис. 3.4):

$$g_1(n) = \sum_{k=0}^{(\log_2 n)-1} 2^k = |V| - 1 = n - 1, \tag{3.4}$$

де k – порядковий номер рівня, на якому фігурують відповідні вузли дерева, – задає порядок модифікації значень на рівні груп з 2^k елементів.

Функція оцінювання загальної кількості дуг у складі графу G:

$$g_2(n) = n + \frac{n}{2} - 2 = \frac{3 \cdot n - 4}{2} = |E|, \tag{3.5}$$

де $n = 2^2, 2^3, \ldots, 2^N$. Зауваження:

– залучаємо саме дуги, а не ребра, щоб відтворити направленість обчислювального процесу;

– у якості початкового значення обрано саме $n = 2^2$, а не $n = 2^1$, оскільки в останньому випадку архітектурна складова досліджуваної ФС вироджується у таку, що відповідає послідовному сценарію, дослідженому вище.

Порядок модифікації значень змінних станів, поданих на рис. 3.4, організовано згідно алгоритму BFS теорії графів: спочатку залучаються 2^0 вершин верхнього рівня (корінь дерева), потім – 2^1 вершин наступного рівня, у якому фігурують суміжні вершини-«нащадки», і т. д. У свою чергу, компонент обчислювальної системи, поданий вершиною $v_1 \in V$, іменуватимемо

«ініціюючим» компонентом обчислювального процесу, а компонент, поданий вершиною $v_8 \in V$, – «агрегуючим» компонентом.

Граф, наведений на рис. 3.4, є поданням сценарію обчислювального процесу, згідно якого «просування» від заданої батьківської вершини до дочірніх відносно неї, і таким чином – до термінальних вершин підграфу G' у складі графу G, окресленого пунктирною областю, є відтворенням процесу поступального розподілу обчислювального навантаження. Вузли у складі підграфу G', «зафарбовані» сірим кольором, у свою чергу, є поданнями компонентів розподіленої системи, на яких безпосередньо виконуються обчислення. Разом із цим, вершиною $v_8 \in V$ у складі графу G відтворюється компонент, до якого надсилаються результати проміжних обчислень.

Для автоматизації процесу одержання ФС, побудованих у відповідності до окресленого сценарію було реалізовано відповідний допоміжний програмний застосунок. Також було автоматизовано процес накопичення та опрацювання результатів проведених досліджень.

Для оцінюванні кількості станів СП, що буде побудовано у процесі ФВ ФС, запропоновано відповідну оціночну функцію:

$$g_3(n) = \sum_{k=1}^{(\log_2 n)-1} \left(2^{2^k} - 1\right) + \beta = |S|, \tag{3.6}$$

де $\beta = 3 = const$ – стале значення, що враховує один початковий стан СП – $s_0 \in S$, передостанній стан СП, що передує активації результуючого компоненту, – $s_{l-1} \in S$, і заключний стан СП – $s_l = R(s_{l-1}) \in S$.

Згідно (3.6), кількість станів СП для $n = 2^2, 2^3, 2^4$ можна чисельно подати наступним чином: $|S| = 6, 21, 276$ відповідно; $D = \{0,1\}$.

Отримані результати проведених експериментальних досліджень одержуваного корисного ефекту від застосування розробленої моделі, викладеної у другому розділі, а також розробленого і представленого у даному розділі методу синтезу ФС, згідно до розглянутого сценарію, із залученням озвучених вище засобів автоматизації, подано у табл. 3.2. У якості показника одержуваного корисного ефекту при цьому залучено кількість рядків результуючої ФС на основі виразних засобів формалізму TLA+.

Таблиця 3.2 – Отримані результати оцінювання корисного ефекту від залучення правила композиції Чарльза Гоара

№ з/п	n	Кількість рядків ФС		Різниця, $x - x'$	α
		Без застосування правила (2.8), x	Із застосуванням правила (2.8), x'		
1	2	3	4	5	6
1	2^2	18	14	4	0,2222
2	2^3	111	92	19	0,1712
3	2^4	1936702	1936428	274	$1{,}4148 \cdot 10^{-4}$

У табл. 3.2 x – кількість рядків результуючої ФС без застосування правила композиції, x' – із застосуванням. Коефіцієнт α при цьому обчислено згідно виразу (3.3).

На відміну від табл. 3.1, у табл. 3.2 спостерігається протилежна ситуація – зменшення одержуваного корисного ефекту із зростанням значення n. У відсотковому вираженні значення α склало від близько 22 % – до близько 0 % відповідно, із збільшенням значення n від $n = 2^2$ до $n = 2^4$.

Варто зазначити, що у процесі проведення досліджень було прийнято рішення відмовитись від подальшого збільшення значення n. Причиною

послугував експоненційний характер зростання простору станів СП від числа змінних станів. Кількісно оцінити таке зростання у площині просторових витрат, супутніх вирішенню задачі ФВ ФС методом TLC, можна, зокрема, наступним чином: для випадку $n=2^3$ розмір відповідного артефакту – текстового файлу ФС – на основі засобів TLA+ склав 7 КБ, при цьому для випадку $n=2^4$ – вже 210727 КБ.

Із табл. 3.2 видно, що значення виразу $(x-x')$ для $n=2^2,2^3,2^4$ є на два меншим, у порівнянні із відповідними оціночними значеннями, обчисленими згідно виразу (3.6). Це пояснюється тим, що вираз $(x-x')$ не враховує початковий і заключний стани СП. Більше того, на відміну від послідовного сценарію (табл. 3.1), залежність значення показника одержуваного корисного ефекту α від числа змінних станів n у даному випадку є не прямо пропорційною, а обернено пропорційною. Значення α (3.3) при цьому прямує до нуля. Це зумовлено тим, що, для випадку подання паралелізму згідно моделі чергування, кількість формалізованих подань переходів між станами СП зростає експоненційно швидше за кількість формалізованих подань станів. У свою чергу, для визначення кількості переходів між станами СП запропоновано і залучено наступну оціночну функцію:

$$g_4(n)=\sum_{k=1}^{(\log_2 n)-1}\left(2^{2^k+k-1}\right)+\gamma=|R|, \tag{3.7}$$

де $\gamma=2=const$ – фіксоване значення, що враховує початковий $(s_0,s_1)\in R$ і заключний $(s_{l-1},s_l)\in R$ переходи СП. Зазначені переходи, у свою чергу, є поданнями подій активації ініціюючого і результуючого компонентів системи, зображеної на рис. 3.4.

Для визначення глибини обходу простору станів СП запропоновано і залучено наступну оціночну функцію:

$$g_5(n) = \sum_{k=0}^{(\log_2 n)-1} \left(2^k\right) + 2 = g_1(n) + 2 = |V| + 1 = n + 1, \qquad (3.8)$$

де $n = 2^1, 2^2, 2^3, \ldots$.

При цьому під глибиною обходу розуміємо не кількість дуг, а кількість станів шляху у складі СП, що їх треба пройти, починаючи із стану $s_0 \in S$, і завершуючи заключним станом $s_l \in S$ включно. Названий шлях, у свою чергу, формалізується як відповідна поведінка $b_i \in B$ (2.2).

Кількість альтернативних шляхів СП, що їх необхідно пройти у процесі ФВ ФС згідно моделі чергування, пропонується оцінювати згідно нижченаведеного виразу:

$$g_6(n) = \prod_{k=1}^{(\log_2 n)-1} \left(2^k\right)! = |B|, \qquad (3.9)$$

де B – множина поведінок, подана виразом (2.2).

Зведемо дані, одержувані на основі запропонованих оціночних функцій (3.4) – (3.9) для $n = 2^2, 2^3, 2^4$, у табличній формі (табл. 3.3).

Дані табл. 3.3 призначені слугувати комплексним показником просторової складності вирішуваної задачі ФВ ФС для заданого значення n. ФС при цьому побудовано згідно досліджуваного сценарію подання паралелізму на основі моделі чергування.

Із табл. 3.3 видно, що для $n = 2^2, 2^3, 2^4$ кількість шляхів СП, що їх потрібно пройти становить $g_6(n) = 2, 48, 1935360$ відповідно. При цьому варто

зауважити, що на рис. 3.4 фігурує графічне подання архітектурної складової ФС, а не відповідної СП, що будується у процесі ФВ ФС.

Таблиця 3.3 – Розрахункові значення, отримані на основі запропонованих оціночних функцій (3.4) – (3.9)

№ з/п	n	Оціночні функції					
		$g_1(n)$, (3.4)	$g_2(n)$, (3.5)	$g_3(n)$, (3.6)	$g_4(n)$, (3.7)	$g_5(n)$, (3.8)	$g_6(n)$, (3.9)
1	2	3	4	5	6	7	8
1	2^2	3	4	6	6	5	2
2	2^3	7	10	21	38	9	48
3	2^4	15	22	276	1062	17	1935360

Як узагальнення до поданих вище оціночних функцій, функції $g_1(n)$, $g_2(n)$ є засобами оцінювання просторових характеристик архітектурної складової безпосередньо ФС. У свою чергу, функції $g_3(n)$,…, $g_6(n)$ адресують названі характеристики відповідних СП, що будуються у процесі ФВ ФС.

Як доповнення до табл. 3.3, у якості графічної демонстрації змісту оціночної функції $g_3(n)$ слугує, зокрема, СП для випадку $n = 2^2$, подана у формі UML-діаграми станів (рис. 3.5).

На рис. 3.5 у якості вершин графу фігурують 6 станів СП: для кожного $s \in S$ у процесі ФВ ФС, згідно виразу (1.1), виконується перевірка істинності темпоральної формули, заданої у ФС на основі виразних засобів формалізму TLA+.

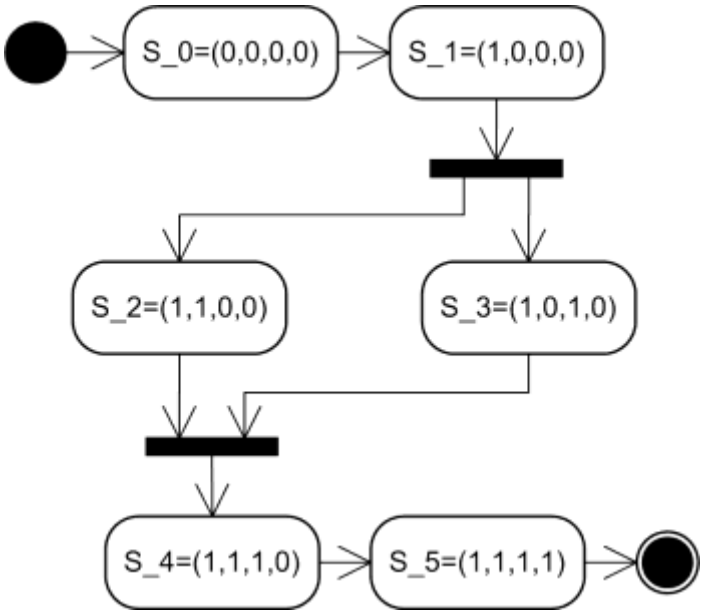

Рисунок 3.5 – Граф системи переходів для випадку $n = 2^2$

З рис. 3.5 видно, що у поданні паралелізму згідно моделі чергування задіяно дві змінні – $v_2, v_3 \in V$. Це обумовлює дві альтернативні послідовності подій: $e_1 \prec e_2 \prec e_3 \prec e_4$ та $e_1 \prec e_3 \prec e_2 \prec e_4$, де $\prec$ – оператор відношення порядку «передувати».

У випадку сценарію $e_1 \prec e_2 \prec e_3 \prec e_4$ подія $(v_2,0) \vee X(v_2,1)$ передує події $(v_3,0) \vee X(v_3,1)$. У свою чергу, у випадку альтернативного сценарію $e_1 \prec e_3 \prec e_2 \prec e_4$, навпаки – подія $(v_3,0) \vee X(v_3,1)$ передує події $(v_2,0) \vee X(v_2,1)$.

Результатом виникнення подій згідно послідовності $e_1 \prec e_2 \prec e_3 \prec e_4$ (2.9) є відповідна «поведінка» $b_1 = s_0, s_1, s_2, s_4, s_5$ (2.3) як послідовність станів. У свою чергу, для альтернативної варіації $e_1 \prec e_3 \prec e_2 \prec e_4$, згідно рис. 3.5, маємо вже наступну послідовність станів: $b_2 = s_0, s_1, s_3, s_4, s_5$. Отже, для даного випадку, $n = 2^2$, маємо $B = \{b_1, b_2\}: |B| = g_6(4) = 2$.

Якщо позначити множини станів, на основі елементів яких будуються відповідні послідовності $b_1, b_2 \in B.$, відповідно, як S_{b_1} і S_{b_2}, відмінність між

зазначеними множинами можна подати, наприклад, наступним чином: $S_{b_1}\Delta S_{b_2}=\{s_2,s_3\}$. У свою чергу, згідно (2.4) і (2.5), формуємо відповідні протоколи процесів нотації CSP: $q(b_1)=p_1\in P$, де $p_1=\langle e_1,e_2,e_3,e_4\rangle$ і $q(b_2)=p_2\in P$, де $p_2=\langle e_1,e_3,e_2,e_4\rangle$.

Для кожного $p_i\in P(i=1,2)$, де для нашого випадку (рис. 3.5) маємо $|P|=|B|=2$, застосуємо до елементів $p_1,p_2\in P$ правило композиції (2.8). У результаті матимемо наступні вирази:

– для $p_1\in P$ – вираз $\dfrac{\{\varphi_0\}e_1\{\varphi_1\},\{\varphi_1\}e_2\{\varphi_2\},\{\varphi_2\}e_3\{\varphi_4\},\{\varphi_4\}e_4\{\varphi_5\}}{\{\varphi_0\}e_1;e_2;e_3;e_4\{\varphi_5\}}$;

– для $p_2\in P$ – вираз $\dfrac{\{\varphi_0\}e_1\{\varphi_1\},\{\varphi_1\}e_3\{\varphi_3\},\{\varphi_3\}e_2\{\varphi_4\},\{\varphi_4\}e_4\{\varphi_5\}}{\{\varphi_0\}e_1;e_3;e_2;e_4\{\varphi_5\}}$.

У чисельниках виразів для $p_1,p_2\in P$ фігурують трійки Гоара. У свою чергу, у знаменниках – результат застосування правила (2.8). Побудова ФС згідно виразу-знаменника, а не чисельника, спрямована на скорочення кількості рядків псевдокоду результуючої ФС – за рахунок виключення формалізованих подань перед- і пост-умов, у даному випадку – $\varphi_1,\varphi_2,...,\varphi_4$.

Зазначені вирази відмінні між собою порядком слідування подій e_2 і e_3. У результаті застосування до них правила імплікації (2.12), маємо наступні аналітичні подання, акцентовані на відмінностях між протоколами $p_1\in P$ і $p_2\in P$:

– для $p_1\in P$ маємо $\dfrac{\varphi_0\to\varphi_1,\{\varphi_1\}e_2\{\varphi_2\},\{\varphi_2\}e_3\{\varphi_4\},\varphi_4\to\varphi_5}{\{\varphi_0\}e_2;e_3\{\varphi_5\}}$;

– для $p_2\in P$ маємо $\dfrac{\varphi_0\to\varphi_1,\{\varphi_1\}e_3\{\varphi_3\},\{\varphi_3\}e_2\{\varphi_4\},\varphi_4\to\varphi_5}{\{\varphi_0\}e_3;e_2\{\varphi_5\}}$.

Отже, застосування правила композиції (2.8) покликане зменшити обсяг коду ФС – для спрощення її сприйняття і аналізу розробником – за рахунок виключення із складу ФС рядків подань розміток проміжних станів СП. У свою

чергу, застосування правила (2.12) призначене слугувати механізмом виокремлення відмінностей між альтернативними протоколами.

Для узагальнення отриманих експериментальних результатів для послідовного сценарію і сценарію подання паралелізму згідно моделі чергування, на основі даних табл. 3.1 і табл. 3.2 було побудовано відповідний зведений графік (рис. 3.6).

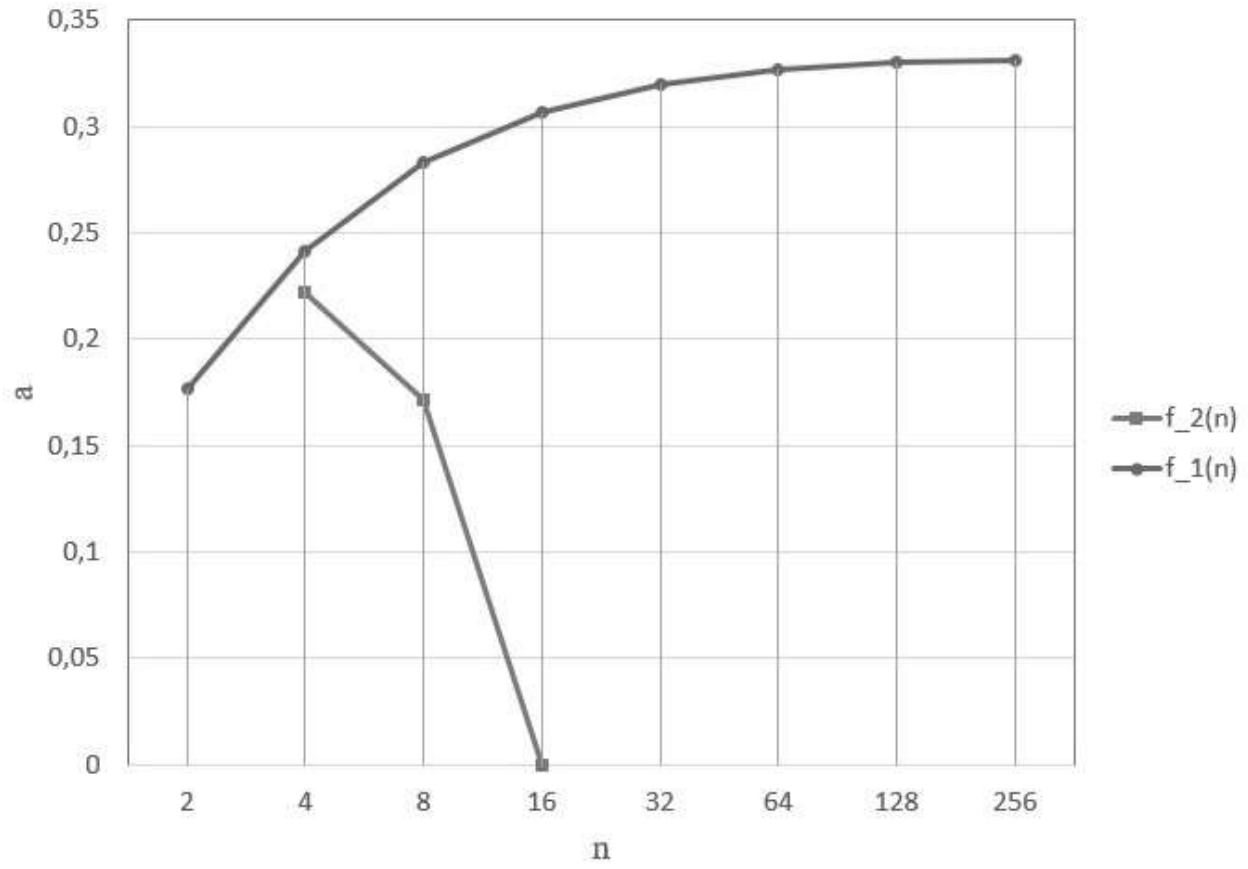

Рисунок 3.6 – Зведений графік залежності одержуваного корисного ефекту від кількості змінних станів

На рис. 3.6 по осі Ox подано значення кількості змінних n. При цьому значення відповідних показників α для охоплених сценаріїв – послідовного і з поданням паралелізму згідно моделі чергування – фігурують у формі значень

функцій – $f_1(n)$ і $f_2(n)$ відповідно. При побудові графіків застосовано кусочно-лінійну інтерполяцію.

З рис. 3.6 видно, що, у той час як функція $f_1(n)$ монотонно зростає із зростанням значення n, функція $f_2(n)$ при цьому стрімко спадає. Таку ситуацію можна охарактеризувати наступним чином:

– для охоплених граничних випадків маємо відмінні динаміки зміни значень одержуваного корисного ефекту із зростанням значення n;

– для випадку послідовного сценарію – із збільшенням значення n одержуваний корисний ефект від залучення розроблених моделі і методу зростає – функція $f_1(n)$;

– для випадку подання паралелізму згідно моделі чергування – із збільшенням значення n одержуваний корисний ефект спадає – функція $f_2(n)$.

У якості прикладу, результуюча ФС для випадку, поданого на рис. 3.5, без застосування правила композиції наведена у додатку Б.

3.5 Деталізація кроків розробленого методу синтезу

У межах підрозділу деталізовано кроки 3 – 5 розробленого методу синтезу ФС, викладені у підрозділі 3.3. При цьому результатом виконання кроку 5 є ФС на основі виразних засобів TLA+, що уможливлює проведення ФВ методом TLC в автоматизованому режимі.

Крок 3: $(a_1, a_2) \in T$. Згідно розробленої моделі (2.1) і рис. 2.4, артефакти опрацьовуються на аналітичному рівні.

Згідно табл. 2.1, артефакти виокремлених типів $a_1, a_2 \in A$ охоплюються на обох ієрархічних рівнях.

До початкового елементу $s_0 \in S$ послідовності (2.3) застосовується функція розмітки станів: $L(s_0)$. У свою чергу, над елементами множини $L(s_0)$ виконується операція кон'юнкції. У результаті одержується вираз вигляду (2.23) як передумова запуску обчислювального процесу згідно знаменнику правила композиції Гоара (2.8). У результаті формується протокол обчислювального процесу на основі засобів CSP згідно виразу (2.5).

Отже, протокол обчислювального процесу згідно виразу (2.5) є формалізованим поданням артефакту типу $a_2 = T(a_1) \in A$, одержуваного у результаті виконання кроку 3 розробленого методу синтезу ФС.

Крок 4: $(a_2, a_3) \in T$. Згідно розробленої моделі (2.1) і рис. 2.4, зазначений крок адресується у якості засобу сполучення аналітичного рівня із рівнем реалізації. Рівень реалізації при цьому опрацьовується у якості площини, на якій фігурують артефакти, залучення яких уможливлює реалізацію процесу ФВ в автоматизованому режимі.

Результуючий у межах зазначеного кроку артефакт типу $a_3 = T(a_2) \in A$ подається засобами алгоритмічної мови PlusCal, що, за рахунок математичної строгості, дозволяє сформувати архітектурну складову цільової ФС типу $a_4 \in A$, одержувану на наступному кроці.

Аналогічно до попереднього кроку, виконуваного згідно запропонованих правил, зведених у табл. 2.1, за поточного кроку також опрацьовуються елементи двох виокремлених ієрархічних рівнів. Відповідні правила, згідно (2.1) і рис. 2.4, викладено у табл. 3.4.

Для сполучення елементів нижнього ієрархічного рівня артефактів типу $a_3 \in A$ – з метою формування елементів верхнього рівня – застосовано правило умовного оператору логіки Гоара. Для цього, для наочності, розглянемо випадок дихотомії, опрацьований у попередньому розділі – у виразі (2.22):

$$\frac{\{\varphi' \wedge ap'_j\}e_j\{\xi\}, \{\varphi' \wedge \neg ap'_j\}e_j^0\{\xi\}}{\{\varphi'\} if\left(ap'_j\right) then\left(e_j\right) else\left(e_j^0\right) endif\{\xi\}}, \tag{3.10}$$

де $ap'_j \in AP' = AP \setminus AP''$ – складова передумови, згідно якої задається варіативність подій: істинність $ap'_j = (v_j, 0)$ є передумовою події e_j, а істинність альтернативної складової $\left(\neg ap'_j\right) = ap''_j = (v_j, 1) \in AP'' = AP \setminus AP'$ – передумовою псевдоподії e_j^0:

$$\varphi \equiv \varphi' \wedge ap'_j \equiv \varphi' \wedge (v_j, 0), \tag{3.11}$$

$$\xi \equiv \varphi' \wedge ap''_j \equiv \varphi' \wedge (v_j, 1). \tag{3.12}$$

Таблиця 3.4 – Співвідношення між складовими артефактів типів $a_2 \in A$ і $a_3 = T(a_2) \in A$ для здійснення переходу $(a_2, a_3) \in T$

Страти	Засоби рівнів: аналітичного, реалізації (рис. 2.4)	
	Формалізм CSP	Формалізм PlusCal
1	2	3
1	Протокол / протоколи процесу / процесів (2.5).	Алгоритм згідно (2.5), (2.10).
2	Подія (2.19).	Еквівалентне подання у формі псевдокоду – модифікація значення елементу множини змінних (2.15).

Введення позначення e_j^0 альтернативної події у виразі (3.10) обумовлене потребою просування модельного часу для кожного із елементів множини змінних (2.15) у процесі ФВ.

Наприклад, для розглянутого вище граничного випадку подання паралелізму згідно моделі чергування за кожного окремого відліку модельного часу модифікується значення лише однієї змінної $v_j \in V$ [131]. При цьому решта $(n-1)$ змінних зберігають попередні значення. У такому випадку опрацювання просування модельного часу для $(n-1)$ змінних можна формалізувати подібно до виразу (2.19) наступним чином:

$$e^0{}_j \equiv \left(\left(v_j, d_k\right) \to X\left(v_j, d_k\right)\right) \equiv \left(\neg\left(v_j, d_k\right) \vee X\left(v_j, d_k\right)\right). \qquad (3.13)$$

З урахуванням залучення темпорального оператору *X*, змістове навантаження виразу (3.13) є наступним: значення змінної $v_j \in V$ для наступного відліку модельного часу не відрізняється від такого для поточного відліку. Зазначена інтерпретація, у свою чергу, характеризується аксіомою пустого оператора Гоара [62]:

$$\overline{\left\{\left(v_j, d_k\right)\right\}} skip \overline{\left\{\left(v_j, d_k\right)\right\}}. \qquad (3.14)$$

Вирази (3.13) і (3.14) адресуємо як формалізовані подання «псевдоподій» – за яких значення відповідних залучених змінних $v_j \in V$ не модифікуються за просування модельного часу у процесі ФВ.

Для формування елементів нижнього ієрархічного рівня у складі артефактів виокремленого типу $a_3 \in A$ на основі виразних засобів алгоритмічної мови PlusCal згідно табл. 3.4, залучимо попередньо опрацьовані вирази-подання подій (2.19) і псевдоподій (3.13) – згідно знаменника правила умовного оператору (3.10) [138]:

$$e'_j \equiv \left(v'_j := if\left(\varphi' \wedge ap'_j\right) then\left(d_h\right) else\left(d_k\right)\right), \qquad (3.15)$$

де e'_j – агрегуюча конструкція, у межах якої вирази (2.19) і (3.13) поєднано на основі засобів алгоритмічної мови PlusCal.

Згідно виразу (3.15), у залежності від булевого значення виразу-умови $\left(\varphi' \wedge ap'_j\right)$, матиме місце або подія (2.19), або псевдоподія (3.13). При цьому до виразу (3.15) залучено також допоміжну складову – $v'_j \in V'$ – змінну-копію відповідної змінної $v_j \in V$ – у якості засобу формалізації наступного кроку модельного часу – щоб відобразити на основі засобів PlusCal ефект від застосування зазначеного темпорального оператору зсуву модельного часу *X*.

Зауваження.

Змістове навантаження змінної-копії $v'_j \in V'$ також можна викласти на основі виразних засобів структури Кріпке, застосованих вище для подання артефактів типу $a_1 \in A$:

– $v_j \in V$ – змінна станів СП, що фігурує у розмітці $L(s)$ поточного стану $s \in S$ СП, на основі елементів якої формується передумова $\left(\varphi' \wedge ap'_j\right)$ для виникнення або події (2.19), або псевдоподії (3.13);

– $v'_j \in V'$ – змінна-копія відповідної змінної $v_j \in V$, призначена фігурувати у розмітці $L(R(s))$ наступного стану $R(s) = s' \in S$ СП. При цьому $(s, s') \in R$.

Графічне подання співвідношень між виокремленими аналітичними конструкціями події (2.19) і псевдоподії (3.13), а також зведеним виразом-репрезентацією (3.15), побудованим з урахуванням виразних можливостей алгоритмічної мови PlusCal і призначеним для формалізації елементів нижнього

ієрархічного рівня у складі артефактів типу $a_3 \in A$ згідно табл. 3.4 наведено на рис. 3.7.

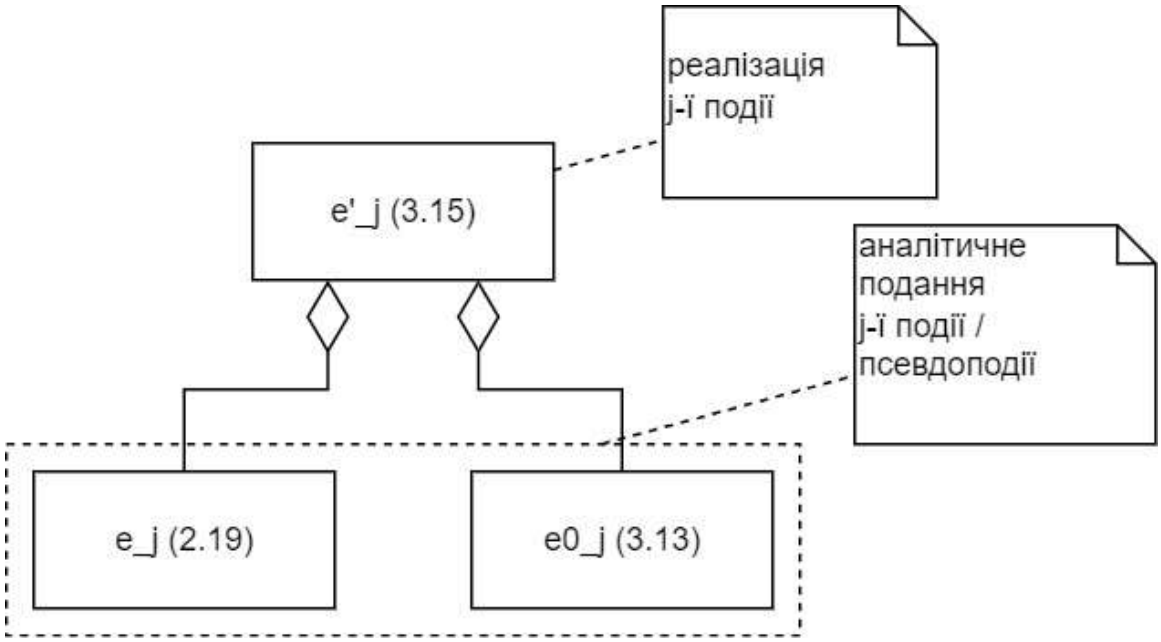

Рисунок 3.7 – Графічне подання застосованого підходу до формування елементів нижнього ієрархічного рівня у складі артефактів типу $a_3 \in A$

UML-діаграма, викладена на рис. 3.7, призначена відобразити дуальний характер застосованого підходу до інтерпретації подій як елементів нижнього ієрархічного рівня артефактів типу $a_3 \in A$. При цьому у якості засобу сполучення виокремлених сутностей аналітичного рівня (2.19) і (3.13) із відповідним дуальним поданням (3.15) рівня реалізації використано відношення агрегування.

Проводячи паралелі між контекстним навантаженням, покладеним в основу виразу (3.15), і таким для відповідних конструкцій мов програмування, доречним вбачається наведення у якості прикладу тернарного оператору «?:» мови програмування С. Його залучення дозволить подати вираз (3.15) більш строго – без викривлення змістового навантаження виразу:

$$e'_j \equiv \left(v'_j := \left(v_j, d_k\right)?\left(d_h\right):\left(d_k\right)\right), \tag{3.16}$$

де $\left(v_j, d_k\right) \in L(s) \subset AP, d_k \neq d_h$ – передумова виникнення події дуального характеру, поданої на рис. 3.7, що приймає істинне значення для поточного стану $s \in S$ СП.

У свою чергу, у результаті виникнення події (2.19) істинне значення вже для наступного стану $R(s) = s' \in S$ СП прийматиме атомарного висловлювання $\left(v_j, d_h\right) \in L(R(s)) \subset AP$.

В іншому випадку – за якого висловлювання-передумова $\left(v_j, d_k\right) \in L(s) \subset AP$ є хибним для поточного стану $s \in S$ – замість події (2.19) матиме місце вже псевдоподія (3.13), у результаті виникнення якої те саме атомарне висловлювання $\left(v_j, d_k\right) \in L(R(s)) \subset AP$ буде істинним і для наступного стану $R(s) = s' \in S$.

У випадку, коли множина допустимих значень змінних (2.16) є булевою, вираз (3.16) можна переписати наступним чином:

$$e_j \equiv \left(v'_j := \left(\varphi' \wedge ap'_j\right)?\left(\neg v_j\right):\left(v_j\right)\right). \tag{3.17}$$

У свою чергу, для формування елементів верхнього ієрархічного рівня табл. 3.4 елементи нижньої страти, сформовані згідно вирізів (3.15) – (3.17), сполучаються згідно протоколу / протоколів обчислювального процесу (2.5). При цьому кількість зазначених протоколів визначається характером архітектурної складової ФС.

З урахуванням опрацьованих у межах попередніх підрозділів граничних випадків – для послідовного сценарію і сценарію подання паралелізму згідно

моделі чергування, для першого випадку маємо $|P|=1$ (2.5), для другого – $|P|=g_6(n)$ (3.9) – як кількість альтернативних шляхів, що їх треба пройти у процесі ФВ.

Крок 5: $(a_3,a_4)\in T$. Виконується згідно розробленої моделі (2.1) і рис. 2.4. Аналогічно до попередніх кроків, застосовується індукційний підхід. При цьому для одержання елементів нижнього ієрархічного рівня артефактів результуючого типу $a_4\in A$ конструкції вигляду (3.16) доопрацьовуються з урахуванням виразних засобів формалізму TLA+: псевдоподія (3.13) реалізується згідно концепції «stuttering step» – шляхом застосування відповідного оператору «unchanged» [89]:

$$e'_j\equiv(v'_j:=(v_j,d_k)?(d_h):(u(v_j))), \tag{3.18}$$

де u – формалізоване подання оператору «unchanged» як засобу реалізації властивості ідемпотентності (додаток А):

$$u(v_j)=v_j. \tag{3.19}$$

Залучення оператору (3.19) безпосередньо, а також у складі виразу (3.18), уможливлює формалізацію фундаментальних конструкцій логіки TLA – «дій» (actions) засобами TLA+ [88, с. 16].

На відміну від формалізованого подання події (3.18), де фігурує єдина змінна $v_j\in V$, а також відповідна копія $v'_j\in V'$, концепція «дії» надіє засоби охоплення кожного із елементів зазначених множин змінних V і V' – шляхом сполучення виразів (3.18) і (3.19) на основі оператору кон'юнкції. Це, у свою чергу, уможливлює формалізацію переходу $(s,s')\in R$ засобами TLA+.

Для кожного із опрацьованих вище граничних випадків – послідовного сценарію і сценарію подання паралелізму згідно моделі чергування – має місце наступний шаблон формалізації «дій» засобами TLA+:

$$act_j \equiv u(v_1) \wedge u(v_2) \wedge \ldots \wedge u(v_{j-1}) \wedge e'_j \wedge u(v_{j+1}) \wedge \ldots \wedge u(v_n), \qquad (3.20)$$

згідно якого за кожного поточного відліку модельного часу у процесі ФВ відбувається одна дія – як кон'юнкція над одним висловлюванням (3.18) і $n-1$ висловлюванням (3.19).

Узагальнений формат подання «дії» одержано шляхом залучення оператору диз'юнкції:

$$act \equiv (u(v_1) \vee e'_1) \wedge (u(v_2) \vee e'_2) \wedge \ldots \wedge (u(v_n) \vee e'_n). \qquad (3.21)$$

Отже, вирази (3.20) і (3.21) є шаблонами подання елементів нижнього ієрархічного рівня для артефактів результуючого типу $a_4 \in A$.

Зауваження:

– серед застосовуваних засобів контролю відповідності елементів нижнього ієрархічного рівня у складі результуючих артефактів типу $a_4 \in A$ з відповідними елементами у складі первинних артефактів типу $a_1 \in A$ залучено оператори різниці і симетричної різниці теорії множин. Граничні випадки, окреслені виразом (3.20), зведено у табл. 3.5.

Таблиця 3.5 – Застосовуваний підхід до співставлення складових елементів нижньої страти для артефактів типів $a_1 \in A$ і $a_4 \in A$ (2.1)

№ з/п	Подання перед- і пост-умов у складі артефактів зазначених типів	
	$a_1 \in A$	$a_4 \in A$
1	2	3
1	$L(s) \setminus L(s') = \{(v_j, d_k)\}$, де $s' = R(s) \in S$.	Як передумова у складі виразу (3.18), залученого до складу виразів (3.20), (3.21).
2	$L(s') \setminus L(s) = \{(v_j, d_h)\}$.	Як пост-умова у складі виразу (3.18), залученого до складу виразів (3.20), (3.21).
3	$L(s) \Delta L(s') = = \{(v_j, d_k), (v_j, d_h)\}$.	Поєднання перед- і пост-умов для випадку виразу (3.20).

У свою чергу, для формування елементів верхнього ієрархічного рівня, у відповідності до протоколу обчислювального процесу (2.5), сполучимо формалізовані подання дій (3.20) і (3.21) на основі оператору кон'юнкції, із залученням зазначеного вище темпорального оператору зсуву модельного часу *X*:

$$\psi_i' \equiv act_1 \wedge X\ act_2 \wedge X^2\ act_3 \wedge \ldots \wedge X^{l-1}\ act_l, \qquad (3.22)$$

де ψ_i' – формалізоване подання протоколу (2.5).

При цьому верхній індекс оператору *X* є засобом зазначення кількості послідовних застосувань оператору: $X^1 \equiv X, X^2 \equiv XX, \ldots, X^{l-1} \equiv (X^{l-2})X$. Такий формат застосування оператору *X* є способом відображення змістового навантаження оператору передування, що фігурує у виразі (2.9), на основі виразних засобів формалізму TLA+.

У свою чергу, контекстне навантаження виразу (3.22) доречно інтерпретувати наступним чином:

– якщо за поточного відліку модельного часу у межах процесу ФВ висловлювання act_1 приймає істинне значення, то для наступного відліку модельного часу істинне значення вже має приймати висловлювання act_2, що подається відповідним виразом $(X\ act_2)$, і т. д.;

– нижній індекс висловлювань act (3.21) у складі виразу (3.22) задає відносний порядок «дій». При цьому передумовою для початкової «дії» act_1 є формалізоване подання φ_0 (2.23) на основі елементів розмітки $L(s_0)$ початкового стану $s_0 \in S$ СП.

З урахуванням пріоритету темпорального оператору *X*, який є вищим за пріоритет логічних операторів, немає потреби подавати вираз (3.22) у формі $\psi_i' \equiv act_1 \wedge (X\ act_2) \wedge (X^2\ act_3) \wedge \ldots \wedge (X^{l-1}\ act_l)$.

Співвідношення між елементами обох виокремлених ієрархічних рівнів для первинних артефактів типу $a_1 \in A$ і результуючих артефактів типу $a_4 \in A$ подано на рис. 3.8. При цьому елемент «перехід» $(s, s') \in R$ залучено у якості формуючого засобу, що дозволяє одержувати наступний елемент $s' = R(s) \in S$ на основі поточного елементу $s \in S$ у складі виразу (2.3).

У свою чергу, для співставлення поведінки (2.3) із результуючим виразом (3.22) застосуємо наступний підхід:

– продублюємо кожен елемент у складі виразу (2.3), починаючи з елементу $s_0 \in S$;

– із зміщенням на один елемент вправо, на основі суміжних елементів $s, s' \in S$ сформуємо відповідні пари елементів $(s, s') \in R$. У результаті отримаємо неоднорідну послідовність наступного вигляду:

$$\sigma_i = s_0, (s_0, s_1), (s_1, s_2), \ldots, (s_{l-2}, s_{l-1}), \tag{3.23}$$

де неоднорідність полягає у наступному:

– першим елементом є початковий стан СП $s_0 \in S$;

– у якості наступних елементів фігурують переходи $(s, s') \in R$, сформовані на основі суміжних станів $s, s' \in S$. При цьому у якості заключного елементу фігурує пара $(s_{l-2}, s_{l-1}) \in R$: другий елемент у складі пари характеризується властивістю рефлексивності: $R(s_{l-1}) = s_{l-1} \in S$ – як відмінна ознака заключного елемента у складі послідовності (3.23).

Рисунок 3.8 – Співвідношення між складовими артефактів виокремлених типів – первинними і результуючими

Вираз (3.23) є прообразом ФС для поведінки (2.3), формалізованої засобами структури Кріпке, а не засобами TLA+. У свою чергу, передумовою

для виконання переходу $(s,s')\in R$ є елемент множини $L(s)\setminus L(s')=\{(v_j,d_k)\}\subset AP$, який є передумовою у складі виразу (3.18). При цьому пост-умовою є елемент множини $L(s')\setminus L(s)=\{(v_j,d_h)\}\subset AP$. З урахуванням вищезазначеного, об'єднання множин $L(s)\setminus L(s')$ і $L(s')\setminus L(s)$ формує множину $L(s)\Delta L(s')=\{(v_j,d_k),(v_j,d_h)\}$, елементи якої уможливлюють формалізацію події (2.19) у складі елементів аналітичного рівня рис. 3.7.

Отже, згідно рис. 3.8, у якості засобів подання артефактів результуючого типу $a_4\in A$ було обрано саме засоби TLA+ – через властивості математичної строгості та модульності. Це уможливило шляхом, залучення логічних операторів $\vee$ і $\wedge$, а також темпорального оператору *X*, провести формування елементів обох виокремлених страт (рис. 3.8):

– елементів (3.20), (3.21) – на основі елементів (3.15), (3.18), рис. 3.7 – у якості елементів нижньої страти;

– елементів (3.22) – на основі елементів (3.20), (3.21) – у якості елементів верхньої страти.

Зауваження:

– у випадку незастосування правила (2.8) вираз (3.23) мав би наступний формат: $\sigma_i=s_0,(s_0,s_1),s_1,(s_1,s_2),s_2,\ldots,s_{l-2},(s_{l-2},s_{l-1})$, де фігурують на $(|S|-2)=|S\setminus\{s_0,s_{l-1}\}|$ елементів $s\in S$ більше. Підтвердженням тому слугує вміст стовпців № 5 табл. 3.2 і табл. 3.3, де елементи стовпця № 5 табл. 3.2 характеризуються виразом $(|S|-2)$, а елементи стовпця № 5 табл. 3.3 – виразом $|S|$.

З метою виокремлення однорідної складової, вираз (3.23) перепишемо наступним чином:

$$\sigma_i=s_0,\sigma_i', \tag{3.24}$$

де $\sigma_i' = (s_0, s_1), (s_1, s_2), \ldots, (s_{l-2}, s_{l-1})$ – зазначена однорідна складова.

Такий крок має на меті отримання прообразу результуючої ФС як артефакту типу $a_4 \in A$ – для випадку однієї динаміки $b_i \in B$ (2.3).

У свою чергу, артефакт типу $a_4 \in A$ на основі виразних засобів TLA+ будується із залученням виразу (2.23) як ФС розмітки початкового стану $L(s_0)$, а також виразу (3.22), – у відповідності до прообразу (3.24):

$$\psi_i \equiv \varphi_0 \wedge G\psi_i' \equiv \varphi_0 \wedge G\left[act_1 \wedge Xact_2 \wedge \ldots \wedge X^{l-1} act_l\right], \qquad (3.25)$$

де ψ_i – результуюча темпоральна формула, отримана індуктивним шляхом згідно викладеного на рис. 3.8 підходу, призначена до ФВ методом TLC або на основі розробленого розвитку зазначеного методу, викладеного у четвертому розділі, у відповідності до постановки задачі ФВ (3.1); G – темпоральний оператор «Globally»; φ_0 – ФС розмітки $L(s_0)$ початкового стану $s_0 \in S$.

Структура виразу (3.25) є демонстрацією відмінних рис залученого формалізму TLA+ – математичної строгості і модульності, що дозволяє подати результуючу ФС у формі єдиної темпоральної формули, індуктивно вибудовуючи останню у відповідності до запропонованого і застосованого підходу (рис. 3.8).

Зауваження:

– вираз (3.25) справедливий для випадку однієї динаміки $b_i \in B$ (2.3); є розширенням виразу (3.22) – за рахунок охоплення також і ФС розмітки $L(s_0)$ початкового стану $s_0 \in S$;

– вираз (3.25) є результуючим шаблоном, згідно якого на основі виразних засобів TLA+ формуються елементи верхнього ієрархічного рівня у складі артефактів типу $a_4 \in A$ (рис. 3.8).

Адаптуємо запропоновану постановку вирішуваної задачі ФВ, поданої виразом (3.1), до виразних можливостей формалізму TLA+. Для цього використаємо метод підстановки, а також техніку декомпозиції, – наступним чином:

– методом підстановки – у виразі (3.1), сформованому згідно озвученого вище запропонованого дуального підходу до постановки задачі ФВ, темпоральну формулу ψ деталізуємо згідно виразу (3.25). У результаті одержимо наступне подання виразу (3.1):

$$\left(M, \sigma_i \models \left(\varphi_0 \wedge G\psi_i'\right)\right) \vee \left(M, \sigma_i \not\models \left(\varphi_0 \wedge G\psi_i'\right)\right), \tag{3.26}$$

де, порівняно із постановкою (3.1), замість поведінки $b_i \in B$ (2.3) залучено результуючу неоднорідну конструкцію σ_i (3.23). Для виокремлення відповідних однорідних складових останньої проведемо її декомпозицію згідно виразу (3.24) у складі першого диз'юнкту виразу (3.26) наступним чином:

$$\left(\left(M, s_0 \models \varphi_0\right) \wedge \left(M, \sigma_i' \models \left(G\psi_i'\right)\right)\right) \vee \left(M, \sigma_i \not\models \left(\varphi_0 \wedge G\psi_i'\right)\right). \tag{3.27}$$

У виразі (3.27) істинність висловлювання $\left(M, s_0 \models \varphi_0\right)$ є передумовою, на підставі якої виконується наступний крок – встановлення у процесі ФВ істинності висловлювання $\left(M, \sigma_i' \models \left(G\psi_i'\right)\right)$. У свою чергу, для висвітлення застосованого підходу до перевірки істинності останнього також застосуємо техніку декомпозиції – по відношенню і до однорідної складової σ_i', і до

відповідної темпоральної формули $(G\psi_i')$, що має приймати істинне значення для кожного із елементів складової σ_i': якщо подати формулу $(G\psi_i')$ згідно виразу (3.25) наступними чином: $G[act_1 \wedge X\ act_2 \wedge ... \wedge X^{l-1} act_l]$, з урахуванням декомпозиції σ_i' на складові, то процес ФВ відбуватиметься у такі кроки:

– перевірка висловлювання $(M,(s_0,s_1)|= act_1)$. Якщо воно є істинним – виконати наступний крок;

– перевірка висловлювання $(M,(s_0,s_1)|= X\ act_2) \equiv (M,(s_1,s_2)|= act_2)$. У випадку його істинності – наступний крок;

– перевірка висловлювання $(M,(s_0,s_1)|= X^2 act_3) \equiv (M,(s_1,s_2)|= X\ act_3) \equiv$ $\equiv (M,(s_2,s_3)|= act_3)$, і т. д. – допоки не відбудеться одна з подій, викладених нижче у формі зауважень.

Зауваження:

– на певному з окреслених вище кроків відповідне висловлювання виявиться хибним. У такому випадку – у контексті прийнятого і застосованого дуального підходу до постановки задачі ФВ (3.1) – істинне значення прийме правий диз'юнкт, що фігурує у складі отриманих виразів (3.26) і (3.27);

– формулу $(G\psi_i')$ буде покроково опрацьовано повністю. У такому випадку істинності набувають ліві диз'юнкти у складі виразів (3.26) і (3.27).

Отже, отримані результуючі вирази (3.26) і (3.27) є формалізованими поданнями вирішуваної задачі ФВ, побудованим у відповідності до запропонованого, застосованого і поданого у формі постановки (3.1) дуального підходу. Відмінність виразів (3.26) і (3.27) при цьому полягає у наступних позиціях:

– враховано виразні можливості TLA+ як засобу опису ФС у формі, придатній до їх автоматизованої ФВ методом TLC, а також на основі

розробленого розвитку зазначеного методу, викладеного у наступному – четвертому – розділі;

– постановка (3.27) справедлива виключно для випадку однієї поведінки (2.3). Це характерно для опрацьованого вище послідовного сценарію як граничного випадку (табл. 3.1), а також для випадку окремого шляху серед множини альтернативних шляхів, загальну кількість яких було оцінено функцією $g_6(n)$ (3.9) – табл. 3.3 – для іншого граничного випадку – подання паралелізму згідно моделі чергування.

Для узагальненого випадку m поведінок (2.3) вираз (3.25) набуває наступного вигляду:

$$\psi \equiv \varphi_0 \wedge G[\psi'_1 \vee \psi'_2 \vee \ldots \vee \psi'_m], \tag{3.28}$$

де оператор диз'юнкції залучено у якості засобу формалізації альтернативності поведінок.

Запропоновані і застосовані правила одержання артефактів результуючого типу $a_4 \in A$ на основі попередньо одержаних артефактів типу $a_3 \in A$ (2.1) адресуємо і застосуємо у якості засобів регламентування аспектів виконання кроку 5 представленого методу синтезу ФС (табл. 3.6).

Згідно розробленої і викладеної у межах попереднього – другого – розділу моделі подання ПАС у формі ФС, озвучені правила призначені до застосування на рівні реалізації (рис. 2.4) – на відміну від запропонованих правил виконання попереднього кроку представленого методу – $(a_2, a_3) \in T$ (табл. 3.4), що є сполучною ланкою між рівнями – аналітичним і реалізації.

Отже, дані, зведені у формі табл. 2.1, табл. 3.4 і табл. 3.6, є поданнями запропонованих правил, що регламентують виконання кроків 3 – 5

представленого методу синтезу ФС, реалізованого у відповідності до стратифікованої моделі подання ПАС, викладеної у другому розділі [142].

Таблиця 3.6 – Співвідношення між складовими артефактів типів $a_3 \in A$ і $a_4 = T(a_3) \in A$ для здійснення переходу $(a_3, a_4) \in T$

Страти	Засоби рівня реалізації (рис. 2.4)	
	Формалізм PlusCal	Формалізм TLA+
1	2	3
1	Алгоритм згідно (2.5), (2.10).	Результуюча ФС (3.25), (3.28).
2	Модифікація значення елементу множини змінних (2.15) згідно (3.16).	Дія (3.20), (3.21).

Заключний крок 6 розробленого методу синтезу ФС, підхід до виконання якого викладено на рис. 3.3, адресується у якості засобу контролю відповідності результуючих артефактів типу $a_4 \in A$ первинним артефактам типу $a_1 \in A$, що реалізується згідно викладеного вище розробленого методу контролю відповідності як допоміжного засобу у складі методу синтезу ФС. Це, у свою чергу, є опосередкованим засобом контролю достовірності результатів ФВ, виконуваної на основі ФС як артефактів типу $a_4 \in A$.

Дієвість прикладного застосування розробленого і представленого у межах розділу методу синтезу ФС, а також розробленого допоміжного методу контролю відповідності результуючої ФС, призначених до застосування згідно моделі подання ПАС, викладеної у попередньому розділі, продемонстровано, у тому числі, на прикладі сценарію синтезу ФС для заданої топології програмно-конфігурованої мережі [143–145].

ВИСНОВКИ ДО РОЗДІЛУ 3

Таким чином, у розділі викладено розроблений метод синтезу ФС ПАС, що базується на представленій у другому розділі моделі подання ПАС.

Методом передбачається залучення структури Кріпке у якості засобу аналітичного подання СП, що будується у процесі ФВ ФС. Складові названої структури запропоновано інтерпретувати на основі виразних засобів числення процесів CSP, що забезпечує механізм формалізації, що передує реалізації ФС на основі засобів темпоральної логіки дій TLA. Остання, у свою чергу, є засобом уможливлення здійснення ФВ ФС методом TLC в автоматизованому режимі.

Результатом застосування розробленого методу є ФС на основі виразних засобів формалізму TLA+ темпоральної логіки дій TLA.

Запропонований метод передбачає застосування трійок, аксіом Гоара, а саме – правил композиції, виведення і умовного оператора. У свою чергу, результатом застосування правила композиції стало зменшення результуючого числа рядків псевдокоду ФС – за рахунок використання у якості передумов виникнення подій не формалізованих подань станів, а формалізованих подань попередніх подій.

Також було залучено алгоритмічну мову PlusCal – у якості засобу формалізованого подання ПАС, що передує одержанню результуючої ФС на основі виразних засобів формалізму TLA+. Це було зроблено для формування архітектурної складової результуючої ФС.

Для кількісного оцінювання одержуваного корисного ефекту від застосування правила композиції Гоара, а також для оцінювання просторової складності задачі ФВ, вирішуваної на основі результуючої ФС, було

запропоновано, обґрунтовано і експериментально досліджено набір відповідних оціночних функцій.

Експериментальні дослідження проведено для граничних випадків: для ФС, побудованих згідно послідовного шаблону; для ФС із поданням паралелізму згідно моделі чергування. Для автоматизації процесу одержання зазначених ФС, із заданим числом змінних станів, було розроблено і застосовано відповідне програмне забезпечення.

Доцільність застосування правила композиції при синтезі ФС на основі засобів TLA+ було підтверджено експериментальним шляхом – на основі озвучених вище шаблонів для граничних випадків. Одержуваний при цьому корисний ефект було оцінено з позиції зменшення числа рядків результуючої ФС на основі виразних засобів TLA+. Для випадку послідовного шаблону, і числа змінних станів $n = 2^1, 2^2, ..., 2^8$, значення показника корисного ефекту склало від близько 18 % до близько 33 % відповідно. Для граничного випадку із поданням паралелізму згідно моделі чергування, і числа змінних станів $n = 2^2, 2^3, 2^4$, значення показника корисного ефекту склало від близько 22 % до близько 0 % відповідно.

Отримані результати проведених досліджень дали підстави вважати, що значення показника одержуваного корисного ефекту залежить як від числа змінних станів ФС, так і від архітектурної складової ФС: для ФС, побудованих згідно послідовного шаблону, спостерігається зростання корисного ефекту від застосування правила композиції із зростанням числа змінних станів; у свою чергу, для випадку подання паралелізму у ФС згідно моделі чергування має місце зворотній процес – зменшення одержуваного корисного ефекту.

Як показник стрімкості зростання просторових витрат на подання ПАС у ФС на основі засобів TLA+ є характер зростання розміру відповідного файлу-артефакту – для граничного випадку подання паралелізму у ФС згідно моделі

чергування: наприклад, для випадку $n = 2^3$ розмір файлу склав 7 КБ; у свою чергу, для випадку $n = 2^4$ – вже 210727 КБ.

Відповідність результуючих артефактів – ФС, одержуваних на основі розробленого методу, первинним артефактам – блок-схемам алгоритмів, UML-діаграмам дій – підтверджено на рівні архітектурної складової ФС – для графів-подань СП, одержуваних аналітичним і експериментальним шляхами: для випадку подання ФС згідно послідовного шаблону контроль відповідності проведено згідно розробленого і представленого у межах розділу допоміжного засобу – методу контролю відповідності – за показниками кількості станів СП, що будується у процесі ФВ, а також за показником глибини обходу простору станів СП. Для іншого граничного випадку – подання паралелізму у ФС згідно моделі чергування було залучено також допоміжні засоби – запропоновані оціночні функції просторових витрат: як у контексті витрат оперативної пам’яті обчислювальної системи на подання ПАС у формі ФС, так і у контексті відповідних витрат на збереження СП, що будується у процесі ФВ. Зазначений розроблений метод контролю відповідності застосовано на заключному кроці розробленого методу синтезу ФС.

Розроблений метод контролю відповідності результуючих ФС первинним артефактам також опрацьовано у якості засобу опосередкованого контролю достовірності результатів формальної верифікації ФС, одержуваних на основі представленого методу синтезу ФС. У випадку, якщо на основі розробленого методу контролю відповідності було підтверджено відповідність результуючих ФС, результати формальної верифікації ФС, виконуваної на основі методу перевірки на моделі TLC, а також на основі розробленого розвитку зазначеного методу, викладеного у наступному – четвертому – розділі, вважатимемо достовірними.

РОЗДІЛ 4
РОЗВИТОК МЕТОДУ ПЕРЕВІРКИ НА МОДЕЛІ

У розділі викладено розроблений розвиток поширеного методу формальної верифікації TLC. Проведений розвиток полягає у комбінуванні методів обходу вершин графу-подання СП у процесі автоматизованої ФВ – методу обходу у ширину (BFS, Breadth-first Search) і методу обходу у глибину (DFS, Depth-first Search) теорії графів. Одержуваний при цьому корисний ефект оцінено за ітеративного підходу до організації процесу ФВ ФС.

Експериментальні дослідження проведено на основі і синтетичних, і предметно орієнтованих сценаріїв, поданих у формі відповідних артефактів. У першому випадку охоплено граничні варіації: коли ФС побудовано згідно послідовного сценарію виникнення подій і коли паралелізм подано у ФС згідно моделі чергування. При цьому у якості предметно орієнтованого аналогу останнього зазначеного граничного випадку розглянуто сценарій організації розподілених Grid-обчислень. У свою чергу, досліджувані предметно орієнтовані артефакти охоплюють наступні області застосування СКП: електроенергетика, аерокосмічна галузь.

При проведенні експериментальних досліджень проаналізовано показники як обчислювальної, так і просторової складностей і базового методу TLC, і розробленого розвитку названого методу.

Для оцінювання корисного ефекту від прикладного застосування розробленого розвитку методу TLC одержано відповідні оціночні функції. Також здійснено оцінювання одержуваного корисного ефекту від залучення мультипоточності, доступної на обчислювальній системі, на якій проведено дослідження.

Сформульовано рекомендації для прикладного застосування розробленого розвитку методу TLC.

4.1 Постановка вирішуваної задачі

Оскільки процес проєктування СКП є, як правило, ітераційним, особливої актуальності набуває скорочення часових витрат на застосування методів МС. Вирішенню цієї задачі на прикладі методу TLC і присвячено даний розділ.

Вирішувана у розділі задача полягає в удосконаленні методу перевірки на моделі TLC з точки зору зниження часових витрат на його застосування.

Ідея в основі застосовуваного підходу до вирішення сформульованої задачі полягає в наступному:

– експериментально дослідити і проаналізувати дві альтернативні реалізації методу TLC, що будуються, відповідно, на основі обходів вершин графу СП в ширину і в глибину;

– дослідження реалізацій методу провести з позицій як просторової, так і обчислювальної складностей вирішуваних при цьому задач ФВ.

Відомо, що, в залежності від способу подання графу $G = \langle S, R \rangle$ у складі структури (1.2), – матрицею чи списком суміжності – витрати ОП на обхід елементів множини S оцінюються, відповідно, як $O(|S^2|)$ чи $O(|S| + |R|)$ [146, с. 591]. При цьому, в залежності як від структури ФС, так і від специфіка реалізації методу обходу вершин графу G, показники просторової складності вирішення задачі ФВ можуть варіюватися. У зв'язку із цим, для формулювання рекомендацій стосовно прикладного застосування реалізацій методу TLC проводяться відповідні експериментальні дослідження.

Для проведення експериментальних досліджень залучаються спеціалізовані програмні засоби, створені для проведення досліджень попереднього розділу, а саме – засоби автоматизованого синтезу ФС – на основі послідовного шаблону і шаблону із представленням паралелізму згідно моделі чергування.

Варіативність експериментальної складової окреслених досліджень подано у табл. 4.1.

Таблиця 4.1 – Напрями експериментального дослідження методу

Шаблон ФС	Реалізація методу TLC на основі	
	BFS	DFS
1	2	3
Послідовний	+	+
Із поданням паралелізму згідно моделі чергування	+	+

Із табл. 4.1 видно, що маємо чотири варіації. Цей крок зроблено для формулювання комплексного судження стосовно показників ефективності роботи методу TLC. У якості таких розглянуто часові витрати на здійснення ФВ ФС, а також витрати ОП.

Більше того, для охоплення прикладного аспекту у якості сценарію предметної області розглядається блок-схема алгоритму роботи БУК (блоку управління конфігурацією) БЦОК (бортового цифрового обчислювального комплексу) КА.

Окрім цього досліджується алгоритм контролю вихідного стану регістрів модуля бортового комп’ютера (БК) пристрою введення/виведення (ПВВ) БЦОК. Для цього алгоритму синтезується ФС, яка перевіряється методом TLC.

4.2 Дослідження методу перевірки на моделі

4.2.1 Аналіз сценарію предметної області

У якості прикладу розглянемо фрагмент блок-схеми алгоритму роботи БУК БЦОК КА.

Доцільність та концепцію застосування темпоральної догіки TLA у якості засобу формалізації ПАС, поданих у формі блок-схем, вже було окреслено [147].

Текстовий опис початкового фрагменту алгоритму:

– значення показника стану контрольно-перевірочної апаратури (КПА) визначає подальший сценарій роботи алгоритму;

– якщо ця ознака рівна 0, виконується запуск ПСЕП (підсистема електропостачання);

– якщо 1 – спочатку виконується запуск БУК, а вже потім – ПСЕП.

Блок-схему описаного алгоритму подано на рис. 4.1.

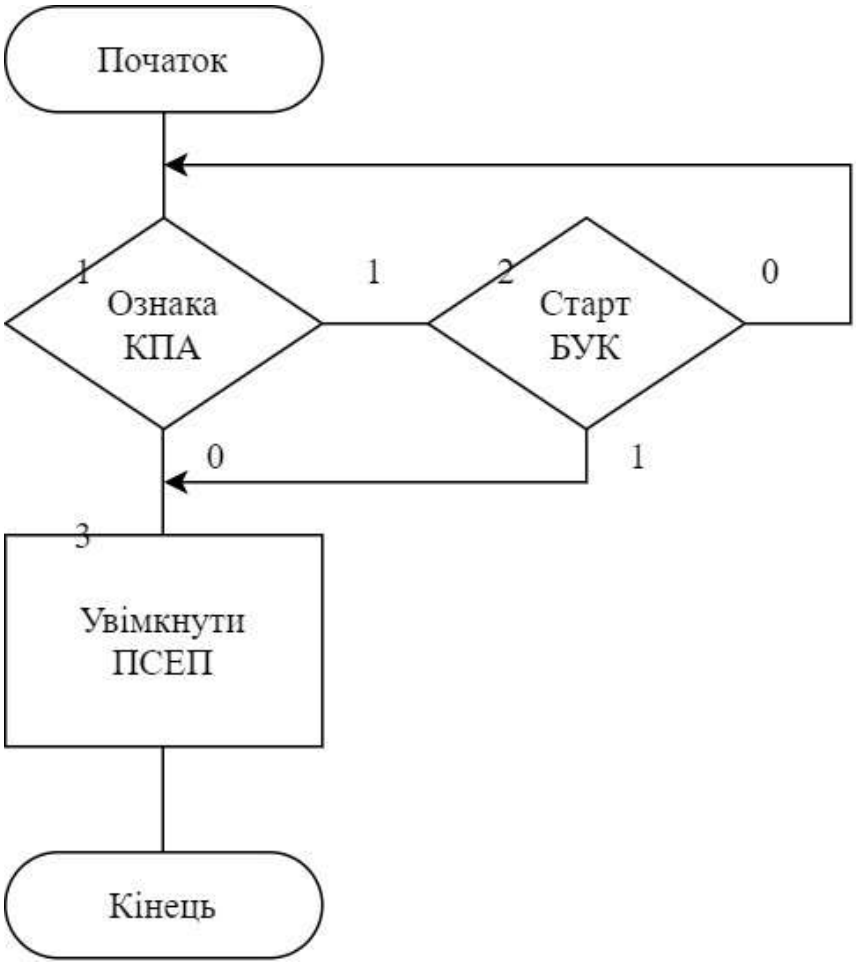

Рисунок 4.1 – Блок-схема фрагменту алгоритму роботи БУК

Згідно алгоритму (рис. 4.1), можливі два сценарії роботи БУК:

– послідовне виконання блоків 1, 3;

– послідовне виконання блоків (1, 2), 3.

Застосуємо запропонований метод синтезу ФС (розділ 2). Для цього представимо рис. 4.1 структурою (1.2). Кожен із нумерованих блоків рис. 4.1 представимо відповідною змінною станів (2.15): $V=\{v_1,v_2,v_3\}$, де *j*-й індекс $v_j \in V,(j=1,2,3)$ відповідає порядковому номеру відповідного блоку. При цьому $D=\{0,1\}$. $AP=\{(v_j,0)\}\cup\{(v_j,1)\}$: $(v_j,0)\in AP$ – фрагмент коду, представлений *j*-м блоком, ще не виконано або результат його виконання відповідає «ні»-гілці (блок умовного переходу); $(v_j,1)\in AP$ – фрагмент коду, представлений *j*-м

блоком, було виконано або результат його виконання відповідає «так»-гілці (блок умовного переходу).

З рис. 4.1 бачимо, що, в залежності від значення ознаки КПА, представленої змінною $v_1 \in V$, маємо два початкових стани СП: $s_0, s_1 \in S_0 \subset S$: $L(s_0) = \{(v_1,0),(v_2,0),(v_3,0)\}$, $L(s_1) = \{(v_1,1),(v_2,0),(v_3,0)\}$. Відповідну UML-діаграму станів подано на рис. 4.2.

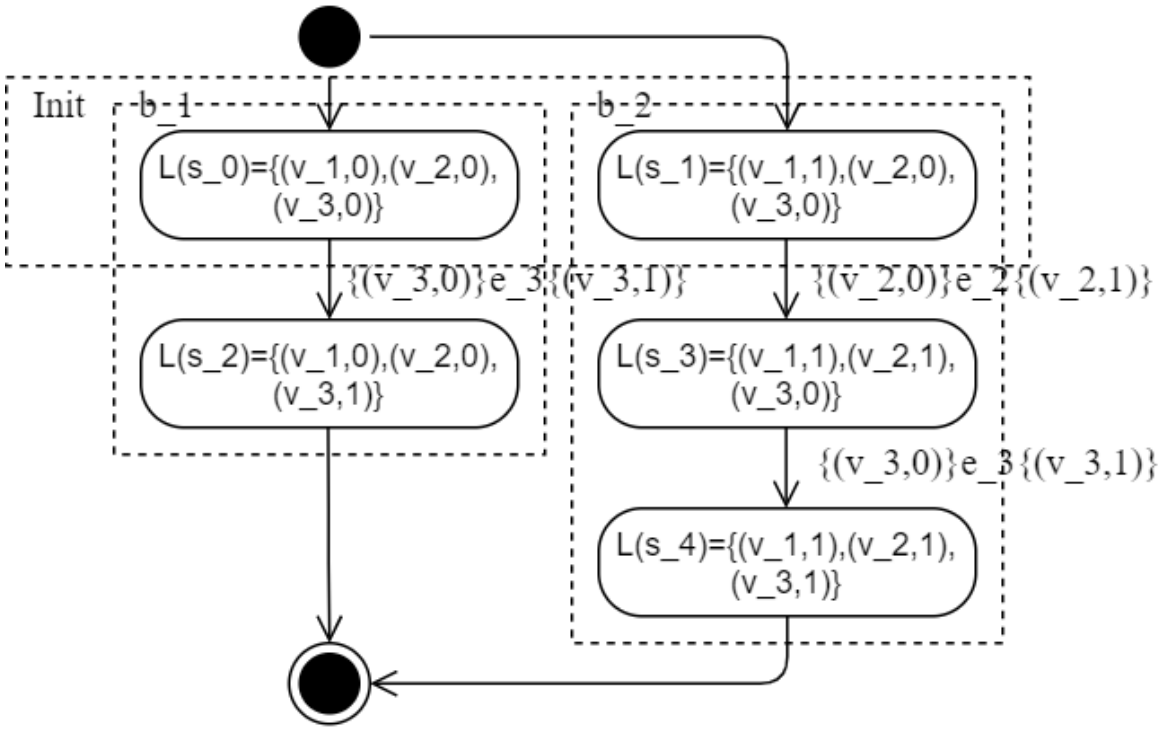

Рисунок 4.2 – UML-діаграма станів СП

На рис. 4.2 відображено структуру (1.2) для рис. 4.1: $|S| = 5$. Пунктирною прямокутною областю Init окреслено множину альтернативних початкових станів СП $S_0 = \{s_0, s_1\} \subset S$: $s_0(v_1) = 0 \in D$, $s_1(v_1) = 1 \in D$, тобто $L(s_0) \Delta L(s_1) = \{(v_1,0),(v_1,1)\}$. Стани $s_0, s_1 \in S$ ініціюють відповідні поведінки $b_1, b_2 \in B$ (2.3): $b_1 = s_0, s_2$, $b_2 = s_1, s_3, s_4$. При цьому елементи множини

$S \setminus S_0 = \{s_2, s_3, s_4\}$ одержуються наступним чином: $s_2 = R(s_0)$, $s_3 = R(s_1)$, $s_4 = R(s_3) = R(R(s_1))$.

4.2.2 Створення і перевірка специфікації

На основі $b_1, b_2 \in B$: $b_1 = s_0, s_2$, $b_2 = s_1, s_3, s_4$, сформуємо відповідні протоколи процесів $p_1, p_2 \in P$ CSP (2.5): $p_1 = \langle e_3 \rangle$, $p_2 = \langle e_2, e_3 \rangle$. Для формалізації елементів наведених структур підготуємо відповідні перед- і пост-умови подій. Для цього застосуємо операцію симетричної різниці: $L(s_0) \Delta L(s_2) = \{(v_3,0),(v_3,1)\}$, де $(v_3,0) \in AP' \subset AP$ – передумова виникнення події e_3, а $(v_3,1) \in AP'' \subset AP$ – пост-умова (2.6), (2.22); аналогічним чином $L(s_1) \Delta L(s_3) = \{(v_2,0),(v_2,1)\}$; $L(s_3) \Delta L(s_4) = L(s_0) \Delta L(s_2) = \{(v_3,0),(v_3,1)\}$. Отже, згідно (2.6), маємо таку формалізацію подій у вигляді трійок Гоара (рис. 3.2): $\{(v_2,0)\} e_2 \{(v_2,1)\}$, $\{(v_3,0)\} e_3 \{(v_3,1)\}$, де, згідно (2.19), $e_2 \equiv \neg(v_2,0) \vee X(v_2,1)$, $e_3 \equiv \neg(v_3,0) \vee X(v_3,1)$.

Коментарі до відповідної ФС на основі TLA+ подано у табл. 4.2. Повну версію ФС подано у додатку В. Дану ФС створено вручну – згідно рис. 4.1.

З табл. 4.2 бачимо, що протокол $p_1 \in P$ формалізується виразом R02, що є TLA-представленням відповідного переходу $(s_0, s_2) \in R$ структури (1.2). Аналогічним чином інтерпретуються і вирази R13, R34. Послідовність подій у межах протоколу $p_2 = \langle e_2, e_3 \rangle \in P$ задається виразом (R13 /\ R34) як кон'юнкція. У рядку 8 подано результуючу темпоральну формулу ψ, де [] – символьне представлення темпорального оператору «Globally». Для успішного проходження ФВ ψ має приймати істинне значення $\forall s \in S$ (1.2).

Таблиця 4.2 – Результуюча формальна специфікація

№ з/п	Фрагмент ФС	Коментарі
1	2	3
1	EXTENDS Naturals VARIABLES v1, v2, v3	Визначили множину змінних станів.
2	Invariant == v1 \in {0,1} /\ v2 \in {0,1} /\ v3 \in {0,1}	Задали множини допустимих значень змінних.
3	Init == v1 \in {0,1} /\ v2 = 0 /\ v3=0	Формалізували множину початкових станів $S_0 = \{s_0, s_1\} \subset S$.
4	R02 == (v3' = IF (v1=0 /\ v3=0) THEN 1-v3 ELSE v3) /\ UNCHANGED <<v1,v2>>	Формалізували перехід $(s_0, s_2) \in R$ на основі події e_3.
5	R13 == (v2' = IF (v1=1 /\ v2=0) THEN 1-v2 ELSE v2) /\ UNCHANGED <<v1>>	Формалізували перехід $(s_1, s_3) \in R$ на основі події e_2.
6	R34 == (v3' = IF v2=1 /\ v3=0 THEN 1-v3 ELSE v3) /\ UNCHANGED <<v1>>	Формалізували перехід $(s_3, s_4) \in R$ на основі події e_3.
7	Next == R02 \/ (R13 /\ R34)	Формалізували поведінки $b_1, b_2 \in B$ згідно відповідних протоколів $p_1, p_2 \in P$.
8	Spec == Init/\[][Next]_<<v1,v2>>	Результуюча формула ψ (1.1).

У результаті автоматизованої перевірки ФС методом TLC було підтверджено коректність останньої – помилок не виявлено.

Для перевірки адекватності синтезованої ФС було застосовано наступний підхід: параметри ФС порівняно із параметрами аналітичного представлення (1.2). Параметри наступні: $|S|$; глибина обходу графу СП. Останній параметр дає можливість перевірити глибину обходу СП для кожної із динамік. Названі параметри одержано автоматизовано – із даних файлу-звіту роботи методу TLC. Отже, перевірка адекватності синтезованої ФС робиться шляхом порівняння характеристик графів – в основі структури (1.2) і в основі результуючої ФС. Для розглянутого прикладу озвучені параметри співпали: $|S| = 5$, а глибини обходів для динамік $b_1, b_2 \in B$ склали 2 і 3 відповідно. Це дає підстави стверджувати, що результуюча ФС є адекватною, а результати ФВ на її основі – достовірними.

Експериментальні дослідження проведено на програмно-апаратній платформі наступної конфігурації: середовище виконання – Java Runtime Environment (64 bit, build 1.8.0_251-b08); версія реалізації методу TLC – 2.14 (від 10 липня 2019 р.); центральний процесор – 4 ядра, 8 потоків, частота – 3,8 ГГц; обсяг ОП – 16 ГБ.

Мета експериментальних досліджень наступна: перевірити синтезовану ФС кожною із двох альтернативних реалізацій методу TLC – на основі обходу вершин графу СП методами обходу BFS і DFS, і порівняти супутні цим процесам часові витратами для виявлення ефективнішої реалізації. При цьому застосовано наступний підхід: результат заміру названих витрат для кожної із варіацій зафіксовано як середнє арифметичне 10 замірів – $\overline{t_{BFS}}$ і $\overline{t_{DFS}}$ відповідно; для автоматизації названого процесу було створено спеціальну програму, код якої подано у додатку Г.

Одержано наступні результати: $\overline{t_{BFS}} = 0{,}895$ с, $\overline{t_{DFS}} = 0{,}355$ с. З одержаних результатів бачимо, що «прискорення» від переходу від BFS- до DFS-реалізації склало 2,52 рази. Отже, застосування DFS-реалізації методу TLC для нашого

випадку є у 2,52 рази ефективнішою (з позиції супутніх часових витрат), у порівнянні із альтернативною BFS-реалізацією.

При цьому варто зазначити, що для одержання значення $\overline{t_{DFS}}$ у коді програми (додаток Г) потрібно вказати також дані стосовно максимальної глибини обходу станів СП (параметр «dfid»). Це вагомий недолік DFS-варіації з позиції автоматизації, оскільки одержати назване значення можна лише попередньо виконавши BFS-обхід. Більше того, спробуємо також оцінити просторову складність вирішуваних при цьому задач перевірки на моделі. У якості відповідного показника застосуємо загальну кількість синтезованих у процесі перевірки станів СП. Для BFS-реалізації вона склала 12 – по 5 для висловлювань R02 і (R13 /\ R34) (табл. 4.2), 2 – початкові стани. Для DFS-реалізації – 14 – початкові стани дублюються для кожної із динамік $b_1, b_2 \in B$. У даному контексті BFS-реалізація є приблизно на 16,7 % ефективнішою, у порівнянні із альтернативною DFS-варіацією.

Підсумувати одержані результати можна наступним чином: DFS-реалізація є приблизно у 2,52 рази ефективнішою за BFS-альтернативу з позиції супутніх процесу ФВ часових витрат. При цьому вона має ряд недоліків (у порівнянні із BFS-альтернативою):

– не є самодостатньою – попередньо потрібно визначити глибину обходу вершин графу СП, застосувавши BFS-реалізацію;

– характеризується більшими вимогами до обсягу доступної пам’яті системи – приблизно на 16,7 %. Це означає, що просторова складність вирішуваної при цьому задачі ФВ є більшою.

4.2.3 Автоматизація процесу синтезу специфікації

У попередньому пункті ФС було створено вручну (табл. 4.2).

У даному пункті розкриємо аспект автоматизації процесу синтезу ФС. Для цього застосуємо підхід в основі запропонованого у попередньому розділі методу, а саме – створимо алгоритмічну метамодель – PlusCal-специфікацію, на основі якої синтезуємо результуючу ФС на мові TLA+. У якості вхідних даних візьмемо алгоритм, поданий на рис. 4.1.

Мета такого кроку наступна: підтвердити припущення стосовно достовірності даних, одержуваних у результаті ФВ ФС, синтезованої згідно запропонованого методу. Для цього порівняємо значення показників для ФС, створеної вручну (табл. 4.2), із значеннями, одержуваними для створюваної автоматизованим шляхом ФС – згідно підходу в основі запропонованого у розділі 2 методу. У якості порівнюваних показників залучимо наступні:

– глибина обходу графу СП і число виявлених при цьому станів – показники адекватності синтезованої ФС;

– часові витрати $\overline{t_{BFS}}$ і $\overline{t_{DFS}}$ – показники обчислювальної складності вирішуваної задачі;

– загальне число синтезованих при ФВ станів – показник просторової складності вирішуваної задачі.

Створену метамодель подано у табл. 4.3, синтезовану на її основі ФС на мові TLA+ – у додатку Д.

Таблиця 4.3 – Алгоритмічне подання ФС засобами PlusCal

№ з/п	Фрагмент псевдокоду	Коментарі
1	2	3
1	(* --algorithm spec variables v1 \in BOOLEAN, v2 \in BOOLEAN, v3 \in BOOLEAN; begin	Задання інваріантів, точки початку алгоритму – begin.

2	Init: either v1 := TRUE;
	or v1 := FALSE; end either;
	v2 := FALSE; v3 := FALSE;
		Формалізація $S_0 = \{s_0, s_1\} \subset S$.

3	r02: if ~v1 /\ ~v3 then v3 := TRUE elsif v1 /\ ~v2 then v2 := TRUE end if;	Формалізація переходів $(s_0, s_2) \in R$, $(s_1, s_3) \in R$ (рис. 4.2).
4	r34: if v2 /\ ~v3 then v3 := TRUE end if; end algorithm; *)	Формалізація переходу $(s_3, s_4) \in R$ (рис. 4.2).

У табл. 4.3 кожна мітка («Init:», «r02:» і «r34:») окреслює події, що матимуть місце у межах відповідного кроку модельного часу. Мітки є прообразами відповідних висловлювань результуючої ФС (додаток Д). Із додатку видно, що синтезована ФС відрізняється від ФС, створеної вручну (табл. 4.2). Це, проте, не суперечить твердженню, що формалізувати певну властивість системи можна різними шляхами.

Результати проведених експериментальних досліджень майже повністю співпали із результатами для випадку табл. 4.2, за виключенням значення $\overline{t_{BFS}}$: було 0,895 с, стало – 0,893 с. Різницю у $2 \cdot 10^{-3}$ с можна вважати несуттєвою.

Отже, за результатами порівняння значень виокремлених показників можна стверджувати, що ФС, одержана згідно запропонованого методу синтезу ФС, є адекватною, а дані-результат ФВ на її основі – достовірними.

Співвідношення у 2,52 рази між значеннями показників $\overline{t_{BFS}}$ і $\overline{t_{DFS}}$ зберіглося.

Зробимо припущення, що показники ефективності застосування тієї чи іншої реалізації методу TLC суттєво варіюються, в залежності від структури ФС (рис. 4.1). У якості фактору, що визначає таку структуру, розглянемо варіативність поведінок, регламентованих ФС. Це означає, що у якості граничних умов розглянемо два сценарії:

– ФС задає послідовний алгоритм роботи системи – без умовних переходів і циклів. Це означає, що ФС задає єдину поведінку системи, яка не передбачає варіативної складової [83];

– ФС задає алгоритм, що передбачає паралелізм; паралелізм при цьому представимо на основі моделі чергування. При цьому матимемо порядку 10^6 альтернативних поведінок системи [22].

Для кожного із сценаріїв дослідимо часові і обчислювальні витрати, пов’язані із застосуванням методу перевірки на моделі TLC. При цьому розглядаються дві альтернативні реалізації методу:

– шляхом обходу вершин графу СП на основі структури Кріпке (1.2) методом обходу в ширину (BFS, Breadth-first Search); застосовується за замовчанням;

– шляхом обходу методом обходу в глибину (DFS, Depth-first Search); недоліком є потреба вказування глибини обходу.

Мета дослідження – виявити переваги і недоліки кожної із реалізації методу TLC за наступними показниками:

– швидкодія – часові витрати на вирішення задачі ФВ;

– придатність до автоматизації.

Це дозволить розвинути названий метод за вищеназваними показниками.

Для проведення експериментальних досліджень методу також було реалізовано відповідну програмну систему, що дозволило синтезувати тестові дані – формальні специфікації (ФС) – як послідовної структури, так і структури з елементами паралелізму.

4.2.4 Дослідження послідовного сценарію

У даному випадку елементи множини V (2.15) представляють собою фрагменти коду, що виконуються послідовно [83].

Побудуємо структуру (1.2).

Для цього, керуючись заданою блок-схемою алгоритму, сформуємо множини V і D (2.15), (2.16): $2 \cdot |V| = |S| - 1$; $D = \{0, 1, 2\}$. У результаті, згідно (2.17), матимемо множину AP:

– $(v_j, 0) \in AP$ – черга виконання ще не дійшла до j-го блоку коду;

– $(v_j, 1) \in AP$ – j-й блок знаходиться у процесі виконання;

– $(v_j, 2) \in AP$ – виконання j-го блоку завершено.

Застосовуваний підхід до проведення експериментальних досліджень:

– синтезувати 10^2 ФС на мові TLA+ для $n = 2^1, 2^2, \ldots, 2^8$;

– оцінити часові витрати на ФВ методом TLC, як на основі BFS-, так і на основі DFS-обходів вершин графу СП, що дозволить підтвердити (чи частково спростувати) висновки стосовно переваг/недоліків кожної із реалізацій методу, зроблені по відношенню до результатів проведеного тематичного дослідження (рис. 4.1).

– проаналізувати одержані результати та сформулювати рекомендації стосовно удосконалення методу TLC.

На основі поданої інтерпретації елементів множини AP сформуємо єдиний початковий стан СП $s_0 \in S_0$, тобто $|S_0| = 1$, $L(s_0) = \{(v_1,0),(v_2,0),...,(v_n,0)\}$. Це означає, що на початковому етапі черга не дійшла до жодного з n блоків.

Згідно (2.23), сформуємо розмітку початкового стану $L(s_0)$ як кон'юнкцію: $\varphi_0 \equiv (v_1,0) \wedge (v_2,0) \wedge ... \wedge (v_n,0)$.

Оскільки $AP = V \times D$ (2.18), $|D| = 3$, а $v'_j = v_j + 1$, де $v_j \in V$, маємо $|S| = 2 \cdot n + 1$ станів СП, де $n = |V|$, тобто $S = \{s_0, s_1, ..., s_{2 \cdot n}\}$.

Наступний крок – формалізувати події (2.19) у межах концепту дії TLA, тобто – подати перехід $(s, s') \in R$ засобами формалізму TLA+ згідно (2.38). Це означає, що на кожному кроці процесу дискретно-подійного моделювання, що відбувається при автоматизованій ФВ, матимуть місце одна подія (2.19) і $n-1$ псевдоподій (2.41).

Отже, для $2 \cdot n + 1$ станів маємо $2 \cdot n$ переходів.

Наступний крок – агрегувати $2 \cdot n$ переходів згідно (2.45) для формалізації *i*-ї динаміки. Наприклад, для $n = 2^1$ матимемо $Next_1 \equiv a_1 \vee a_2 \vee a_3 \vee a_4$, де a_1 – ФС $(s_0, s_1) \in R$, a_2 – ФС $(s_1, s_2) \in R$, a_3 – ФС $(s_2, s_3) \in R$, a_4 – ФС $(s_3, s_4) \in R$. Для $n = 2^2$ матимемо $Next_1 \equiv a_1 \vee a_2 \vee ... \vee a_8$, і т. д.

Отже, для структури Кріпке для $n = 2^1$ матимемо наступні характеристики:

– $V = \{v_1, v_2\}$;

– згідно (2.15) – (2.17), $AP = \{(v_1,0),(v_1,1),(v_1,2),(v_2,0),(v_2,1),(v_2,2)\}$;

– множина станів $S = \{s_0, s_1, s_2, s_3, s_4\}$, де $s_0 \in S$: $L(s_0) = \{(v_1,0),(v_2,0)\}$ – початковий стан, $s_1 = R(s_0) \in S$: $L(s_1) = \{(v_1,1),(v_2,0)\}$ – наступний стан, і т. д., $s_4 = R(s_3) \in S$: $L(s_4) = \{(v_1,2),(v_2,2)\}$ – заключний стан СП на основі (1.2).

Відповідна ФС на мові TLA+ подана у табл. 4.4.

Таблиця 4.4 – ФС, синтезована згідно запропонованого методу

№ з/п	Фрагмент ФС	Коментарі
1	2	3
1	EXTENDS Naturals VARIABLES v1, v2	Визначити множину змінних станів;
2	Invariant == /\ v1 \in (0..2) /\ v2 \in (0..2)	Задати інваріанти;
3	Init == (v1=0) /\ (v2=0)	Задати початковий стан системи;
4	R_01 == /\ v1' = IF Init THEN v1+1 ELSE v1 /\ UNCHANGED <<v2>> R_12 == /\ v1' = IF R_01 THEN v1+1 ELSE v1 /\ UNCHANGED <<v2>> R_23 == /\ v2' = IF R_12 THEN v2+1 ELSE v2 /\ UNCHANGED <<v1>> R_34 == /\ v2' = IF R_23 THEN v2+1 ELSE v2 /\ UNCHANGED <<v1>>	Формалізувати події і псевдоподії у межах концепту дії;
5	Next == R_01 \/ R_12 \/ R_23 \/ R_34	Формалізувати процес;
6	Spec == Init/\[][Next]_<<v1,v2>>	Рез. формула.

Відповідна діаграма станів СП, заданої ФС (табл. 3.4), представлена на рис. 4.3.

Для $n = 2^2, 2^3$ і т. д. структура (1.2) будується аналогічним чином. Для $n = 2^8$, відповідно, матимемо $2 \cdot n + 1 = 513$ станів.

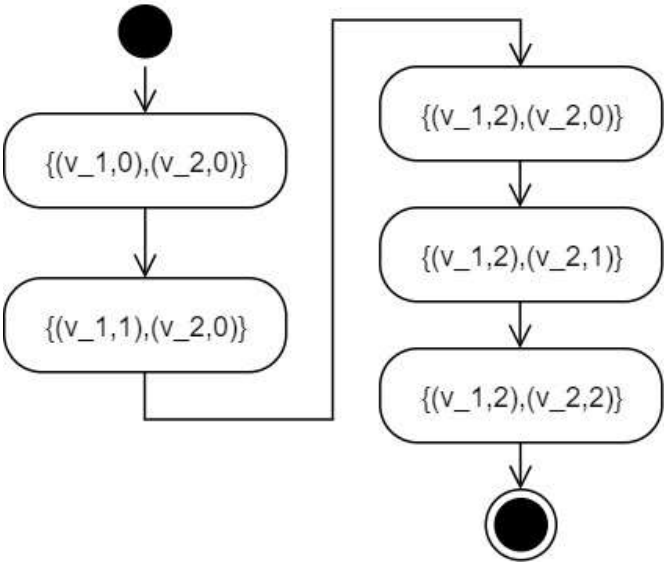

Рисунок 4.3 – Діаграма станів СП

Наступний крок – одержати результуючу темпоральну формулу згідно (2.48), здійсненність якої підлягає перевірці $\forall s \in S$ (1.2). Для цього залучено обидві реалізації методу перевірки на моделі – BFS і DFS.

Експериментальні дослідження проведено на програмно-апаратній платформі наступної конфігурації: процесор – AMD K10, 3,0 ГГц; 2 ГБ ОП стандарту DDR3; операційна система – MS Windows 7; середовище виконання – Java Runtime Environment v. 1.7 (JRE).

При проведенні експериментальних досліджень застосовувалися реалізації методу TLC версії 2.14.

Одержані експериментальні дані подано у табл. 4.5 [83].

У табл. 4.5 показники $\bar{t}_{BFS}$ і $\bar{t}_{DFS}$ є середніми арифметичними 10 замірів. Значення $|S|$ є також глибиною обходу СП, заданої ФС, що є числом пройдених станів – від $s_0 \in S$ – до $s_l \in S$, включаючи $s_0, s_l \in S$, а не числом дуг між відповідними вершинами графу СП; стовпець $\bar{t}_{BFS} / \bar{t}_{DFS}$ відображає відношення між часовими витратами на здійснення ФВ методом TLC на основі BFS- і DFS-

реалізацій відповідно. Більше того, значення $|S|$, у даному випадку, задається у якості вхідного параметру для запуску DFS-обходу.

Таблиця 4.5 – Результати перевірки реалізацій методу TLC

№ з/п	n	$\|S\|$	$\bar{t}_{BFS}$, с	$\bar{t}_{DFS}$, с	$\bar{t}_{BFS}/\bar{t}_{DFS}$
1	2	3	4	5	6
1	2^1	5	0,934	0,420	2,224
2	2^2	9	0,952	0,450	2,115
3	2^3	17	1,029	0,540	1,906
4	2^4	33	1,154	0,770	1,499
5	2^5	65	1,412	3,170	0,446
6	2^6	129	2,970	35,750	0,083
7	2^7	257	19,210	-	-
8	2^8	513	-	-	-

Із табл. 4.5 видно, що $|S| = 2 \cdot n + 1$; для $n \le 2^4$ відношення $\bar{t}_{BFS}/\bar{t}_{DFS}$ є на користь DFS-реалізації методу TLC. Наприклад, для $n = 2^1$ DFS-реалізація методу є більше, ніж удвічі, ефективнішою за BFS-реалізацію, а для $n = 2^4$ – вже лише на близько 50 %.

При $n \ge 2^5$ картина змінюється на протилежну – більш ефективною з позиції супутніх ФВ часових витрат стає BFS-реалізація. Наприклад, для $n = 2^5$ більше, ніж удвічі, ефективнішою є вже BFS-реалізація, а для $n = 2^6$ ця перевага стає вже 12-кратною. Більше того, вагомим недоліком DFS-реалізації з позиції автоматизації є потреба вказувати глибину обходу простору станів СП.

Окремої уваги заслуговує просторова складність задач ФВ, вирішуваних при застосуванні BFS- і DFS-реалізацій методу. Наприклад, у випадку DFS-

реалізації методу, для $n \geq 2^7$ має місце нестача ОП – обсяг у 256 МБ, доступний JRE, вичерпано. Аналогічна ситуація справедлива і для BFS-реалізації методу, але вже для $n \geq 2^8$. З цього можна зробити висновок, що задача ФВ, вирішувана на основі BFS-реалізації методу TLC, має ліпшу просторову складність, і є пріоритетнішою з позиції витрат ОП.

Експоненційний характер росту часових витрат від значення n подано на рис. 4.4 [83; 148–152].

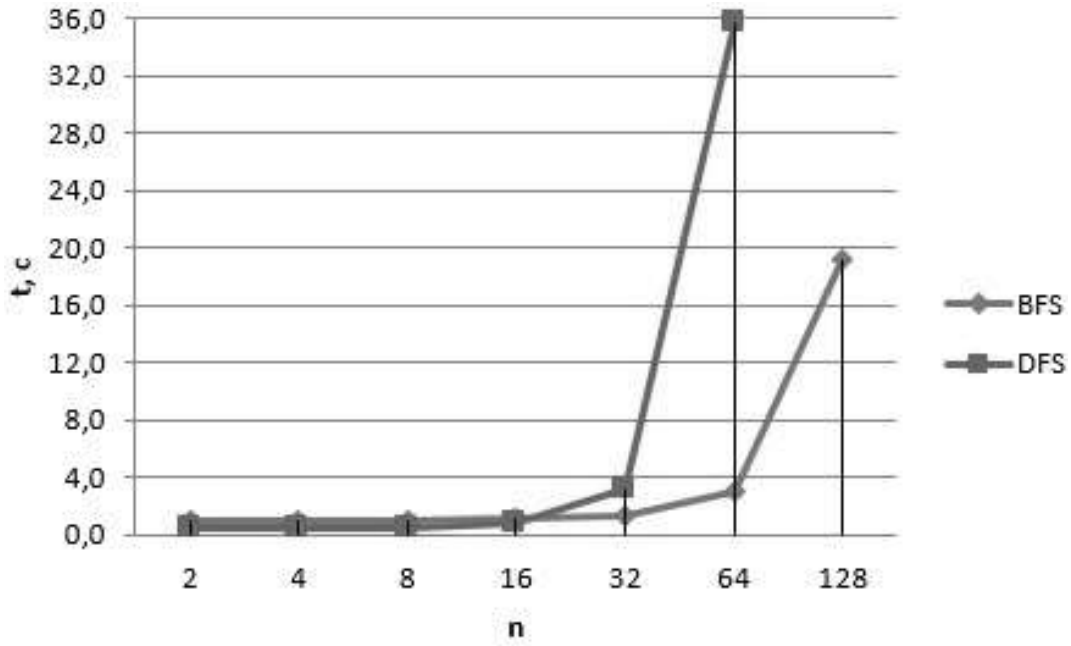

Рисунок 4.4 – Графіку зростання часових витрат на ФВ

Для візуалізації даних рис. 4.4 застосовано кусково-лінійну інтерполяцію. З рис. 4.4 видно, що стрімке зростання часових витрат для DFS-реалізації методу спостерігається для $n \geq 2^5$, а для BFS-реалізації – для $n \geq 2^6$.

Для підвищення достовірності одержуваних даних то подолання обмежень доступної ОП експеримент було повторно проведено на наступній обчислювальній платформі: центральний процесор – 4 ядра, 8 потоків, частота – 3,8 ГГц; обсяг ОП – 16 ГБ (табл. 4.6).

Таблиця 4.6 – Доповнення результатів табл. 4.5

№ з/п	n	$\lvert S \rvert$	$\bar{t}_{BFS}$, с	$\bar{t}_{DFS}$, с	$\bar{t}_{BFS} / \bar{t}_{DFS}$
1	2	3	4	5	6
1	2^1	5	0,880	0,310	2,839
2	2^2	9	0,890	0,390	2,282
3	2^3	17	0,930	0,460	2,022
4	2^4	33	1,070	0,670	1,597
5	2^5	65	1,430	1,620	0,883
6	2^6	129	2,160	17,390	0,124
7	2^7	257	6,390	226,970	0,028
8	2^8	513	38,980	3687,330	0,011

Характер даних табл. 4.6 подібний до такого, поданого у табл. 4.5. При цьому варто зазначити, що для BFS- і DFS-реалізацій при $n = 2^8$ витрати ОП перевищили, відповідно, 4 і 5 ГБ. У свою чергу, для $n = 2^1$ перевага DFS-реалізації вже є майже трикратною (у порівнянні із даними табл. 4.5), а для $n = 2^5$ – значення витрат є співставними.

Вищезазначене можна узагальнити наступним чином:

– застосування BFS-реалізації методу TLC є більш пріоритетним за наступними позиціями: з позиції автоматизації – дає можливість не зазначати глибину обходу СП; для $n \geq 2^5$ (табл. 4.5, табл. 4.6) – супроводжується кращою ефективністю – у порівнянні із альтернативною DFS-реалізацією; визначається нижчими вимогами до обсягу наявної ОП;

– застосування DFS-реалізації методу TLC є більш пріоритетним для $n \leq 2^4$, за умови, що глибина обходу простору станів СП є відомою;

– у випадку застосування ітеративного підходу до розроблення системи [85], доречним вбачається комбінування BFS- і DFS-реалізацій. Наприклад, у випадку $n \le 2^4$, за умови незмінної архітектури СП, на першому проході ФВ доречно застосувати BFS-реалізацію методу TLC, що дозволить визначити глибину обходу, а потім виконати серію DFS-обходів, що дасть змогу скоротити супутні ФВ часові витрати.

Для оцінювання просторової складності вирішуваних задач ФВ співставимо кількості станів, згенерованих методом TLC, для кожної із реалізації – BFS та DFS (табл. 4.7) [153, 154].

Таблиця 4.7 – Показники просторової складності

№ з/п	n	$\lvert S \rvert$	$\lvert S^*_{BFS} \rvert$	$\lvert S^*_{DFS} \rvert$	$\frac{\lvert S \rvert}{\lvert S^*_{BFS} \rvert}$	$\frac{\lvert S \rvert}{\lvert S^*_{DFS} \rvert}$	$\frac{\lvert S^*_{DFS} \rvert}{\lvert S^*_{BFS} \rvert}$	$size$, байтів
1	2	3	4	5	6	7	8	9
1	2^1	5	21	41	0,238	0,122	1,952	681
2	2^2	9	73	289	0,123	0,031	3,959	1331
3	2^3	17	273	2177	0,062	0,008	7,974	3293
4	2^4	33	1057	16897	0,031	0,002	15,986	10230
5	2^5	65	4161	133121	0,016	$4{,}88 \cdot 10^{-4}$	31,993	35702
6	2^6	129	16513	1056769	0,008	$1{,}22 \cdot 10^{-4}$	63,996	132896
7	2^7	257	65793	8421377	0,004	$3{,}05 \cdot 10^{-5}$	127,998	527085
8	2^8	513	262657	67239937	0,002	$0{,}76 \cdot 10^{-5}$	255,999	2169856

У табл. 4.7 $\lvert S \rvert$ – загальне число станів СП, виявлених у процесі ФВ методом TLC; $\lvert S^*_{BFS} \rvert$ – загальне число станів, згенерованих при ФВ на основі BFS-обходу, $\lvert S^*_{DFS} \rvert$ – DFS-обходу; $\lvert S \rvert / \lvert S^*_{BFS} \rvert$, $\lvert S \rvert / \lvert S^*_{DFS} \rvert$ – відносна частка

виявлених станів від згенерованих. Відношення $|S|/|S^*_{BFS}|$ і $|S|/|S^*_{DFS}|$ розглядатимемо у якості показників ефективності вирішення задачі ФВ, відповідно, на основі BFS- і DFS-обходу; $size$ – розмір файлу ФС.

Із табл. 4.7 видно, що значення показника $|S|/|S^*_{DFS}|$ наближається до 0 приблизно у $2^k (k = 1,2....,8)$ разів швидше за значення показника $|S|/|S^*_{BFS}|$. Це свідчить про наступне: згідно відношення $|S^*_{DFS}|/|S^*_{BFS}|$, просторову складність задачі ФВ, вирішуваної DFS-реалізацією методу TLC, можна оцінити як у 2^k разів гіршу за альтернативну BFS-реалізацію. Іншими словами, значення $|S^*_{DFS}|/|S^*_{BFS}|$ пропорційне значенню n. Також має місце наступне співвідношення: $|S^*_{DFS}|/|S^*_{BFS}| \approx 2^k = n$. Отже, характер залежності $|S^*_{DFS}|/|S^*_{BFS}|$ від k є експоненційним (рис. 4.5). Це означає, що залучення DFS-реалізації методу TLC висуває у n разів вищі вимоги до обсягу наявної ОП обчислювальної системи.

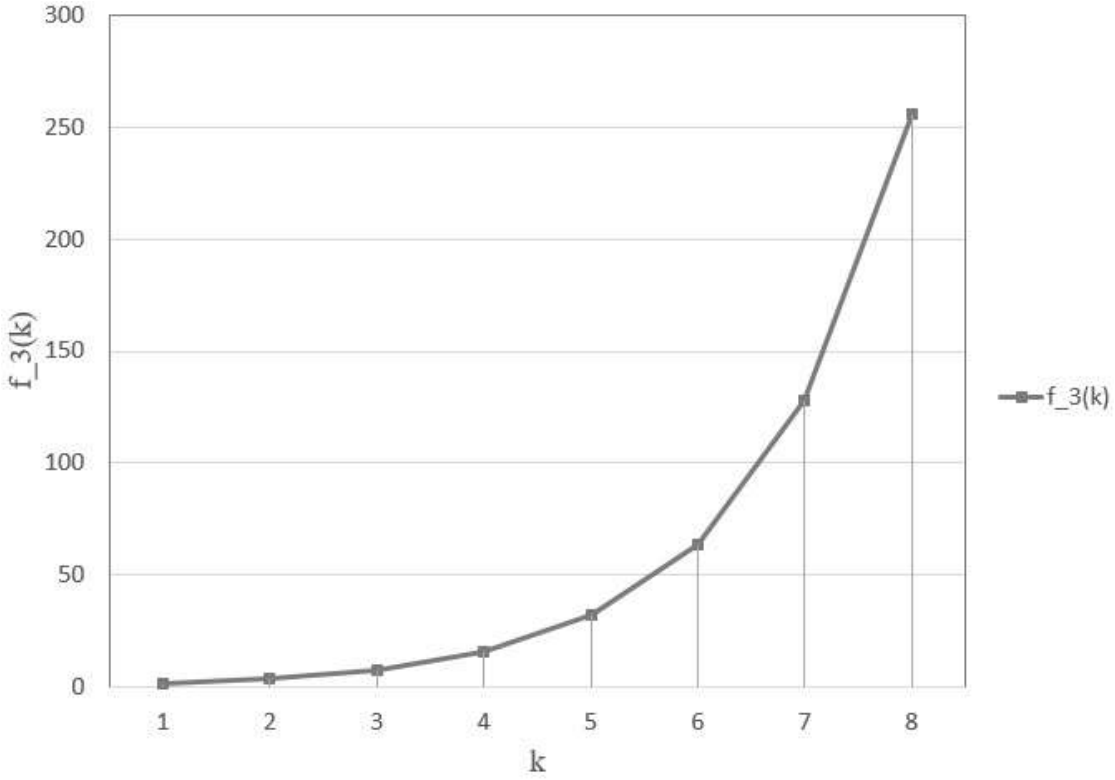

Рисунок 4.5 – Графік залежності $\left|S^*_{DFS}\right|/\left|S^*_{BFS}\right|$ від k

На рис. 4.5 функція $f_3(k)$ демонструє характер залежності значення $\left|S^*_{DFS}\right|/\left|S^*_{BFS}\right|$ від значення k (табл. 3.7). При цьому було зафіксовано, що, для $k=8$ $(n=256)$, для здійснення ФВ DFS-реалізацією методу TLC знадобилося більше 5 ГБ ОП.

Характер залежності значень $\left|S\right|/\left|S^*_{BFS}\right|$ і $\left|S\right|/\left|S^*_{DFS}\right|$ від k подано на рис. 4.6.

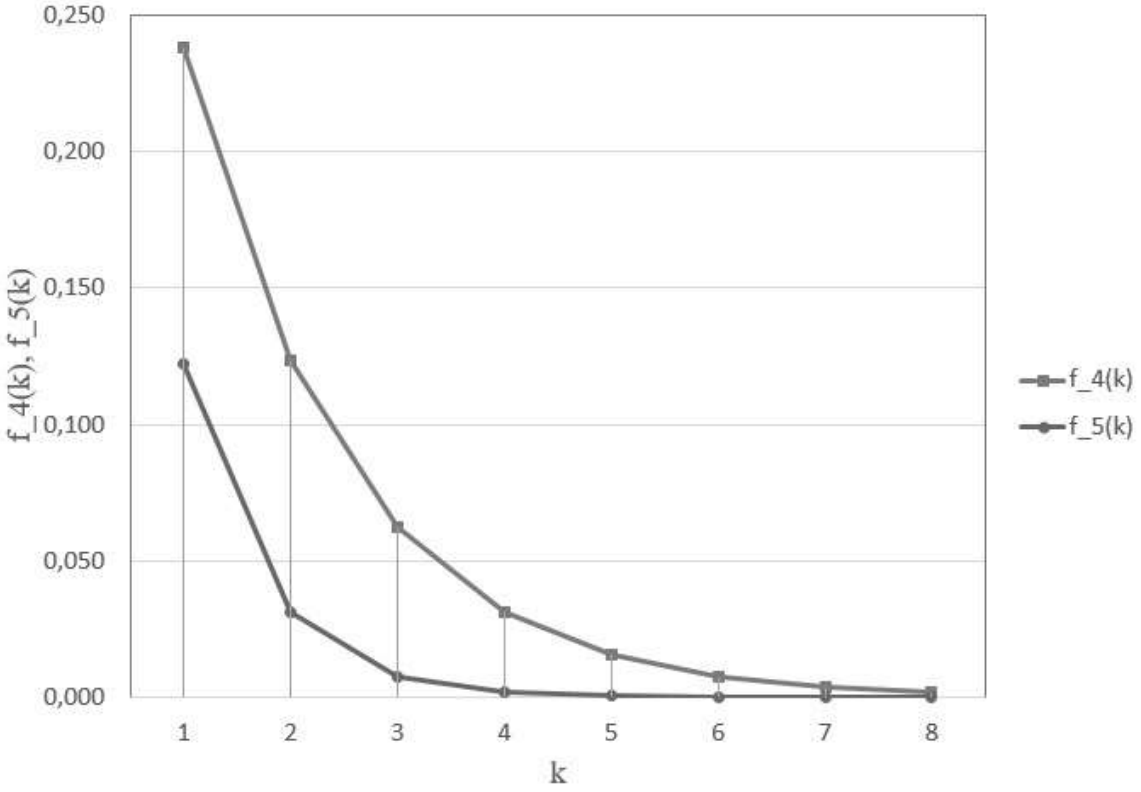

Рисунок 4.6 – Графік залежності $|S|/|S^*_{BFS}|$ і $|S|/|S^*_{DFS}|$ від k

На рис. 4.6 функція $f_4(k)$ демонструє залежність $|S|/|S^*_{BFS}|$ від k, функція $f_5(k)$ – залежність $|S|/|S^*_{DFS}|$ від k. З рис. 4.6 видно, що функція $f_5(k)$ убуває приблизно у 2^k швидше за функцію $f_4(k)$. Це є підтвердженням гіршої просторової складності вирішення задачі ФВ на основі DFS-реалізації методу TLC.

4.2.5 Дослідження сценарію із поданням паралелізму

Розглянемо сценарій, коли паралелізм представлено моделлю чергування [22]. При цьому множина значень D (2.16) визначається наступним чином: $D = \{0,1\}$, тобто є множиною булевих значень.

На основі множини D сформуємо множину AP (2.17): $(v_j,0)\in AP$ – фрагмент коду не виконано, $(v_j,1)\in AP$ – виконано.

Змоделюємо сценарій обчислювального процесу, що має структуру на основі бінарного дерева, де корінь дерева – змінна стану СП $v_0\in V$, що представляє фрагмент коду, який ініціює обчислювальний процес. Архітектуру відповідної програмної системи подано на рис. 2.7.

Представимо архітектуру програмної системи графом $G=\langle V,E\rangle$, де V – множина вершин – множина змінних станів (2.15), E – множина дуг, елементи якої задають порядок активації змінних станів. Це означає, що порядковий номер рівня архітектури (починаючи зверху) відображає відносний порядок зміни значень елементами множини V. При цьому елементи одного рівня призначені функціонувати паралельно.

Зауваження:

– дослідження проведемо для $n=2^2,2^3,2^4$;

– завжди матимемо парну кількість вершин у складі G, оскільки G': $|V\setminus\{v_n\}|=2^l-1$, де l – число рівнів бінарної купи G';

– $\forall v_j\in V\setminus\{v_n\}$: $p(v_{2\cdot j})=v_j$ і $p(v_{2\cdot j+1})=v_j$, де p – функція визначення батьківської вершини для заданої дочірньої.

Бінарне дерево у складі G представимо підграфом $G'=\langle V\setminus\{v_n\},E'\rangle$, де до $E'\subset E$ не входять дуги, що сполучають термінальні вершини з вершиною-надбудовою $v_n\in V$ у складі $G=\langle V,E\rangle$. Отже, у контексті G' вершина $v_1\in V$ є коренем дерева, а у контексті надграфу G – джерелом. Вершина $v_n\in V$ при цьому є витоком. З позиції обчислювального процесу, з точки зору розподілених обчислень, вершина $v_1\in V$ представляє ініціатора названого процесу – засіб розподілу обчислювального навантаження, вершини $v_2\in V$ –

$v_{(n/2)-1} \in V$ – засоби розподілу задач на підзадачі, термінальні вершини $v_{n/2} \in V$ – $v_{n-1} \in V$ – засоби виконання безпосередньо обчислень, вершина-виток $v_n \in V$ – засіб опрацювання проміжних даних. При цьому доречно простежити аналогію із Грід-сервісами, де функції елементів $v_1 \in V$ – $v_{(n/2)-1} \in V$ і $v_n \in V$ реалізуються компонентом CE (Computing Element), а елементів $v_{n/2} \in V$ – $v_{n-1} \in V$ – компонентами WN (Worker Node) [155]. Більше того, на кожному із WN-вузлів може бути також розгорнуто інтерфейс MPI (Message Passing Interface), що, з урахуванням потенційної мультиядерності WN-вузлів, дозволяє розпаралелювати вирішення задач/підзадач і нарівні окремого WN-вузла розподіленої системи.

Озвучений підхід дозволяє інтерпретувати зміст рис. 2.7 для $n = 2^3$ наступним чином: елементи $v_1, v_8 \in V$ – представлення компоненту CE, елементи $v_2, v_3 \in V$ – двох двоядерних компонентів WN, $v_4 \in V$ – $v_7 \in V$ – кожного із обчислювальних ядер окремо. Отже, з позиції стратифікованого підходу до аналізу ФС, термінальні вершини підграфу G' є представниками нижнього ієрархічного рівня, а вершини $v_1, v_8 \in V$, $v_2, v_3 \in V$ – представниками верхнього рівня.

Вищезазначене є обґрунтуванням репрезентативності досліджуваної архітектури ФС для предметної області розподілених комп'ютерних систем, зокрема таких, що реалізуються на основі Грід-сервісів.

Наприклад, для $n = 2^2$ діаграма дій матиме вигляд, поданий на рис. 4.7.

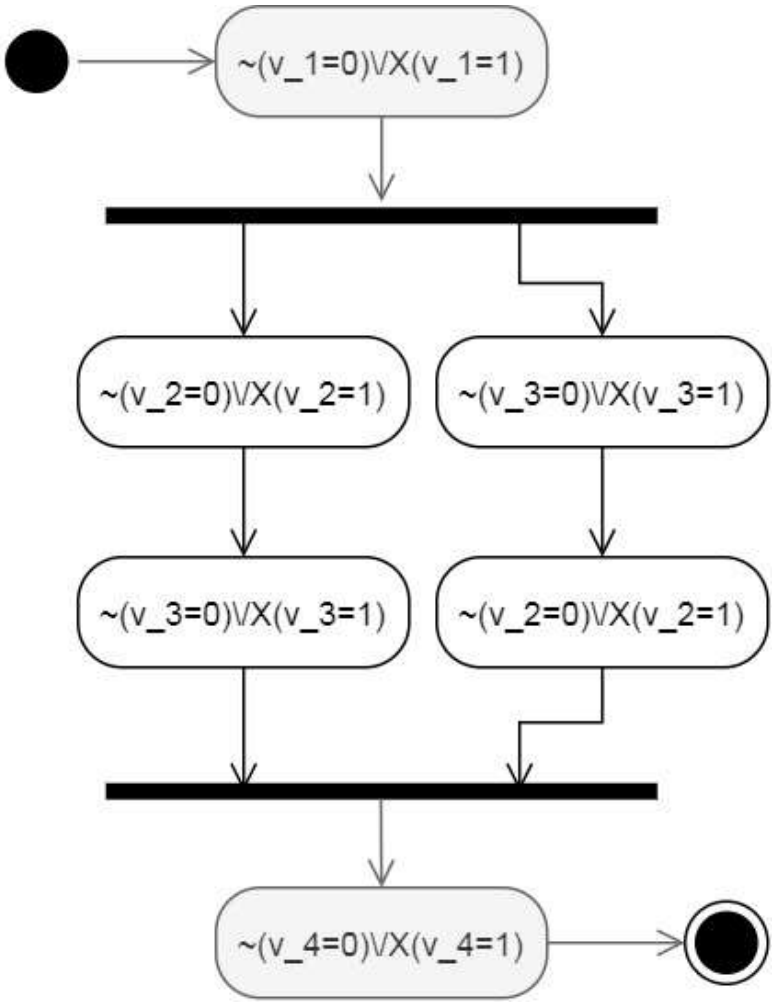

Рисунок 4.7 – Діаграма дій для СП, де вершинами є події

На рис. 4.7 вершинами позначено події, що призводять до змін станів СП. Відповідну діаграму станів подано на рис. 2.8.

На рис. 4.7 символ «~» позначає логічний оператор «Ні».

Маємо чотири події, що складають два альтернативних протоколи процесів – $p_1, p_2 \in P$ (2.5): $p_1 = \langle e_1, e_2, e_3, e_4 \rangle$, $p_2 = \langle e_1, e_3, e_2, e_4 \rangle$. Відмінність обумовлюється послідовністю виникнення подій e_2 і e_3. Ці події виникають псевдопаралельно, що подається у ФС згідно моделі чергування.

При проведенні експериментальних досліджень конфігурація тестової платформи лишилася незмінною. Просторові характеристики вирішуваних при цьому задач для $n = 2^2, 2^3, 2^4$ подано у табл. 2.4.

Для $n = 2^2, 2^3$ загальна кількість станів СП, синтезованих при перевірці згідно BFS- і DFS-обходах, різниться (табл. 4.8).

Таблиця 4.8 – Додаткові просторові характеристики вирішуваних задач

Кількість Станів СП	Значення n			
	2^2		2^3	
	BFS	DFS	BFS	DFS
1	2	3	4	5
Синтезовано	13	25	1009	3601
Знайдено	6		21	
Глибина обходу	5		9	

Із табл. 4.8 видно, що просторові характеристики вирішуваної задачі ФВ за DFS-обходу є гіршими за такі для BFS-обходу. Так, число синтезованих станів СП для $n = 2^2$ у випадку DFS-реалізації методу в 1,92 рази більше за відповідне число станів для BFS-реалізації, а для $n = 2^3$ це співвідношення вже складає 3,57. Це свідчить про те, що і асимптотичні просторові характеристики DFS-задачі є гіршими за BFS-альтернативу. Підтвердженням цьому слугує і характер зростання часових витрат на здійснення ФВ (табл. 4.9).

Із табл. 4.9 видно, що для $n = 2^2, 2^3$ значення відносного показника $\bar{t}_{BFS} / \bar{t}_{DFS}$ є близькими до відповідних значень, поданих у табл. 4.5. При цьому для $n = 2^4$, по причині, представленій у табл. 2.4, – значення функції $g_5(n)$, мав місце факт нестачі доступної реалізації методу TLC ОП. У результаті цього, у режимі виконання, при вирішенні задачі ФВ було одержано відповідне

повідомлення. Залежність часу, що пройшов від моменту запуску процедури ФВ до моменту одержання названого повідомлення для $n = 2^4$, подано на рис. 4.8.

Таблиця 4.9 – Часові витрати на ФВ ФС згідно шаблону із паралелізмом

№ з/п	n	$\bar{t}_{BFS}$, с	$\bar{t}_{DFS}$, с	$\bar{t}_{BFS} / \bar{t}_{DFS}$
1	2	3	4	5
1	2^2	0,926	0,425	2,179
2	2^3	1,094	0,623	1,756
3	2^4	-	-	-

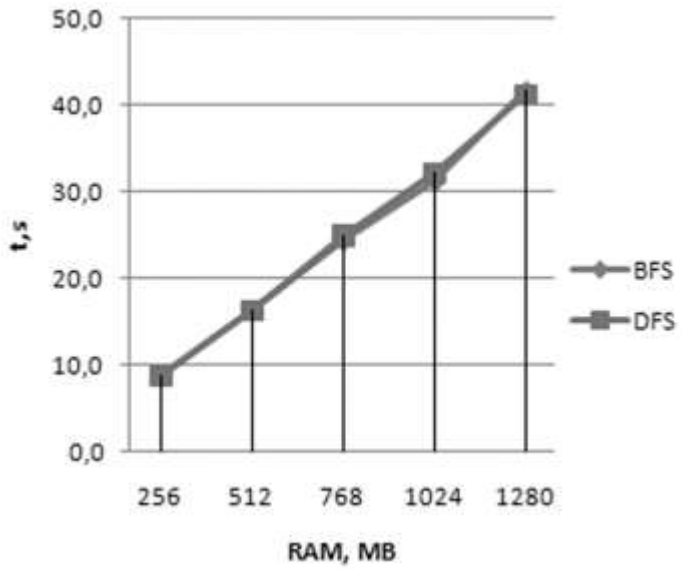

Рисунок 4.8 – Залежність часових витрат на ФВ від обсягу доступної ОП

З рис. 4.8 видно, що в даному аспекті помітної різниці між реалізаціями методу не спостерігається. Більше того, збільшення обсягу пам’яті до 8 ГБ не

змінює ситуації: вирішення задачі ФВ припиняється через нестачу ОП за 23 хв і 19 с.

4.3 Розвиток методу і валідація результатів

Проведений розвиток методу TLC полягає у комбінуванні методів обходу вершин графу-подання СП у процесі автоматизованої ФВ ФС – методу обходу у ширину (BFS) і методу обходу у глибину (DFS) теорії графів – за ітеративного підходу до організації процесу ФВ ФС.

Проведений розвиток полягає у виконанні наступних кроків:

1. Здійснення BFS-обходу простору станів СП, що дасть змогу визначити глибину обходу. На наступних кроках отримане значення залучається у якості параметру DFS-обходів.

2. Виконання серії DFS-обходів, за рахунок чого досягається скорочення супутніх процесу ФВ ФС результуючих часових витрат. Одержуваний при цьому корисний ефект визначається, зокрема, архітектурною складовою ФС, числом змінних ФС, числом ітерацій здійснення DFS-обходів.

Експериментальні дослідження проведено на основі штучних і предметно орієнтованих сценаріїв, поданих у формі відповідних артефактів – блок-схем алгоритмів, UML-діаграм дій. Предметно орієнтовані сценарії при цьому охоплюють галузь електроенергетики, аерокосмічну галузь.

Валідацію результатів розвитку методу проведено на прикладі модуля БК ПВВ комплекту БЦОК. Для цього у якості вихідних даних було залучено блок-схему алгоритму контролю вихідного стану регістрів ПВВ.

4.3.1 Специфіка проведеного розвитку методу

Ідея в основі проведеного розвитку методу TLC – знизити часові витрати на ФВ ФС за ітераційного підходу до організації процесу проєктування ПАС СКП, керуючись наступними параметрами структури ФС:

– число змінних станів СП;

– глибина обходу простору станів.

Підґрунтям для розробленої модифікації методу TLC є експериментальні дані, подані, зокрема, у табл. 4.5, табл. 4.6 і табл. 4.9, а саме – відношення значень часових витрат на застосування методу, в залежності від значення n.

Із табл. 4.5, табл. 4.6 видно, що для $n \le 2^4$ застосування DFS-обходу вершин графу СП є ефективнішим за застосування BFS-обходу. У випадку табл. 4.9 це твердження справедливе для $n < 2^4$. При цьому для BFS-реалізації методу не потрібно знати глибину обходу – вагомий аргумент з точки зору автоматизації.

Ідея в основі розробленого удосконалення методу – забезпечити зниження часових витрат на застосування методу, із збереженням придатності до автоматизації. Для здійснення цього застосовується комбінований (комплексний) підхід, призначений до застосування за ітераційного підходу до організації процесу проєктування ПАС СКП. Суть запропонованого підходу полягає у наступному:

– на початковій ітерації процесу ФВ виконується BFS-обхід простору станів СП, що дає змогу визначити глибину обходу;

– за умови незмінності архітектурної складової ФС, при $n \le 2^4$, всі наступні ітерації перевірки ФС виконуються на основі DFS-обходу, що дозволяє скоротити часові витрати.

4.3.2 Оцінювання корисного ефекту

4.3.2.1 Опис сценарію предметної області

Придатність проведеного удосконалення методу TLC до цільового використання підтверджено шляхом валідації. Для цього у якості предметної області розглянуто космічну галузь, де предметом дослідження є система критичного призначення, представлена програмною складовою БЦОК. Відповідний сценарій предметної області полягає у дослідженні алгоритму контролю вихідного стану регістрів модуля БК ПВВ БЦОК.

Характеристики вихідних даних наступні:

– число змінних станів – 18;

– число блоків блок-схеми – 77, з яких число блоків умовного переходу – 30. Відносна частка блоків умовного переходу – $30/77 = 0{,}39$. Таке значення приблизно удвічі перевищує табличне значення – 22,3 – для суміші Шатл (табл. 1.1). Блоки умовного переходу з'єднані послідовно попарно.

Характеристики синтезованої ФС:

– число рядків PlusCal-коду – 209;

– число рядків TLA+ коду – 607.

Із співвідношення числа рядків PlusCal- і TLA+ коду маємо, що PlusCal-подання ФС є приблизно утричі компактнішим.

Характеристики СП, синтезованої згідно ФС:

– глибина обходу – 52;

– число станів СП – $1{,}474552 \cdot 10^6$.

Показники ефективності застосування базової реалізації методу TLC –на основі BFS-обходу:

– число згенерованих станів СП – показник просторової складності вирішуваної задачі ФВ ФС – $2{,}260984 \cdot 10^6$;

– часові витрати на ФВ ФС – 17,127 с.

При DFS-обході часові витрати склали 9 хв і 41 с. При цьому було згенеровано $73{,}046458 \cdot 10^6$ станів СП.

Фрагмент блок-схеми алгоритму, використаної у якості вихідних даних, подано на рис. 4.9 [156].

Результати ФВ відповідної ФС методом TLC показали, що виконання перевірки блоку 4 є надлишковим, і, як результат, дії блоку 7 ніколи не буде виконано. З позиції формальної інтерпретації це означає, що стан СП, який передує діям блоку 7, ніколи не буде досягнуто. Такий результат означає, що блок-схему, разом із відповідною реалізацією алгоритму, доречно спростити.

Озвучений результат є показовим з позиції того, що демонструє корисний ефект від застосування методу ФВ, який проявляється у поліпшенні сприйняття розробником структури артефактів процесу проєктування.

ФС згідно рис. 4.9 подано у додатку Е, де мітками позначено відповідні блоки блок-схеми.

На рис. 4.9 пунктирною областю (позначено як Z) окреслено фрагмент блок-схеми, що повторюється 15 разів: $5 \cdot 15 = 75$ блоків. Блоки 6 і 7 при цьому демонструють альтернативні заключні кроки алгоритму. Після 15 зазначених фрагментів вони одержують, відповідно, номери 76 і 77. Кожен наступний фрагмент відрізняється від попереднього діями, виконуваними у блоках 2 і 5, де скобками < i > позначено опосередковану адресацію.

На рис. 4.9 фігурують три змінних стану: “flag”, “ISH”, “RRP_IRQ5”. Окрім того, аби імітувати виконання/невиконання блоків 6 і 7, введено додаткову змінну “done”: $s(done) \in \{0,1,2\}$.

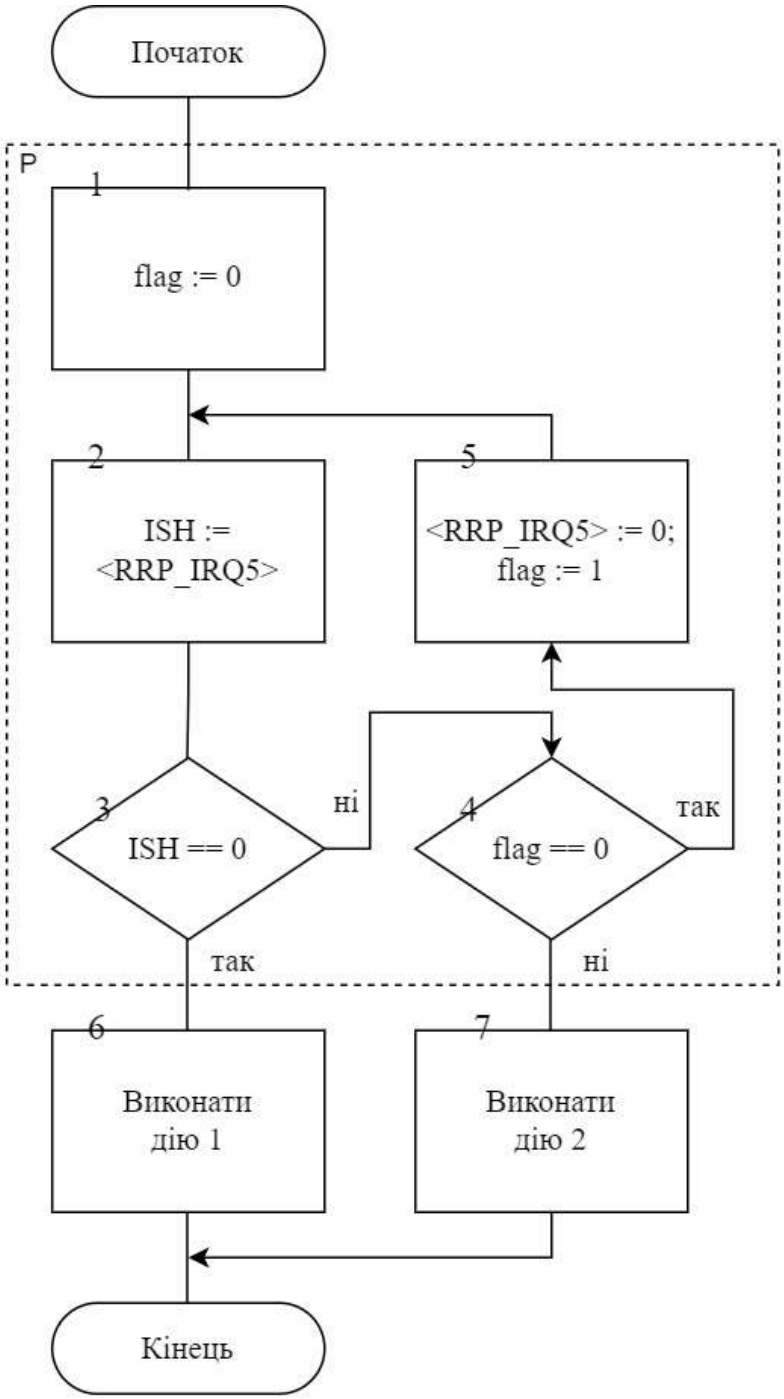

Рисунок 4.9 – Фрагмент блок-схеми алгоритму контролю вихідного стану регістрів ПВВ

Згідно рис. 4.9, частка операторів умовного переходу для розглянутого випадку складає $(2 \cdot 15)/77 \approx 0{,}39$. Це суттєво розходиться із даними табл. 1.1 (0,22).

4.3.2.2 Експериментальне дослідження проведеного розвитку

Методика проведення експерименту:

– провести ФВ ФС для фрагменту рис. 4.9 методами BFS і DFS; при цьому початковий прогін для заданого n виконується BFS-реалізацією методу TLC, що дасть змогу визначити глибину обходу СП – необхідний параметр для застосування DFS-реалізації методу TLC.

– індуктивним шляхом збільшувати число Z-блоків від 1 до 15, фіксуючи при цьому значення n, що дозволить виявити граничну точку, у якій застосування DFS-обходу при ФВ ФС ще буде доцільним.

ФВ ФС згідно рис. 4.9 проведено двома шляхами: на основі BFS- і DFS-обходів. Здобуті результати подано у табл. 4.10 (значення замірів вказано для однопоточної реалізації методу) і табл. 4.11 [157–161].

У табл. 4.10 значення часових витрат є середнім арифметичним 10 замірів; стовпець *depth* містить значення глибини обходу СП.

Варто при цьому зазначити, що у табл. 4.10 і, відповідно, у ФС введено додаткову змінну – v_n: $s(v_n) \in \{0,1,2\}$ – для ідентифікації виконання/невиконання кожного із двох заключних блоків блок-схеми. Ця змінна фігурує у кожному із 15 експериментів.

Із табл. 4.10 видно, що для $n \geq 10$ застосування DFS-реалізації методу TLC вже є недоцільним. При цьому для $n = 8$ DFS-реалізація ще є на 27% ефективнішою за альтернативну BFS-реалізацію – за часовими витратами. Отже, значення $n = 8$ вважатимемо граничним, за якого здійснення DFS-обходу ще є доцільним.

Дані табл. 4.10 використаємо наступним чином (за аналогією із асимптотичними оцінками):

– рядок 1 ($n = 4$) – для одержання оціночної функції нижньої границі – найліпший випадок – «оцінка знизу»;

– рядок 3 ($n = 7$) – для оцінки у середньому;

– рядок 5 ($n = 8$) – для одержання оціночної функції верхньої границі – найгірший випадок – «оцінка зверху».

Таблиця 4.10 – Показники обчислювальної складності

№ з/п	n	$\bar{t}_{BFS}$, с	$\bar{t}_{DFS}$, с	$\bar{t}_{BFS}/\bar{t}_{DFS}$	$\bar{t}_{DFS}/\bar{t}_{BFS}$	$depth$, вершин
1	2	3	4	5	6	7
1	4	0,893	0,437	2,043	0,489	10
2	6	0,929	0,477	1,948	0,513	15
3	7	0,957	0,554	1,727	0,579	18
4	7	0,969	0,578	1,676	0,596	19
5	8	1,001	0,788	1,270	0,787	22
6	10	1,140	1,505	0,757	1,320	27
7	11	1,448	2,577	0,562	1,780	30
8	11	1,468	2,674	0,549	1,822	31
9	12	1,682	4,818	0,349	2,864	34
10	14	2,460	19,751	0,125	8,029	39
11	15	3,552	44,575	0,080	12,549	42
12	15	3,560	53,226	0,067	14,951	43
13	16	5,370	105,340	0,051	19,616	46
14	17	9,666	273,970	0,035	28,344	49
15	18	17,127	581,860	0,029	33,973	52

Вищесказане означає, що застосування розробленого удосконалення методу TLC є доцільним для досліджуваного сценарію для $|V| = 4,...,8$, і для числа блоків блок-схеми від 7 = (5 + 2) до 27 = $(5 \cdot 5) + 2$. Це узгоджується із

попередніми синтетичними результатами досліджень для послідовного сценарію і сценарію із поданням паралелізму згідно моделі чергування (табл. 4.5 – табл. 4.9): граничний випадок настає швидше, ніж за виключно послідовного сценарію, і повільніше, ніж за сценарію подання паралелізму згідно моделі чергування.

Таблиця 4.11 – Показники просторової складності

№ з/п	n	$\lvert S \rvert$	$\lvert S^*_{BFS} \rvert$	$\lvert S^*_{DFS} \rvert$	$\frac{\lvert S \rvert}{\lvert S^*_{BFS} \rvert}$	$\frac{\lvert S \rvert}{\lvert S^*_{DFS} \rvert}$	$\frac{\lvert S^*_{DFS} \rvert}{\lvert S^*_{BFS} \rvert}$	$depth$, вершин
1	2	3	4	5	6	7	8	9
1	4	82	130	700	0,631	0,117	5,385	10
2	6	342	534	4519	0,640	0,076	8,463	15
3	7	702	1086	11189	0,646	0,063	10,303	18
4	7	712	1096	11927	0,650	0,060	10,882	19
5	8	1432	2200	28156	0,651	0,051	12,798	22
6	10	5742	8814	141343	0,651	0,041	16,036	27
7	11	11502	17646	317237	0,652	0,036	17,978	30
8	11	11512	17656	328775	0,652	0,035	18,621	31
9	12	23032	35320	726652	0,652	0,032	20,573	34
10	14	92142	141294	3367327	0,652	0,027	23,832	39
11	15	184302	282606	7287605	0,652	0,025	25,787	42
12	15	184312	282616	7471943	0,652	0,025	26,438	43
13	16	368632	565240	16049788	0,652	0,023	28,395	46
14	17	737272	1130488	34311398	0,652	0,021	30,351	49
15	18	1474552	2260984	73046458	0,652	0,020	32,307	52

У табл. 4.11 із відношення $|S|/|S^*_{BFS}|$ видно, що частка виявлених станів СП до згенерованих при перевірці BFS-реалізацією методу є відносно сталою, і знаходиться у діапазоні 0,631 … 0,652. Це свідчить про стійкість BFS-реалізації методу з позиції просторової складності. Цього, у свою чергу, не можна стверджувати відносно DFS-реалізації методу, де відносна частка виявлених станів СП зменшується при збільшенні n – від 0,105 – до 0,020.

Зведені результати оцінювання обчислювальної та просторової складностей методу TLC на прикладі розглянутого сценарію – на основі даних табл. 4.10 та табл. 4.11 – подано на рис. 4.10.

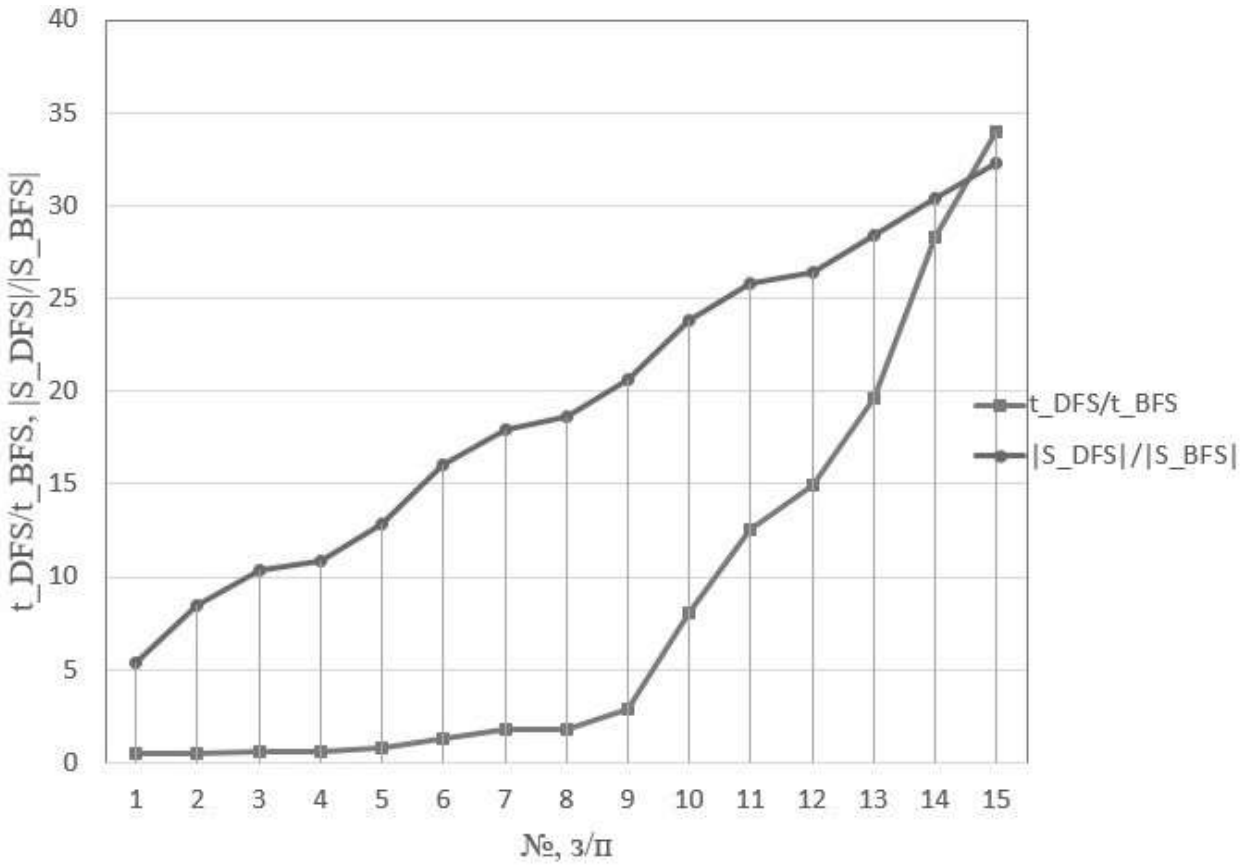

Рисунок 4.10 – Графік значень відносних показників $\bar{t}_{DFS}/\bar{t}_{BFS}$ і $|S^*_{DFS}|/|S^*_{BFS}|$ від архітектурної складової ФС і числа змінних станів

На рис. 4.10 покажчик $\bar{t}_{DFS}/\bar{t}_{BFS}$ (табл. 3.10) демонструє відношення між часовими витратами на застосування реалізацій методу, покажчик $\left|S^*_{DFS}\right|/\left|S^*_{BFS}\right|$ – між витратами ОП. При цьому для № 1…5, з позиції скорочення часових витрат, ефективнішою є саме DFS-реалізація методу: $\bar{t}_{DFS}/\bar{t}_{BFS}<1$. Для № 6…15 за цим показником ліпшою є вже BFS-альтернатива. У свою чергу, з позиції просторової складності, DFS-реалізація методу є гіршою за BFS-реалізацію у будь-якому випадку: значення $\left|S^*_{DFS}\right|/\left|S^*_{BFS}\right|$ зростає від 6,015 до 32,307 із зростанням значення n.

4.3.2.3 Узагальнення результату оцінювання

На прикладі розглянутого сценарію (рис. 4.9), оцінемо корисний ефект від прикладного застосування проведеного удосконалення методу на основі даних табл. 4.10 для $x=1,2,...,\left(10^3+1\right)$, де x – число ітерацій ФВ ФС.

Зауваження:

– у якості правої границі діапазону значень x має місце саме вираз $\left(10^3+1\right)$, а не 10^3,оскільки на початковій ітерації завжди залучається BFS-реалізація методу – через необхідність визначити глибину обходу простору станів СП.

Методика оцінювання наступна:

– оцінювання проведемо для верхньої (найліпшого випадку) і нижньої (найгіршого випадку) границь. Для цього, згідно табл. 4.10, оціночні функції будуються для $n=4$ і $n=8$ відповідно – границі діапазону значень $[4;8]$, на якому часові витрати на ФВ на основі DFS-реалізації методу є нижчими за відповідні витрати на основі BFS-реалізації;

– у якості базового розглянемо випадок, коли на кожній ітерації ФВ ФС застосовується BFS-реалізація методу TLC;

– корисний ефект оцінюватимемо як відносне значення прискорення для $x \in [1,2,...,(10^3+1)]$ за умови, що на першій ітерації завжди виконуватиметься BFS-обхід, а на всіх наступних – DFS-обходи – для одержання зазначеного ефекту.

Запропонована оціночна функція має наступний вигляд:

$$\xi(x) = \begin{cases} (\bar{t}_{BFS} / \bar{t}_{BFS}) = 1, \; x = 1, \\ x \cdot \bar{t}_{BFS} \Big/ \left(\bar{t}_{BFS} + \sum_{i=2}^{x} t_{DFS_i} \right), \; x > 1, \end{cases} \tag{4.1}$$

де $\bar{t}_{BFS}$ – табличне значення (середнє арифметичне) часових витрат при ФВ BFS-обходом для заданого значення n (табл. 4.10), t_{DFS_i} – часові витрати на здійснення кожної наступної (з 10^3) ФВ шляхом DFS-обходу.

Для обчислення значень верхньої границі $\xi_1(x)$ ($n = 4$, $depth = 10$) маємо $\bar{t}_{BFS} = 0{,}893$ с, нижньої границі $\xi_2(x)$ ($n = 8$, $depth = 22$) – $\bar{t}_{BFS} = 1{,}001$ с (табл. 4.10).

Для підвищення достовірності одержуваних даних було вирішено розширити число замірів для $depth = 10$ і $depth = 22$ (табл. 4.10) з 10 до 10^3. Значення функцій подано на рис. 4.11.

При побудові графіку рис. 4.11 застосовано кусочно-лінійну інтерполяцію. Фрагмент вихідних даних для побудови графіку подано у табл. Ж.1 і табл. Ж.2 додатку Ж. Код програми обчислення значень функцій $\xi_1(x)$ і $\xi_2(x)$ згідно (4.1) подано у лістингу Ж.1.

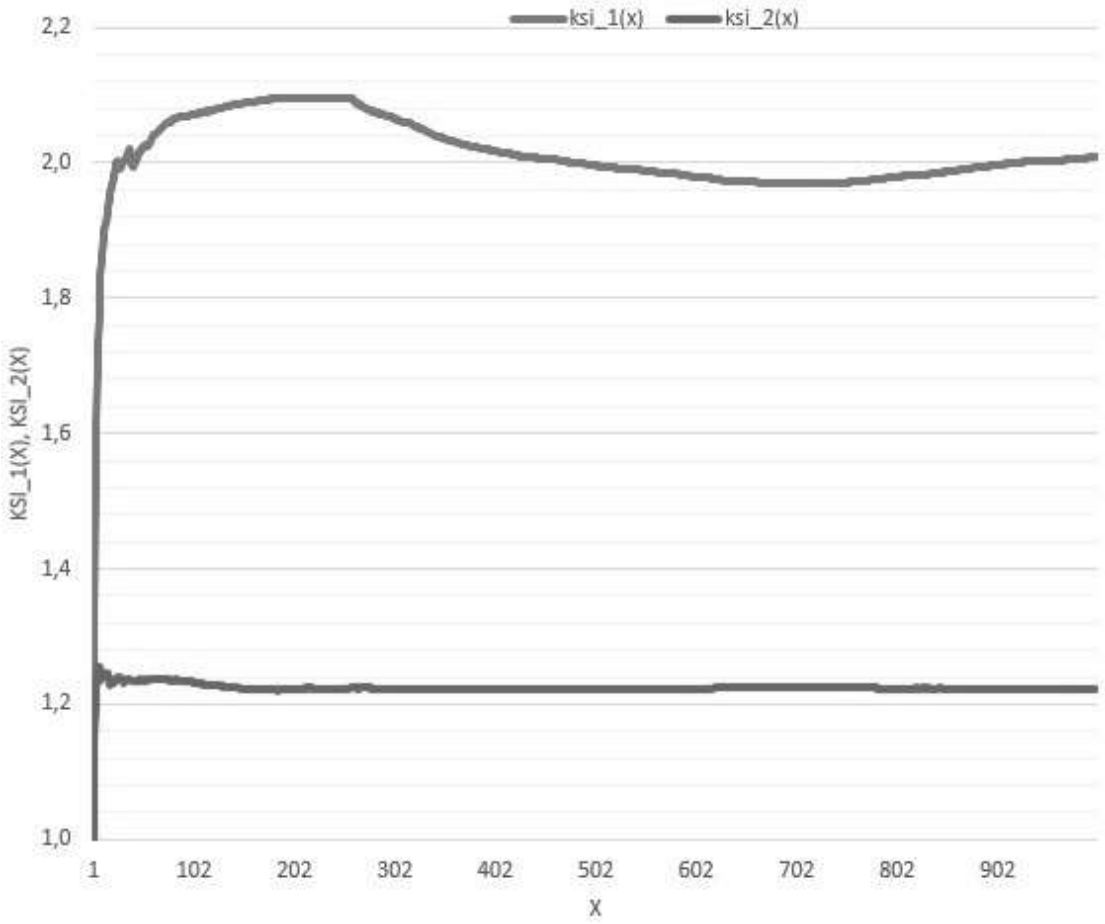

Рисунок 4.11 – Обчислені значення прискорення від застосування проведеного удосконалення методу

Висунуто наступну нульову гіпотезу:

$$H_0 : \lim_{x \to \infty} \xi(x) = (\bar{t}_{BFS} / \bar{t}_{DFS}). \tag{4.2}$$

Застосований підхід до перевірки гіпотези H_0.

Перевірку гіпотези H_0 вирішено проводити шляхом послідовного розв'язання задач апроксимації і екстраполяції:

– вирішення задачі апроксимації дозволить отримати наближену функцію $\xi'(x)$, що дасть змогу розв'язати задачу екстраполяції. Для перевірки

успішності вирішення задачі апроксимації виконано обчислення значення t-критерію;

– вирішення задачі екстраполяції, у свою чергу, має на меті підтвердити висунуту гіпотезу H_0 шляхом підстановки значення $x > 10^3$ до функції $\xi'(x)$.

Для перевірки H_0 висунуто допоміжну гіпотезу H_0': про рівність вибірок $\xi(x)$ і $\xi'(x)$, що підтверджується при вирішенні задачі апроксимації – шляхом обчислення значення t-критерію.

Для перевірки H_0 число замірів для обчислення $\bar{t}_{BFS}$ було розширено до 10^3.

4.3.2.4 Перевірка нульової гіпотези для нижньої границі

На основі значень функції $\xi_2(x)$ отримано апроксимуючу функцію $\xi_2'(x)$ – як поліном четвертого ступеня:

$$\xi_2'(x) = a + b \cdot x + c/x + d \cdot x^2 + e/x^2 + f \cdot x^3 + g/x^3 + h \cdot x^4, \qquad (4.3)$$

для якої коефіцієнт детермінації $R^2 = 0{,}957$. Значення коефіцієнтів подано у табл. 4.12, де σ – середнє квадратичне відхилення.

Таблиця 4.12 – Значення коефіцієнтів для $\xi_2'(x)$

№ з/п	Коефіцієнт	Значення коефіцієнту	№ з/п	Коефіцієнт	Значення коефіцієнту
1	2	3	4	5	6
1	a	1,237988264	5	e	-0,95167447
2	b	-0,00013431	6	f	-3,7648e-10
3	c	0,160404728	7	g	0,553304272
4	d	3,6019e-07	8	h	1,34087e-13

Для перевірки успішності вирішення задачі апроксимації було висунуто нульову гіпотезу H_0 про рівність середніх для вибірок $\xi(x)$ і $\xi'(x)$. Для підтвердження/спростування гіпотези H_0 виконано обчислення значення t-критерію згідно наступної формули:

$$t = \left|M[\xi(x)] - M[\xi'(x)]\right| \Big/ \sqrt{\frac{\sigma[\xi(x)]^2 + (\sigma[\xi'(x)])^2}{x}}, \qquad (4.4)$$

де $M[\xi(x)]$ – середнє арифметичне для $\xi(x)$, $M[\xi'(x)]$ – середнє арифметичне для $\xi'(x)$, $\sigma[\xi(x)]$ – середнє квадратичне відхилення для $\xi(x)$, $\sigma[\xi'(x)]$ – середнє квадратичне відхилення для $\xi'(x)$; $x = 10^3 + 1$, оскільки, згідно (4.1), має місце складова $\bar{t}_{BFS}$.

Результат вирішення задачі апроксимації для $\xi_2(x)$ подано на рис. 4.12.

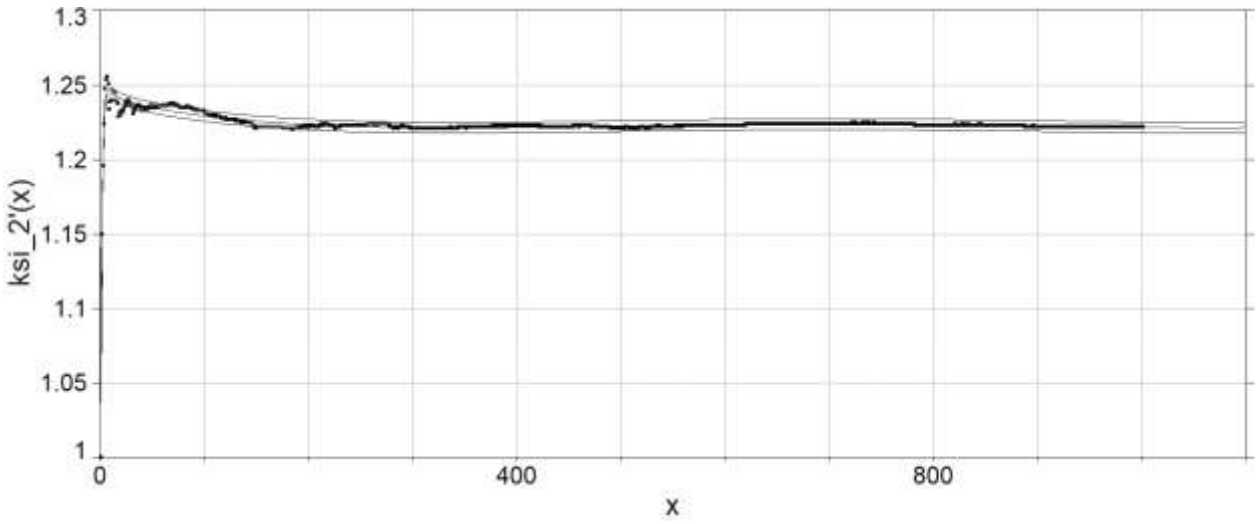

Рисунок 4.12 – Графік апроксимуючої функції $\xi_2'(x)$, довірчі інтервали для довірчої імовірності 0,95

Отримані значення статистичних показників зведено у табл. 4.13.

Таблиця 4.13 – Значення статистичних показників при побудові $\xi_2'(x)$

№ з/п	Показник	Значення показника	№ з/п	Показник	Значення показника
1	2	3	4	5	6
1	$M[\xi_2(x)]$	1,223930	4	$M[\xi_2'(x)]$	1,223884
2	$\sigma[\xi_2(x)]$	$7{,}601\cdot10^{-5}$	5	$\sigma[\xi_2'(x)]$	$7{,}284\cdot10^{-5}$
3	t_2	0,127847	6	t^*	1,962

У табл. 4.13 t_2 – розрахункове значення t-критерію, t^* – табличне значення. Отже, маємо наступну нерівність t-критерію для довірчої імовірності 0,95 і (10^3+1) ступенів свободи:

$$t_2 < t^*, \tag{4.5}$$

Отже, нерівність (4.5) є підтвердженням гіпотези H_0' про рівність вибірок $\xi_2(x)$ і $\xi_2'(x)$. Це дає підстави застосовувати апроксимуючу функцію $\xi_2'(x)$ як оціночну функцію для нижньої границі корисного ефекту від прикладного застосування проведеного удосконалення методу.

Програмний код обчислення значення t-критерію подано у лістингу Ж.1.

Справедливість нерівності (4.5) дає підстави переходити до наступного кроку – вирішення задачі екстраполяції, що дасть змогу підтвердити / спростувати гіпотезу H_0 (4.2).

4.3.2.5 Перевірка нульової гіпотези для верхньої границі

У результаті вирішення задачі апроксимації для верхньої границі було отримано поліном дев'ятого порядку (рис. 4.13):

$$\begin{aligned}\xi_1'(x) = a + b\cdot\ln x + c\cdot(\ln x)^2 + d\cdot(\ln x)^3 + e\cdot(\ln x)^4 + \\ + f\cdot(\ln x)^5 + g\cdot(\ln x)^6 + h\cdot(\ln x)^7 + i\cdot(\ln x)^8 + j\cdot(\ln x)^9,\end{aligned} \quad (4.6)$$

де $R^2 = 0{,}995$, значення коефіцієнтів подано у табл. 4.14.

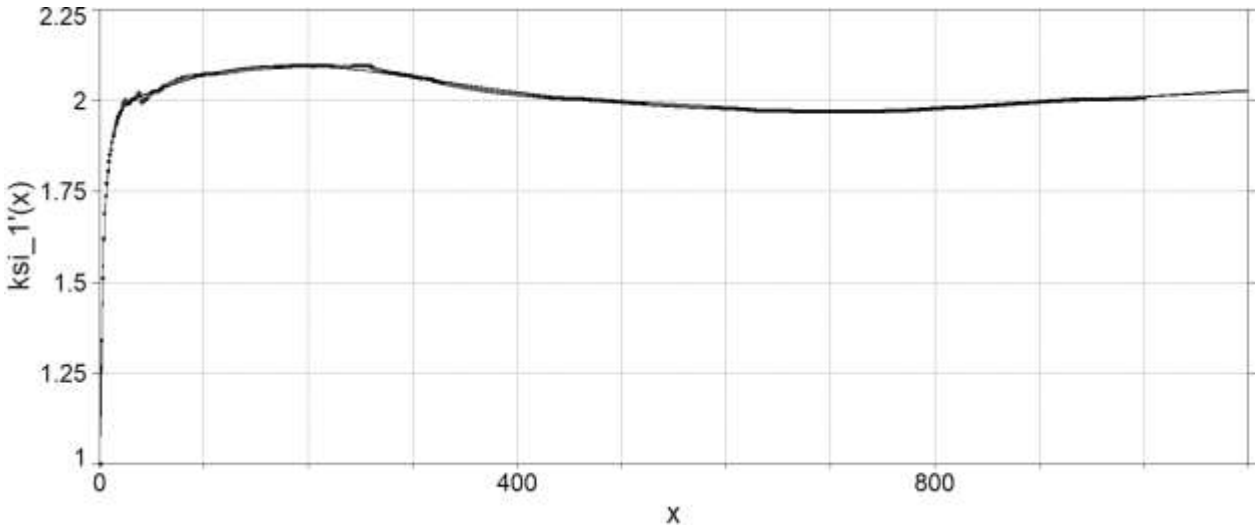

Рисунок 4.13 – Графік апроксимуючої функції $\xi_1'(x)$

Для рис. 4.13 довірчі інтервали побудовано для довірчої імовірності 0,95.

Таблиця 4.14 – Значення коефіцієнтів для $\xi_1'(x)$

№ з/п	Коефіцієнт	Значення коефіцієнту	№ з/п	Коефіцієнт	Значення коефіцієнту
1	2	3	4	5	6
1	a	1,000214225	6	f	-0,55597656
2	b	0,052512779	7	g	0,129289625
3	c	1,567945107	8	h	-0,01768063
4	d	-2,12507041	9	i	0,001308658
5	e	1,429643245	10	j	-4,0386e-05

Отримані значення статистичних показників зведено у табл. 4.15.

Таблиця 4.15 – Значення статистичних показників при побудові $\xi_1'(x)$

№ з/п	Показник	Значення показника	№ з/п	Показник	Значення показника
1	2	3	4	5	6
1	$M[\xi_1(x)]$	2,012785	4	$M[\xi_1'(x)]$	2,018247
2	$\sigma[\xi_1(x)]$	$4{,}099853 \cdot 10^{-3}$	5	$\sigma[\xi_1'(x)]$	$3{,}912761 \cdot 10^{-3}$
3	t_1	1,929606	6	t^*	1,962

Згідно даним табл. 4.15, аналогічно до нерівності (4.5), маємо нерівність $t_1 < t^*$. Це є підтвердженням гіпотези H_0' про рівність вибірок $\xi_1(x)$ і $\xi_1'(x)$. Це дає підстави характеризувати отриманий результат як успішний, і використовувати апроксимуючу функцію $\xi_1'(x)$ (4.6) у якості "оцінки зверху" (верхньої границі, найліпшого випадку) корисного ефекту від впровадження результатів проведеного розвитку поширеного методу формальної верифікації TLC з позиції зниження супутніх його застосуванню часових витрат.

Для чисельного подання узагальненої оцінки корисного ефекту від впровадження результатів проведеного розвитку методу запропоновано наступний інтервал:

$$[\min(\xi_2'(x)); \max(\xi_1'(x))], \tag{4.7}$$

де $x \in \{2,3,\ldots,(10^3+1)\}$, $n = |V| \in \{4,5,\ldots 8\}$.

У результаті, згідно (4.7), було здобуто наступну чисельну оцінку проведеного удосконалення методу:

$$[1{,}149;\ 2{,}098], \tag{4.8}$$

тобто корисний ефект від впровадження проведеного удосконалення методу TLC можна оцінити діапазоном значень від 15 % (при $n = 8$) – до 110 % (при $n = 4$).

4.4 Дослідження впливу реалізації мультипоточності

Ставиться завдання оцінити корисний ефект від залучення мультипоточності до BFS- та DFS-реалізацій методу TLC. У якості відповідного показника застосовується коефіцієнт прискорення:

$$\alpha = \bar{t}_1 / \bar{t}_{tc}\ , \tag{4.9}$$

де $\bar{t}_1$ – середній час, витрачений на ФВ методом TLC за однопоточної реалізації, $\bar{t}_{tc}$ – за мультипоточної реалізації, tc – кількість обчислювальних потоків, що одночасно виконуються.

При вирішенні поставленого завдання досліджуються дві альтернативні реалізації методу – на основі BFS- і DFS-обходів. У якості предмету дослідження розглядається сценарій, поданий у попередньому пункті. Це дозволить сформулювати рекомендації стосовно прикладного мультипоточного застосування кожної із реалізацій методу.

Методика проведення дослідження:

– провести ФВ ФС, фрагмент якої подано на рис. 4.9, спочатку BFS-, а потім – DFS-реалізацією методу TLC, що дасть змогу визначити глибину обходу СП – необхідний параметр для застосування DFS-реалізації методу;

– виміряти та порівняти часові витрати на використання одно- і мультипоточної реалізацій методу, розрахувавши коефіцієнти прискорення для мультипоточних реалізацій.

На прикладі СП ФС із $n = 15$ і $depth = 42$ (табл. 4.10) було виміряно часові витрати на ФВ ФС для $tc = 1,2,4,8$. Здобуті результати подано у табл. 4.16 і на рис. 4.14 [162–165].

Таблиця 4.16 – Результати дослідження мультипоточного застосування реалізацій методу

№ з/п	*tc*	Реалізація методу				$\bar{t}_{DFS} / \bar{t}_{BFS}$
		BFS		DFS		
		$\bar{t}_{BFS}$, с	α	$\bar{t}_{DFS}$, с	α	
1	2	3	4	5	6	7
1	1	3,552	1,000	44,575	1,000	12,549
2	2	3,072	1,156	28,384	1,570	9,240
3	4	2,966	1,198	19,989	2,230	6,739
4	8	2,815	1,262	20,526	2,172	7,292

У табл. 4.16 кожне подане значення часових витрат є середнім арифметичним 10 замірів.

Із табл. 4.16 видно, що введення мультипоточності в BFS-реалізацію методу супроводжується незначним ефектом – близько 26 % для 8 потоків. При цьому, для розглянутого випадку, в абсолютному вираженні BFS-реалізація методу є від 12,549 до 6,739 разів ефективнішою. Це можна обґрунтувати як

специфікою структури досліджуваної ФС, так і особливостями як алгоритмічної складової, так і програмних реалізацій відповідних обходів.

Графічне подання здобутих даних представлене на рис. 4.14, де показано залежність значень коефіцієнтів прискорення від кількості обчислювальних потоків. Застосовано кусочно-лінійну інтерполяцію.

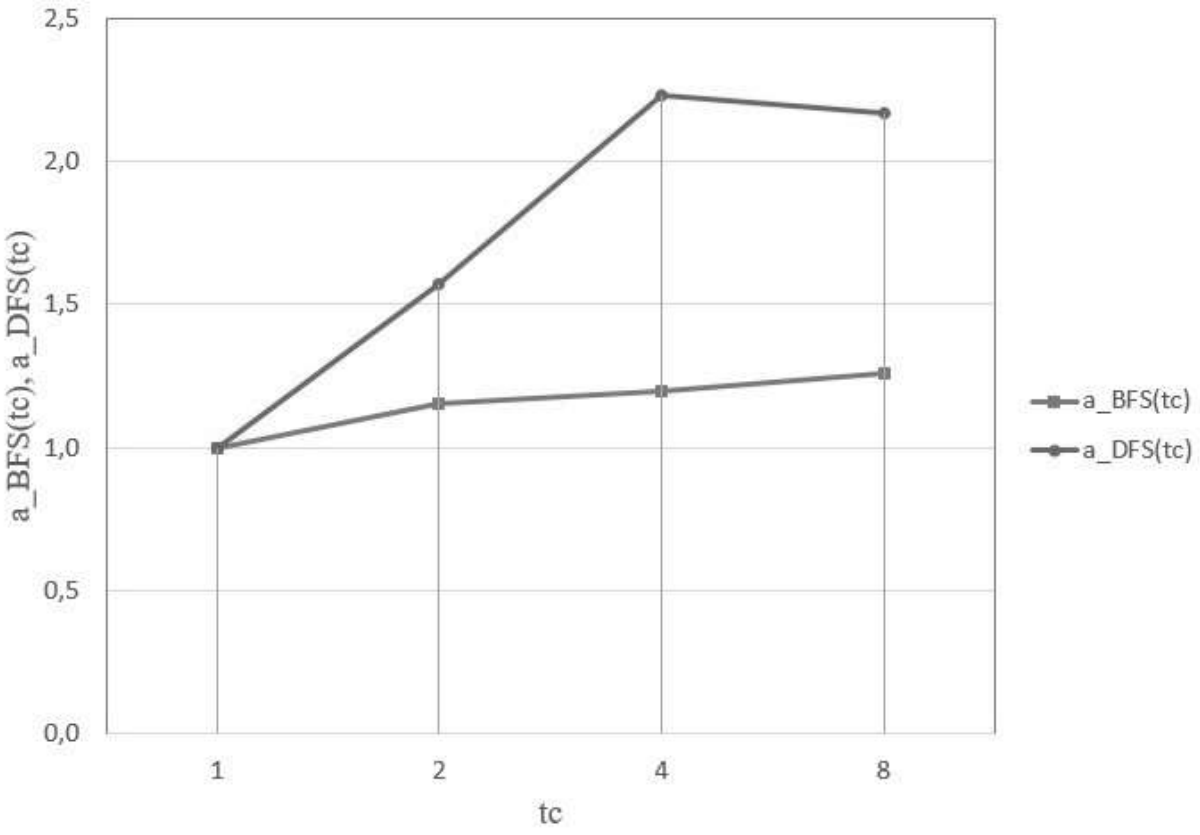

Рисунок 4.14 – Графік залежності коефіцієнту прискорення від числа програмних потоків

З рис. 4.14 видно, що застосування мультипоточної DFS-реалізації методу супроводжується кращим ефектом: для $\bar{t}_{tc} = 4$ одержано прискорення $\alpha = 2{,}230$. При цьому для $\bar{t}_{tc} = 8$ здобуто гірший результат – $\alpha = 2{,}172$. Це можна пояснити специфікою тестової платформи – чотири обчислювальних ядра з реалізацією

технології “Simultaneous Multithreading”, що дозволяє одночасно виконувати два програмних потоки на одному обчислювальному ядрі [166].

Узагальнену оцінку корисного ефекту від введення паралелізму для обох (BFS і DFS) реалізацій методу TLC отримано на основі табл. 4.16 згідно наступного правила:

$$\xi = \max(\alpha_{XFS}), \tag{4.10}$$

де $\alpha_{XFS} \in \{\alpha_{BFS}, \alpha_{DFS}\}$.

Отже, для BFS-випадку маємо $\alpha_{BFS} = 1{,}262$ (або близько 26 %), для DFS-випадку $\alpha_{DFS} = 2{,}230$ (або 123 %).

Узагальнити названу оцінку для обох реалізацій методу можна наступним чином:

$$\xi \in [1{,}262;\ 2{,}230], \text{ для } tc = 2,3,\ldots,8. \tag{4.11}$$

Таким чином, введення мультипоточності до DFS-реалізації методу TLC супроводжується істотно більшим корисним ефектом, у порівнянні із альтернативною BFS-реалізацією. Такий крок є обґрунтованим у випадку співставних часових витрат для однопоточних BFS- і DFS-реалізацій методу.

4.5 Дослідження сценарію галузі енергетики

У якості артефакту, до якого застосовано розроблений і представлений до захисту розвиток методу TLC, адресовано UML-діаграму дій – подання дій оновлення реєстру міжнародних ідентифікаційних кодів, виконуваних згідно гармонізованої моделі європейського ринку електроенергії [167]. Названа діаграма дій окреслює інформаційні потоки, що відбуваються між центральним і місцевим органами видачі ідентифікаційних кодів учасникам ринку електричної енергії, що взаємодіють згідно гармонізованої моделі.

Окреслений сценарій набуває особливої актуальності у контексті інтеграції енергетичного сектору економіки України до складу європейського ринку електричної енергії (рис. 4.15). Окрім зазначеного, дана інтеграція розглядається як важливий крок з позиції підвищення стійкості (резилієнтності) енергетики України [168, 169].

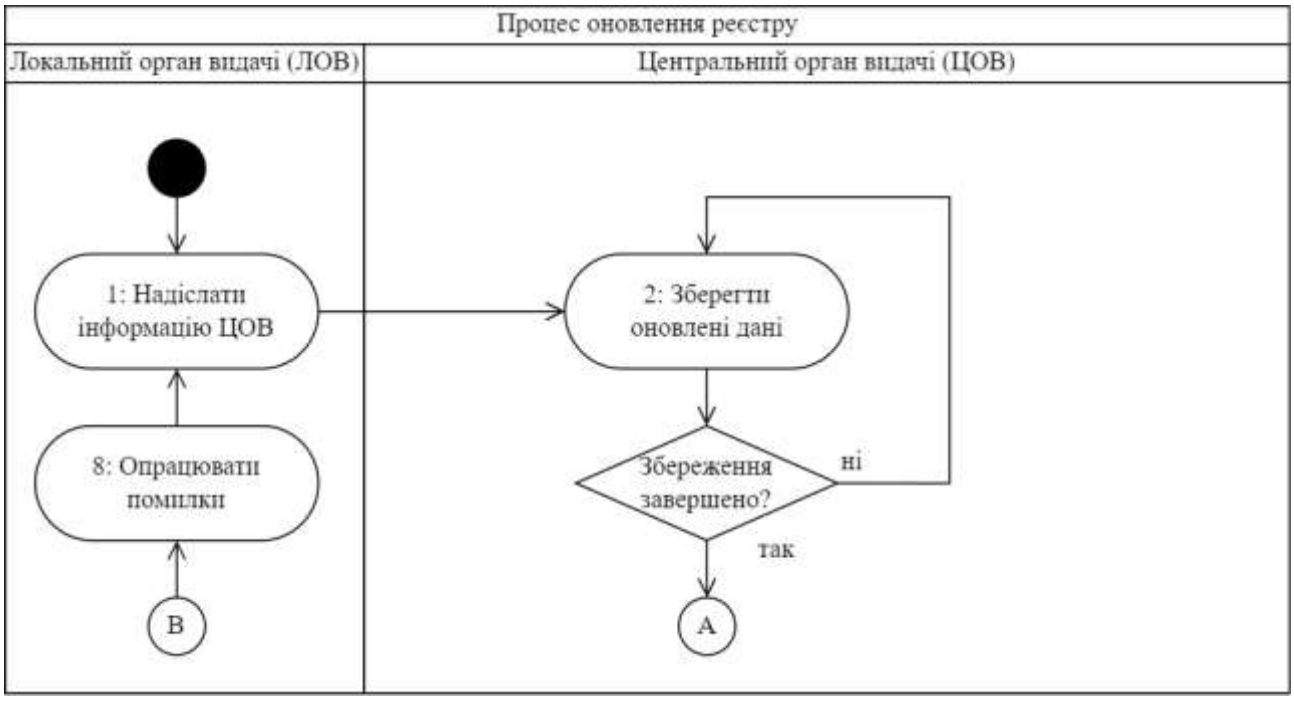

Рисунок 4.15 – Фрагмент рольової моделі оновлення реєстру ідентифікаційних кодів учасників європейського ринку електричної енергії

На рис. 4.15 подано фрагмент UML-діаграми дій з оновлення даних реєстру ідентифікаційних кодів учасників європейського ринку електричної енергії:

1. «Надіслати інформацію ЦОВ (центральному органу видачі кодів)» – локальний орган видачі (ЛОВ) кодів реєстру звертається до ЦОВ, надаючи при цьому інформацію стосовно створення/змін у реєстраційній інформації (активація/дезактивація) коду.

2. Збереження оновлених даних на стороні ЦОВ.

І т. д.

Решту фрагменту подано у додатку Е (рис. Е.1). Дії при цьому пронумеровано у порядку їх виконання. Для випадку Fork-конструкцій альтернативні дії пронумеровано зліва направо.

Повна діаграма, фрагменти якої подано на рис. 4.15 і рис. Е.1, містить 23 дії, не враховуючи подань операторів умовних переходів, яких 6. Названу діаграму розглянуто як первинний артефакт, на основі якого було одержано результуючий артефакт – відповідну ФС – із залученням розроблених моделі і методу, викладених у другому і третьому розділах відповідно.

ФС побудовано згідно наступного підходу:

– кожну дію подано відповідною булевою змінною: $|V'| = 23$. Окрім цього, було введено чотири додаткові змінні, три з яких – подання умовних переходів: одна змінна – подання оператору, що фігурує на рис. 4.15; одна змінна – подання групи умовних переходів, що фігурують на рис. Е.1 (додаток Е); одна змінна – подання умовного переходу, не охопленому у межах фрагментів, поданих на рис. 4.15 і рис. Е.1. У результаті маємо додаткову множину змінних станів, яку позначимо як V'': $|V''| = 3$. Окрім зазначеного, було введено ще одну змінну, призначення якої – слугувати допоміжним засобом індикації

ініціації/завершення процесу поданого UML-діаграмою. У результаті маємо додаткову множину з одного елемента: $\{v^*\}$. Таким чином, результуючу множину змінних станів було одержано наступним чином: $V = V' \cup V'' \cup \{v^*\}$: $|V| = 27$;

– контекстне навантаження атомарних висловлювань наступне: $(v_i, 0) \in AP$ – дію ще не було виконано; $(v_i, 1) \in AP$ – дію вже було виконано.

Просторові витрати на подання ФС наступні:

– для випадку PlusCal-подання – 348 рядків коду;

– для випадку результуючого подання на основі засобів TLA+ – 508 рядків коду.

Отримані результати проведеної ФВ методом TLC виявили потребу доопрацювання UML-діаграми дії з позиції несуперечності. За результатами здійснених доопрацювань – шляхом повторного проведення ФВ доопрацьованої ФС на основі розробленого розвитку методу – було підтверджено несуперечність результуючої ФС. У свою чергу, на основі розробленого методу контролю відповідності результуючої ФС було підтверджено відповідність останньої первинному графічному поданню – UML-діаграмі дій.

Просторові характеристики відповідної СП наступні: $|S| = 290$; глибина обходу простору станів СП склала $|V| + 1 = 28$.

Дослідження впливу введення мультипоточності на задану реалізацію методу TLC було розширено. Отримані результати зведено у табл. 4.17, де охоплено показники і обчислювальних, і просторових витрат, супутніх здійсненню процесу ФВ ФС.

Значення, подані у табл. 4.17, є середнім арифметичним 10^2 замірів.

Із табл. 4.17 видно, що показник $\left|S^*_{BFS}\right| = const$ для $tn = 2^0, 2^1, \ldots, 2^3$. Це свідчить про сталість BFS-реалізації методу TLC з позиції просторових витрат, у залежності від числа залучених програмних потоків. Для DFS-реалізації,

однак, маємо протилежну ситуацію: із збільшенням значення tn у 8 разів просторові витрати теж зростають – приблизно удвічі: $\xi_{DFS} = \left(\overline{S^*_{DFS_1}} \Big/ \overline{S^*_{DFS_{tn}}}\right)$.

Таблиця 4.17 – Експериментальні значення показників обчислювальних і просторових витрат

№ з/п	tn	$f_{BFS}(tn)$, мс	$\alpha_{BFS}(tn)$	$\lvert S^*_{BFS}\rvert$	$f_{DFS}(tn)$, мс	$\alpha_{DFS}(tn)$	$\lvert S^*_{DFS}\rvert$	ξ_{DFS}
1	2	3	4	5	6	7	8	9
1	2^0	1214,000	1,000	336,0	638,400	1,000	4194,000	1,000
2	2^1	1157,800	1,049	336,0	632,900	1,009	5382,890	1,283
3	2^2	1149,100	1,056	336,0	665,300	0,960	7267,050	1,733
4	2^3	1143,800	1,061	336,0	682,700	0,935	8209,850	1,958

Із табл. 4.17 видно, що отриманий у результаті проведених досліджень корисний ефект від введення мультипоточності для даного випадку є істотно меншим, у порівнянні із розглянутим вище сценарієм (табл. 4.16):

– для випадку BFS-реалізації методу він склав не більше 6 %;

– для випадку DFS-реалізації – навпаки – спостерігалось уповільнення швидкодії процесу ФВ ФС до близько 7 %.

Такі результати дають підстави вважати, що одержуваний у результаті залучення мультипоточності корисний ефект істотним чином залежить від архітектурної складової досліджуваного артефакту; і для формулювання у даному контексті рекомендацій для розробників – стосовно доцільності залучення мультипоточності до складу тієї чи іншої реалізації методу TLC – небезпідставним вбачається розширення набору досліджуваних артефактів.

ВИСНОВКИ ДО РОЗДІЛУ 4

Таким чином, у розділі викладено розроблений розвиток поширеного формального методу перевірки на моделі TLC.

Проведений розвиток полягає у сполученні методів обходу у ширину і у глибину теорії графів у процесі здійснення ФВ ФС за ітеративного підходу до організації процесу проєктування ПАС СКП.

Експериментальні дослідження проведено у напрямах оцінювання значень показників обчислювальних і просторових витрат, супутніх застосуванню і базового методу TLC, і розробленого розвитку зазначеного методу. У якості опрацьованих артефактів при цьому залучено формалізовані подання і граничних синтетичних сценаріїв, і предметно-орієнтованих – для галузі енергетики, аерокосмічної галузі.

Граничні сценарії опрацьовано для окреслювання меж одержуваного корисного ефекту від застосування розробленого розвитку методу TLC. При цьому для кількісного оцінювання супутніх просторових витрат було запропоновано і застосовано відповідні оціночні функції.

Для автоматизації процесу проведення досліджень реалізовано і залучено відповідні допоміжні програмні засоби, де реалізовано складові викладених у попередніх розділах моделі і методу.

Отримано, у тому числі, наступні результати:

1. Адресуючи предметну область аерокосмічної галузі, у якості досліджуваного артефакту охоплено, зокрема, фрагмент блок-схеми алгоритму роботи БУК БЦОК КА. При цьому, при дослідженні базового методу TLC, було встановлено, що BFS-реалізація названого методу супроводжувалася меншими просторовими витратами на здійснення ФВ ФС, у порівнянні із альтернативною DFS-реалізацією, – на близько 16,7 %. У свою чергу, застосування DFS-

реалізації методу TLC супроводжувалося у близько 2,5 рази вищою ефективністю за показником супутніх ФВ часових витрат, у порівнянні із альтернативною реалізацією на основі BFS-обходу. При цьому DFS-реалізація методу супроводжується суттєвим недоліком з позиції автоматизації – для її застосування необхідно вказати глибину обходу простору станів СП, у той час як для BFS-реалізації методу значення даного параметру встановлювати і зазначати не потрібно. Більше того, застосування BFS-реалізації методу дає можливість визначити глибину обходу.

Вищезазначену специфіку узагальнено наступним чином: BFS-реалізація методу TLC ліпше придатна до автоматизації процесу її застосування, у порівнянні із альтернативою – DFS-реалізацією. Разом із цим, застосування DFS-реалізації методу супроводжувалося істотно вищою ефективністю за показником супутніх процесу ФВ ФС часових витрат. На основі отриманих результатів було висунуто припущення, що корисний ефект від застосування заданої реалізації методу TLC визначається, у тому числі, наступними факторами: кількістю змінних станів, що фігурують у ФС, архітектурною складовою ФС. Для перевірки даного припущення проведено дослідження граничних випадків, отримані результати яких подано нижче.

2. У якості граничного (найліпшого) випадку – «оцінка знизу» – досліджено штучний сценарій, згідно якого ФС синтезовано із архітектурною складовою послідовного характеру – без умовних переходів.

Експериментальні дослідження проведено для числа змінних станів $n = 2^1, 2^2, ..., 2^8$. Відповідні ФС отримано автоматизованим шляхом – із залученням розробленого допоміжного програмного забезпечення, де реалізовано положення розробленої моделі подання ФС (розділ 2), у відповідності до кроків розробленого методу синтезу ФС (розділ 3).

Дослідження базового методу TLC проведено у розрізі оцінювання обчислювальних і просторових витрат, супутніх процесу ФВ ФС – по

відношенню до двох альтернативних реалізацій методу – BFS і DFS. На підставі отриманих результатів було встановлено, що для заданої ФС має місце граничне значення n, за якого ефективність DFS-реалізації методу TLC є вищою, у порівнянні із альтернативною BFS-реалізацією методу, – за показником супутніх обчислювальних витрат. Назване граничне значення, у свою чергу, залежить від архітектурної складової досліджуваної ФС. Для досліджуваного випадку таке значення становило $n \le 2^4$. У свою чергу, для випадку $n = 2^1$ DFS-реалізація методу супроводжувалася від близько 2,2 до близько 2,8 разів вищою ефективністю; для випадку $n = 2^4$ – від близько 1,5 до близько 1,6 разів.

Згідно до зазначених вище отриманих результатів, було встановлено обернено пропорційну тенденцію, згідно якої, із зростанням значення n, перевага DFS-реалізації методу над альтернативною BFS-реалізацією – з позиції супутніх обчислювальних витрат – зменшується – допоки не буде досягнуто граничного значення n.

Подальше збільшення значення n призводить до протилежної ситуації, за якої ліпшою ефективністю за критерієм обчислювальних витрат вже характеризується саме BFS-реалізація методу TLC: наприклад, для випадку $n = 2^5$ отримані результати досліджень продемонстрували вищу ефективність залучення саме BFS-реалізації – у діапазоні від близько 1,1 до близько 2,2 разів; у свою чергу, для випадку $n = 2^8$ перевага BFS-реалізації вже наближалась до 100-кратної.

Адресуючи відзначену вище специфіку, було прийнято рішення розвинути базовий метод TLC таким чином, щоб, з урахуванням граничного значення n, досягти підвищення ефективності роботи методу TLC за ітеративного підходу до реалізації процесу ФВ ФС – за показником супутніх названому процесу часових витрат.

Варто також зауважити, що за показником супутніх процесу ФВ ФС просторових витрат DFS-реалізація методу TLC супроводжувалася істотно нижчою ефективністю, у порівнянні із альтернативною BFS-реалізацією методу, – від близько двох – до близько 256 разів – для $n = 2^1, 2^2, ..., 2^8$ відповідно. При цьому було зафіксовано, що для випадку $n = 2^8$ при проведенні ФВ ФС шляхом застосування DFS-реалізації методу було залучено більше 5 ГБ ОП.

3. У якості другого граничного (найгіршого) випадку – «оцінки зверху» – досліджено штучний сценарій, у відповідності до якого ФС синтезовано із поданням паралелізму згідно моделі чергування. Дослідження проведено для випадків $n = 2^2, 2^3, 2^4$. При цьому для випадку $n = 2^4$ експеримент було припинено – через нестачу доступної обчислювальній системі ОП. Зокрема, 8 ГБ ОП було вичерпано за 23 хв і 19 с. У свою чергу, для випадків $n = 2^2$ і $n = 2^3$ ефективність DFS-реалізації методу виявилась вищою за ефективність BFS-реалізації – у близько 2,18 і 1,76 разів відповідно – за показником супутніх процесу ФВ ФС часових витрат. При цьому за показником просторових витрат мала місце протилежна ситуація – ефективність BFS-реалізації була вищою у близько 1,92 і 3,57 разів відповідно.

4. У якості предметно-орієнтованого артефакту досліджено подання сценарію предметної області із галузі енергетики – блок-схему алгоритму контролю вихідного стану регістрів пристрою введення-виведення бортового цифрового обчислювального комплексу. Згідно отриманих результатів проведених досліджень було, у тому числі, підтверджено несуперечність ПАС, а також встановлено граничне значення $n = 8$, за якого все ще було доцільним сполучати методи обходу у глибину і у ширину теорії графів – згідно розробленого розвитку методу TLC. Залучаючи дане значення, проведено оцінювання граничних значень корисного ефекту, одержуваного у результаті застосування розробленого розвитку методу TLC. Шляхом вирішення задачі

апроксимації було отримано відповідні оціночні функції меж одержуваного корисного ефекту від прикладного застосування зазначеного розвитку методу – «оцінку зверху» (найгірший випадок) і «оцінку знизу» (найліпший випадок). Із залученням цих функцій, одержуваний корисний ефект оцінено діапазоном відносних значень від близько 15 % (для випадку $n = 8$) до близько 110 % (для випадку $n = 4$). Для випадку $n > 8$ було встановлено, що застосування проведеного розвитку методу буде недоцільним.

5. У якості предметно-орієнтованого артефакту галузі енергетики досліджено UML-діаграму дій – подання рольової моделі оновлення реєстру ідентифікаційних кодів учасників європейського ринку електричної енергії. У результаті застосування розробленого розвитку методу TLC зазначений артефакт було доопрацьовано з позиції досягнення несуперечності. Повторне застосування названого розвитку методу дозволило у цьому пересвідчитись.

6. З урахуванням поточних реалій – мультипоточності обчислювальних систем, проведено дослідження корисного ефекту, одержуваного у результаті введення мультипоточності до складу BFS- і DFS-реалізацій методу TLC.

Дослідження проведено для двох випадків – предметно-орієнтованих сценаріїв – на основі артефактів аерокосмічної галузі і галузі енергетики. У першому випадку у якості артефакту досліджено блок-схему алгоритму, у другому – UML-діаграму дій.

У першому випадку істотно більшого корисного ефекту було досягнуто по відношенню до DFS-реалізації методу TLC – значення коефіцієнту прискорення склали близько 1,57; 2,23 і 2,17 разів – для 2; 4 і 8 програмних потоків відповідно. При цьому для випадку залучення 8 потоків спостерігалося зниження зазначеного ефекту, у порівнянні із використанням 4 потоків. У свою чергу, найліпше зафіксоване значення зазначеного коефіцієнту для BFS-реалізації методу склало близько 1,26 – для випадку 8 потоків.

Для випадку дослідження у якості артефакту UML-діаграми дій отриманий корисний ефект був істотно меншим: до близько 6 % пришвидшення – для випадку введення мультипоточності до складу BFS-реалізації методу TLC; до близько 7 % уповільнення – для випадку залучення мультипоточності до складу DFS-реалізації методу.

Отже, отримані результати проведеного дослідження корисного ефекту від введення мультипоточності до складу реалізацій методу TLC можна охарактеризувати як такі, що істотним чином залежать від архітектурної складової відповідного артефакту. При цьому, однак, варто відзначити, що ліпші результати було отримано саме для випадку BFS-реалізації методу.

Таким чином, одержуваний корисний ефект від прикладного застосування розробленого розвитку методу TLC істотним чином визначається специфікою досліджуваного артефакту заданої предметної області – залежить, у тому числі, від кількості ітерацій процесу ФВ ФС, кількості змінних ФС, архітектурної складової ФС.

РОЗДІЛ 5
РОЗРОБЛЕННЯ МОДЕЛІ ЯК СТРАТИФІКОВАНОЇ АРХІТЕКТУРИ

У розділі викладено розроблену модель як стратифіковану архітектуру подання ПАС у формі складеної комп'ютерної моделі, призначену слугувати засобом уможливлення охоплення у формалізованому поданні також і складових досліджуваного показника НФХ, із відтворенням архітектурної складової ФС, несуперечність якої попередньо вже було підтверджено на основі формального методу перевірки на моделі TLC або на основі розробленого і викладеного у попередньому, четвертому, розділі розвитку зазначеного методу, призначеного до застосування за ітеративного підходу до організації процесу ФВ.

Представлена модель, на відміну від розробленої моделі, викладеної у другому розділі, є інструментом, що надає засоби подання, у тому числі, складових досліджуваного показника НФХ у процесі проєктування ПАС, оперуючи при цьому як оціночними значеннями, так і фактичними.

Стратифікований підхід до розроблення моделі застосовано з міркувань забезпечення наступних властивостей похідних від неї конструкцій:

– гнучкості відтворення архітектурної складової ФС, несуперечність яких вже було попередньо підтверджено згідно розробленого підходу (рис. 2.1), – на основі розроблених і викладених у другому – четвертому розділах моделі подання ПАС, методу синтезу ФС і розвитку методу TLC;

– модульності і структурованості результуючої складеної комп'ютерної моделі – похідної конструкції від представленої моделі як стратифікованої архітектури, призначеної слугувати засобом уможливлення проведення контролю досліджуваного показника НФХ вже на етапі проєктування ПАС.

Для реалізації зазначених вище властивостей застосовано математичний апарат DEVS, у тому числі конструкції «атомарної» і «складеної» моделей – у відповідності до ієрархічного підходу:

– у якості елементів нижнього ієрархічного рівня у складі представленої моделі як ієрархічної архітектури залучено елементи, сформовані згідно конструкції «атомарної» моделі математичного апарату DEVS;

– елементи наступних страт прийнято рішення подавати згідно конструкції «складеної» моделі математичного апарату DEVS: елементи формуються шляхом встановлення сполучень між елементами попередніх страт, і реалізації механізму обміну повідомленнями між ними.

У якості досліджуваного показника НФХ при проєктуванні ПАС прийнято рішення опрацьовувати часові витрати, супутні реалізації кроків ПАС. На таке рішення вплинули, у тому числі, наступні аспекти:

– особливості математичного апарату DEVS, реалізованого згідно положень теорії часових автоматів, що включає засоби подання модельного часу;

– наявність вже існуючих засобів – програмних реалізацій, що включають також і засоби проведення замірів реального часу у процесі здійснення імітаційного дискретно-подійного моделювання.

Зауваження:

– аналогічним чином можуть бути опрацьовані, у тому числі, і матеріальні витрати як показник НФХ.

5.1 Формулювання вирішуваної задачі

У межах розділу досліджувані артефакти як форми подання ПАС на етапі її проєктування опрацьовуються з позиції спадковості їх архітектурної складової: у відповідності до концепції в основі озвученого вище поняття багатовимірної верифікації [170]. Згідно до цього введемо в обіг поняття первинного (ПА), вихідного (ВА) і результуючого артефактів (РА), кожне з яких асоціюється із відповідним типом артефактів.

Типом ПА охоплено графічні подання ПАС: у формі блок-схем алгоритмів, UML-діаграм дій, станів. У свою чергу, серед перелічених прикладів, у контексті виконуваних досліджень, UML-діаграми станів було опрацьовано як похідні від артефактів, поданих першими двома прикладами:

– оскільки діаграми станів попередньо було одержано одним з наступних шляхів: або шляхом безпосереднього аналізу решти перелічених представників типу ПА розробником, або за результатами автоматизованого опрацювання відповідних формалізованих подань.

Спільною рисою артефактів типу ПА є, у тому числі, і вагомий недолік з позиції ФБ – вплив людського фактору на результати їх аналізу є безпосереднім і вирішальним. У свою чергу, зниження ролі даного фактору можливе за рахунок введення формалізації, що сприяє і підвищенню рівня строгості подання і сприйняття артефактів розробником, і уможливленню автоматизації процесів їх опрацювання. Це є обґрунтуванням доцільності виокремлення також і іншого згаданого вище типу – ВА. При цьому артефакти типу ПА адресовано як «первинні», оскільки саме вони розглядаються як вихідні конструкції, по відношенню до яких застосовується розроблений у межах дисертації комплекс методів і засобів, сполучених на основі представленого підходу, викладеного у

межах другого розділу. При реалізації цього підходу до обігу вводиться решта виокремлених типів – ВА і РА.

Артефакти типу ВА.

У якості представників даного типу опрацьовано ФС, одержувані згідно розроблених і викладених у межах другого і третього розділів моделі подання ПАС у формі ФС і методу синтезу ФС згідно цієї моделі відповідно.

Представники типу ВА призначені слугувати засобами уможливлення проведення контролю досліджуваних артефактів у процесі проєктування ПАС за показником несуперечності ПАС в автоматизованому режимі. Відповідними прикладами є ФС на основі засобів алгоритмічної мови PlusCal, а також ФС на основі засобів формалізму TLA+. При цьому PlusCal-подання, у відповідності до згаданого вище підходу, передують за послідовністю появи артефактам на основі засобів TLA+.

Аналогічно до проведеного розмежування представників охопленого вище типу ПА, артефакти типу ВА також ранжовано за показником «первинності» у межах типу: первинними є подання на основі засобів PlusCal, а похідними від них – ФС на основі засобів TLA+, до яких безпосередньо застосовується формальний метод перевірки на моделі TLC або розроблений розвиток цього методу, представлений у четвертому розділі.

Засоби PlusCal призначені для попередньої формалізації архітектурної складової ФС, а засоби TLA+ – розширюють ці подання до форми, що безпосередньо уможливлює автоматизацію процесу ФВ.

Отже, відмінною рисою артефактів виокремленого типу ВА є наявність формалізації у контексті уможливлення проведення ФВ в автоматизованому режимі для здійснення контролю артефактів при проєктуванні ПАС за показником несуперечності ПАС. У свою чергу, як засіб узагальнення артефактів, виразні засоби подання яких включають також і засоби

формалізованого представлення досліджуваного показника НФХ, було виокремлено також і тип РА.

Артефакти типу РА.

На оперуванні артефактами даного типу базуються викладена у межах поточного розділу розроблена модель як стратифікована архітектура, а також розроблений метод контролю досліджуваного показника НФХ при проєктуванні ПАС, представлений у наступному розділі. Для цього типу, за аналогією до двох розглянутих вище, також проведено ранжування відповідних артефактів-представників: у даному випадку – за рівнями ієрархії.

Виокремлено наступні групи елементів типу РА:

– артефакти, реалізовані згідно конструкції «атомарної» моделі DEVS, як елементи нижнього ієрархічного рівня – формуючі елементи, на основі яких реалізуються похідні від них елементи наступних ієрархічних рівнів, виокремлених у складі розробленої моделі як ієрархічної архітектури;

– артефакти, сформовані у відповідності до конструкції «складеної» моделі DEVS, що реалізуються шляхом встановлення сполучень між елементами попередньої і даної груп, у відповідності до архітектурної складової розроблюваної ПАС.

Взаємодію елементів ієрархічних рівнів реалізовано на основі механізму обміну повідомленнями.

Окреслений вище підхід прийнято з метою одержання модульних, реконфігурованих ієрархічних конструкцій, що є придатними до відтворення архітектурної складової ПАС на основі відповідних ФС, несуперечність яких вже було попередньо підтверджено шляхом формальної верифікації на основі методу TLC або на основі розробленого розвитку даного методу, викладеного у четвертому розділі.

Властивість модульності сприяє зручності реконфігурування артефактів типу РА, у тому числі у частині варіювання рівня деталізації останніх.

Графічне подання виокремлених типів артефактів, із встановленими відношеннями між розглянутими типами, виконано на основі UML-нотації (рис. 5.1) [171].

Рисунок 5.1 – Графічне подання властивості спадковості виокремлених типів артефактів

На рисунку 5.1 для подання властивості спадковості виокремлених типів артефактів залучено відношення «Extends» (розширює) виразних засобів UML. У це подання вкладено наступний зміст:

– тип, що розширює базовий по відношенню до нього тип, успадковує і архітектурну складову, і змістове навантаження, – як основоположні сутності, що формують контекст поняття «артефакт» (рис. 1.1). Згідно цієї концепції реалізується відтворення, у тому числі, архітектурної складової артефактів типу

ВА на основі засобів подання похідних від них артефактів типу РА, одержуваних у відповідності до викладеної у межах розділу розробленої моделі як стратифікованої архітектури.

На рисунку 5.1 у вигляді коментарів наведено форми подання, а також відповідні виразні засоби, для артефактів розглянутих типів.

Якщо розглядати змістове навантаження UML-діаграми рисунку 5.1 як покроковий процес, для якого встановлені відношення – подання кроків, а об’єкти – відповідні перед- і пост-умови, то результатом виконання першого кроку мають бути артефакти типу ВА, а результатом здійснення другого кроку – артефакти типу РА. Передумовою виконання другого кроку при цьому є несуперечність ПАС як досліджуваний показник ФХ. У свою чергу, артефакти типу РА є засобами, на основі яких проводиться контроль досліджуваного показника НФХ при проєктуванні ПАС.

Під контролем при цьому розуміється здійснення перевірки значення досліджуваного показника НФХ на відповідність встановленим обмеженням. Згідно концепції в основі представленої моделі як стратифікованої архітектури, зазначену перевірку пропонується виконувати шляхом проведення дискретно-подійного імітаційного моделювання на основі ієрархічної комп’ютерної моделі як похідної від представленої моделі конструкції. Така похідна ієрархічна конструкція, у свою чергу, реалізується на основі засобів DEVS, а також засобів високорівневої мови програмування – Java.

Базуючись на вищезазначеному, у межах поточного розділу до вирішення ставиться наступна задача:

– розроблення моделі як стратифікованої архітектури, застосування якої при проєктуванні ПАС уможливило б проведення контролю досліджуваних артефактів типу РА за показником НФХ, базуючись на відповідних артефактах типу ВА, контроль яких вже було проведено за показником ФХ – несуперечністю ПАС, шляхом залучення розроблених і викладених у

попередніх другому – четвертому розділах моделі подання ПАС, методу синтезу ФС, методу контролю відповідності результуючих ФС, а також розробленого розвитку методу TLC.

Дану постановку вирішуваної у межах розділу задачі виконано у відповідності до озвученої вище концепції багатовимірної верифікації [64]. При цьому, згідно до розробленого комплексного підходу, представленого у межах другого розділу, контроль досліджуваних показників і ФХ, і НФХ, має проводитись вже на етапі проєктування у складі етапів процесу розроблення ПАС.

Для уможливлення цього у частині контролю показника НФХ опрацюванню підлягають, у тому числі, наступні аспекти одержання артефактів типу РА на основі артефактів типу ВА (рис. 5.1):

– відтворення архітектурної складової (структури та зав’язків) артефакту типу ВА у межах результуючого артефакту типу РА;

– доповнення результуючих формалізованих подань типу РА поданнями складових досліджуваного показника НФХ. У якості відповідних значень, у свою чергу, допускається оперування як оціночними, так і фактичними значеннями. Зауваження: фактичні значення допускається отримувати безпосередньо у процесі дискретно-подійного імітаційного моделювання на основі результуючої комп’ютерної моделі – артефакту типу РА: шляхом залучення вбудованих програмних засобів вимірювання часових витрат, супутніх проведенню обчислень у складі артефактів типу РА. Зазначені обчислення також допустимо відтворювати у форматі «один-до-одного».

– забезпечення можливості варіювання рівня деталізації подання ПАС у формі артефакту типу РА. Цього планується досягати шляхом виокремлення ієрархічних рівнів і оперування при цьому конструкціями атомарної і складеної моделей DEVS.

5.2 Теоретичні засади в основі розробленої моделі

Для регламентації контекстного навантаження складових застосовуваного при викладені розробленої стратифікованої моделі понятійного апарату встановимо відношення між зазначеними складовими, із залученням виразних засобів мови UML, зокрема залучивши відношення «реалізація» (пунктирна стрілка) і «агрегування» (ромбовидна стрілка) (рис. 5.2) [85, 172]:

– відношення «реалізація» призначене акцентувати увагу, що артефакт є реалізацією – формою подання – прийнятого розробником ПР: графічною (UML-діаграма дій, блок-схема алгоритму тощо) або формалізованою (формалізовані подання ПАС). Обґрунтування важливості проведення контролю показників і НФХ при проєктуванні здійснено, зокрема, у [172, 173];

– у свою чергу, залучення саме відношення «агрегування», а не «композиції», означає, що відповідні блоки діаграми можуть фігурувати у складі і інших конструкцій.

Згідно рис. 1.1, за точку відліку береться ПАС, формою подання якої є артефакт – графічне або формалізоване подання. При цьому артефакт типу ВА, одержуваний на основі виразних засобів алгоритмічної мови PlusCal і побудований у відповідності до моделі, викладеної у другому розділі, адресується у якості вихідної конструкції, у відповідності до якої, із збереженням архітектурної складової названої конструкції, одержуємо результуючий артефакт типу РА [86, 176, 177]. Для цього залучається викладена у поточному розділі стратифікована модель – у якості шаблону (прототипу) цільового артефакту типу РА, побудованого на основі засобів DEVS, що уможливить здійснення контролю заданого показника НФХ розроблюваної ПАС СКП вже на етапі проєктування процесу розроблення, у відповідності до

запропонованого комплексного підходу, викладеного у другому розділі. Наприклад, часову затримку (час відгуку компонента розподіленої системи) було проаналізовано у якості показника НФХ [178–182].

Викладений підхід (рис. 5.1), згідно якого будується розроблена модель, пройшов відповідну апробацію [185]. Застосування підходу орієнтоване на відтворення архітектурної складової артефакту типу ВА, несуперечність якого вже було підтверджено методом перевірки на моделі TLC або на основі розробленого розвитку зазначеного методу, у межах результуючого артефакту типу РА – складеної комп'ютерної моделі, побудованої на основі засобів математичного апарату DEVS.

Прикладами артефактів типу ВА, несуперечність яких вже було підтверджено, є опрацьовані у четвертому розділі наступні ФС: для блок-схеми алгоритму контролю вихідного стану регістрів ПВВ БЦОК [165], а також для UML-діаграми дій з оновлення ідентифікаційних ключів учасників ринку електричної енергії [167].

Для досягнення модульності і структурованості результуючих рішень викладену у межах розділу модель побудовано згідно положень теорії ієрархічних багаторівневих систем [130].

Ієрархічні рівні (страти) виокремлено у відповідності до конструкцій «атомарної» і «складеної» моделей математичного апарату DEVS.

Атомарні моделі опрацьовано у якості елементів нижнього ієрархічного рівня. Змістова складова відповідних елементів формує рівень деталізації результуючої конструкції [114, 177]:

$$AM_j = \langle X, ST, Y, \delta_{ext}, \delta_{\text{int}}, \lambda, ta \rangle,\ j = 1,2,\ldots,n \in N, \qquad (5.1)$$

де X – множина зовнішніх по відношенню до атомарної моделі подій, що призводять до зміни її стану із поточною міткою $st \in ST = \{"busy","passive"\}$, де $"busy" \in ST$ – модель перебуває у стані опрацювання повідомлення, спричиненому подією $x \in X$, $"passive" \in ST$ – модель перебуває у стані очікування; Y – множина подій, які модель продукує; $\delta_{ext} : Q \times X \to ST$ – зовнішня функція переходу: $Q = \{(st, e) | st \in ST, 0 \le e \le ta(st)\}$, де Q – загальна множина станів, e – модельний час, що пройшов від моменту останнього переходу, $ta : ST \to R^{+}_{0,\infty}$ – функція просування модельного часу; $\delta_{\text{int}} : ST \to ST$ – внутрішня функція переходу; $\lambda : ST \to Y$ – функція виходу.

Наявність елементу ta у складі структури (5.1) уможливлює опрацювання часових витрат у якості досліджуваного показника НФХ при проведенні дискретно-подійного імітаційного моделювання.

Як альтернатива, замість часових витрат допустимо також опрацьовувати вартісні витрати, накопичуючи результуюче значення шляхом обміну повідомленнями між атомарними моделями компонентів у процесі моделювання.

Підхід до побудови представленої у межах розділу моделі подання ПАС полягає у виконанні нижченаведених кроків.

Крок 1. У відповідності до обраного розробником рівня деталізації створюваної комп'ютерної моделі на основі засобів DEVS, згідно структури (5.1) формуються елементи нижнього – базового – ієрархічного рівня зазначеної складеної конструкції.

У загальному випадку, у тому числі – при проведенні досліджень розробленої моделі, застосовано підхід «один-до-одного» – коли змінній, що фігурує у ФС на основі засобів PlusCal і TLA+, ставиться у відповідність атомарна модель (5.1). Як результат, маємо наступне: $\left|\{AM_j\}\right| = |V| = n$. У

такому випадку досягається найвищий рівень деталізації створюваної комп'ютерної моделі як засобу контролю досліджуваного показника НФХ.

У випадку потреби варіювання рівня деталізації застосовується абстрагування, за якого структура (5.1) ставиться у відповідність певній підмножині $V^* \subset V$. При цьому зміст результуючого артефакту – комп'ютерної моделі – програмної реалізації згідно (5.1) – визначається контекстним навантаженням залучених елементів підмножини V^*, а архітектура – складом цих елементів і встановленими між ними зв'язками.

Крок 2. Базуючись на виокремлених елементах нижнього ієрархічного рівня, формуються елементи наступних страт – шляхом встановлення зв'язків між елементами множини $\{AM_i\}$.

Зазначений крок реалізується наступним чином:

– якщо позначити через $k = 1,2,...,n \in N$ порядковий номер страти, починаючи з нижнього ієрархічного рівня, то елемент $(k+1)$-го рівня відносно k-го формується шляхом сполучення елементів k-го рівня. Для цього залучаються відповідні «порти». Кожен «порт» при цьому опрацьовується як сутність, що визначає тип повідомлень, що з/до нього надходять.

Надсилання/надходження повідомлень з/до портів адресуються як «події», виникнення яких уможливлюється за рахунок просування модельного часу.

Для страт із номерами $k = 2,3,...,n \in N$ відповідні елементи формуються на основі конструкції «складеної» моделі DEVS, яку подано наступним чином [174, 175]:

$$CM^k = \left\langle INP^{k-1}, OUTP^{k-1}, *MS^{k-1}, set \right\rangle, \qquad (5.2)$$

де верхній індекс k позначає порядковий номер страти, починаючи з другого знизу рівня; при цьому складові структури асоціюються з елементами на рівень нижче:

– для випадку $k=2$ зазначені елементи розміщуються на початковому рівні, і будуються згідно виразу (5.1);

– для випадку $k>2$ зазначені елементи розміщуються на наступних рівнях, і будуються згідно виразу (5.2);

– INP^{k-1} ($OUTP^{k-1}$) – множина усіх вхідних (вихідних) портів – засобів зв'язку елементів $(k-1)$-го ієрархічного рівня;

– $*MS^{k-1}=\{*M_j^{k-1}\}$ – узагальнене позначення множини комп'ютерних моделей, побудованих на основі засобів математичного апарату DEVS, і розміщених на $(k-1)$-му ієрархічному рівні; при цьому кожний елемент зазначеної множини, для взаємодії з рештою елементів вказаного рівня, містить один або більше елементів множини INP^{k-1} та/або множини $OUTP^{k-1}$; замість узагальнюючого умовного позначення «*» при цьому може фігурувати літера «A» (для випадку $k=2$) або літера «C» (для випадку $k>2$);

– $set:*MS^{k-1}\times OUTP^{k-1}\to *MS^{k-1}\times INP^{k-1}$ – функція встановлення зав'язків між портами елементів спільного рівня з номером $(k-1)$.

Як узагальнення вищезазначеному, на рівні структур (5.2), на відміну від структур (5.1), вже оперуємо поняттям «порт» – сутністю, що визначає тип відповідних подій, а не поняттям «подія» безпосередньо.

Показовими прикладами у даному випадку є композитні вебсервіси як форми організації розподіленої комп'ютерної системи для реалізації заданих ФХ [116] і [117]. Для зазначених прикладів представлена модель була реалізована як дворівнева ієрархічна конструкція. У свою чергу, «атомарні» комп'ютерні моделі відповідно до (5.1) було побудовано у режимі «один-до-одного»:

– кожному атомарному вебсервісу у складі композитного вебсервісу поставлено у відповідність «атомарну» модель DEVS (5.1);

– результуючому композитному вебсервісу як засобу реалізації досліджуваної ФХ поставлено у відповідність «складену» модель DEVS (5.2);

У даному контексті, як приклад, у наступному – шостому – розділі опрацьовано сценарій розподіленого обчислення значення π через арктангенси.

У якості супутнього досліджуваного показника НФХ при цьому було охоплено часові витрати, супутні реалізації ФХ. Аналогічний підхід було застосовано і на прикладі побудови складеної моделі для фрагменту блок-схеми алгоритму управління конфігурацією бортового цифрового обчислювального комплексу космічного апарату [177].

Як механізм реалізації успадковування архітектурної складової артефакту типу ВА, несуперечність якого вже було підтверджено на основі базового методу TLC або згідно розробленого розвитку зазначеного методу, викладеного у попередньому – четвертому – розділі (рис. 5.1), застосуємо подійно-орієнтований підхід:

– у тому числі, шляхом співвіднесення подій, що мають місце на суміжних $(k-1)$-й і k-й стратах; події при цьому опрацьовано як змушуючі чинники зміни поточного стану результуючого складеного артефакту типу РА, із прив'язкою до поточного рівня із номером $(k-1)$.

Для зміщення від оперування подіями $(k-1)$-го рівня – до опрацювання подій k-го рівня, по відношенню до подій, що мають місце на $(k-1)$-му рівні, застосуємо правило композиції (2.8). При цьому контекстне навантаження подій, що тепер адресуються, було заміщено новим:

– у якості події вже розглядаємо не модифікацію значення змінної $v_j \in V$, а, у тому числі, – факт надходження на порт $inp_{*M_j}^{k-1} \in INP^{k-1}$ відповідної моделі $*M_j^{k-1} \in *MS^{k-1}$, побудованої на основі засобів DEVS, повідомлення,

надісланого з вихідного порту $outp_{*M_r}^{k-1} \in OUTP^{k-1}$ іншого (r-го; $r \neq j$) елементу $*M_r^{k-1} \in *MS^{k-1}$ поточного ієрархічного рівня із номером $(k-1)$. Зазначене надходження уможливлено за рахунок застосування функції set у якості засобу сполучення портів елементів рівня $(k-1)$ (5.2). При цьому активація елементу $*M_r^{k-1} \in *MS^{k-1}$ передує активації елементу $*M_j^{k-1} \in *MS^{k-1}$, у відповідності до порядку виникнення подій, встановленого згідно протоколу обчислювального процесу (2.5) на основі засобів CSP.

З урахуванням вищезазначеного, змістове навантаження перед- і пост-умов виникнення подій також було змінено, у порівнянні з таким, що мало місце для розробленої моделі формалізованого подання ПАС, викладеної у другому розділі, за якої у якості перед- і пост-умов було опрацьовано формалізовані подання розміток станів СП:

– натомість, у випадку поточної представленої моделі у якості перед- і пост-умов також залучено події – елементи множин X і Y у складі структури (5.1).

– у свою чергу, у якості дій, яким передують передумови як елементи множини X послідовно залучимо згадані вище функцій у складі структури (5.1) – у наступному порядку: $\delta_{ext}, \delta_{\text{int}}, \lambda$; як результат, замість трійок Гоара вигляду (2.6) матимемо наступні конструкції-засоби сполучення подій:

$$\{x\}\delta_{ext}, \delta_{\text{int}}, \lambda\{y\}, \tag{5.3}$$

де $x \in X, y \in Y$. У свою чергу, порядок запису функцій $\delta_{ext}, \delta_{\text{int}}, \lambda$ задає порядок їх виклику: тобто істинності набуває висловлювання $\delta_{ext} \prec \delta_{\text{int}} \prec \lambda$, де $\prec$ – оператор передування. Отже, події-елементи протоколу обчислювального

процесу (2.5) пропонується подавати засобами DEVS на основі трійок Гоара згідно виразу (5.3).

Для відтворення засобами DEVS безпосередньо протоколу (2.5) як цілісної конструкції застосуємо попередньо використане у другому розділі правило композиції Гоара, подане вище у формі виразу (2.8), оперуючи вже конструкціями вигляду (5.3):

$$\frac{\left\{x_1^{AM_1}\right\}\left\{\delta_{ext}^{AM_1},\delta_{\text{int}}^{AM_1},\lambda^{AM_1}\left\{y_1^{AM_1}\right\}\right\},\ldots,\left\{x_l^{AM_n}\right\}\left\{\delta_{ext}^{AM_n},\delta_{\text{int}}^{AM_n},\lambda^{AM_n}\left\{y_l^{AM_n}\right\}\right\}}{\left\{x_1^{AM_1}\right\}\left\{\left(\delta_{ext}^{AM_1},\delta_{\text{int}}^{AM_1},\lambda^{AM_1}\right),\ldots,\left(\delta_{ext}^{AM_n},\delta_{\text{int}}^{AM_n},\lambda^{AM_n}\right)\right\}\left\{y_l^{AM_n}\right\}}, \qquad (5.4)$$

де верхній індекс AM_j елементів у складі структури (5.1) ідентифікує відповідну атомарну модель.

Застосування правила (5.4) на основі засобів DEVS має на меті одержання у знаменнику виразу послідовності, сформованої із трійок вигляду $(\delta_{ext},\delta_{\text{int}},\lambda)$, у відповідності до послідовності елементів протоколу (2.5), побудованого на основі засобів CSP.

Правилу, поданому виразом (5.4), надамо дуального характеру:

– це і засіб відтворення послідовності подій, якими попередньо оперували при проведенні контролю несуперечності ПАС як досліджуваного показника ФХ – у другому – четвертому розділах; і засіб виокремлення граничних перед- і пост-умови, що фігурують у знаменнику, для проведення шляхом індукції зміщення від поточної страти із порядковим номером $(k-1)$ до наступної страти із порядковим номером k.

При цьому було прийнято наступні припущення:

– повідомлення, що надсилаються / приймаються елементами спільного ієрархічного рівня, є засобами інкапсуляції складових досліджуваного показника НФХ;

– повідомлення, що передаються каналом зв'язку, встановленим між елементами спільного ієрархічного рівня функцією *set* (5.2), не втрачаються у процесі передачі при проведенні дискретно-подійного імітаційного моделювання засобами DEVS;

– повідомлення, що опрацьовується у межах деякого елементу $*M_j^{k-1} \in *MS^{k-1}$ як результат виникнення події надходження зазначеного повідомлення на порт $inp_{*M_j}^{k-1} \in INP^{k-1}$, не втрачається у процесій його опрацювання при здійсненні дискретно-подійного імітаційного моделювання.

Зауваження стосовно прийнятих припущень:

– згідно прийнятих припущень, маємо до застосування детерміновану модель виникнення і опрацювання подій, що відбуваються у процесі імітаційного моделювання на основі засобів DEVS. Такий підхід можна вважати прийнятним, коли у якості досліджуваного показника НФХ фігурують, у тому числі, часові або матеріальні витрати, супутні реалізації ФХ згідно ПАС;

– з іншої точки зору, якщо у якості досліджуваного показника НФХ буде фігурувати, наприклад, інтенсивність відмов, – можна застосувати стохастичну модель виникнення і опрацювання подій, із заданим законом розподілу імовірностей;

– у виразі (5.4) у якості верхніх індексів фігурують умовні позначення саме конструкцій (5.1), а не (5.2), – з метою виокремлення ініціюючих і завершальних трійок подій рівня $(k-1)$, що відтворюються також і на рівні k як значущі пограничні послідовності подій, – у відповідності до послідовності елементів протоколу обчислювального процесу (2.5). У свою чергу, послідовності, подані конструкціями (5.3), які не було віднесено розробником / розробниками ні до складу ініціюючих, ні до складу завершальних на рівні $(k-1)$, на рівні k не враховуються. При цьому кількість таких конструкцій є

показником стрімкості зростання рівня абстрагування елементів рівня k відносно елементів рівня $(k-1)$.

Отже, у межах розділу було викладено розроблену модель як стратифіковану архітектуру, призначену слугувати засобом уможливлення проведення контролю досліджуваного показника НФХ вже на етапі проєктування у складі етапів процесу розроблення ПАС. Зазначений контроль, у свою чергу, призначений до реалізації шляхом проведення імітаційного дискретно-подійного моделювання на основі засобів математичного апарату DEVS – на основі розробленого і викладеного у наступному – шостому – розділі методу контролю.

5.3 Підхід до контролю адекватності моделі

Для контролю адекватності розробленої моделі як стратифікованої архітектури застосовано опосередкований підхід. Для реалізації цього підходу у якості передумови мають бути здійснені наступні кроки, виконувані, у свою чергу, згідно висхідного підходу:

1. У відповідності до представленої моделі, викладеної у попередньому підрозділі, на основі засобів математичного апарату DEVS, охоплених у межах змістових навантажень конструкцій «атомарної» і «складеної» моделей DEVS, на основі засобів мови програмування Java як засобів реалізації початково формуються елементи виокремленого нижнього ієрархічного рівня (5.1).

2. Елементи наступних ієрархічних рівнів поступально формалізуються індуктивним шляхом згідно виразу (5.2): шляхом встановлення зв’язків між елементами попередніх страт як засобів уможливлення реалізації механізму обміну повідомленнями. У свою чергу, програмна їх реалізація також

виконується на основі засобів мови програмування Java, – допоки не буде побудовано результуючу комп'ютерну модель, на основі якої, згідно розробленого методу, викладеного у наступному розділі, шляхом проведення дискретно-подійного імітаційного моделювання проводиться накопичення і контроль досліджуваного показника НФХ при проєктуванні ПАС.

Як узагальнення до окреслених вище кроків, представлена модель як ієрархічна архітектура, поступально сформована на основі виразів (5.1) і (5.2), опрацьовується як метамодель, згідно якої, покроково, індуктивним шляхом, у залежності від кількості формалізованих і проконтрольованих за показником несуперечності поведінок $|B|$ (2.2), формується один або декілька елементів верхньої страти. І вже по відношенню до єдиного / кількох результуючих елементів верхнього ієрархічного рівня проводиться опосередкований контроль адекватності розробленої моделі як ієрархічної архітектури. Для цього залучається розроблений метод, викладений у наступному – шостому – розділі.

Контроль адекватності розробленої моделі виконується наведеним нижче чином.

У відповідності до положень розглянутого у першому розділі стандарту IEEE 1012-2016 (рис. 1.2), згідно яких валідація може бути реалізована і шляхом імітаційного моделювання, і шляхом тестування, дану дуальну концепцію застосуємо у контексті процесу контролю значення досліджуваного показника НФХ при проєктуванні ПАС: дії, виконувані згідно архітектурної складової ПАС, несуперечність якої як досліджуваний показник ФХ вже було підтверджено на основі формального методу перевірки на моделі TLC або на основі розробленого розвитку даного методу, викладеного у четвертому розділі, відтворюються один-до-одного засобами озвучених вище конструкцій «атомарної» і «складеної» моделей DEVS у формі комп'ютерної моделі DEVS, згідно представленої у межах розділу розробленої моделі як стратифікованої

архітектури. За результатами проведення дискретно-подійного імітаційного моделювання, на основі механізму обміну повідомленнями між складовими результуючої складеної моделі DEVS, одержуємо у формі повідомлення результат реалізації ФХ згідно ПАС, при проєктуванні ПАС. Даний результат, у свою чергу, співставляється із відповідним результатом, одержуваним шляхом тестування результуючої програмної реалізації. Факт співпадіння результатів, зафіксованих у формі повідомлення при проєктуванні ПАС, із відповідними результатами тестування розглядатимемо як опосередковане підтвердження адекватності розробленої моделі як стратифікованої архітектури.

Демонстрацію зазначеного підтвердження адекватності викладено у наступному – шостому – розділі, присвяченому представленню і дослідженню розробленого методу контролю показника НФХ, що базується на даній моделі. Для цього у якості форми реалізації досліджуваної ПАС опрацьовано композитний вебсервіс, для якого ФХ реалізовано згідно моделі централізованого координування складових компонентів.

У контексті окресленого вище, серед іншого, постають супутні виконанню кроків ПАС часові витрати, які у межах дисертації опрацьовуються у якості досліджуваного показника НФХ. Варто, однак, зауважити, що аналогічним чином як показник можна опрацьовувати, наприклад, супутні матеріальні витрати.

Стосовно часових витрат доречно відзначити, що результуючі агреговані значення можуть бути одержані, у тому числі, наступними шляхами:

1. Оперуючи оціночними значеннями відповідних складових. Такі складові при цьому подаються у формі повідомлень, поширюваних і модифікованих у процесі імітаційного дискретно-подійного моделювання згідно механізму обміну повідомленнями. При цьому, однак, постає питання достовірності оціночних даних.

2. Оперуючи фактичними значеннями складових, одержуваними у процесі проведення дискретно-подійного імітаційного моделювання, шляхом виконання замірів часових витрат, супутніх здійсненню обчислень у межах кожного із елементів нижнього ієрархічного рівня, побудованих згідно виразу (5.1).

3. Маніпулюючи «фактично одержуваними» значеннями складових. У виокремлене словосполучення при цьому вкладається контекстне навантаження, відмінне від такого для попереднього пункту: фактично виконувані фрагменти обчислень заміщуються

4. Шляхом комбінування вищерозглянутих варіацій.

Отже, у частині накопичення результуючого значення досліджуваного показника НФХ допустимі альтернативні шляхи, перелічені вище. При цьому підтвердження адекватності запропонованої моделі згідно висвітленого підходу виконано із застосуванням відповідного розробленого методу контролю значення показника НФХ, представленого у наступному, шостому, розділі.

ВИСНОВКИ ДО РОЗДІЛУ 5

Таким чином, у розділі викладено розроблену модель – стратифіковану архітектуру, де ієрархічний підхід застосовано у якості інструменту досягнення архітектурної відповідності результуючої складеної імітаційної моделі вихідній ФС, несуперечність якої вже було підтверджено шляхом формальної верифікації – методом перевірки на моделі TLC або згідно розробленого і викладеного у попередньому розділі розвитку зазначеного методу, призначеного до застосування за ітеративного підходу до організації процесу ФВ і викладеного у попередньому розділі.

Механізм спадковості як засіб досягнення зазначеної архітектурної відповідності реалізовано шляхом оперування конструкціями «атомарної» і «складеної» моделей математичного апарату DEVS. Названі конструкції залучено у межах озвученого стратифікованого підходу, із охопленням, у тому числі, засобів формалізованого подання складових досліджуваного показника НФХ. У якості останнього опрацьовано часові витрати, супутні реалізації ФХ згідно ПАС, несуперечність якої вже було підтверджено. Зазначено, що у якості можливої альтернативи можуть фігурувати, у тому числі, оціночні та/або фактичні значення складових супутніх матеріальних витрат.

Розроблену модель реалізовано шляхом виокремлення ієрархічних рівнів – страт: таким чином, що елементи нижнього рівня подано як атомарні моделі DEVS, а елементи наступних рівнів – як складені моделі. При цьому було відзначено, що елементи нижньої страти опрацьовано у якості «формуючих» засобів рівня деталізації результуючої складеної ієрархічної конструкції.

Розроблена модель призначена до застосування у якості складової комплексу засобів, що залучаються згідно розробленого підходу, викладеного у другому розділі.

Модель побудовано як результат виокремлення трьох типів артефактів, і віднесено до складу засобів продукування артефактів результуючого типу. При цьому було визначено наступні типи: первинні артефакти – графічні подання ПАС (у тому числі, у формі блок-схем алгоритмів, UML-діаграм дій); вихідні артефакти – формалізовані подання ПАС (на основі виразних засобів формалізмів PlusCal і TLA+) як засоби уможливлення контролю несуперечності ПАС – досліджуваного показника ФХ; результуючі артефакти (на основі засобів математичного апарату DEVS) як засоби уможливлення контролю показника НФХ при проєктуванні ПАС.

Кроки з виокремлення типів артефактів здійснено як з точки зору слідування положенням концепції багатовимірної верифікації, так і з точки зору уможливлення включення розробленої моделі до складу комплексу засобів, застосовуваних згідно розробленого підходу, викладеного у другому розділі.

Представлена стратифікована модель призначена слугувати засобом уніфікації складених комп’ютерних моделей, побудованих на основі виразних засобів математичного апарату DEVS, що залучаються згідно розробленого підходу у якості засобів контролю досліджуваного показника НФХ при проєктуванні ПАС. У свою чергу, результуючі уніфіковані комп’ютерні моделі, побудовані на основі засобів DEVS у відповідності до розробленої стратифікованої моделі, адресуються у якості артефактів, до яких безпосередньо застосовується розроблений метод контролю значення досліджуваного показника НФХ при проєктуванні ПАС, викладений у наступному – шостому – розділі.

РОЗДІЛ 6
РОЗРОБЛЕННЯ МЕТОДУ КОНТРОЛЮ ПОКАЗНИКА НЕФУНКЦІОНАЛЬНИХ ХАРАКТЕРИСТИК

У розділі викладається розроблений метод здійснення контролю значення досліджуваного показника НФХ, що подається у складі артефактів, якими оперує розробник на етапі проєктування у складі етапів процесу розроблення ПАС системи критичного призначення.

Названий метод призначений слугувати засобом уможливлення здійснення зазначеного контролю шляхом проведення імітаційного дискретно-подійного моделювання на основі комп'ютерних моделей, побудованих у відповідності до стратифікованої моделі подання ПАС, викладеної у попередньому – п'ятому – розділі. За рахунок залучення даної моделі є можливість оперувати у процесі моделювання при проєктуванні ПАС як оціночними, так і фактичними значеннями названого показника НФХ.

Одержання у процесі моделювання результуючого значення показника НФХ здійснюється шляхом агрегування (накопичення) останнього на основі механізму обміну повідомленнями між елементами спільного ієрархічного рівня, у відповідності до архітектурної складової досліджуваної ПАС.

При проведенні дослідження розробленого методу у якості показника НФХ опрацьовано часові витрати, супутні виконанню кроків ПАС для реалізації ФХ. Одержуваний у результаті застосування розробленого методу корисний ефект оцінено на основі показника скорочення супутніх використанню методу часових витрат за рахунок оперування оціночними значеннями складових досліджуваного показника.

6.1 Формулювання вирішуваної задачі та застосовуваний підхід

У розділі вирішується задача розроблення методу контролю значення досліджуваного показника НФХ при проєктуванні ПАС, призначеного слугувати засобом автоматизованої перевірки відповідних артефактів, що, разом із проведеним розвитком методу TLC, викладеним у четвертому розділі, а також із розробленими методами і моделями, представленими у інших попередніх розділах, має формувати комплексний інструментарій забезпечення ФБ при проєктуванні ПАС.

У відповідності до положень багатовимірної верифікації, метод має характеризуватися наступними властивостями: безшовність сполучення із методом контролю ФХ за показником несуперечності ПАС; наявність механізму варіювання рівня деталізації артефактів, що опрацьовуються у якості вихідних конструкцій, по відношенню до яких застосовується розроблений і викладений у межах розділу метод [175].

Поставлена задача вирішується згідно нижченаведеного запропонованого підходу, який позиціонується у якості складової комплексу засобів контролю показників ФХ і НФХ при проєктуванні ПАС. При цьому у якості досліджуваного показника НФХ опрацьовується значення часових витрат, яке накопичується у процесі роботи представленого методу. Згідно методу передбачається проведення імітаційного дискретно-подійного моделювання на основі засобів математичного апарату DEVS.

Застосування методу базується на залученні наступних складових:

– ФС як артефакт типу ВА (рис. 5.1), несуперечність якого вже було підтверджено на основі методу TLC, або на основі розробленого розвитку даного методу, викладеного у четвертому розділі [175], використовується у якості конструкції, що подається на вхід методу;

– задані часові обмеження на виконання кроків ПАС для реалізації ФХ: концептуальна складова даної позиції подібна до такої, що має місце за модульного тестування, виконуваного при реалізації ПАС, за якого перевищення вказаних часових обмежень є підставою вважати тест непройденим.

Запропонований і застосований підхід до реалізації кроків розробленого методу, якому присвячено даний розділ, базується на оперуванні конструкціями «атомарної» (АМ) і «складеної» (СМ) моделей математичного апарату DEVS.

6.2 Викладення застосовуваного підходу

У контексті підходу до застосування розробленого методу оперуватимемо наступними типами артефактів (рис. 5.1):

– у якості артефакту типу ПА – блок-схема алгоритму або UML-діаграма дій як графічне подання ПАС на етапі проєктування;

– у якості артефактів типу ВА – ФС на основі виразних засобів PlusCal і TLA+. Призначення засобів PlusCal – сформувати архітектурну складову для ФС на основі виразних засобів TLA+. Останні, у свою чергу, доповнюють змістову складову ФС, несуперечність якої має бути підтверджена шляхом проведення ФВ на основі методу TLC або на основі розробленого розвитку даного методу, викладеного у четвертому розділі [134];

– наявність артефакту типу ВА на основі виразних засобів TLA+, несуперечність якого вже було підтверджено, опрацьовується у якості передумови до застосування розробленого методу, для якого артефакт типу ВА на основі виразних засобів PlusCal залучається у якості вихідної конструкції (рис. 6.1).

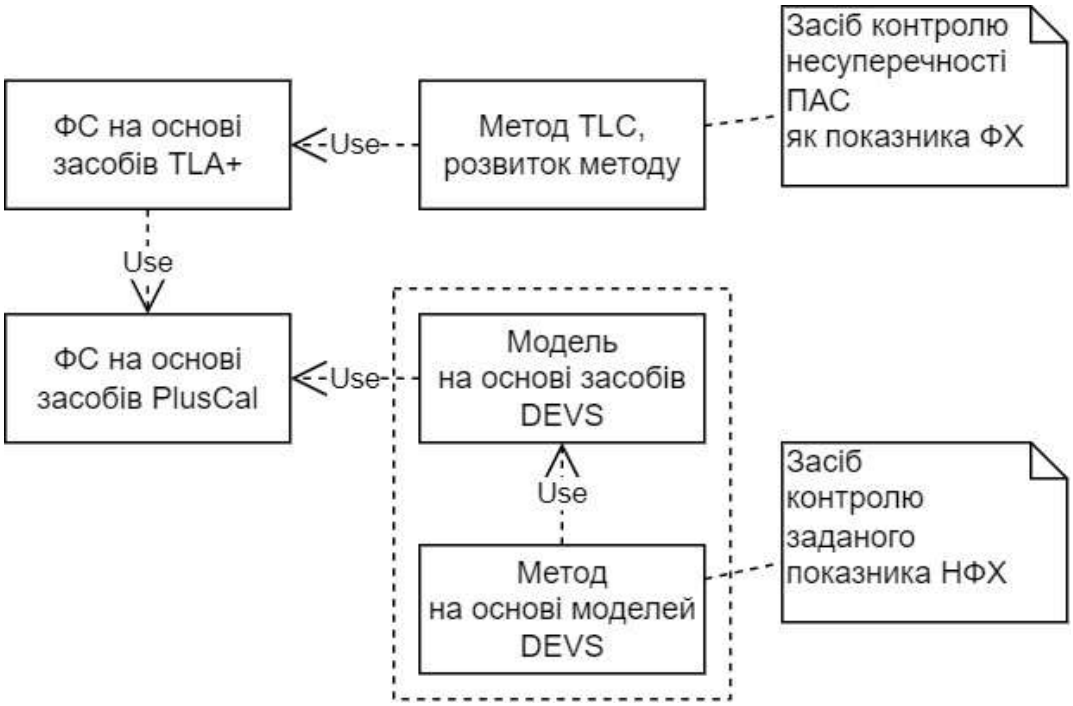

Рисунок 6.1 – Подання підходу до застосування розробленого методу

На рисунку 6.1 пунктиром окреслено науковий здобуток, винесений на захист і викладений у п’ятому і поточному розділах [176, 177].

Розроблений метод полягає у проведенні дискретно-подійного імітаційного моделювання на основі результуючої складеної комп’ютерної моделі, побудованої згідно виразу (5.2), як елементу верхнього ієрархічного рівня, виокремленого згідно розробленої моделі, представленої у п’ятому розділі.

Розроблений метод полягає у виконанні кроків, деталізованих нижче [173, 175–189].

Крок 1. Побудова складеної комп’ютерної моделі. У відповідності до виразу (5.2), згідно підходу, поданого на рисунку 6.1, на основі виразних засобів DEVS, сполучених у межах концепції «складеної» моделі, будується комп’ютерна модель, рівень деталізації якої визначається змістовим навантаженням, покладеним в основу атомарних моделей DEVS (5.1).

Крок 2. Вибір механізму просування модельного часу, що полягає у використанні у якості відліків оціночних та/або фактичних значень. Функція *ta* у складі виразу (5.1) реалізується одним із двох наступних шляхів: як засіб оперування оціночними значеннями складових досліджуваного показника НФХ – часових та/або матеріальних витрат; як засіб оперування фактичними значеннями. Останні, у свою чергу, одержуються, наприклад, шляхом виконання у процесі імітаційного дискретно-подійного моделювання програмної реалізації функції δ_{int} (5.1), і фіксації часових відліків, супутніх такому виконанню.

Зауваження:

– у якості альтернативного показника НФХ, замість оціночних значень часових витрат допускається також залучення оціночних значень матеріальних витрат, супутніх виконанню кроків ПАС.

Крок 3. Проведення імітаційного дискретно-подійного моделювання, у процесі якого результуюче значення досліджуваного показника накопичується шляхом обміну повідомленнями між компонентами складеної моделі (5.2).

Для викладення суті розробленого методу розглянемо фрагмент блок-схеми алгоритму роботи БУК БЦОК КА (рис. 4.1), для якого шляхом ФВ відповідної ФС попередньо вже було підтверджено його несуперечність. У якості досліджуваного показника НФХ охопимо при цьому часові витрати.

Результат виконання кроків 1 і 2 методу представлено на рисунку 6.2.

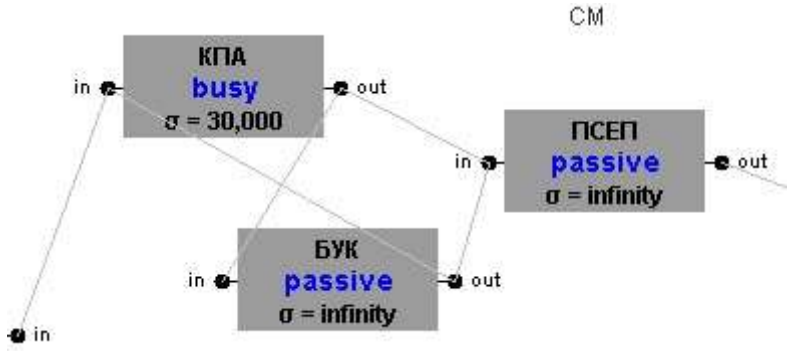

Рисунок 6.2 – Графічне подання результуючої складеної моделі

На рисунку 6.2 представлено архітектурну складову результуючої складеної моделі DEVS для розглянутого випадку, побудовану у відповідності до розробленої моделі, викладеної у п’ятому розділі. Фігурують атомарні моделі наступних компонентів: КПА, БУК, ПСЕП, які у ФС на основі виразних засобів PlusCal і TLA+ було подано відповідними змінними станів СП (рис. 4.2).

Як узагальнення, на рисунку 6.2 зведено результат виконання перших двох кроків розробленого методу. У свою чергу, виконання заключного кроку дозволить отримати результуюче значення досліджуваного показника шляхом проведення дискретно-подійного імітаційного моделювання.

На рисунку 6.2 представлена дворівнева ієрархічна комп’ютерна модель, побудована на основі засобів математичного апарату DEVS, у відповідності до моделі, викладеної у п’ятому розділі. Нижній ієрархічний рівень охоплює три атомарні моделі DEVS ($n = |V| = 3$), сформовані згідно виразу (5.1). Загальна кількість комп’ютерних моделей при цьому становить $(n+1)$, оскільки має місце також і єдиний елемент верхнього ієрархічного рівня, формалізований згідно виразу (5.2).

Змістове навантаження елементів нижнього ієрархічного рівня є наступним: $ST = \{"busy","passive"\}$, де мітка стану $"busy" \in ST$ означає, що відповідна атомарна модель перебуває у стані опрацювання модельних даних; натомість, мітка $"passive" \in ST$ означає, що елемент нижньої страти перебуває у стані очікування модельних даних для їх опрацювання, про це свідчить також і висловлювання $\sigma = "\inf"$. У свою чергу, вирази на кшталт $\sigma = 30{,}000$ є засобами сповіщення розробника, що результат застосування функції λ (5.1) буде одержано на вихідному порті відповідної моделі після вказаної затримки в одиницях модельного часу.

6.3 Дослідження розробленого методу

У якості досліджуваного сценарію предметної області опрацьовано синтетичний випадок проведення розподілених обчислень із залученням конструкцій атомарних і складеного вебсервісів.

Запропонований і застосований підхід до проведення досліджень полягає у наступному:

– за основу береться концептуальна складова в основі стандарту IEEE 1012-2016 (рис. 1.1), згідно якої процес валідації може бути реалізований як шляхом імітаційного моделювання, так і шляхом тестування. У свою чергу, у контексті дисертаційного дослідження, імітаційне моделювання і тестування адресуються у якості альтернативних підходів до здійснення контролю значення заданого показника НФХ – часові витрати, супутні реалізації ПАС. Також прикладом можуть слугувати супутні матеріальні витрати;

– одержання результуючого оціночного значення заданого показника НФХ здійснюється шляхом накопичення (агрегування) значень відповідних складових у процесі імітаційного дискретно-подійного моделювання на основі засобів математичного апарату DEVS, оперуючи при цьому конструкціями атомарних і складеної/складених DEVS-моделей. Назване накопичення уможливлюється у результаті сполучення зазначених конструкцій, із організацією їх взаємодії на основі механізму обміну повідомленнями;

– згідно представленого методу, у якості складових, що внаслідок проведення моделювання формують результуюче оціночне значення заданого показника, можуть залучатись як оціночні, так і фактичні значення. Останні, у свою чергу, можуть бути отримані шляхом застосування штатних засобів мови

програмування Java для вимірювання часових витрат, супутніх реалізації ФХ-складової.

Згідно представленого підходу, ідея в основі проведеного дослідження розробленого методу полягає у наступному:

– названий метод застосовується у двох варіантах реалізації: із залученням реалізації, за якої оперуємо виключно оціночними значеннями складових заданого показника НФХ; із залученням реалізації, де оперуємо фактичними (виміряними) значеннями;

– для обох вищезазначених випадків фіксуються результуючі часові витрати, супутні застосуванню розробленого методу. Для цього залучаються відповідні програмні засоби автоматизації процесу фіксації таких витрат;

– судження стосовно ефективності певної реалізації методу виноситься з позиції співставлення результуючих значень часових витрат для обох реалізацій.

Нижче подається синтетичний сценарій прикладного застосування розробленого методу, за якого досліджувану систему реалізовано у формі композитного вебсервісу. При цьому у якості компонентів фігурують атомарні вебсервіси. Такий крок було здійснено з наступного міркування:

– щоб подавати і результуючу систему, і відповідні компоненти, засобами математичного апарату DEVS у форматі «один-до-одного», оперуючи при цьому конструкціями «атомарної» і «складеної» DEVS-моделей.

Зазначений крок має на меті спростити опрацювання аспекту контролю адекватності результуючої складеної DEVS-моделі на архітектурному рівні.

6.3.1 Дослідження сценарію розподілених обчислень

Розглядається сценарій розподіленого обчислення значення π через арктангенси [117, 141]. Компоненти розподіленої системи при цьому реалізовано у формі атомарних вебсервісів, а результуючу систему на їх основі

– у формі композитного вебсервісу. Архітектуру результуючої системи подано на рис. 6.3.

Атомарні вебсервіси при цьому подано множиною $\{AWS_1,...,AWS_4\}$, де призначення елементів підмножини $\{AWS_1,...,AWS_3\}$ – обчислення значень арктангенсів, а елемент AWS_4, у свою чергу, слугує засобом накопичення результуючого значення π.

На рис. 6.3 для встановлення зв'язків між компонентами системи застосовано саме відношення «агрегування», а не відношення «композиції», – аби наголосити на наступному аспекті:

– компоненти системи можуть бути залучені до складу і інших конфігурацій.

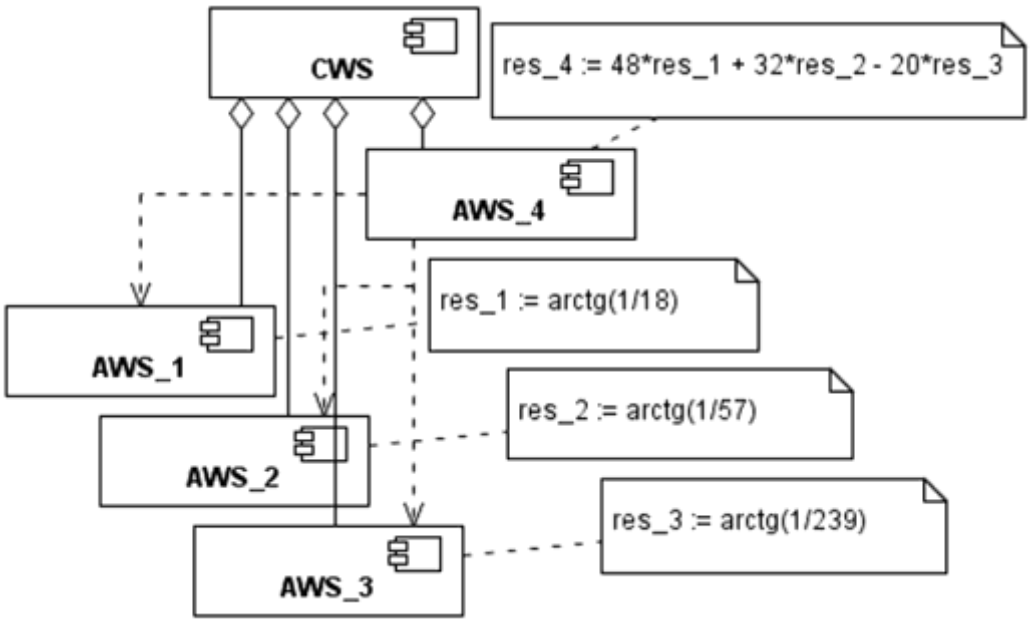

Рисунок 6.3 – Архітектурне подання результуючої системи у формі UML-діаграми компонентів

Результати проведених досліджень реалізацій розробленого методу згідно викладеного вище підходу подано на рис. 6.4, а також зведено у табл. 6.1.

Отримана реалізація композитного вебсервісу уможливила розрахунок шуканого значення. У свою чергу, результуюча складена комп'ютерна модель на основі засобів DEVS також дозволила отримати назване значення: на основі механізму обміну повідомленнями між компонентами при проведенні дискретно-подійного імітаційного моделювання. Це також надає підстави стверджувати стосовно адекватності розробленої стратифікованої моделі, викладеної у п'ятому розділі, у відповідності до якої було побудовано результуючу складену модель композитного вебсервісу на основі засобів DEVS.

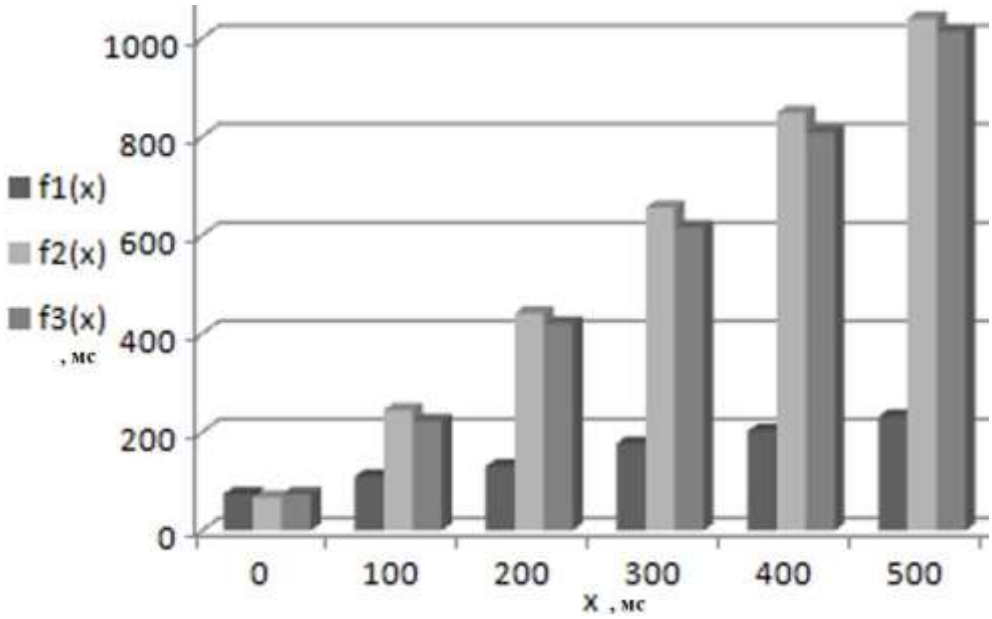

Рисунок 6.4 – Зведений графік оціночних і фактичних значень часових витрат – досліджуваного показника НФХ

На рис. 6.4 по осі Ox подано значення часових витрат – x (у мс), супутніх реалізації ФХ-складової кожним із *AWS* у складі результуючого *CWS* (рис. 6.3). Це значення встановлено однаковим для всіх компонентів, із фіксованим кроком у 100 мс: 0, 100,…, 500 мс.

По осі Oy подано значення функцій від x – $f_1(x),...,f_3(x)$: $f_1(x)$ – функція залежності фактичних часових витрат, супутніх застосуванню реалізації методу,

за якої оперуємо оціночними значеннями складових показника; $f_2(x)$ – функція залежності фактичних результуючих часових витрат для випадку, коли оперуємо фактичними значеннями складових показника; примітка: для останнього випадку фактичні значення складових результуючої затримки було одержано із залученням вбудованих засобів мови програмування Java; $f_3(x)$ – функція залежності результуючого оціночного агрегованого значення показника, отриманого у результаті проведення імітаційного дискретно-подійного моделювання на основі засобів математичного апарату DEVS.

Отримані експериментальні значення функцій $f_1(x),...,f_3(x)$ зведено у табл. 6.1.

Таблиця 6.1 – Зведені результати проведених досліджень реалізацій розробленого методу

№ з/п	x, мс	$f_1(x)$, мс	$f_2(x)$, мс	$f_3(x)$, мс	$f_3(x)/f_1(x)$
1	2	3	4	5	6
1	0,000	74,600	67,823	74,000	0,992
2	100,000	109,700	244,846	222,000	2,024
3	200,000	130,600	441,542	420,000	3,216
4	300,000	177,100	655,630	616,000	3,478
5	400,000	202,400	848,369	808,000	3,992
6	500,000	231,600	1040,265	1013,000	4,374

У табл. 6.1, у стовпці 6, у формі відношення $f_3(x)/f_1(x)$ подано показник корисного ефекту, одержуваного у результаті застосування реалізації розробленого методу, що будується на оперуванні оціночними значеннями, замість реалізації методу, заснованої на оперуванні фактичними значеннями

складових. На охопленому інтервалі значень x цей показник можна оцінити діапазоном відносних значень від близько 0,99 – до близько 4,37.

Для випадку $f_3(x)/f_1(x)=0{,}992$ спостерігаємо ситуацію, за якої оперування фактичними значеннями складових показника є більш пріоритетним – з позиції супутніх часових витрат. У свою чергу, для граничного випадку $f_3(x)/f_1(x)=4{,}374$ маємо протилежну ситуацію – оперування оціночними значеннями супроводжується істотно вищою ефективністю. При цьому, однак, постає питання достовірності одержуваних результуючих оціночних значень. Для стверджувальної відповіді на це питання було проведено перевірку адекватності результуючої складеної DEVS-моделі – на основі статистичних критеріїв t та F – для довірчої імовірності 0,95. Для цього було опрацьовано множини значень функцій $f_2(x)$ і $f_3(x)$.

За результатами аналізу даних табл. 6.1 можна зробити наступні висновки:

– із зростанням фактичних часових витрат корисний ефект від оперування оціночними значеннями також зростає. При цьому постає питання контролю достовірності оціночних значень;

– для одержання фактичних значень заданого показника НФХ, у відповідності до архітектурної складової, поданої на рис. 6.3, було реалізовано відповідний композитний вебсервіс, оперуючи при цьому виразними засобами WS-BPEL (Web Services Business Process Execution Language) [117]. Отриманий результат подано на рис. 6.5.

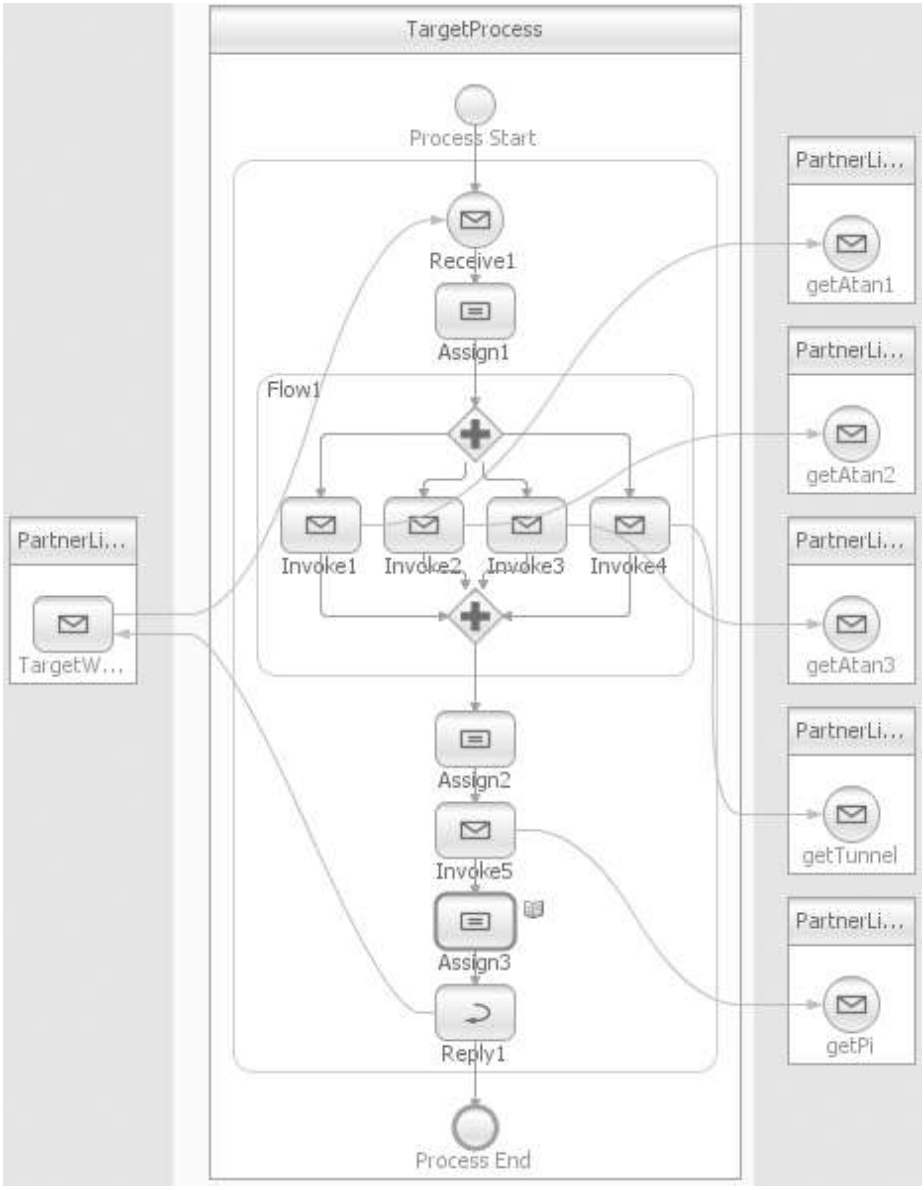

Рисунок 6.5 – Діаграма дій для побудованої реалізації композитного вебсервісу згідно рисунку 6.3

Взаємодія компонентів реалізованої системи здійснюється згідно моделі оркестрування – централізованого керування складовими компонентами у відповідності до діаграми дій, поданої на рисунку 6.5, де діями «Invoke» представлено виклики атомарних вебсервісів.

Практичне значення отриманих при проведенні досліджень результатів висвітлено у наступному підрозділі.

6.4 Відомості стосовно впровадження результатів

У відповідності до запропонованої розширеної постановки задачі ФВ (3.1), одержуваний корисний ефект від прикладного застосування розроблених і винесених на захист методів і супутніх засобів було зафіксовано для наступних нижченаведених випадків:

– для випадку опрацьованих артефактів – блок-схем алгоритмів – з аерокосмічної галузі було підтверджено несуперечність зазначених артефактів. Це дає підстави стверджувати стосовно підвищення рівня довіри розробників до подань прийнятих проєктних рішень. У тому числі, це сприяє уніфікації сприйняття зазначених артефактів учасниками колективу розробників. Підтвердження несуперечності проведено на основі розробленого розвитку методу TLC;

– для випадку опрацьованого артефакту – UML-діаграми дій – з галузі енергетики було виявлено суперечності у діаграмі дій для сценарію оновлення реєстру міжнародних кодів для учасників ринку електричної енергії. Це дозволило на наступному кроці доопрацювати зазначений артефакт – шляхом видалення двох зайвих переходів. У результаті було отримано доопрацьовану UML-діаграму дій, для якої було підтверджено несуперечність на основі розробленого розвитку методу TLC;

– практичне значення від застосування розроблених моделі і методу, викладених у поточному і попередньому розділах, як засобів контролю результуючого значення заданого показника полягає у скороченні часових витрат, супутніх здійсненню названого контролю для випадку оперування оціночними значеннями відповідних складових.

Практичне значення здобутих результатів підтверджено на підставі 6 документів, отриманих, у тому числі, у відповідності до вирішених задач НДР № 0121U110615 «Розроблення методів та засобів верифікації артефактів процесу проєктування систем критичного призначення» (2021−2022 рр.; науковий керівник), профінансованої у межах гранту НАН України дослідницьким лабораторіям/групам молодих вчених НАН України для проведення досліджень за пріоритетними напрямами розвитку науки і техніки (2021−2022 рр.). Було отримано: 1 – акт впровадження у робочий процес; 3 – листи підтримки від організацій та установ; 2 – листи підтвердження впровадження у навчальний процес:

– акт впровадження у робочий процес ТОВ «НВП «ХАРТРОН-ЮКОМ» (аерокосмічна галузь);

– лист підтримки від Державного центру кіберзахисту Державної служби спеціального зв’язку та захисту інформації України (ДЦКЗ Держспецзв’язку);

– лист підтримки від Громадської спілки «Міжнародна рада з великих електроенергетичних систем СІГРЕ в Україні»;

– лист підтримки від Державного підприємства «Державний науково-технічний центр з ядерної та радіаційної безпеки»;

– лист підтвердження впровадження отриманих результатів у навчальний процес Навчально-наукового інституту енергозбереження та енергоменеджменту Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського»;

– лист підтвердження впровадження отриманих результатів у навчальний процес факультету Інформаційних технологій Національного університету біоресурсів і природокористування України (НУБіП України).

Копії зазначених вище документів зведено у додатку З.

ВИСНОВКИ ДО РОЗДІЛУ 6

Таким чином, у розділі викладено запропонований метод контролю значення заданого показника НФХ, що фігурує у розроблюваній ПАС СКП. Представлений метод призначений до застосування вже на етапі проєктування процесу розроблення ПАС СКП – у якості складової розробленого комплексного підходу, викладеного у першому розділі.

При проведенні експериментального дослідження розробленого методу у якості зазначеного НФХ-показника залучено часові витрати – часові обмеження, що накладаються на виконання дій, зазначених у ПАС.

Розроблений метод реалізовано у два етапи:

1. Синтез набору DEVS-специфікації на основі вихідного PlusCal-подання – згідно розробленої і викладеної у попередньому розділі стратифікованої моделі. Озвучений процес синтезу реалізовано у висхідному порядку: спочатку одержуємо атомарні DEVS-моделі – елементи нижнього ієрархічного рівня; потім, оперуючи математичним апаратом в основі концепції складеної DEVS-моделі, формуємо елементи наступних рівнів ієрархічної конструкції – допоки не буде одержано один або декілька результуючих елементів верхнього ієрархічного рівня, у якому відтворено архітектурну складову (структуру та зв’язки) вихідного PlusCal-подання.

2. Проведення дискретно-подійного імітаційного моделювання – на основі результуючого артефакту – елементу верхньої страти згаданої вище ієрархічної конструкції. При цьому взаємодія компонентів у складі результуючої архітектури – складеної DEVS-моделі – побудовано згідно моделі обміну повідомленнями. У свою чергу, однією з відмінних рис розробленого методу є можливість оперування НФХ-складовою як у режимі один-до-одного, так і на

основі оціночних значень. У першому випадку фрагменти програмної компоненти ПАС подано у складі атомарних DEVS-моделей – елементів нижньої страти – у режимі один-до-одного. У другому випадку – до елементів названої страти заносяться оціночні значення. При цьому було експериментально показано, що оперування оціночними значеннями характеризується суттєвою позитивною рисою – істотно знижуються часові витрати, супутні прикладному застосуванню розробленого методу.

Було показано, що прикладне застосування розробленого методу характеризується наступними відмінними рисами:

1. Властивість спадковості. У результуючій складеній DEVS-моделі – елементі верхнього ієрархічного рівня – відтворюється архітектурна складова формальної специфікації на основі виразних засобів TLA+, несуперечність якої вже було підтверджено методом перевірки на моделі TLC або розробленим і викладеним у четвертому розділі розвитком названого методу. Це дає підстави розглядати результуючу складену DEVS-модель як похідну конструкцію від вже верифікованої специфікації TLA+. Принципова відмінність такої конструкції від специфікації TLA+, у свою чергу, полягає у можливості подання НФХ-складової. При проведенні експериментальних досліджень розробленого методу у якості названої складової залучено часові витрати на реалізацію ПАС.

2. Модульність. Компоненти результуючої складеної DEVS-моделі, розміщені на нижній (нижніх) стратах ієрархічної архітектури, можна досліджувати як безпосередньо (відокремлено) – шляхом дискретно-подійного імітаційного моделювання, так і формувати на їх основі альтернативні конструкції – шляхом встановлення зав’язків між компонентами.

3. Гнучкість варіювання рівня деталізації результуючої складеної DEVS-моделі. Досягається як шляхом додавання / видалення ієрархічних рівнів, так і шляхом підбору рівня деталізації елементів нижньої страти.

Варто зауважити, що результати проведеного експериментального дослідження розробленого методу – на прикладі композитного вебсервісу – доречно розглядати у якості демонстрації корисного ефекту, одержуваного у результаті оперування не фактичними значеннями складових результуючого значення заданого показника НФХ (часових витрат у нашому випадку), а оціночними значеннями. Проявом названого ефекту стало зниження фактичних часових витрат на одержання результуючого значення заданого показника НФХ. На залученому діапазоні вихідних даних кількісні показники одержуваного корисного ефекту склали від близько 0,99 до близько 4,37 – скорочення часових витрат на одержання шуканого результуючого значення показника НФХ при проєктуванні ПАС СКП. При цьому контроль значення заданого показника ПАС СКП полягає у порівнянні одержуваного результуючого оціночного / фактичного значення заданого показника ПАС СКП із заданим граничним значенням цього показника.

У розділі також зведено інформацію стосовно практичного значення отриманих результатів, результатів впровадження, із зазначенням копій відповідних документальних підтверджень, викладених у додатку З, серед яких – акт впровадження, листи підтвердження впровадження, листи підтримки від організацій та установ.

ВИСНОВКИ

У дисертаційній роботі вирішено важливу науково-технічну проблему забезпечення контролю артефактів процесу проєктування ПАС систем критичного призначення стосовно несуперечності артефактів. Проблему вирішено за рахунок розроблення, дослідження і застосування формальних методів, моделей, розвитку методу TLC, а також підходу до їх комплексного використання.

Отримано наступні основні результати:

1. На основі аналізу розроблено підхід, де вперше, у формі комплексного рішення, сполучено засоби контролю показників і функціональних, і нефункціональних характеристик досліджуваних артефактів процесу проєктування ПАС. У якості названих артефактів опрацьовано, у тому числі, блок-схеми алгоритмів, UML-діаграми дій, станів, похідні формалізовані подання на основі виразних засобів числення послідовних процесів, що взаємодіють Ч. Гоара, алгоритмічної мови PlusCal, формалізму TLA+, математичного апарату DEVS. У якості допоміжних засобів створено і залучено засоби автоматизації процесів одержання похідних формалізованих подань і опрацювання результатів досліджень.

2. Розроблено модель подання ПАС у вигляді формальної специфікації, де вперше застосовано правило композиції Ч. Гоара у якості засобу скорочення кількості рядків коду специфікації. Названу модель залучено у якості засобу уніфікації формальних специфікацій як форм подання ПАС. Це дозволило автоматизувати процес постачання формальних специфікацій для здійснення на їх основі формальної верифікації методом перевірки на моделі TLC, а також на основі розробленого розвитку цього методу, за показником несуперечності ПАС.

Розроблену модель побудовано із залученням виразних засобів числення послідовних процесів, що взаємодіють Ч. Гоара, алгоритмічної мови PlusCal і формалізму TLA+ темпоральної логіки дій TLA Л. Лемпорта.

Отримані результати проведених досліджень розробленої моделі дали підстави стверджувати, що застосування правила композиції Ч. Гоара дозволило скоротити кількість рядків коду результуючих формальних специфікацій на частку від близько 18 % – до близько 33 % (для найліпшого випадку – послідовний сценарій: від 2 – до 256 змінних станів); від близько 22 % – до близько 0 % (для найгіршого випадку – із поданням паралелізму згідно моделі чергування: від 4 – до 16 змінних станів). Зазначені випадки опрацьовано як граничні. У якості додаткового показника просторових витрат на подання ПАС у формалізованому вигляді розглянуто також розмір відповідного файлу-артефакту. Наприклад, для граничного випадку із поданням паралелізму згідно моделі чергування, для 8 змінних станів, він склав близько 7 КБ, а для 16 змінних – вже близько 210 МБ. При цьому для випадку 16 змінних було зафіксовано факт нестачі наявної оперативної пам’яті обчислювальної системи при проведенні формальної верифікації.

3. Розроблено метод синтезу формальних специфікацій на основі графічних подань ПАС, де вперше комплексно охоплено і аналітичний рівень опрацювання результуючих формалізованих подань, і рівень реалізації. Метод побудовано на основі розробленої моделі подання ПАС. Такий крок, на відміну від альтернативних рішень, забезпечує прозорий механізм опрацювання складових результуючої формальної специфікації та зв’язків між ними на рівнях аналітичному і програмної реалізації. Для цього було залучено математичний апарат темпоральної логіки дій TLA, відповідний формалізм TLA+ та алгоритмічну мову PlusCal. Застосування виразних засобів PlusCal опрацьовано як допоміжний крок, що полягає у попередньому формуванні архітектурної складової результуючої формальної специфікації. Останню, у свою чергу,

сформовано на основі виразних засобів TLA+, що уможливило проведення формальної верифікації методом перевірки на моделі TLC або на основі розробленого розвитку даного методу в автоматизованому режимі.

4. Вперше розроблено метод контролю відповідності результуючих формальних специфікацій, одержуваних на основі зазначеного вище запропонованого методу синтезу, первинним артефактам – графічним поданням ПАС. Метод призначений слугувати допоміжним засобом, застосовуваним на заключному кроці методу синтезу. Метод контролю базується на співставленні показників архітектурних складових системи переходів для подань ПАС на рівнях аналітичному і реалізації. Такими показниками є глибина обходу простору станів системи переходів, що будується у процесі формальної верифікації, а також загальна кількість станів системи переходів. Отримані результати застосування розробленого методу контролю підтвердили відповідність результуючих формальних специфікацій первинним артефактам за названими показниками. Окрім зазначеного, розроблений метод є також засобом контролю достовірності результатів формальної верифікації методом перевірки на моделі у частині допустимості поширення зроблених висновків за результатами також і на первинні артефакти.

5. Набув подальшого розвитку поширений метод формальної верифікації TLC. Розвиток проведено у частині підвищення ефективності роботи методу за ітеративного підходу до організації процесу формальної верифікації специфікацій – за показником зниження супутніх результуючих часових витрат: на першій ітерації при обході простору станів системи переходів застосовано метод обходу у ширину теорії графів, що дало змогу визначити глибину обходу; на наступних ітераціях – метод обходу у глибину, що дозволило скоротити результуючі часові витрати, супутні ітеративному процесу формальної верифікації специфікацій.

Експериментальні дослідження базового методу TLC проведено для граничних синтетичних випадків, а також по відношенню до предметно-орієнтованих артефактів процесу проєктування ПАС систем критичного призначення – для галузі енергетики, аерокосмічної галузі: за напрямами оцінювання обчислювальних і просторових витрат, супутніх залученню альтернативних реалізацій базового методу – на основі методів обходу у ширину і у глибину теорії графів.

Дослідження одержуваного корисного ефекту від проведеного розвитку базового методу TLC здійснено у залежності від кількості виконуваних ітерацій для граничних значень кількості змінних станів стосовно предметно-орієнтованого сценарію аерокосмічної галузі. Зазначений ефект склав від близько 15 % – до близько 110 %.

Також проведено оцінювання одержуваного корисного ефекту від введення мультипоточності до складу програмних реалізацій базового методу TLC. Він склав від близько 26 % – до близько 123 %, у залежності від кількості залучених програмних потоків, кількості змінних станів і застосованого методу обходу простору станів системи переходів.

6. Розроблено модель – стратифіковану архітектуру, де ієрархічний підхід вперше застосовано у якості засобу досягнення архітектурної відповідності результуючої складеної комп’ютерної моделі вихідній формальній специфікації, несуперечність якої вже було підтверджено шляхом формальної верифікації методом перевірки на моделі. Ієрархічний підхід реалізовано на основі засобів математичного апарату DEVS: шляхом оперування конструкціями «атомарної» і «складеної» моделей DEVS. Дослідження розробленої моделі проведено шляхом опрацювання у якості показника нефункціональних характеристик часових витрат, супутніх реалізації кроків ПАС. Адекватність розробленої моделі підтверджено шляхом проведення дискретно-подійного імітаційного

моделювання, і співставлення отриманих результатів, у тому числі – результуючих агрегованих значень показника, із результатами тестування.

7. Розроблено метод контролю значення досліджуваного показника нефункціональних характеристик, де вперше враховано можливість оперувати і оціночними, і фактичними значеннями складових названого показника, – вже на етапі проєктування ПАС. Метод реалізовано у відповідності до зазначеної вище моделі: шляхом проведення дискретно-подійного імітаційного моделювання на основі засобів математичного апарату DEVS. Накопичення значення показника у процесі моделювання реалізовано на основі механізму обміну повідомленнями між компонентами результуючої складеної ієрархічної комп'ютерної моделі. У якості досліджуваного показника нефункціональних характеристик опрацьовано часові витрати, супутні реалізації кроків ПАС. Отримані результуючі накопичені значення показника співставлено із відповідними значеннями, одержуваними шляхом тестування. Встановлено, що оперування оціночними значеннями складових показника, замість фактичних значень, у залежності від специфіки обчислювальних дій, виконуваних згідно ПАС, дозволяє істотно скоротити часові витрати, супутні реалізації процесу контролю значення показника.

8. Практичне значення отриманих результатів проведених дисертаційних досліджень було підтверджено на основі наступних документів: акту впровадження, листів підтвердження впровадження, листів підтримки від організацій та установ.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Харченко В. С. Аналіз проблем IT-інженерії безпеки: проект TEMPUS-SAFEGUARD. *Радіоелектронні і комп'ютерні системи*, 2010. № 7 (48). С. 297–300.

2. Конорев Б. М., Манжос Ю. С., Харченко В. С., Алексеев Ю. Г., Сергиенко В. В., Чертков Г. Н. *Инвариантно-ориентированная оценка качества программного обеспечения космических систем: монография* / под ред. Б. М. Конорева, В. С. Харченко. Харьков : Государственный центр регулирования качества поставок и услуг, Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ», 2009. 224 с. ISBN: 978-966-662-193-4

3. Омельчук Л. Л. *Формальні методи специфікації програм*: навч. посібник. К. : УкрІНТЕІ, 2010. 78 с.

4. Yefymenko N., Kudermetov R. Quaternion models of a rigid body rotation motion and their application for spacecraft attitude control. *Acta Astronautica*, 2022. Vol. 194. P. 76–82. DOI: https://doi.org/10.1016/j.actaastro.2022.01.029

5. Петрик В. Л. Экспертиза программного обеспечения информационно-управляющих систем с использованием дескрипторного семантического пространства. *Радіоелектронні і комп'ютерні системи*, 2007. № 2 (21). С. 29–35.

6. ДСТУ EN 61508-3:2019. *Функційна безпечність електричних, електронних, програмованих електронних систем, пов'язаних із безпекою. Частина 1. Загальні вимоги* (EN 61508-1:2010, IDT; IEC 61508-1:2010, IDT). [Чинний від 2019-09-01]. URL: https://zakon.rada.gov.ua/rada/show/v0249774-19#Text. (Accessed: 19.03.2022).

7. ISO 26262:2018. Road vehicles. Functional safety. Part 1: Vocabulary. [Published: December 2018]. URL: https://www.iso.org/standard/68383.html (Accessed: 05.10.2020).

8. Про затвердження Вимог з ядерної та радіаційної безпеки до інформаційних та керуючих систем, важливих для безпеки атомних станцій: наказ Державної інспекції ядерного регулювання від 22.07.2015 № 140, із змінами, внесеними згідно з Наказом Державної інспекції ядерного регулювання № 508 від 25.11.2019. URL: https://zakon.rada.gov.ua/laws/term/34229 (дата звернення: 26.03.2020).

9. Railway applications - Communication, signalling and processing systems - Software for railway control and protection systems (CENELEC - EN 50128), by European Committee for Electrotechnical Standardization (CENELEC), June 2020. URL: https://standards.globalspec.com/std/14317747/EN%2050128 (дата звернення: 12.04.2021).

10. Youn W. K., Hong S. B., Oh K. R., Ahn O. S. Software certification of safety-critical avionic systems: DO-178C and its impacts. *IEEE Aerospace and Electronic Systems Magazine*, 2015. Vol. 30, No. 4. P. 4–13. DOI: https://doi.org/10.1109/MAES.2014.140109

11. Харченко В. С., Скляр В. В., Конорев Б. М., Алексеев Ю. Г., Чертков Г. Н., Засуха С. А., Семенов Л. П. Оценка и обеспечение качества программных средств / под ред. Б. М. Конорева, В. С. Харченко. Харьков : Государственный центр регулирования качества поставок и услуг, Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ», 2007. 244 с.

12. Летичевський О. О. Символьні методи у верифікації та тестуванні високонадійних систем. *Радіоелектронні і комп'ютерні системи*, 2016. № 5(79). С. 78–83. URL: http://nbuv.gov.ua/UJRN/recs_2016_5_14

13. Broy M. A logical approach to systems engineering artifacts and traceability: from requirements to functional and architectural views. *Engineering dependable software systems* : NATO Science for Peace and Security Series – D: Information and Communication Security / eds. M. Broy, D. Peled, G. Kalus. Amsterdam : IOS Press, 2013. Vol. 34. P. 1–48. DOI: https://doi.org/10.3233/978-1-61499-207-3-1

14. Manna Z., Pnueli A. *Temporal verification of reactive systems: safety*. Springer-Verlag : Berlin, Heidelberg, 1995. 530 p.

15. Sharvia S., Papadopoulos Y. Integrating model checking with HiP-HOPS in model-based safety analysis. *Reliability engineering & system safety*, 2015. Vol. 135. P. 64–80.

16. Reinertsen D. G. *The principles of product development flow: second generation lean product development* : 1st ed. Redondo Beach, CA : Celeritas Publishing, 2009. 294 p.

17. Amilon J., Lidström C., Gurov D. Deductive verification based abstraction for software model checking. In: T. Margaria, B. Steffen (eds). Leveraging Applications of Formal Methods, Verification and Validation. Verification Principles. *ISoLA 2022. Lecture Notes in Computer Science*, 2022. Vol. 13701. Springer, Cham. P. 7–28. DOI: https://doi.org/10.1007/978-3-031-19849-6_2

18. Clarke E. M., Grumberg O., Kroening D., Peled D., Veith H. *Model checking: 2nd ed*. Massachusetts: The MIT Press, 2018.

19. Clarke E. M. Model checking, In: S. Ramesh, G. Sivakumar (eds). Foundations of Software Technology and Theoretical Computer Science. FSTTCS 1997. *Lecture Notes in Computer Science*, Vol. 1346. Springer, Berlin, Heidelberg. DOI: https://doi.org/10.1007/BFb0058022

20. Resch S., Paulitsch M. Using TLA+ in the Development of a Safety-Critical Fault-Tolerant Middleware. *Software Reliability Engineering Workshops* :

Proc. 2017 IEEE International Symposium (Toulouse, France, 23–26 October 2017). P. 146–152. DOI: https://doi.org/10.1109/ISSREW.2017.43

21. Pakonen A. Model-checking I&C logics – insights from over a decade of projects in Finland. In *12th Nuclear Plant Instrumentation, Control and Human-Machine Interface Technologies, NPIC&HMIT 2021. American Nuclear Society (ANS)*, 2021. P. 792–801 DOI: https://doi.org/10.13182/T124-34322

22. Shkarupylo V. V., Tomičić I., Kasian K. M., Alsayaydeh J. A. J. An Approach to increase the Effectiveness of TLC Verification with Respect to the Concurrent Structure of TLA+ Specification. *International Journal of Software Engineering and Computer Systems*, 2018. Vol. 4, No. 1. P. 48–60. ISSN: 2289-8522. DOI: https://doi.org/10.15282/ijsecs.4.1.2018.4.0037 (**Directory of Open Access Journal, Google Scholar**)

23. Das M., Mohan B. R., Guddeti R. M. R. Formal specification and verification of drone system using TLA+: a case study. *2022 IEEE/ACIS 23rd International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, Taichung, Taiwan, 07–09 December 2022. DOI: https://doi.org/10.1109/SNPD54884.2022.10051801

24. ДСТУ ISO 9000:2015. Системи управління якістю. Основні положення та словник термінів. [Чинний від 01.07.2016]. Вид. офіц. Київ : ДП «УкрНДНЦ», 2016. 45 с. URL: https://dnaop.com/html/62656/doc-%D0%94%D0%A1%D0%A2%D0%A3_ISO_9000_2015 (дата звернення: 18.11.2023).

25. IEEE 1012-2016. IEEE Standard for system, software, and hardware verification and validation. [Approved: 28 September 2017]. DOI: https://doi.org/10.1109/IEEESTD.2017.8055462. (дата звернення: 18.11.2023).

26. Leucker M., Schallhart C. A brief account of runtime verification. *The journal of logic and algebraic programming*. 2009. Vol. 78, No. 5. P. 293–303.

27. Falcone Y., Havelund K., Reger G. *A Tutorial on Runtime Verification*. NATO Science for Peace and Security Series – D: Information and Communication Security. Vol. 34: Engineering Dependable Software Systems. IOS Press, 2013. P. 141–175. DOI: https://www.doi.org/10.3233/978-1-61499-207-3-141

28. Marmsoler D., Petrovska A. Detecting architectural erosion using runtime verification. *Interaction and Concurrency Experience* : 12th international scientific meeting (Lyngby, Denmark, June 20–21, 2019). EPTCS 304, 2019. P. 97–114.

29. Marmsoler D. Hierarchical Specification and Verification of Architectural Design Patterns. *Fundamental Approaches to Software Engineering* : 21st International Conference (Thessaloniki, Greece, April 14–20, 2018). Thessaloniki, 2018. P. 149–168. DOI: https://doi.org/10.1007/978-3-319-89363-1_9

30. Garlan D., Allen R., Ockerbloom J. M. Architectural mismatch: why reuse is still so hard. *IEEE Software*, 2009. Vol. 26, No. 4. P. 66–69.

31. Myers G. J. Software reliability: principles and practices. NY : Wiley, 1976. 360 p.

32. Rozier K. Y. Linear Temporal Logic symbolic model checking. *Computer Science Review*, 2011. Vol. 5, No. 2. P. 163–203.

33. Curcin V., Ghanem M. M., Guo Y. Analysing scientific workflows with Computational Tree Logic. *Cluster Computing*, 2009. Nol. 12, No. 4. P. 399–419. DOI: 10.1007/s10586-009-0099-6

34. Kwiatkowska M., Norman G., Parker D. PRISM 4.0: verification of probabilistic real-time systems. *Computer Aided Verification* : International Conference. Part of LNCS (Snowbird, UT, USA, 14-20 July 2011), 2011. Vol. 6806. P. 585–591.

35. Kapus T. Using PRISM model checker as a validation tool for an analytical model of IEEE 802.15.4 networks. *Simulation Modelling Practice and Theory*, 2017. Vol. 77. P. 367–378. DOI: https://doi.org/10.1016/j.simpat.2017.08.002

36. Dimitrova R., Fioriti L. M. F., Hermanns H., Majumdar R. Probabilistic CTL∗: The Deductive Way. *Tools and Algorithms for the Construction and Analysis of Systems* : 22nd International Conference. Part of LNCS (Eindhoven, The Netherlands, April 2-8, 2016). 2016. Vol. 9636. P. 280–296. DOI: https://doi.org/10.1007/978-3-662-49674-9_16

37. Abraham E., Jansen N., Wimmer R., Katoen J.-P., Becker B. DTMC model checking by SCC reduction. *Quantitative Evaluation of Systems* : 2010 Seventh International Conference, IEEE (Williamsburg, VA, USA, 15-18 Sept. 2010). 2010. P. 37–46. DOI: https://doi.org/10.1109/QEST.2010.13

38. Dehnert C., Junges S., Katoen J.-P., Volk M. A Storm is coming: a modern probabilistic model checker. *Computer Aided Verification, CAV 2017* : 29th International Conference. Part of LNCS (Heidelberg, Germany, July 24–28, 2017). 2017. Vol. 10427, Part II. P. 592–600.

39. Fabarisov T., Yusupova N., Ding K., Morozov A., Janschek K. The efficiency comparison of the prism and storm probabilistic model checkers for error propagation analysis tasks. *Industry 4.0*, 2018. Vol. 3, No. 5. P. 229–231.

40. Agha G., Palmskog K. A survey of statistical model checking. *ACM Transactions on Modeling and Computer Simulation*, 2018. Vol. 28, No. 1. P. 1–39. DOI: https://doi.org/10.1145/3158668

41. Kraibi K., Ben Ayed R., Rehm J., Collart-Dutilleul S., Bon P., Petit D. Event-B Decomposition Analysis for Systems Behavior Modeling. *Software Technologies (ICSOFT 2019)* : 14th International Conference (Prague, Czech Republic, July 26–28, 2019). Vol. 1. P. 278–286. DOI: https://doi.org/10.5220/0007929602780286

42. Butler M., Körner P., Krings S., Lecomte T., Leuschel M., Mejia L.-F., Voisin L. The first twenty-five years of industrial use of the B-method. *Formal Methods for Industrial Critical Systems, FMICS 2020* : 25th Int. Conf. / eds. M. ter Beek, D. Ničković (Vienna, Austria, September 2–3, 2020). 2020. Lecture Notes in

Computer Science, Vol. 12327. Springer, Cham. P. 189–209. DOI: https://doi.org/10.1007/978-3-030-58298-2_8

43. Biere A., Cimatti A., Clarke E.M., Strichman O., Zhu Y. Bounded model checking. *Advances in computers*, 2003. Vol. 58, No. 11. P. 117–148.

44. Albarghouthi A., Dillig I., Gurfinkel A. Maximal specification synthesis. *ACM SIGPLAN Notices*, 2016. Vol. 51, No. 1. P. 789–801. DOI: https://doi.org/10.1145/2914770.2837628

45. Deng X., Dwyer M.B., Hatcliff J., Mizuno M. Invariant-based specification, synthesis, and verification of synchronization in concurrent programs. *Software Engineering, ICSE '02* : 24th International Conference (Orlando, Florida, May, 2002). 2002. New York, NY, United States : Association for Computing Machinery. P. 442–452. DOI: https://doi.org/10.1145/581339.581394

46. Cook S.A. The complexity of theorem-proving procedures. Theory of computing, STOC '71 : Proceedings of the third annual ACM symposium, May 1971. P. 151–158. DOI: https://doi.org/10.1145/800157.805047

47. Prasad M., Biere A., Gupta A. A survey of recent advances in SAT-based formal verification. *International Journal on Software Tools for Technology Transfer*, 2005. Vol. 7. P. 156–173. DOI: https://doi.org/10.1007/s10009-004-0183-4

48. Bjorner N. Z3 and SMT in industrial R&D. *Formal Methods, FM 2018*, as Part of the Federated Logic Conference, FloC 2018 : 22nd International Symposium (Oxford, UK, July 15–17, 2018). Part of LNCS. Vol. 10951. Springer, Cham. P. 675–678. DOI: https://doi.org/10.1007/978-3-319-95582-7_44

49. Ishii D., Tomita T., Aoki T., Ngo T.Q., Do T.B.N., Takai H. SMT-Based Model Checking of Industrial Simulink Models / Eds. A. Riesco, M. Zhang. Formal Methods and Software Engineering. ICFEM 2022. *Lecture Notes in Computer Science*, Vol. 13478. Springer, Cham, 2022. DOI: https://doi.org/10.1007/978-3-031-17244-1_10

50. Leinenbach D., Santen T. Verifying the Microsoft Hyper-V Hypervisor with VCC / Eds. A. Cavalcanti, D.R. Dams. *FM 2009: Formal Methods. FM 2009. Lecture Notes in Computer Science*, Vol. 5850. Springer, Berlin, Heidelberg, 2009. P. 806–809. DOI: https://doi.org/10.1007/978-3-642-05089-3_51

51. Hoare C. A. R. Communicating sequential processes. *Communications of the ACM*, 1978. Vol. 21, No. 8. P. 666–677.

52. Gurov D., Lidström C., Nyberg M., Westman J. Deductive functional verification of safety-critical embedded c-code: an experience report / Eds. L. Petrucci, C. Seceleanu, A. Cavalcanti. *Critical Systems: Formal Methods and Automated Verification*. AVoCS 2017, FMICS 2017. Lecture Notes in Computer Science. Vol. 10471. Springer, Cham. DOI: https://doi.org/10.1007/978-3-319-67113-0_1

53. Lee J., Yu G., Bae K. Efficient SMT-based model checking for signal temporal logic. *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, Melbourne, Australia, November 15–19, 2021. P. 343–354. DOI: https://doi.org/10.1109/ASE51524.2021.9678719

54. Biere A., Cimatti A., Clarke E. M., Fujita M., Zhu Y. Symbolic model checking using SAT procedures instead of BDDs. *Design Automation Conference, DAC '99* : 36th annual ACM/IEEE Conference (New Orleans, Louisiana, USA, June, 1999). 1999. P. 317–320. DOI: https://doi.org/10.1145/309847.309942

55. Alur R., Dill D. L. A theory of timed automata. *Theoretical Computer Science*, 1994. Vol. 126, No. 2. P. 183–235. DOI: https://doi.org/10.1016/0304-3975(94)90010-8

56. Kim J. H., Larsen K. G., Nielsen B., Mikucionis M., Olsen P. Formal Analysis and Testing of Real-Time Automotive Systems Using UPPAAL Tools. *Formal Methods for Industrial Critical Systems, FMICS'2015* : International Workshop. Part of LNCS (Oslo, Norway, June 22-23, 2015). 2015. Vol. 9128. P. 47–61.

57. Shkarupylo V. V., Kudermetov R. K., Polska O. V. On the approaches to cyber-physical systems simulation. *Advances in Cyber-Physical Systems (ACPS)*, 2018. Vol. 3, No. 1. P. 51–54. ISSN: 2524-0382 (Print), 2707-0069 (Online). DOI: https://doi.org/10.23939/acps2018.01.051 (**фахове видання**)

58. Shkarupylo V., Kudermetov R. On the aspects of cyber-physical systems modeling with UPPAAL. *Simulation-2018: 6th Int. conference*, September 12–14, 2018: theses. Kyiv: Pukhov Institute for Modelling in Energy Engineering, 2018. P. 267–269.

59. Летичевський О. О. Алгебраїчне моделювання та його застосування. *Вісник Національної академії наук України*, 2021. № 3. С. 59–66. DOI: https://www.doi.org/10.15407/visn2021.03.059

60. Czerwiński W., S. Lasota, Lazić R. S., Leroux J., Mazowiecki F. The reachability problem for Petri nets is not elementary. *STOC 2019*: Proc. 51st Annual ACM SIGACT Symposium on Theory of Computing (Phoenix, AZ, USA June, 2019). New York, NY, United States: Association for Computing Machinery, 2019. P. 24–33. DOI: https://doi.org/10.1145/3313276.3316369

61. Bonneland F., Dyhr J., Jensen P.G., Johannsen M., Srba J. Simplification of CTL formulae for efficient model checking of Petri Nets. Application and Theory of Petri Nets and Concurrency, PETRI NETS 2018 : 39th International Conference (Bratislava, Slovakia, June 24-29, 2018). 2018. Lecture Notes in Computer Science, Vol. 10877. Springer, Cham. P. 143–163. DOI: https://doi.org/10.1007/978-3-319-91268-4_8

62. Hoare C. A. R. An axiomatic basis for computer programming. *Communications of the ACM*, 1969. Vol. 12, No. 10. P. 576–583.

63. Stroud C. E., Wang L.-T., Chang Y.-W. *Introduction to Electronic design automation, in Electronic Design Automation: Synthesis, Verification, and Test* / edited by L.-T. Wang, Y.-W. Chang, K.-T. Cheng. Morgan Kaufmann, 2009. P. 1–38.

64. Jenihhin M., Lai X., Ghasempouri T., Raik J. Towards multidimensional verification: where functional meets non-functional. *NORCHIP and International Symposium of System-on-Chip (SoC)* : 2018 IEEE Nordic Circuits and Systems Conference (Tallinn, Estonia, 30-31 Oct. 2018). P. 1–7. URL: https://arxiv.org/ftp/arxiv/papers/1908/1908.00314.pdf

65. Lang J., Prasetya I.S.W.B. Model checking a C++ software framework: a case study. *Foundations of Software Engineering* : 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium (Tallinn, Estonia, August 26-30, 2019). P. 1026–1036. DOI: https://doi.org/10.1145/3338906.3340453

66. ADAPRO: A general-purpose application framework for multi-threaded applications with support for a configuration file and remote control. Written in C++14. URL: https://gitlab.cern.ch/adapos/adapro (дата звернення: 16.10.2019).

67. Su J., Chen W.-H. Model based fault diagnosis system verification using reachability analysis. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019. Vol. 49, No. 4. P. 742–751. DOI: https://doi.org/10.1109/TSMC.2017.2710132

68. Kokologiannakis M., Sagonas K. Stateless model checking of the Linux kernel's read-copy update (RCU). *International journal on software tools for technology transfer*, 2019. Vol. 21, No. 3. P. 287–306.

69. Musuvathi M., Qadeer S. Fair stateless model checking. *ACM SIGPLAN Notices*, 2008. Vol. 43, No. 6. P. 362–371.

70. Kokologiannakis M., Lahav O., Sagonas K., Vafeiadis V. Effective stateless model checking for C/C++ concurrency. *Proceedings of the ACM on Programming Languages*, January 2018. Vol. 2, No. POPL, Article 17. P. 1–32. DOI: https://doi.org/10.1145/3158105

71. Aichernig B. K., Tappler M. Probabilistic black-box reachability checking (extended version). *Formal Methods in System Design*, 2019. In press. P. 1–33.

72. Yang Y., Zu Q., Ke W., Zhang M., Li X. Real-time system modeling and verification through labeled transition system analyzer. *IEEE Access*, 2019. Vol. 7. P. 26314–26323. DOI: https://doi.org/10.1109/ACCESS.2019.2899761

73. Nardone V., Santone A., Tipaldi M., Liuzza D., Glielmo L. Model checking techniques applied to satellite operational mode management. *IEEE Systems Journal*, 2019. Vol. 13, No. 1. P. 1018–1029. DOI: https://doi.org/10.1109/JSYST.2018.2793665

74. Milner R. A Calculus of Communicating Systems. Berlin Heidelberg : Springer-Verlag, 1980. 174 p. DOI: https://doi.org/10.1007/3-540-10235-3

75. Омельчук Л. Л. Формальні методи специфікації програм: навч. посібник. К. : УкрІНТЕІ, 2010. 78 с.

76. Katz G., Barrett C., Dill D. L., Julian K., Kochenderfer M. J. Reluplex: an efficient SMT solver for verifying deep neural networks. *Computer aided verification, CAV 2017* : 29th International Conference (Heidelberg, Germany, July 24–28, 2017). Proceedings, Part I. P. 97–117.

77. Шкарупило В. В., Євдокимов В. Ф., Душеба В. В. Застосування формальних методів для перевірки систем критичного призначення. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2019. Том 30 (69), № 6. С. 188–193. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI https://doi.org/10.32838/2663-5941/2019.6-1/34 (**фахове видання**)

78. Шкарупило В. В., Євдокимов В. Ф., Душеба В. В. Аспекти застосування методів перевірки на моделі при проектуванні систем критичного призначення. *Безпека енергетики в епоху цифрової трансформації: науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України* : програма та матеріали, 20 грудня 2019 р. Київ : ІПМЕ ім. Г. Є. Пухова НАН України, 2019. С. 94–96. URL:

https://ipme.kiev.ua/wp-content/uploads/2019/12/Програма-КБЕЕЦ-2019.pdf (дата звернення: 06.08.2023)

79. Конорев Б. М., Сергієнко В. В., Туркін І. Б. Доказова незалежна верифікація та прогнозування прихованих дефектів критичного програмного забезпечення на базі диверсного вимірювання інваріантів. *Інженерія програмного забезпечення*, 2011. №1 (5). С. 5–15.

80. ECSS-E-00A. Space engineering. Policy and principles. [Cancelled]. The Netherlands : ESA Publications Division, 1996. 46 p.

81. ECSS-S-ST-00C – ECSS system. Description, implementation and general requirements. [Чинний від 2008-07-31]. AG Noordwijk, The Netherlands : ESA Requirements and Standards Division, 2008. 34 p. URL: https://ecss.nl/standard/ecss-s-st-00c-description-implementation-and-general-requirements-31-july-2008/ (дата звернення: 16.09.2019)

82. Ferrari D. *Computer Systems Performance Evaluation*. Prentice-Hall, 1978. 554 p.

83. Shkarupylo V. V., Tomičić I., Kasian K. M. The investigation of TLC model checker properties. *Journal of Information and Organizational Sciences*, 2016. Vol. 40, No. 1. P. 145–152. ISSN: 1846-9418 (Online), 1846-3312 (Print). DOI: https://doi.org/10.31341/jios.40.1.7 (**Web of Science Core Collection:** https://www.webofscience.com/wos/woscc/full-record/WOS:000409240900008 ; **Scopus, Q4:** https://www.scopus.com/record/display.uri?eid=2-s2.0-84975057117&origin=resultslist)

84. Шкарупило В. Дослідження методу перевірки на моделі TLC. *Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2020 : VIII Міжнародна науково-практична Інтернет-конференція* (м. Київ, Україна, 14–15 травня, 2020). 2020. Київ : НУБіП України. С. 84–86. URL:

http://econference.nubip.edu.ua/index.php/grpi/grpi20/paper/view/2306/317 (дата звернення: 06.08.2023)

85. Larman C. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. 3rd Edition. Pearson, 2004. 736 p.

86. Lamport L. The PlusCal algorithm language / eds. M. Leucker, C. Morgan, *Theoretical Aspects of Computing – ICTAC 2009. ICTAC 2009. Lecture Notes in Computer Science*, Vol. 5684. Springer, Berlin, Heidelberg. P. 36–60. DOI: https://doi.org/10.1007/978-3-642-03466-4_2

87. Shkarupylo V., Kudermetov R., Timenko A., Polska O. On the Aspects of IoT Protocols Specification and Verification. *Problems of Infocommunications. Science and Technology* : 2019 International Scientific-Practical Conference (Kyiv, Ukraine, October 8–11, 2019). P. 93–96. DOI: https://doi.org/10.1109/PICST47496.2019.9061406 (**Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85083637232&origin=resultslist)

88. Lamport L. *Specifying systems: The TLA+ language and tools for hardware and software engineers*. Boston, MA, United States : Addison-Wesley Longman Publishing Co., Inc., 2002. 364 p. URL: https://dl.acm.org/doi/10.5555/579617 (дата звернення: 18.11.2023)

89. Lamport L. Specifying concurrent systems with TLA+. Calculational System Design: Marktoberdorf summer school materials / eds. M. Broy, R. Steinbroggen. Amsterdam : IOS Press, Jan. 2000. Vol. 173 of NATO Science Series, III: Computer and Systems Sciences. P. 183–247.

90. Newcombe C. Why Amazon chose TLA+. *Abstract State Machines, Alloy, B, TLA, VDM, and Z* : ABZ 2014 Int. Conf., Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 2014. Vol. 8477. P. 25–39.

91. Newcombe C., Rath T., Zhang F., Munteanu B., Brooker M., Deardeuff M. How Amazon web services uses formal methods. *Communications of the ACM*, 2015. Vol. 58, No. 4. P. 66–73. DOI: https://doi.org/10.1145/2699417

92. Sputh B., Verhulst E., Mezhuyev V. OpenComRTOS: Formally developed RTOS for Heterogeneous Systems. 2010. DOI: http://doi.org/10.13140/2.1.1488.0006

93. Verhulst E., Boute R. T., Faria J. M. S., Sputh B. H. C., Mezhuyev V. Formal Development of a Network-Centric RTOS: Software Engineering for Reliable Embedded Systems. Springer Publishing Company, Inc., 2011. 236 p.

94. Taibi T., Herranz A., Moreno-Navarro J. J. Stepwise refinement validation of design patterns formalized in TLA+ using the TLC model checker. *Journal of Object Technology*, 2009. Vol. 8, No. 2. P. 137–161. URL: https://www.jot.fm/issues/issue_2009_03/article3.pdf (дата звернення: 06.10.2023)

95. Beers R. Pre-RTL formal verification: an Intel experience. *Design Automation Conference, DAC '08*: Proceedings of the 45th annual Conference (Anaheim, California, June 2008). New York, NY, United States : Association for Computing Machinery, 2008. P. 806–811. DOI: https://doi.org/10.1145/1391469.1391675

96. Kuppe M. A., Lamport L., Ricketts D. The TLA+ Toolbox. *Formal Integrated Development Environment, F-IDE 2019* : 5th Workshop (Porto, Portugal, October 7, 2019). EPTCS 310, 2019. P. 50–62. DOI: http://doi.org/10.4204/EPTCS.310.6

97. Yin J.-Q., Zhu H.-B., Fei Y. Specification and verification of the Zab protocol with TLA+. *Journal of Computer Science and Technology*, 2020. Vol. 35, No. 6. P. 1312–1323. DOI: https://doi.org/10.1007/s11390-020-0538-7

98. Shkarupylo V.V. On the applicability of model checking techniques in the Internet of Things domain. *Тиждень науки-2018*: науково-практ. конф., 16–20 квітня 2018 р.: тези доп. Запоріжжя: ЗНТУ, 2018. С. 967–968. URL:

https://zp.edu.ua/uploads/dept_s&r/2018/conf/1/TN2018.pdf (дата звернення: 06.08.2023)

99. Shkarupylo V., Polska O., Shcherbak N. On the classification of model checking methods for the Internet of Things. *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: IX Міжнародна науково-практична конференція*, 3–5 жовтня 2018 р.: тези доп. Запоріжжя: ЗНТУ, 2018. С. 77–78.

100. Kim Y.-M., Kang M. Formal verification of SDN-based firewalls by using TLA+. *IEEE Access*, 2020. Vol. 8. P. 52100–52112. DOI: https://doi.org/10.1109/ACCESS.2020.2979894

101. Lamport L. Checking a multithreaded algorithm with +CAL. In: S. Dolev, S. (Eds.) Distributed Computing. DISC 2006. *Lecture Notes in Computer Science*, Vol. 4167. Springer, Berlin, Heidelberg. P. 151–163. DOI: https://doi.org/10.1007/11864219_11

102. Zeller P., Bieniusa A., Ferreira C. Teaching practical realistic verification of distributed algorithms in Erlang with TLA+. *Erlang 2020*: Proceedings of the 19th ACM SIGPLAN International Workshop on Erlang (August, 2020). New York: Association for Computing Machinery, 2020. P. 14–23. DOI: https://doi.org/10.1145/3406085.3409009

103. Hackett F., Rowe J., Kuppe M.A. Understanding Inconsistency in Azure Cosmos DB with TLA+. *Proc. 2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, Melbourne, Australia, May 14–20, 2023. DOI: https://doi.org/10.1109/ICSE-SEIP58684.2023.00006

104. Konnov I., Kukovec J., Tran T.-H. TLA+ model checking made symbolic. *Proceedings of the ACM on Programming Languages*, 2019. Vol. 3, No. OOPSLA. P. 1–30. DOI: https://doi.org/10.1145/3360549

105. Konnov I., Kuppe M., Merz S. Specification and Verification with the TLA+ Trifecta: TLC, Apalache, and TLAPS. In: Margaria, T., Steffen, B. (eds) Leveraging Applications of Formal Methods, Verification and Validation. Verification Principles. ISoLA 2022. *Lecture Notes in Computer Science*, 2022. Vol. 13701. Springer, Cham. https://doi.org/10.1007/978-3-031-19849-6_6

106. Heimdahl M.P.E., George D., Weber R. Specification test coverage adequacy criteria = specification test generation inadequacy criteria. *High Assurance Systems Engineering*: Eighth IEEE International Symposium (Tampa, FL, USA, March 25–26, 2004). 2004. P. 178–186, DOI: https://doi.org/10.1109/HASE.2004.1281742

107. Pakonen A., Buzhinsky I. Verification of fault tolerant safety I&C systems using model checking. *Industrial Technology, ICIT 2019*: 2019 IEEE International Conference (Melbourne, Australia, 2019). 2019. P. 969–974. DOI: https://doi.org/10.1109/ICIT.2019.8755014

108. Gay G., Rajan A., Staats M., Whalen M., Heimdahl M.P.E. The effect of program and model structure on the effectiveness of MC/DC test adequacy coverage. *ACM Transactions on Software Engineering and Methodology*, 2016. Vol. 25, No. 3. P. 25:1–25:34. DOI: https://doi.org/10.1145/2934672

109. Cohen E., Dahlweid M., Hillebrand M., Leinenbach D., Moskal M., Santen T., Schulte W. VCC: A Practical System for Verifying Concurrent C / Eds. S. Berghofer, T. Nipkow, C. Urban, M. Wenzel. *Theorem Proving in Higher Order Logics. TPHOLs 2009*. *Lecture Notes in Computer Science*. 2009. Vol. 5674. Berlin, Heidelberg: Springer. P. 23–42. DOI: https://doi.org/10.1007/978-3-642-03359-9_2

110. Ge N., Jenn E., Breton N., Fonteneau Y. Integrated formal verification of safety-critical software. *International Journal on Software Tools for Technology Transfer (STTT)*, 2018. Vol. 20, No. 4. P. 423–440. DOI: https://doi.org/10.1007/s10009-017-0475-0

111. Nouri A., Bensalem S., Bozga M., Delahaye B., Jegourel C., Legay A. Statistical model checking QoS properties of systems with SBIP. *International Journal on Software Tools for Technology Transfer (STTT)*, 2015. Vol. 17, No. 2. P. 171–185. DOI: https://doi.org/10.1007/s10009-014-0313-6

112. Ghezzi C., Sharifloo A. M. Model-based verification of quantitative non-functional properties for software product lines. *Information and Software Technology*, 2013. Vol. 55, No. 3. P. 508–524. DOI: https://doi.org/10.1016/j.infsof.2012.07.017

113. Singh P., Singh L. Verification of safety critical and control systems of nuclear power plants using Petri nets. *Annals of Nuclear Energy*, 2019. Vol. 132. P. 584–592. DOI: https://doi.org/10.1016/j.anucene.2019.06.027

114. Huang L., Kang E.-Y. Formal verification of safety & security related timing constraints for a cooperative automotive system / Eds. R. Hähnle, W. van der Aalst. *Fundamental Approaches to Software Engineering. FASE 2019. Lecture Notes in Computer Science*. 2019. Vol. 11424. Springer, Cham. P. 210–227. DOI: https://doi.org/10.1007/978-3-030-16722-6_12

115. Weissnegger R., Pistauer M., Kreiner C., Römer K., Steger C. A novel design method for automotive safety-critical systems based on UML/MARTE. *Proceedings of the 2015 Forum on specification & Design Languages* (Barcelona Spain, September 14–16, 2015). Belmont, France, 2015. P. 177–184.

116. Weissnegger R., Schuss M., Kreiner C., Pistauer M., Römer K., Steger C. Simulation-based verification of automotive safety-critical systems based on EAST-ADL. *Procedia Computer Science*, 2016. Vol. 83. P. 245–252. DOI: https://doi.org/10.1016/j.procs.2016.04.122

117. Correa T., Becker L. B., Farines J.-M., Bodeveix J.-P., Filali M., Vernadat F. Supporting the design of safety critical systems using AADL. *Proc. 2010 15th IEEE International Conference on Engineering of Complex Computer Systems*

(Oxford, UK, March 22–26, 2010). P. 331–336. DOI: https://doi.org/10.1109/ICECCS.2010.56

118. Van Tendeloo Y., Vangheluwe H. An evaluation of DEVS simulation tools, *SIMULATION*, 2017. Vol. 93, No. 2. P. 103–121. DOI: https://doi.org/10.1177/0037549716678330

119. Дімітрієва Д.О., Шкарупило В.В. Огляд інструментів використання формальних методів та засобів при проєктуванні систем критичного призначення. *Інформаційні технології: економіка, техніка, освіта '2021*: Збірник матеріалів XI Міжнародної науково-практичної конференції молодих вчених, 11–12 листопада 2021 року, НУБіП України, Київ. С. 164–165. URL: https://drive.google.com/file/d/10iRiRUwpXqTY510LzL1j0BeDt-Kpx4Ab/view?usp=sharing (дата звернення: 06.08.2023)

120. Shkarupylo V. A Simulation-driven Approach for Composite Web Services Validation. *Proc. 27th Int. Central European Conference on Information and Intelligent Systems, CECIIS 2016* (Varazdin, Croatia, September 21–23, 2016). P. 227–231. URL: https://www.webofscience.com/wos/woscc/full-record/WOS:000595003500030 (**Web of Science Core Collection**)

121. Shkarupylo V. A Technique of DEVS-Driven Validation. *Proc. XIIIth Int. Conf. on Modern Problems of Radio Engineering, Telecommunications, and Computer Science, TCSET'2016* (Lviv-Slavske, Ukraine, February 23–26, 2016). P. 495–497. DOI: 10.1109/TCSET.2016.7452097 (**Web of Science Core Collection:** https://www.webofscience.com/wos/woscc/full-record/WOS:000381804300127 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-84969263650&origin=resultslist)

122. Шкарупило В.В., Скрупский С.Ю., Кудерметов Р.К. DEVS-модель как средство валидации композитных веб-сервисов распределенной системы. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво*, 2011. № 7. С. 61–67.

123. Naumchev A., Meyer B. Seamless requirements. *Computer Languages, Systems & Structures*, 2017. Vol. 49. P. 119–132. DOI: https://doi.org/10.1016/j.cl.2017.04.001

124. Куликовська Н.А., Руденко В.В., Тіменко А.В., Шкарупило В.В. Дослідження часу збирання додатків, побудованих на основі сучасних стратегій розроблення. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2023. Том 34 (73), № 4. С. 65–70. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32782/2663-5941/2023.4/11 (**фахове видання**)

125. Шкарупило В.В., Кудерметов Р.К., Польська О.В., Тіменко А.В. Щодо доцільності перевірки протоколів взаємодії компонентів систем інтернету речей. *Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2019*: матеріали VII Міжнародної науково-практичної конференції, 15–16 травня 2019 р. Київ: НУБіП України, 2019. С. 59–61. URL: https://lib.lntu.edu.ua/sites/default/files/2021-03/Київ_Конференція_НУБІП_2019_UA.pdf (дата звернення: 06.08.2023)

126. Шкарупило В.В., Блінов І.В., Душеба В.В., Тіменко А.В. Дуальний підхід до формалізації функціональних характеристик систем критичного призначення. *European scientific discussions : 9th International scientific and practical conference. Potere della ragione Editore* (м. Рим, Італія, 18–20 липня, 2021 р.). С. 143–149. URL: https://sci-conf.com.ua/wp-content/uploads/2021/07/EUROPEAN-SCIENTIFIC-DISCUSSIONS-18-20.07.2021.pdf (дата звернення: 06.08.2023)

127. Томашевський В.М. Моделювання систем. Київ : Видавнича група BHV, 2005. 352 с.

128. Gupta H.V., Clark M.P., Vrugt J.A., Abramowitz G., Ye M. Towards a comprehensive assessment of model structural adequacy. *Water Resources Research*, 2012. Vol. 48, No. 8. P. 1–16. DOI: https://doi.org/10.1029/2011WR011044

129. Babai L. Graph isomorphism in quasipolynomial time. *Theory of Computing, STOC '16*: 48th annual ACM symposium (Cambridge, MA, USA, June 18–21, 2016). P. 684–697.

130. Mesarovic M. D., Macko D., Takahara Y. *Theory of hierarchical, multilevel, systems*. New York : Academic Press, 1970. 294 p.

131. Alsayaydeh J. A. J., Shkarupylo V., Hamid M. S. B., Skrupsky S., Oliinyk A. Stratified model of the Internet of Things infrastructure. *Journal of Engineering and Applied Sciences*. 2018. Vol. 13, No. 20. P. 8634–8638. ISSN: 1816-949x (Print), 1818-7803 (Online). DOI: https://medwelljournals.com/abstract/?doi=jeasci.2018.8634.8638 (**Scopus, Q3**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85056326734&origin=resultslist)

132. Shkarupylo V., Polska O. The Approach to SDN Network Topology Verification on a Basis of Temporal Logic of Actions. Proc. 14th Int. Conf. on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET'2018 (Lviv-Slavske, Ukraine, February 20–24, 2018). P. 183–186. DOI: https://doi.org/10.1109/TCSET.2018.8336182 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000465121700033 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85047524592&origin=resultslist)

133. Shkarupylo V. V., Timenko A. V. On the expediency of stratification to foster the reconfigurability of formal specifications. *Тенденції та вектор розвитку науки в сучасному світі: VI Міжнародна науково-практична інтернет-конференція*: тези доповідей, Дніпро, 30 квітня 2018 р. Ч. 1. Дніпро: НБК, 2018. С. 46–49. URL: https://ispic.ngo-seb.com/assets/files/6_conf_30.04.18_P.1.pdf (дата звернення: 06.08.2023)

134. Shkarupylo V. , Kudermetov R., Golub T., Polska O., Tiahunova M. Towards Model Checking of the Internet of Things Solutions Interoperability. *Problems of Infocommunications. Science and Technology*: proc. 2018 IEEE International Scientific and Practical Conference (Kharkiv, Ukraine, October 9–12, 2018). P. 465–468. DOI: https://doi.org/10.1109/INFOCOMMST.2018.8632037 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000458659100087 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85062879597&origin=resultslist)

135. Шкарупило В.В., Чемерис О.А., Душеба В.В., Кудерметов Р.К., Польська О.В. Метод синтезу формальних специфікацій на основі трійок Хоара. *Наукові праці ДонНТУ, Серія "Інформатика, кібернетика та обчислювальна техніка"*, 2020. № 1(30). С. 49-57. ISSN: 1996-1588. DOI: 10.31474/1996-1588-2020-1-30-49-57 (**фахове видання**)

136. Шкарупило В.В. Про застосування правила композиції при синтезі формальних специфікацій. *Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*, 15 травня 2020 р. Київ : ІПМЕ ім. Г.Є. Пухова НАН України, 2020. С. 20–21. URL: https://zenodo.org/record/3813710 (дата звернення: 06.08.2023)

137. Shkarupylo V., Chemerys O., Dusheba V., Kudermetov R., Oliinyk A. On Hoare triples applicability to dependable system specification synthesis. *Dependable Systems, Services and Technologies, DESSERT'2020* : The 11th International Conference (Kyiv, Ukraine, May 14–18, 2020). P. 371–375. DOI: https://doi.org/10.1109/DESSERT50317.2020.9125074 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000619228000064 ; **Scopus:**

https://www.scopus.com/record/display.uri?eid=2-s2.0-85087906543&origin=resultslist)

138. Shkarupylo V. V., Timenko A. V. On the interoperability and consistency aspects with respect to the Internet of Things domain. Engineering sciences: development prospects in countries of Europe at the beginning of the third millennium: Collective monograph. Vol. 2. Riga : Izdevnieciba “Baltija Publishing”. 2018. P. 466–485. ISBN 978-9934-571-63-3 (**розділ колективної монографії**)

139. Timenko A.V., Shkarupylo V.V., Oliinyk A.O., Hrushko S.S. Formal Model for Checking the Interoperability Between the Components of the IoT system. *Problemele Energeticii Regionale*, 2019. Vol. 40, No. 1-1. P. 69–78. ISSN: 1857-0070. DOI: https://zenodo.org/record/3239196 (**Web of Science Core Collection:** https://www.webofscience.com/wos/woscc/full-record/WOS:000472596400007)

140. Dahl O. -J., Dijkstra E. W., Hoare C. A. R. *Structured programming*: 10th printing. London, UK : Academic Press Ltd., 1982. P. 1-82.

141. Шкарупило В. В., Кудерметов Р. К., Польська О.В. DEVS-орієнтована методика валідації композитних веб-сервісів. *Радіоелектроніка, інформатика, управління*, 2015. № 4. С. 79–86. ISSN: 1607-3274 (Print), 2313-688X (Online). DOI: 10.15588/1607-3274-2015-4-12 URL: http://ric.zntu.edu.ua/article/view/60404 (дата звернення: 06.08.2023) (**фахове видання категорії А**)

142. В. В. Шкарупило, І. В. Блінов. Сценарії, методи та засоби формальної верифікації артефактів процесу проєктування систем критичного призначення: монографія. Вінниця: ГО «Європейська наукова платформа», 2021. 104 с. ISBN 978-617-8037-55-0. DOI https://doi.org/10.36074/smtzfvappskp-monograph.2021 (**колективна монографія**)

143. Shkarupylo V.V., Mazur D. Software defined networks basics. Internet of Things for Industry and Human Application. In Volumes 1-3. Volume 2. Modelling and Development / V. S. Kharchenko (ed.). Ministry of Education and

Science of Ukraine, National Aerospace University KhAI, 2019. P. 135–164. ISBN: 978-617-7361-82-3

144. Shkarupylo V.V. SDN programming and simulation of SDN composing, configuring and scaling. Internet of Things for Industry and Human Application. In Volumes 1-3. Volume 2. Modelling and Development / V. S. Kharchenko (ed.). Ministry of Education and Science of Ukraine, National Aerospace University KhAI, 2019. P. 165–193. ISBN: 978-617-7361-82-3

145. Shkarupylo V.V., Kudermetov R.K., Skarga-Bandurova I.S., Velykzhanin A.Yu., Shumova L.O., Mazur D.S., Kharchenko V.S., Uzun D.D., Uzun Y.O., Hodovaniuk P.A. Software defined networks and Internet of Things: Practicum / Kudermetov R.K. (Ed.) – Ministry of Education and Science of Ukraine, Zaporizhzhia National Technical University, Volodymyr Dahl East Ukrainian National University, National Aerospace University "KhAI", 2019. 129 p. ISBN: 978-617-7361-93-9

146. Cormen T.H., Leiserson C.E., Rivest R.L., Stein C. *Introduction to algorithms: 3rd ed*. Cambridge, Massachusetts: The MIT Press, 2009. 1320 p.

147. Шкарупило В. В. Концепція формальної верифікації UML-діаграм методами Model Checking. *Моделювання: XXXIV науково-технічна конференція*, 13–14 січня 2015 р.: тези доп. К.: ІПМЕ ім. Г. Є. Пухова НАН України, 2015. С. 13.

148. Shkarupylo V.V. An in-depth look at TLC model checker. *Тиждень науки-2016*: зб. тез доп. науково-практ. конф., 18–22 квітня 2016 р. Запоріжжя: ЗНТУ, 2016. С. 523–524. URL: https://zp.edu.ua/uploads/conference/2016/TN2016_T1.pdf (дата звернення: 06.08.2023)

149. Шкарупило В. В. Особливості використання методу формальної верифікації TLC. *Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова*

НАН України: тези доп., м. Київ, 12 січня 2016 р. С. 31. DOI: http://dx.doi.org/10.5281/zenodo.2545399

150. Shkarupylo V. TLC model checking and the concurrency in specification. *Proc. Tenth International Scientific-Practical Conference "INTERNET-EDUCATION-SCIENCE-2016", IES-2016* (Vinnytsia, Ukraine, October 11–14, 2016). P. 89–91. URL: http://ir.lib.vntu.edu.ua/handle/123456789/13390 (дата звернення: 06.08.2023)

151. Шкарупило В.В., Скрупський С.Ю. Комбінований підхід до застосування методу перевірки на моделі TLC. *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: X Ювілейна міжнародна науково-практична конференція, присвячена 120-річчю з дня заснування Національного університету «Запорізька політехніка»* (Запоріжжя, Україна, 7–9 жовтня, 2020). Запоріжжя: НУ «ЗП». С. 95–97. URL: http://rtt.zntu.edu.ua/data/Tezy_NUZP_2020.pdf (дата звернення: 06.08.2023)

152. Shkarupylo V.V., Tomičić I., Arapin D.V. The concurrency representation in TLA+ specification. *Proc. VIIIth Int. scientific-practical conf. on Modern problems and achievements of radio engineering (electronics), telecommunications and information technology* (Zaporizhzhya, Ukraine, September 21–23, 2016). P. 118–119. URL: http://rtt.zntu.edu.ua/data/Tezy_ZNTU_2016.pdf (дата звернення: 06.08.2023)

153. Шкарупило В.В., Кудерметов Р.К., Польська О.В. Дослідження просторової складності алгоритмів в основі методу верифікації TLC. *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: X Ювілейна міжнародна науково-практична конференція, присвячена 120-річчю з дня заснування Національного університету «Запорізька політехніка»* (Запоріжжя, Україна, 7–9 жовтня, 2020). Запоріжжя: НУ «ЗП». С. 93–95. URL: http://rtt.zntu.edu.ua/data/Tezy_NUZP_2020.pdf (дата звернення: 06.08.2023)

154. Шкарупило В.В., Чемерис О.А., Душеба В.В. Оцінювання просторової складності задачі формальної верифікації, вирішуваної методом перевірки на моделі. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2020. Том 31 (70), № 5. С. 147–151. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/2663-5941/2020.5/24 (**фахове видання**)

155. Методичні вказівки до лабораторних робіт з дисципліни "Грід обчислення та хмарні технології" для студентів спеціальності 123 "Комп'ютерна інженерія" всіх форм навчання / Укл. С.Ю. Скрупський, В.В. Шкарупило. Запоріжжя: ЗНТУ, 2018. 64 с. URL: http://eir.zp.edu.ua/handle/123456789/2990 (дата звернення: 06.08.2023) (**методичні вказівки**)

156. Shkarupylo V., Alsayaydeh J.A.J, Tomičić I., Chemeris A., Dusheba V. A technique for checking the adequacy of formal model. *ARPN Journal of Engineering and Applied Sciences*, August 2021. Vol. 16, No. 16. P. 1707–1719. ISSN: 1819-6608. URL: http://www.arpnjournals.org/jeas/research_papers/rp_2021/jeas_0821_8670.pdf (**Scopus, Q3**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85118181893&origin=resultslist)

157. Шкарупило В.В., Блінов І.В. Щодо застосування методу перевірки на моделі при проєктуванні інформаційно-технологічних систем суб'єктів ринку електроенергії. *XXXIX науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, присвячена 40-річчю Інституту, Дню науки в Україні та з нагоди відзначення 30-ї річниці незалежності України* (м. Київ, Україна, 12 травня, 2021). Київ: ІПМЕ ім. Г.Є. Пухова НАН України. С. 7–9. URL: https://drive.google.com/file/d/1QOydMJU3nHOjXZ92vcLF2zAclu8w1RcG/view?usp=share_link (дата звернення: 06.08.2023)

158. Shkarupylo V., Blinov I., Chemeris A., Dusheba V., Alsayaydeh J., Oliinyk A. Iterative Approach to TLC Model Checker Application. *Proc. 2021 IEEE KhPI Week on Advanced Technology* (Kharkiv, Ukraine, September 13–17, 2021). DOI: https://doi.org/10.1109/KhPIWeek53812.2021.9570055 (**Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85118943601&origin=resultslist)

159. Шкарупило В.В., Душеба В.В. Щодо аспектів контролю несуперечності програмно-алгоритмічної складової систем критичного призначення. *Продовольча та екологічна безпека в умовах війни та повоєнної відбудови, присвячена 125-річчю заснування Національного університету біоресурсів і природокористування України: виклики для України та світу*: мат. Міжн. наук.-практ. конф., секція 5: Інженерія, енергетика та інформаційні технології в умовах війни та післявоєнній відбудові країни (м. Київ, 25 трав. 2023 р.): тези доп. Київ: НУБіП України, 2023. С. 170–172. URL: https://nubip.edu.ua/sites/default/files/u381/sekciya_5.pdf (дата звернення: 06.08.2023)

160. Шкарупило В.В., Блінов І.В., Душеба В.В. Дослідження методу верифікації TLC при вирішенні задач енергетики. *XLI Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України* (м. Київ, Україна, 17 травня, 2023 р.). С. 21–22. URL: https://ipme.kiev.ua/wp-content/uploads/2023/05/Матеріали-конференції-2023.pdf (дата звернення: 06.08.2023)

161. Шкарупило В., Блінов І., Кучанський В., Давидюк А., Дімітрієва Д. Методи і засоби контролю артефактів процесу проєктування програмно-алгоритмічної складової систем критичного призначення: монографія / за заг. ред. В. В. Шкарупила. *Publishing House «European Scientific Platform»*, 2023, 120 с. ISBN: 978-617-8126-22-3 DOI: https://doi.org/10.36074/mzkapppasskp-monograph.2023 (**колективна монографія**)

162. Шкарупило В.В., Чемерис О.А., Душеба В.В., Кудерметов Р.К. Дослідження мультипоточної реалізації методу перевірки на моделі для темпоральної логіки дій. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2020. Том 31 (70), № 6, Ч. 1. С. 173–177. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/TNU-2663-5941/2020.6-1/28 (**фахове видання**)

163. Шкарупило В.В., Чемерис О.А., Душеба В.В. Дослідження впливу мультипоточності на швидкодію методу перевірки на моделі. *Безпека енергетики в епоху цифрової трансформації*: Друга науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України (Київ, Україна, 28–29 грудня, 2020). Київ : ІПМЕ ім. Г.Є. Пухова НАН України, 2020. С. 75–77. URL: https://ipme.kiev.ua/wp-content/uploads/2021/01/Програма-та-матеріали-КБЕЕЦ-2020.pdf (дата звернення: 06.08.2023)

164. Шкарупило В.В., Блінов І.В., Душеба В.В., Кучанський В.В. Щодо мультипоточного застосування формального методу перевірки на моделі TLC. *Topical issues of modern science, society and education. Proceedings of the 2nd International scientific and practical conference. SPC "Sci-conf.com.ua"*. Kharkiv, Ukraine. 2021. P. 231–236. URL: https://sci-conf.com.ua/wp-content/uploads/2021/09/TOPICAL-ISSUES-OF-MODERN-SCIENCE-SOCIETY-AND-EDUCATION-5-7.09.21.pdf (дата звернення: 06.08.2023)

165. Shkarupylo V.V., Blinov I.V., Chemeris A.A., Dusheba V.V., Alsayaydeh J.A.J. On Applicability of Model Checking Technique in Power Systems and Electric Power Industry. In: Zaporozhets A. (eds) Systems, Decision and Control in Energy III. Studies in Systems, Decision and Control, vol 399. Springer, Cham, 2022. ISBN 978-3-030-87675-3. DOI: https://doi.org/10.1007/978-3-030-87675-3_1 (**Scopus:** https://www.scopus.com/record/display.uri?eid=2-s2.0-85120868312&origin=resultslist&sort=plf-f **; розділ колективної монографії**)

166. Тіменко А.В., Шкарупило В.В., Скрупський С.Ю., Смолій В.В. Дослідження шляхів підвищення пропускної спроможності підсистеми пам'яті сучасної обчислювальної системи. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2020. Том 31 (70), Ч. 1, № 2. С. 208–212. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/2663-5941/2020.2-1/32 (**фахове видання**)

167. Shkarupylo V., Blinov I., Dusheba V., Alsayaydeh J. A. J. Case Driven TLC Model Checker Analysis in Energy Scenario. *CEUR Workshop Proceedings*, 2023. Vol. 3392. P. 65–75. ISSN 1613-0073. DOI: https://doi.org/10.32782/cmis/3392-6 (**Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85160296577&origin=resultslist&sort=plf-f)

168. Шкарупило В.В., Душеба В.В., Тіменко А.В. Огляд рівнів забезпечення резилієнтності у галузі енергетики. *Survivability & Resilience – 2023: collection of materials of the international scientific and practical conference*, Kyiv, October 19, 2023, PIMEE of NAS of Ukraine. 2023. P. 33–34. URL: https://ipme.kiev.ua/konferencii/zhivuchist-ta-rezilyentnist-2023/

169. Шкарупило В.В., Душеба В.В. Аспекти введення мультипоточності до реалізації методу формальної верифікації TLC. *Безпека енергетики в епоху цифрової трансформації*: П'ята науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України, Київ, Україна, 22 листопада, 2023 р. Київ : ІПМЕ ім. Г.Є. Пухова НАН України. С. 121–122. URL: https://ipme.kiev.ua/konferencii/naukovo-praktichna-konferenciya-bevect-2023/ (дата звернення: 23.11.2023)

170. Шкарупило В.В., Душеба В.В. Спадковість артефактів у контексті багатовимірної верифікації. *Тиждень науки-2022*: науково-практ. конф., 18–22 квітня 2022 р.: тези доп. Запоріжжя: НУ "Запорізька політехніка", 2022. С. 789–

791. URL: https://zp.edu.ua/uploads/dept_s&r/2022/conf/4.1/TN_2022.pdf (дата звернення: 06.08.2023)

171. Шкарупило В.В., Душеба В.В. Модельно-орієнтований підхід до синтезу формалізованих подань. *XL Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*, 11 травня 2022 р.: тези доп. Київ: ІПМЕ ім. Г.Є. Пухова НАН України, 2022. С. 20–22. URL: https://bit.ly/3cPGyFk (дата звернення: 06.08.2023)

172. Шкарупило В.В., Душеба В.В., Скрупський С.Ю., Блінов І.В. Стратифікована модель подання нефункціональних характеристик системи критичного призначення при проєктуванні. *Електронне моделювання*, 2022. Т. 44, № 2 (2022). С. 90–106. ISSN: 0204-3572. URL: https://www.emodel.org.ua/uk/archive-ukr/2022/44-2-u/c-90-106 DOI: https://doi.org/10.15407/emodel.44.02.090 (**фахове видання**)

173. Дімітрієва Д.О., Шкарупило В.В. Огляд нефункціональних характеристик систем критичного призначення. *XL Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*, 11 травня 2022 р.: тези доп. Київ: ІПМЕ ім. Г.Є. Пухова НАН України, 2022. С. 78–79. URL: https://bit.ly/3cPGyFk (дата звернення: 06.08.2023)

174. Concepcion A.I., Zeigler B.P. DEVS formalism: a framework for hierarchical model development. *IEEE Transactions on Software Engineering*, 1988. Vol. 14, No. 2. P. 228–241. DOI: https://doi.org/10.1109/32.4640

175. Shkarupylo V., Skrupsky S., Oliinyk A., Kolpakova T. Development of stratified approach to software defined networks simulation. *Eastern-European Journal of Enterprise Technologies. Information and controlling systems*, 2017. Vol. 5, No. 9 (89). P. 67–73. ISSN: 1729-3774 (Print), 1729-4061 (Online). DOI: https://doi.org/10.15587/1729-4061.2017.110142 (**Scopus, Q3**:

https://www.scopus.com/record/display.uri?eid=2-s2.0-85031750626&origin=resultslist; **фахове видання категорії А**)

176. Шкарупило В.В., Блінов І.В. Модельно-орієнтований підхід до формалізації нефункціональних характеристик систем критичного призначення, зокрема у природокористуванні. *Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні 2021*: IX Міжнародна науково-практична Інтернет-конференція (м. Київ, Україна, 13–14 травня, 2021). Київ: НУБіП України. С. 55–57. URL: https://drive.google.com/file/d/1IPmtWaLu85W3c9CsrYXYqx2_iFYIq4q2/view?usp=sharing (дата звернення: 06.08.2023)

177. Шкарупило В. В., Чемерис О. А., Душеба В. В., Кудерметов Р. К., Польська О. В. Модельно-орієнтований підхід до контролю показників нефункціональних характеристик під час проєктування. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2021. Том 32 (71), Ч. 1, № 1. С. 166–171. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/2663-5941/2021.1-1/27 (**фахове видання**)

178. Polska O.V., Kudermetov R.K., Shkarupylo.V.V. Discovery and selection of web-services. *Electrotechnic and computer systems*, 2015. No. 19(95). P. 169–173. ISSN: 2221-3937 (Print), 2221-3805 (Online). URL: http://nbuv.gov.ua/UJRN/etks_2015_19_39 (дата звернення: 06.08.2023) (**фахове видання**)

179. Kudermetov R., Polska O., Shkarupylo V., Shcherbak N. Quality of services in scientific workflows. *Electrotechnic and Computer Systems,* 2018. Vol. 28, No. 104. P. 170–177. ISSN: 2221-3805 (Print), 2221-3937 (Online). DOI: 10.15276/eltecs.28.104.2018.20 URL: https://eltecs.op.edu.ua/index.php/journal/article/view/155/42 (**фахове видання**)

180. Polska O.V., Kudermetov R.K., Shkarupylo V.V. The approach for QoS based web service selection with user's preferences. *Наукові праці Донецького національного технічного університету, серія: «Проблеми моделювання та автоматизації проектування»*, 2020. №2 (16). С. 19–27. ISSN: 2074-7888. DOI: 10.31474/2074-7888-2020-2-19-27 URL: http://pmap.donntu.edu.ua/sites/upload/articles/pmap_2020_19-27.pdf (дата звернення: 06.08.2023) (**фахове видання**)

181. Polska O.V., Kudermetov R.K., Shkarupylo V.V. An approach web service selection by quality criteria based on sensitivity analysis of MCDM methods. *Radio Electronics, Computer Science, Control*, 2021. No. 2. P. 133–143. ISSN: 1607-3274 (Online), 2313-688X (Print). DOI: https://doi.org/10.15588/1607-3274-2021-2-14 (**Web of Science Core Collection:** https://www.webofscience.com/wos/woscc/full-record/WOS:000673377700014 **; фахове видання категорії А**)

182. Polska O.V., Kudermetov R.K., Zolotukhina O.A., Shkarupylo V.V. A UML profile for quality-based web service selection using logic scoring of preference method. *Telecommunication and information technologies*, 2021. No. 1 (2021). P. 65–78. ISSN: 2412-4338. DOI: https://doi.org/10.31673/2412-4338.2021.016578 (**фахове видання**)

183. Shkarupylo V.V. An Approach to DEVS-driven Simulation of Software-defined Networks. *Тиждень науки-2017*: науково-практ. конф., 18–21 квітня 2017 р.: тези доп. Запоріжжя: ЗНТУ, 2017. С. 652. URL: https://zp.edu.ua/uploads/dept_s&r/2017/conf/1/TN2017.pdf (дата звернення: 06.08.2023)

184. Shkarupylo V.V., Timenko A.V. An approach to the Internet of things simulation on a basis of discrete event system specification. *Proc. Int. research and practice conference on Modern methods, innovations, and experience of practical*

application in the field of technical sciences (Radom, Republic of Poland, Dec. 27–28, 2017). P. 32–34.

185. Шкарупило В.В., Душеба В.В. Підхід до синтезу формалізованих подань нефункціональних характеристик на етапі проєктування. *Безпека енергетики в епоху цифрової трансформації* : ІІІ науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України : матеріали, 22 грудня 2021 р. Київ : ІПМЕ ім. Г.Є.Пухова НАН України, 2021. С. 128–130. URL: https://ipme.kiev.ua/wp-content/uploads/2021/12/Матеріали-КБЕЕЦ-2021-1.pdf (дата звернення: 06.08.2023)

186. Блінов І.В., Парус Є.В, Шкарупило В.В. Структура та моделі інформаційної взаємодії учасників ринку електричної енергії: монографія. Вінниця: ГО «Європейська наукова платформа», 2021. 114 с. ISBN 978-617-8037-31-4 DOI: https://doi.org/10.36074/stmivyree-monograph.2021 (**колективна монографія**)

187. Шкарупило В.В., Душеба В.В., Тіменко А.В., Казакова Н.О. Аспекти досягнення функційної безпечності при розробленні систем критичного призначення. *Актуальні проблеми та перспективи розвитку фундаментальних, прикладних, загальнотехнічних та безпекових наук*: ІІІ Всеукраїнської науково-практичної конференції пам’яті академіка Академії наук вищої освіти, професора Анатолія Володимировича Касперського (м. Київ, Україна, 21 червня 2023 р.): тези доп. Київ: УДУ ім. Михайла Драгоманова, 2023. С. 394–395. URL: https://drive.google.com/file/d/1nS1d9cf9EUNUZDnY0ZWLkQTaBlb0xoIN/view (дата звернення: 06.08.2023)

188. Шкарупило В.В., Тіменко А.В. Складові методу контролю показників нефункціональних характеристик розроблюваної комп'ютерної системи при проєктуванні. *XLIX Міжнародна науково-практична інтернет-*

конференція «Проблеми та перспективи розвитку сучасної науки в країнах Європи та Азії», 31 серпня 2022 р.: тези доп., Переяслав, 2022. С. 69–71.

189. Борукаєв З.Х., Блінов І.В., Остапченко К.Б., Чемерис О.А., Шкарупило В.В. Моделі та засоби автоматизації систем організаційного управління енергоринком: монографія / за заг. ред. З.Х. Борукаєва. Вінниця: ГО «Європейська наукова платформа», 2022. 122 с. ISBN: 978-617-8037-82-6 DOI: https://doi.org/10.36074/mtzasoye-monograph.2022 (**колективна монографія**)

ДОДАТОК А
СПЕЦИФІКАЦІЇ ЗГІДНО ПОСЛІДОВНОГО ШАБЛОНУ

Лістинг А.1 – Приклад формальної специфікації TLA+ згідно послідовного шаблону для двох змінних станів

```
// Для n=2^1
---------------- MODULE sec2 ----------------
EXTENDS Naturals
VARIABLES   v1,v2
Invariant == v1 \in (0..2) /\ v2 \in (0..2)
Init == v1 = 0 /\ v2 = 0
S1   == v1 = 1 /\ v2 = 0
S2   == v1 = 2 /\ v2 = 0
S3   == v1 = 2 /\ v2 = 1
R1_01 == /\ v1' = IF Init THEN v1+1 ELSE v1
         /\ UNCHANGED <<v2>>
R1_12 == /\ v1' = IF S1 THEN v1+1 ELSE v1
         /\ UNCHANGED <<v2>>
R2_23 == /\ v2' = IF S2 THEN v2+1 ELSE v2
         /\ UNCHANGED <<v1>>
R2_34 == /\ v2' = IF S3 THEN v2+1 ELSE v2
         /\ UNCHANGED <<v1>>
Next == R1_01 \/ R1_12 \/ R2_23 \/ R2_34
Spec == Init /\[][Next]_<<v1,v2>>
=====
```

Лістинг А.2 – Приклад формальної специфікації TLA+ згідно послідовного шаблону для чотирьох змінних станів

```
// Для n=2^2
---------------- MODULE sec4 -----------------
EXTENDS Naturals
VARIABLES v1, v2, v3, v4
Invariant ==  /\ v1 \in (0..2)  /\ v2 \in (0..2)  /\ v3 \in (0..2)
/\ v4 \in (0..2)
Init == (v1=0) /\ (v2=0) /\ (v3=0) /\ (v4=0)
S1 == (v1=1) /\ (v2=0) /\ (v3=0) /\ (v4=0)
S2 == (v1=2) /\ (v2=0) /\ (v3=0) /\ (v4=0)
S3 == (v1=2) /\ (v2=1) /\ (v3=0) /\ (v4=0)
S4 == (v1=2) /\ (v2=2) /\ (v3=0) /\ (v4=0)
S5 == (v1=2) /\ (v2=2) /\ (v3=1) /\ (v4=0)
S6 == (v1=2) /\ (v2=2) /\ (v3=2) /\ (v4=0)
S7 == (v1=2) /\ (v2=2) /\ (v3=2) /\ (v4=1)
R_01 ==   /\ v1' = IF Init THEN v1+1 ELSE v1
         /\ UNCHANGED <<v2,v3,v4>>
R_12 ==   /\ v1' = IF S1 THEN v1+1 ELSE v1
         /\ UNCHANGED <<v2,v3,v4>>
R_23 ==   /\ v2' = IF S2 THEN v2+1 ELSE v2
         /\ UNCHANGED <<v1,v3,v4>>
R_34 ==   /\ v2' = IF S3 THEN v2+1 ELSE v2
         /\ UNCHANGED <<v1,v3,v4>>
R_45 ==   /\ v3' = IF S4 THEN v3+1 ELSE v3
         /\ UNCHANGED <<v1,v2,v4>>
R_56 ==   /\ v3' = IF S5 THEN v3+1 ELSE v3
         /\ UNCHANGED <<v1,v2,v4>>
R_67 ==   /\ v4' = IF S6 THEN v4+1 ELSE v4
         /\ UNCHANGED <<v1,v2,v3>>
R_78 ==   /\ v4' = IF S7 THEN v4+1 ELSE v4
         /\ UNCHANGED <<v1,v2,v3>>
Next == R_01 \/ R_12 \/ R_23 \/ R_34 \/ R_45 \/ R_56 \/ R_67 \/
R_78
Spec == Init/\[][Next]_<<v1,v2,v3,v4>>
=====
```

ДОДАТОК Б
СПЕЦИФІКАЦІЇ ІЗ ПОДАННЯМ ПАРАЛЕЛІЗМУ ЗГІДНО МОДЕЛІ ЧЕРГУВАННЯ

Лістинг Б.1 – Приклад формальної специфікації TLA+ для чотирьох змінних станів

```
// Для n=2^2
--------------- MODULE flow4 -------------------
VARIABLES v1, v2, v3, v4
Invariant ==   /\ v1 \in BOOLEAN   /\ v2 \in BOOLEAN   /\ v3 \in
BOOLEAN  /\ v4 \in BOOLEAN
Init == v1=FALSE /\ v2=FALSE /\ v3=FALSE /\ v4=FALSE
S_1 ==  /\ v1=TRUE /\ v2=FALSE /\ v3=FALSE /\ v4=FALSE
S_2 ==  /\ v1=TRUE /\ v2=TRUE /\ v3=FALSE /\ v4=FALSE
S_3 ==  /\ v1=TRUE /\ v2=FALSE /\ v3=TRUE /\ v4=FALSE
S_4 ==  /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=FALSE
R_0 == v1' = IF Init THEN ~v1 ELSE v1
R_1 == v2' = IF S_1 THEN ~v2 ELSE v2
R_2 == v3' = IF S_1 THEN ~v3 ELSE v3
R_3 == v3' = IF S_2 THEN ~v3 ELSE v3
R_4 == v2' = IF S_3 THEN ~v2 ELSE v2
R_5 == v4' = IF S_4 THEN ~v4 ELSE v4
Next == /\ R_0
 /\      \/ (/\ R_1/\ R_3)
         \/ (/\ R_2/\ R_4)
         /\ R_5
Spec == Init/\[][Next]_<<v1,v2,v3,v4>>
=========
```

Лістинг Б.2 – Приклад формальної специфікації TLA+ для восьми змінних станів

```
// Для n=2^3
--------------- MODULE flow8 --------------------
VARIABLES v1, v2, v3, v4, v5, v6, v7, v8
Invariant ==   /\ v1 \in BOOLEAN   /\ v2 \in BOOLEAN   /\ v3 \in
BOOLEAN   /\ v4 \in BOOLEAN   /\ v5 \in BOOLEAN   /\ v6 \in BOOLEAN
/\ v7 \in BOOLEAN   /\ v8 \in BOOLEAN
Init == v1=FALSE /\ v2=FALSE /\ v3=FALSE /\ v4=FALSE /\ v5=FALSE /\
v6=FALSE /\ v7=FALSE /\ v8=FALSE
S_1 ==   /\ v1=TRUE /\ v2=FALSE /\ v3=FALSE /\ v4=FALSE /\ v5=FALSE
/\ v6=FALSE /\ v7=FALSE /\ v8=FALSE
S_2 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=FALSE /\ v4=FALSE /\ v5=FALSE
/\ v6=FALSE /\ v7=FALSE /\ v8=FALSE
S_3 ==   /\ v1=TRUE /\ v2=FALSE /\ v3=TRUE /\ v4=FALSE /\ v5=FALSE
/\ v6=FALSE /\ v7=FALSE /\ v8=FALSE
S_4 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=FALSE /\ v5=FALSE /\
v6=FALSE /\ v7=FALSE /\ v8=FALSE
S_5 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=TRUE /\ v5=FALSE /\
v6=FALSE /\ v7=FALSE /\ v8=FALSE
S_6 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=FALSE /\ v5=TRUE /\
v6=FALSE /\ v7=FALSE /\ v8=FALSE
S_7 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=TRUE /\ v5=TRUE /\
v6=FALSE /\ v7=FALSE /\ v8=FALSE
S_8 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=FALSE /\ v5=FALSE /\
v6=TRUE /\ v7=FALSE /\ v8=FALSE
S_9 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=TRUE /\ v5=FALSE /\
v6=TRUE /\ v7=FALSE /\ v8=FALSE
S_10 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=FALSE /\ v5=TRUE /\
v6=TRUE /\ v7=FALSE /\ v8=FALSE
S_11 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=TRUE /\ v5=TRUE /\
v6=TRUE /\ v7=FALSE /\ v8=FALSE
S_12 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=FALSE /\ v5=FALSE
/\ v6=FALSE /\ v7=TRUE /\ v8=FALSE
S_13 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=TRUE /\ v5=FALSE /\
v6=FALSE /\ v7=TRUE /\ v8=FALSE
S_14 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=FALSE /\ v5=TRUE /\
v6=FALSE /\ v7=TRUE /\ v8=FALSE
S_15 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=TRUE /\ v5=TRUE /\
v6=FALSE /\ v7=TRUE /\ v8=FALSE
S_16 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=FALSE /\ v5=FALSE
/\ v6=TRUE /\ v7=TRUE /\ v8=FALSE
S_17 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=TRUE /\ v5=FALSE /\
v6=TRUE /\ v7=TRUE /\ v8=FALSE
S_18 ==   /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=FALSE /\ v5=TRUE /\
v6=TRUE /\ v7=TRUE /\ v8=FALSE
```

Продовження лістингу Б.2

```
S_19 ==  /\ v1=TRUE /\ v2=TRUE /\ v3=TRUE /\ v4=TRUE /\ v5=TRUE /\
v6=TRUE /\ v7=TRUE /\ v8=FALSE
R_0 == v1' = IF Init THEN ~v1 ELSE v1
R_1 == v2' = IF S_1 THEN ~v2 ELSE v2
R_2 == v3' = IF S_1 THEN ~v3 ELSE v3
R_3 == v3' = IF S_2 THEN ~v3 ELSE v3
R_4 == v2' = IF S_3 THEN ~v2 ELSE v2
R_5 == v4' = IF S_4 THEN ~v4 ELSE v4
R_6 == v5' = IF S_4 THEN ~v5 ELSE v5
R_7 == v6' = IF S_4 THEN ~v6 ELSE v6
R_8 == v7' = IF S_4 THEN ~v7 ELSE v7
R_9 == v5' = IF S_5 THEN ~v5 ELSE v5
R_10 == v6' = IF S_5 THEN ~v6 ELSE v6
R_11 == v7' = IF S_5 THEN ~v7 ELSE v7
R_12 == v4' = IF S_6 THEN ~v4 ELSE v4
R_13 == v6' = IF S_6 THEN ~v6 ELSE v6
R_14 == v7' = IF S_6 THEN ~v7 ELSE v7
R_15 == v6' = IF S_7 THEN ~v6 ELSE v6
R_16 == v7' = IF S_7 THEN ~v7 ELSE v7
R_17 == v4' = IF S_8 THEN ~v4 ELSE v4
R_18 == v5' = IF S_8 THEN ~v5 ELSE v5
R_19 == v7' = IF S_8 THEN ~v7 ELSE v7
R_20 == v5' = IF S_9 THEN ~v5 ELSE v5
R_21 == v7' = IF S_9 THEN ~v7 ELSE v7
R_22 == v4' = IF S_10 THEN ~v4 ELSE v4
R_23 == v7' = IF S_10 THEN ~v7 ELSE v7
R_24 == v7' = IF S_11 THEN ~v7 ELSE v7
R_25 == v4' = IF S_12 THEN ~v4 ELSE v4
R_26 == v5' = IF S_12 THEN ~v5 ELSE v5
R_27 == v6' = IF S_12 THEN ~v6 ELSE v6
R_28 == v5' = IF S_13 THEN ~v5 ELSE v5
R_29 == v6' = IF S_13 THEN ~v6 ELSE v6
R_30 == v4' = IF S_14 THEN ~v4 ELSE v4
R_31 == v6' = IF S_14 THEN ~v6 ELSE v6
R_32 == v6' = IF S_15 THEN ~v6 ELSE v6
R_33 == v4' = IF S_16 THEN ~v4 ELSE v4
R_34 == v5' = IF S_16 THEN ~v5 ELSE v5
R_35 == v5' = IF S_17 THEN ~v5 ELSE v5
R_36 == v4' = IF S_18 THEN ~v4 ELSE v4
R_37 == v8' = IF S_19 THEN ~v8 ELSE v8
Next == /\ R_0
 /\      \/ (/\ R_1/\ R_3/\ R_5/\ R_9/\ R_15/\ R_24)
         \/ (/\ R_1/\ R_3/\ R_5/\ R_9/\ R_16/\ R_32)
         \/ (/\ R_1/\ R_3/\ R_5/\ R_10/\ R_20/\ R_24)
         \/ (/\ R_1/\ R_3/\ R_5/\ R_10/\ R_21/\ R_35)
```

Продовження лістингу Б.2

```
        \/ (/\ R_1/\ R_3/\ R_5/\ R_11/\ R_28/\ R_32)
        \/ (/\ R_1/\ R_3/\ R_5/\ R_11/\ R_29/\ R_35)
        \/ (/\ R_1/\ R_3/\ R_6/\ R_12/\ R_15/\ R_24)
        \/ (/\ R_1/\ R_3/\ R_6/\ R_12/\ R_16/\ R_32)
        \/ (/\ R_1/\ R_3/\ R_6/\ R_13/\ R_22/\ R_24)
        \/ (/\ R_1/\ R_3/\ R_6/\ R_13/\ R_23/\ R_36)
        \/ (/\ R_1/\ R_3/\ R_6/\ R_14/\ R_30/\ R_32)
        \/ (/\ R_1/\ R_3/\ R_6/\ R_14/\ R_31/\ R_36)
        \/ (/\ R_1/\ R_3/\ R_7/\ R_17/\ R_20/\ R_24)
        \/ (/\ R_1/\ R_3/\ R_7/\ R_17/\ R_21/\ R_35)
        \/ (/\ R_1/\ R_3/\ R_7/\ R_18/\ R_22/\ R_24)
        \/ (/\ R_1/\ R_3/\ R_7/\ R_18/\ R_23/\ R_36)
        \/ (/\ R_1/\ R_3/\ R_7/\ R_19/\ R_33/\ R_35)
        \/ (/\ R_1/\ R_3/\ R_7/\ R_19/\ R_34/\ R_36)
        \/ (/\ R_1/\ R_3/\ R_8/\ R_25/\ R_28/\ R_32)
        \/ (/\ R_1/\ R_3/\ R_8/\ R_25/\ R_29/\ R_35)
        \/ (/\ R_1/\ R_3/\ R_8/\ R_26/\ R_30/\ R_32)
        \/ (/\ R_1/\ R_3/\ R_8/\ R_26/\ R_31/\ R_36)
        \/ (/\ R_1/\ R_3/\ R_8/\ R_27/\ R_33/\ R_35)
        \/ (/\ R_1/\ R_3/\ R_8/\ R_27/\ R_34/\ R_36)
        \/ (/\ R_2/\ R_4/\ R_5/\ R_9/\ R_15/\ R_24)
        \/ (/\ R_2/\ R_4/\ R_5/\ R_9/\ R_16/\ R_32)
        \/ (/\ R_2/\ R_4/\ R_5/\ R_10/\ R_20/\ R_24)
        \/ (/\ R_2/\ R_4/\ R_5/\ R_10/\ R_21/\ R_35)
        \/ (/\ R_2/\ R_4/\ R_5/\ R_11/\ R_28/\ R_32)
        \/ (/\ R_2/\ R_4/\ R_5/\ R_11/\ R_29/\ R_35)
        \/ (/\ R_2/\ R_4/\ R_6/\ R_12/\ R_15/\ R_24)
        \/ (/\ R_2/\ R_4/\ R_6/\ R_12/\ R_16/\ R_32)
        \/ (/\ R_2/\ R_4/\ R_6/\ R_13/\ R_22/\ R_24)
        \/ (/\ R_2/\ R_4/\ R_6/\ R_13/\ R_23/\ R_36)
        \/ (/\ R_2/\ R_4/\ R_6/\ R_14/\ R_30/\ R_32)
        \/ (/\ R_2/\ R_4/\ R_6/\ R_14/\ R_31/\ R_36)
        \/ (/\ R_2/\ R_4/\ R_7/\ R_17/\ R_20/\ R_24)
        \/ (/\ R_2/\ R_4/\ R_7/\ R_17/\ R_21/\ R_35)
        \/ (/\ R_2/\ R_4/\ R_7/\ R_18/\ R_22/\ R_24)
        \/ (/\ R_2/\ R_4/\ R_7/\ R_18/\ R_23/\ R_36)
        \/ (/\ R_2/\ R_4/\ R_7/\ R_19/\ R_33/\ R_35)
        \/ (/\ R_2/\ R_4/\ R_7/\ R_19/\ R_34/\ R_36)
        \/ (/\ R_2/\ R_4/\ R_8/\ R_25/\ R_28/\ R_32)
        \/ (/\ R_2/\ R_4/\ R_8/\ R_25/\ R_29/\ R_35)
        \/ (/\ R_2/\ R_4/\ R_8/\ R_26/\ R_30/\ R_32)
        \/ (/\ R_2/\ R_4/\ R_8/\ R_26/\ R_31/\ R_36)
        \/ (/\ R_2/\ R_4/\ R_8/\ R_27/\ R_33/\ R_35)
        \/ (/\ R_2/\ R_4/\ R_8/\ R_27/\ R_34/\ R_36)
        /\ R_37
Spec == Init/\[][Next]_<<v1,v2,v3,v4,v5,v6,v7,v8>>
=======
```

ДОДАТОК В
СПЕЦИФІКАЦІЯ ФРАГМЕНТУ АЛГОРИТМУ РОБОТИ БУК

Лістинг В.1 – ФС із трьома змінними станів

```
-------- MODULE spec --------
EXTENDS Naturals
VARIABLES v1,v2,v3
\* 5 states, depth = 3
\* obtained manually
Invariant == v1 \in {0,1} /\ v2 \in {0,1} /\ v3 \in {0,1}
Init == v1 \in {0,1} /\ v2 = 0 /\ v3=0
R02 == /\ v3' = IF v1=0 /\ v3=0 THEN 1-v3 ELSE v3
       /\ UNCHANGED <<v1,v2>>
R13 == /\ v2' = IF v1=1 /\ v2=0 THEN 1-v2 ELSE v2
       /\ UNCHANGED <<v1>>
R34 == /\ v3' = IF v2=1 /\ v3=0 THEN 1-v3 ELSE v3
       /\ UNCHANGED <<v1>>

Next == R02 \/ (R13 /\ R34)

Spec == Init/\[][Next]_<<v1,v2,v3>>
========
```

ДОДАТОК Г
ЗАСІБ АВТОМАТИЗОВАНОЇ ФІКСАЦІЇ ЧАСОВИХ ВИТРАТ

Лістинг Г.1 – Програмна реалізація-засіб автоматизації процесу фіксації часових витрат

```
@echo off
setlocal
set /a num=0
:m1
set /a num =%num%+1
set t0=%time: =0%

cls
java    -d64    -XX:MaxHeapSize=8192m    -XX:+UseParallelGC    -cp
D:\me\TLA\TLAToolbox-1.6.0-
win32.win32.x86_64\toolbox\plugins\org.lamport.tlatools_1.0.0.20190
7102009 tlc2.TLC -config spec.cfg -dfid 42 -workers 8 spec.tla >
DFS_out_log.txt
set t=%time: =0%

set /a h=1%t0:~0,2%-100
set /a m=1%t0:~3,2%-100
set /a s=1%t0:~6,2%-100
set /a c=1%t0:~9,2%-100
set /a starttime = %h% * 360000 + %m% * 6000 + 100 * %s% + %c%

set /a h=1%t:~0,2%-100
set /a m=1%t:~3,2%-100
set /a s=1%t:~6,2%-100
set /a c=1%t:~9,2%-100
set /a endtime = %h% * 360000 + %m% * 6000 + 100 * %s% + %c%

set /a runtime = %endtime% - %starttime%
set runtime = %s%.%c%
echo Started at %t0%

echo %runtime%0>>DFS_res_log.txt
if %num% equ 1 goto end
timeout 1
goto m1
:end
echo dfs done!
```

ДОДАТОК Д
ПРИКЛАД СИНТЕЗОВАНОЇ СПЕЦИФІКАЦІЇ

Лістинг Д.1 – Результуюча формальна специфікація, отримана в автоматизованому режимі на основі PlusCal-подання

```
\* BEGIN TRANSLATION
\* obtained in an automated manner
VARIABLES v1, v2, v3
vars == << v1, v2, v3 >>

Invar ==
        /\ v1 \in BOOLEAN
        /\ v2 \in BOOLEAN
        /\ v3 \in BOOLEAN

Init ==
        /\ \/ v1 = TRUE
           \/ v1 = FALSE
        /\ v2 = FALSE
        /\ v3 = FALSE
r02 ==
       /\ IF ~v1 /\ ~v3
             THEN /\ v3' = TRUE
                  /\ v2' = v2
             ELSE /\ IF v1 /\ ~v2
                        THEN /\ v2' = TRUE
                        ELSE /\ TRUE
                             /\ v2' = v2
                  /\ v3' = v3
       /\ v1' = v1
r34 ==
       /\ IF v2 /\ ~v3
             THEN /\ v3' = TRUE
             ELSE /\ TRUE
                  /\ v3' = v3
       /\ UNCHANGED << v1, v2 >>

Next == r02 \/ r34
Spec == Init /\[][Next]_vars
========
```

ДОДАТОК Е
СПЕЦИФІКАЦІЇ ДЛЯ ТЕМАТИЧНОГО ДОСЛІДЖЕННЯ

Лістинг Е.1 – Формальна специфікація на основі засобів PlusCal для чотирьох змінних станів

```
-------- MODULE spec --------
EXTENDS Naturals
(* --algorithm example
variables
\*p.185 - variables and flags
        flag \in {0,1},         \*v1
        ISH \in {0,1,2,100},    \*v2 -- 2 instead of 01
        RRP_IRQ5 \in {0,1},     \*v3
\*finalize flags
        done \in {0,1,2},   \* v18 -- for 1, 7, 8 blocks
\*------------------------
begin
\* initial state
\* precondition to start
 m1:  done := 0; goto m2;
\*-----
\*p.185 - 20 blobks: 2 - 21; ISH \in {0, 1, 100}
\*1
 m2:    flag := 0; goto m3;
 m3:    ISH := RRP_IRQ5; goto m4;
 m4:    if ISH # 0 then
 m5:        if flag = 0 then
 m6:            RRP_IRQ5 := 0;
                flag := 1;
                goto m3;
            else goto m7;  \* 1
            end if;
        else
            goto m8;
        end if;
\*2
\* for m77, m78 blocks; plus mine m79 block for 'stop' flag
 m7:   done := 1;
 m8:   done := 2;
end algorithm; *)
========
```

Лістинг Е.2 – Результуюча формальна специфікація на основі засобів TLA+ для чотирьох змінних станів

```
\* BEGIN TRANSLATION
VARIABLES flag, ISH, RRP_IRQ5, done, pc
vars == << flag, ISH, RRP_IRQ5, done, pc >>

Invariant == /\ flag \in {0,1}
        /\ ISH \in {0,1,2,100}
        /\ RRP_IRQ5 \in {0,1}
        /\ done \in {0,1,2}

Init == (* Global variables *)
        /\ flag \in {0,1}
        /\ ISH \in {0,1,2,100}
        /\ RRP_IRQ5 \in {0,1}
        /\ done \in {0,1,2}
        /\ pc = "m1"

m1 == /\ pc = "m1"
      /\ done' = 0
      /\ pc' = "m2"
      /\ UNCHANGED << flag, ISH, RRP_IRQ5 >>

m2 == /\ pc = "m2"
      /\ flag' = 0
      /\ pc' = "m3"
      /\ UNCHANGED << ISH, RRP_IRQ5, done >>

m3 == /\ pc = "m3"
      /\ ISH' = RRP_IRQ5
      /\ pc' = "m4"
      /\ UNCHANGED << flag, RRP_IRQ5, done >>

m4 == /\ pc = "m4"
      /\ IF ISH # 0
            THEN /\ pc' = "m5"
            ELSE /\ pc' = "m78"
      /\ UNCHANGED << flag, ISH, RRP_IRQ5, done >>

m5 == /\ pc = "m5"
      /\ IF flag = 0
            THEN /\ pc' = "m6"
            ELSE /\ pc' = "m77"
      /\ UNCHANGED << flag, ISH, RRP_IRQ5, done >>
```

Продовження лістингу Е.2

```
m6 == /\ pc = "m6"
      /\ RRP_IRQ5' = 0
      /\ flag' = 1
      /\ pc' = "m3"
      /\ UNCHANGED << ISH, done >>

m77 == /\ pc = "m77"
       /\ done' = 1
       /\ pc' = "m78"
       /\ UNCHANGED << flag, ISH, RRP_IRQ5 >>

m78 == /\ pc = "m78"
       /\ done' = 2
       /\ pc' = "Done"
       /\ UNCHANGED << flag, ISH, RRP_IRQ5 >>

(* Allow infinite stuttering to prevent deadlock on termination. *)
Terminating == pc = "Done" /\ UNCHANGED vars

Next == m1 \/ m2 \/ m3 \/ m4 \/ m5 \/ m6 \/ m77 \/ m78
           \/ Terminating

Spec == Init /\ [][Next]_vars

Termination == <>(pc = "Done")

\* END TRANSLATION

============
```

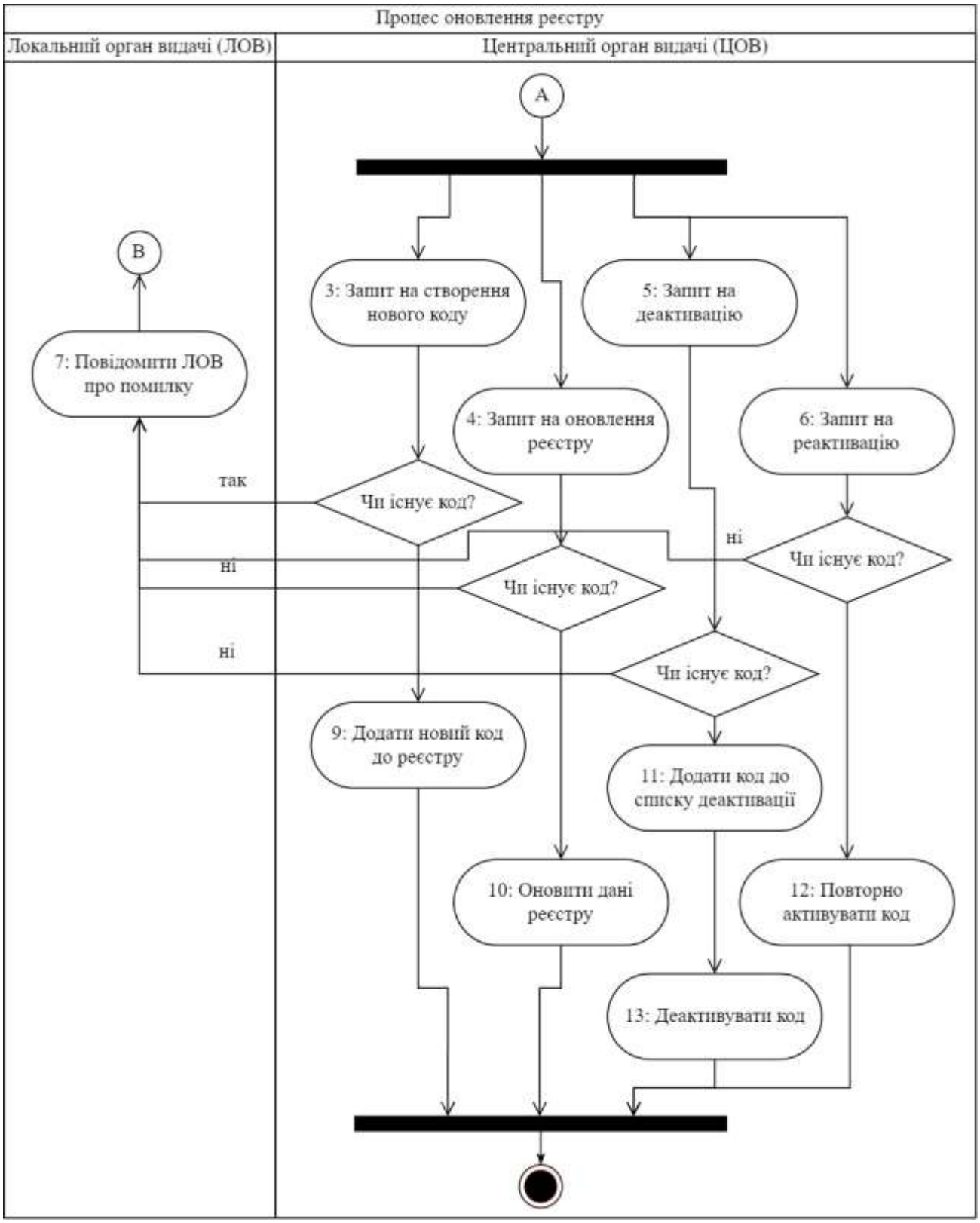

Рисунок Е.1 – Продовження фрагменту рольової моделі оновлення реєстру ідентифікаційних кодів учасників європейського ринку електричної енергії

ДОДАТОК Ж
ПАРАМЕТРИ І ЗНАЧЕННЯ ОЦІНОЧНИХ ФУНКЦІЙ

Таблиця Ж.1 – Вихідні дані, значення функції $\xi_1(x)$ ($n = 4$, $depth = 10$)

№ з/п	№ заміру										
	1	2	3	4	5	6	7	8	9	10	11
1	0,893	-	-	-	-	-	-	-	-	-	-
2	0,893	0,44	-	-	-	-	-	-	-	-	-
3	0,893	0,44	0,44	-	-	-	-	-	-	-	-
4	0,893	0,44	0,44	0,43	-	-	-	-	-	-	-
5	0,893	0,44	0,44	0,43	0,44	-	-	-	-	-	-
6	0,893	0,44	0,44	0,43	0,44	0,44	-	-	-	-	-
7	0,893	0,44	0,44	0,43	0,44	0,44	0,44	-	-	-	-
8	0,893	0,44	0,44	0,43	0,44	0,44	0,44	0,43	-	-	-
9	0,893	0,44	0,44	0,43	0,44	0,44	0,44	0,43	0,43	-	-
10	0,893	0,44	0,44	0,43	0,44	0,44	0,44	0,43	0,43	0,44	-
11	0,893	0,44	0,44	0,43	0,44	0,44	0,44	0,43	0,43	0,44	0,44

Таблиця Ж.2 – Вихідні дані, значення функції $\xi_2(x)$ ($n = 8$, $depth = 22$)

№ з/п	№ заміру										
	1	2	3	4	5	6	7	8	9	10	11
1	1,001	-	-	-	-	-	-	-	-	-	-
2	1,001	0,74	-	-	-	-	-	-	-	-	-
3	1,001	0,74	0,77	-	-	-	-	-	-	-	-
4	1,001	0,74	0,77	0,76	-	-	-	-	-	-	-
5	1,001	0,74	0,77	0,76	0,74	-	-	-	-	-	-
6	1,001	0,74	0,77	0,76	0,74	0,78	-	-	-	-	-
7	1,001	0,74	0,77	0,76	0,74	0,78	0,79	-	-	-	-
8	1,001	0,74	0,77	0,76	0,74	0,78	0,79	0,82	-	-	-
9	1,001	0,74	0,77	0,76	0,74	0,78	0,79	0,82	0,90	-	-
10	1,001	0,74	0,77	0,76	0,74	0,78	0,79	0,82	0,90	0,78	-
11	1,001	0,74	0,77	0,76	0,74	0,78	0,79	0,82	0,90	0,78	0,80

Лістинг Ж.1 – Програмна реалізація обчислення функцій $\xi_1(x)$, $\xi_1'(x)$, $\xi_2(x)$, $\xi_2'(x)$, а також t-критерію

```
#include <iostream>
#include <iomanip>
#include <math.h>
#include <float.h>
#include <ctime>
#include <cmath>
#include <cstdlib>
#include <locale.h>
using namespace std;

//get up estim
double get_up_approx(int x) {
        double a = 1.000214225;
        double b = 0.052512779;
        double c = 1.567945107;
        double d = -2.12507041;
        double e = 1.429643245;
        double f = -0.55597656;
        double g = 0.129289625;
        double h = -0.01768063;
        double i = 0.001308658;
        double j = -4.0386e-05;
        return a + b * log(x) + c * pow(log(x), 2) + d *
pow(log(x), 3) + e * pow(log(x), 4) + f * pow(log(x), 5) + g *
pow(log(x), 6) +
          h * pow(log(x), 7) + i * pow(log(x), 8) + j * pow(log(x),
9);
}
// calculate up estim max
double calc_up_max(int n) {
        double max = 0.0, cur = 0.0;
        for (int i = 0; i < n; i++) {
          cur = get_up_approx(i + 2);
          if (cur > max)
               max = cur;
        }
        return max;
}

//get down estim
double get_down_approx(int x) {
        double a= 1.237982533;
        double b = -0.00013423;
```

Продовження лістингу Ж.1

```
        double c = 0.160477316;
        double d = 3.59863e-07;
        double e = -0.95187466;
        double f = -3.7607e-10;
        double g = 0.553437737;
        double h = 1.33924e-13;
        return a + b * x + c / x + d * x * x + e / pow(x,2) + f *
pow(x,3) + g / pow(x, 3) + h * pow(x,4);
}
// calculate down estim min
double calc_down_min(int n) {
        double min = 1000.0, cur = 0.0;
        for (int i = 0; i < n; i++) {
          cur = get_down_approx(i + 2);
          if (cur < min)
               min = cur;
        }
        return min;
}

//calculate t-criterion
double get_t(double* exp_down, double* exp_up, int n) {
        FILE*                          fp                          =
fopen("D:\\me\\DocDis\\exp\\algos\\t_calc_res.txt", "w");
        //bottom estim
        double avg_approx = 0.0, avg = 0.0, t;
        double disp = 0.0, disp_approx = 0.0;
        // up estim
        double avg_approx_up = 0.0, avg_up = 0.0, t_up;
        double disp_up = 0.0, disp_approx_up = 0.0;
        // measures number
        double x = (double)n + 1.0;
        // calc approx avg
        for (int i = 0; i <= n; i++) {
          avg_approx += get_down_approx(i + 1);
          avg_approx_up += get_up_approx(i + 1);
        }
        avg_approx /= x;
        avg_approx_up /= x;
        // calc exp avg
        for (int i = 0; i < n; i++) {
          avg += exp_down[i];
          avg_up += exp_up[i];
        }
        avg += 1.0;
```

Продовження лістингу Ж.1

```
        avg_up += 1.0;
        avg /= x;
        avg_up /= x;
        // calc dispersion
        disp += pow(1.0 - avg, 2); // popravka na + 1
        disp_up += pow(1.0 - avg_up, 2); // popravka na + 1
        for (int i = 0; i < n; i++) {
          disp += pow(exp_down[i] - avg, 2);
          disp_up += pow(exp_up[i] - avg_up, 2);
        }
        disp /= x;
        disp_up /= x;

        // calc approx dispersion
        for (int i = 0; i <= n; i++) {
          disp_approx += pow(get_down_approx(i + 1) - avg_approx,
2);
          disp_approx_up    +=    pow(get_up_approx(i    +    1)    -
avg_approx_up, 2);
        }
        disp_approx /= x;
        disp_approx_up /= x;

        // calc t criterion
        t = abs(avg - avg_approx);
        t_up = abs(avg_up - avg_approx_up);
        t /= sqrt((disp + disp_approx)/n);
        t_up /= sqrt((disp_up + disp_approx_up) / n);
        // show bottom characteristics
        cout << "bottom stat data:\n";
        cout  <<"bottom  avg:"<<  avg  <<  "\t  avg  approx:  "  <<
avg_approx << endl;
        cout << "bottom disp: "<< disp << "\t disp approx: " <<
disp_approx << "\t bottom t-criterion: " << t << endl;
        cout << "\n\n";
        // show up characteristics
        cout << "up stat data:\n";
        cout  <<  "up  avg:"  <<  avg_up  <<  "\t  avg  approx:  "  <<
avg_approx_up << endl;
        cout << "up disp: " << disp_up << "\t disp approx: " <<
disp_approx_up << "\t up t-criterion: " << t_up << endl;
        //write to file
        cout << "writing to file <t_calc_res.txt>...\n";
        //write bottom estim
        fprintf(fp, "bottom estim (lower bound):\n");
```

Продовження лістингу Ж.1

```
        fprintf(fp, "bottom avg: %.9lf\t avg approx: %.9lf\n", avg,
avg_approx);
        fprintf(fp, "bottom disp: %.9lf\t disp approx: %.9lf\t
bottom t-criterion: %.9lf\n", disp, disp_approx, t);
        //write up estim
        fprintf(fp, "up estim (upper bound):\n");
        fprintf(fp, "up avg: %.9lf\t avg approx: %.9lf\n", avg_up,
avg_approx_up);
        fprintf(fp, "up disp: %.9lf\t disp approx: %.9lf\t up t-
criterion: %.9lf\n", disp_up, disp_approx_up, t_up);
        // close file
        fclose(fp);
        return t;
}

int main()
{
        FILE* fp1, * fp2;
        const int n = 1000; // number of measures
        const double up_bfs_avg = 0.893; //sec
        const double down_bfs_avg = 1.001; //sec
        double  down_min=1000.0,  up_max=0.0;  //  to  build  estim
intervals
        int x, y;
        int z; // to extrapolate
        double acc = 0, acc2 = 0; // to accumulate
        double* up = new double[n];
        double* down = new double[n];
        double* up_res = new double[n]; up_res[0] = 1;
        double* down_res = new double[n]; down_res[0] = 1;
        fp1                                                          =
fopen("D:\\me\\DocDis\\exp\\algos\\DFS_res_1000_up.txt","r");
        fp2                                                          =
fopen("D:\\me\\DocDis\\exp\\algos\\DFS_res_1000_down.txt", "r");
        if (fp1 == NULL || fp2 == NULL)
          return -1;
        // fill input data for best case
        int i = 0, size = n;
        while (size--) {
          fscanf(fp1, "%d", &x);
          fscanf(fp2, "%d", &y);
          up[i] = double(x) / double(n);
          down[i] = double(y) / double(n);
          up_res[i] = 0;
```

Продовження лістингу Ж.1

```
          down_res[i] = 0;
          i++;
        }
        fclose(fp1);
        fclose(fp2);
        // accumulate denominator
        for (int i = 0; i < n; i++) {
          acc += up[i];
          acc2 += down[i];
          up_res[i] = acc;
          down_res[i] = acc2;
        }
        // calculate ksi_1, ksi_2
        for (int i = 0; i < n; i++) {
          up_res[i] = (double(i) + 2.0) * up_bfs_avg / (up_res[i] +
up_bfs_avg);   // ksi_1
          // find up_max - experimental
          if (up_res[i] > up_max)
               up_max = up_res[i];
          down_res[i]  =  (double(i)  +  2.0)  *  down_bfs_avg  /
(down_res[i] + down_bfs_avg); // ksi_2
          // find down_min - experimental
          //if (down_res[i] < down_min)
               down_min = down_res[i];
        }
        // write the result to "DFS_res_1000_up_res.txt" file
        fp1                                                          =
fopen("D:\\me\\DocDis\\exp\\algos\\DFS_res_1000_up_res.txt", "w");
        fp2                                                          =
fopen("D:\\me\\DocDis\\exp\\algos\\DFS_res_1000_down_res.txt",
"w");
        fputs("1,000\n", fp1);
        fputs("1,000\n", fp2);
        for (int i = 0; i < n; i++) {
          fprintf(fp1, "%.3f\n", up_res[i]);
          fprintf(fp2, "%.3f\n", down_res[i]);
        }
        fclose(fp1);
        fclose(fp2);
        // fill for statistics
        fp1                                                          =
fopen("D:\\me\\DocDis\\exp\\algos\\DFS_res_1000_up_res_stat.txt",
"w");
```

Продовження лістингу Ж.1

```
        fp2                                                    =
fopen("D:\\me\\DocDis\\exp\\algos\\DFS_res_1000_down_res_stat.txt",
"w");
        fprintf(fp1,"%d\t%.3f\n", 1, 1.0f);
        fprintf(fp2,"%d\t%.3f\n", 1, 1.0f);
        for (int i = 0; i < n; i++) {
          fprintf(fp1, "%d\t%.3f\n", i+2, up_res[i]);
          fprintf(fp2, "%d\t%.3f\n", i+2, down_res[i]);
        }
        //experimental min-max
        printf("exp.  min:  %lf\t  exp.  max:  %lf\n",  down_min,
up_max);
        //approx min-max
        printf("approx.   min:   %lf\t   approx.   max:   %lf\n",
calc_down_min(n), calc_up_max(n));

        fclose(fp1);
        fclose(fp2);
        // calc approx
        get_t(down_res, up_res, n);
        // extrapolate
        cout << "enter x...\n";
        cin >> z;
        cout << get_up_approx(z)<<endl;

        cout << "done!\n";
        delete[] up;
        delete[] down;
        delete[] up_res;
        delete[] down_res;
        return 0;
}
```

ДОДАТОК З
ДОКУМЕНТАЛЬНІ ПІДТВЕРДЖЕННЯ

АКТ ВПРОВАДЖЕННЯ У РОБОЧИЙ ПРОЦЕС

ЗАТВЕРДЖУЮ
Генеральний директор
ТОВ «НВП «ХАРТРОН-ЮКОМ»
_____ Романовський О.В.
підпис ПІБ
_____ 2021 р.

АКТ
про впровадження (використання)
результатів дисертаційної роботи
Шкарупила Вадима Вікторовича
на здобуття наукового ступеня доктора технічних наук
за спеціальністю 05.13.05 – комп'ютерні системи та компоненти

Комісія у складі:
Голова комісії головний конструктор Сфименко М.В.
посада, ПІБ
Члени комісії начальник лабораторії Фіногенов Ю.М.
посада, ПІБ
начальник групи Подмастер'єв С.В.
посада, ПІБ

встановила впровадження у робочий процес ТОВ «НВП «ХАРТРОН-ЮКОМ» результатів дисертаційної роботи Шкарупила В.В., присвяченої розробленню методів та засобів контролю артефактів процесу проєктування програмно-алгоритмічного забезпечення підсистеми керування орієнтацією космічного апарату.

У робочому процесі ТОВ «НВП «ХАРТРОН-ЮКОМ» використано такі результати дисертаційної роботи: метод автоматизованого синтезу формальних специфікацій одержуваних артефактів процесу проєктування (блок-схем алгоритмів) та відповідна модель; розвиток методу формальної верифікації TLC – у якості засобу контролю функціональних характеристик розроблюваної системи при проєктуванні; метод та відповідна модель – у якості засобів контролю нефункціональних характеристик розроблюваної системи при проєктуванні.

Голова комісії _____
Члени комісії _____
_____
_____

ЛИСТ ПІДТРИМКИ ВІД ДЦКЗ ДЕРЖСПЕЦЗВ'ЯЗКУ

Прим. № 1

ДЕРЖСПЕЦЗВ'ЯЗКУ

ДЕРЖАВНИЙ ЦЕНТР КІБЕРЗАХИСТУ ДЕРЖАВНОЇ СЛУЖБИ СПЕЦІАЛЬНОГО ЗВ'ЯЗКУ ТА ЗАХИСТУ ІНФОРМАЦІЇ УКРАЇНИ
(ДЦКЗ Держспецзв'язку)

вул. Юрія Іллєнка, 83-Б, м. Київ, 04119, Україна, тел.: (044) 281-88-01,
e-mail: info@scpc.gov.ua, код згідно з ЄДРПОУ 36947982

02.11.2022 № 64/04-1404 На № _____ від _____

Заступнику директора з наукової роботи, д.т.н., старшому науковому співробітнику
ІПМЕ ім. Г.Є. Пухова НАН України

Олександру ЧЕМЕРИСУ

вул. Генерала Наумова, 15, м. Київ, 03164
pme@ipme.kiev.ua

Щодо НДР «Артефакт»

Шановний пане Олександре!

Фахівцями Центру в рамках засідання Науково-технічного семінару ДЦКЗ Держспецзв'язку (протокол від 03.11.2022 № 64/4-1392) розглянуто складові НДР «Розроблення методів та засобів верифікації артефактів процесу проєктування систем критичного призначення» (шифр: «Артефакт»), реєстраційний № 0121U110615 (науковий керівник:к.т.н., доц. Шкарупило В.В.), що виконувалась за Програмно-цільовою та конкурсною тематикою НАН України – Гранти НАН України дослідницьким лабораторіям/групам молодих вчених НАН України для проведення досліджень за пріоритетними напрямами розвитку науки і техніки, у 2021-2022 рр. Проведений розгляд показав наукову та практичну значимість отриманих результатів.

Внаслідок виконання названої НДР отримано низку важливих науково-прикладних результатів, призначених до залучення на етапі проєктування процесу розроблення програмно-алгоритмічної складової систем критичного призначення, що знаходять застосування, зокрема, на ринку електричної енергії, в аерокосмічній, енергетичній галузях тощо.

Важливою складовою проведеної роботи є розроблені нові методи і засоби контролю показників функціональних і нефункціональних характеристик

програмно-алгоритмічної складової системи критичного призначення вже на етапі проєктування процесу розроблення, зокрема, систем кіберзахисту, а також комплексний підхід до прикладного застосування названих методів і засобів.

У контексті актуальних подій, зовнішньої агресії по відношенню до нашої держави, досліджено також і процеси проєктування систем кіберзахисту та безпеки інформації, визначено артефакти кіберзахисту та підходи до їх верифікації як складових забезпечення функційної безпечності систем критичного призначення, що особливо актуально в умовах воєнного стану і післявоєнного відновлення України.

Прикладне застосування отриманих результатів виконання НДР дозволяє сприяти досягненню заданого рівня функційної безпечності розроблюваної програмно-алгоритмічної складової системи критичного призначення вже на етапі проєктування процесу розроблення, у відповідності до положень актуального стандарту ДСТУ EN 61508-1:2019 «Функційна безпечність електричних, електронних, програмованих електронних систем, пов'язаних із безпекою», що підтверджує практичну значимість роботи та її актуальність.

З повагою

Заступник начальника центру Олександр ПЕТРУШКЕВИЧ

Андрій КОВАЛЕНКО, 281 88 30

ВПРОВАДЖЕННЯ У НАВЧАЛЬНИЙ ПРОЦЕС ІЕЕ НТУУ «КПІ ІМ. ІГОРЯ СІКОРСЬКОГО»

Академіку-секретарю
відділення фізико-технічних проблем енергетики
НАН України
Кириленко О.В.

Щодо використання
науково-практичних результатів
в навчальному процесі

Шановний Олександре Васильовичу!

В межах програмно-цільової та конкурсної тематики НАН України (гранти НАН України дослідницьким лабораторіям/групам молодих вчених НАН України) в Інституті проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, із залученням фахівців Інституту електродинаміки НАН України, протягом 2021-2022 років виконувався науковий проєкт «Розроблення методів та засобів верифікації артефактів процесу проєктування систем критичного призначення». У ході реалізації проєкту виконавцями було отримано низку важливих практичних результатів щодо забезпечення заданого рівня функційної безпечності розроблюваної програмно-алгоритмічної складової систем критичного призначення вже на етапі проєктування процесу їх розроблення.

Важливою складовою роботи визначення області застосування розроблених методіт та моделей, зокрема і для забезпечення функціонування систем керування окремими сегментами ринку електричної енергії України, що в сучасних умовах часто потребують зміни вимог до правил функціонування ринку, міждержавної торгівлі електричною енергією та розподілу пропускної спроможності згідно вимог ENTSO-E, підвищення надійності роботи ОЕС України та внесення відповідних змін до алгоритмів роботи відповідних інформаційно-технологічних систем. Визначення складових таких систем та підвищення їх функційної безпечності особливо актуально для об'єктів критичної інфраструктури, робота яких керується Оператором системи передачі.

Результати роботи використані у робочій програмі навчальної дисципліни підготовки другого магістерського рівня «Системи ринків електричної енергії» в якості окремих тем або їх складових за спеціальністю 141 «Електроенергетика, електротехніка та електромеханіка» освітньої програми «Енергетичний менеджмент, електропостачання та інжиніринг електротехнічних комплексів» у 2022/2023 навчальному році.

З повагою,
Директор
ІЕЕ НТУУ «КПІ ім. Ігоря Сікорського»
д.т.н., професор **Сергій ДЕНИСЮК**

ВПРОВАДЖЕННЯ У НАВЧАЛЬНИЙ ПРОЦЕС ФАКУЛЬТЕТУ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ НУБІП УКРАЇНИ

Академіку-секретарю
Відділення фізико-технічних проблем енергетики
НАН України
О.В. Кириленко

Щодо використання
науково-практичних результатів
у навчальному процесі

Шановний Олександре Васильовичу!

У межах програмно-цільової та конкурсної тематики НАН України – Гранти НАН України дослідницьким лабораторіям/групам молодих вчених НАН України для проведення досліджень за пріоритетними напрямами розвитку науки і техніки, у 2021-2022 рр. виконувалась НДР «Розроблення методів та засобів верифікації артефактів процесу проєктування систем критичного призначення» (шифр: Артефакт), реєстраційний № 0121U110615 (науковий керівник: к.т.н., доц. Шкарупило В.В.).

Актуальність та практична значимість результатів, отриманих дослідницькою лабораторією, не викликають сумнівів, особливо за умов воєнного стану в Державі, з акцентом на післявоєнне відновлення України. У даному контексті напрацювання у напрямі розвитку методів та засобів формальної верифікації, зокрема у розрізі сприяння кібербезпеці об'єктів критичного призначення, вбачаються особливо вагомими. У зв'язку із цим, названі теоретико-практичні напрацювання було введено у якості складової навчального процесу, здійснюваного згідно робочих програм наступних навчальних дисциплін: «Технології безпечного програмування», що викладається за спеціальністю 125 «Кібербезпека» освітнього ступеня «Бакалавр»; «Системне програмування», що викладається за спеціальностями 123 «Комп'ютерна інженерія» і 125 «Кібербезпека» освітнього ступеня «Бакалавр».

З повагою,

завідувач кафедри комп'ютерних систем,
мереж та кібербезпеки
факультету інформаційних технологій
Національного університету біоресурсів
і природокористування України,
академік ГО "Національна Академія
наук вищої освіти України" Дмитро КАСАТКІН

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

підпис засвідчую — (Глушкова А.С.)

ЛИСТ ПІДТРИМКИ ВІД УКРАЇНСЬКОГО НАЦІОНАЛЬНОГО КОМІТЕТУ CIGRE В УКРАЇНІ

Громадська спілка «Міжнародна рада з великих електроенергетичних систем СІГРЕ в Україні»
Український Національний Комітет
Міжнародної ради з великих електроенергетичних систем
Тел./факс 38 (044) 456 24 69

cigre Ukraine
cigre.ukraine@ukr.net
www.cigre.org.ua

Public Association «International Council on Large Electric Systems CIGRE in Ukraine»
Ukrainian National Committee
International Council on Large Electric Systems
Tel./fax 38 (044) 456 24 69

01-100 від 09.11.22

Академіку-секретарю
відділення фізико-технічних проблем енергетики
НАН України
Кириленко О.В.

Щодо підтримки
наукового проєкту

В Інституті проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, із залученням фахівців Інституту електродинаміки НАН України, протягом 2021-2022 років виконувався науковий проєкт «Розроблення методів та засобів верифікації артефактів процесу проєктування систем критичного призначення». За результатами реалізації проєкту було отримано важливі практичні результати щодо забезпечення заданого рівня функційної безпечності розроблюваної програмно-алгоритмічної складової систем критичного призначення під час проєктування процесу їх розроблення.

Запропоновані методи та засоби контролю показників як функціональних, зокрема несуперечності, так і нефункціональних характеристик (часових затримок) програмно-алгоритмічних складових систем критичного призначення на етапі проєктування процесу розроблення таких систем Розроблені методи та засоби є важливими в умовах необхідності постійної зміни вимог до правил функціонування ринку електричної енергії, міждержавної торгівлі електричною енергією та розподілу пропускної спроможності згідно вимог ENTSO-E, підвищення надійності роботи ОЕС України та внесення відповідних змін до алгоритмів роботи відповідних інформаційно-технологічних систем. Крім того, в межах проєкту запропоновано підходи до верифікації артефактів кіберзахисту систем критичного призначення, до яких, безумовно, відносяться інформаційно-технологічні системи операторів систем розподілу та оператора системи передачі, виробників електричної енергії, зокрема і АЕС.

Впровадження та використання запропонованих рішень дозволить підвищити функційну безпечність розроблюваних систем критичного призначення, що використовуються в ОЕС України в частині їх програмної складової ще на етапі їх проєктування.

З повагою,
Президент
Українського національного
комітету CIGRE

Олександр СВЕТЕЛІК

ЛИСТ ПІДТРИМКИ ВІД ДП «ДЕРЖАВНИЙ НАУКОВО-ТЕХНІЧНИЙ ЦЕНТР З ЯДЕРНОЇ ТА РАДІАЦІЙНОЇ БЕЗПЕКИ»

SSTC NRS | ДЕРЖАВНЕ ПІДПРИЄМСТВО ДЕРЖАВНИЙ НАУКОВО-ТЕХНІЧНИЙ ЦЕНТР З ЯДЕРНОЇ ТА РАДІАЦІЙНОЇ БЕЗПЕКИ

ДЕРЖАВНА ІНСПЕКЦІЯ ЯДЕРНОГО РЕГУЛЮВАННЯ УКРАЇНИ

НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ

Державне підприємство
"Державний науково-технічний центр з ядерної та радіаційної безпеки"
Вул. Василя Стуса, 35-37
м. Київ, 03142, Україна, а/с 124
тел.: (044) 450-05-00, факс: (044) 452-89-90
e-mail: nrs@sstc.ua
www.sstc.ua

Заступнику директора
з наукової роботи
Інституту проблем моделювання
в енергетиці ім. Г.Є. Пухова
Національної академії наук України,
д.т.н., с.н.с.
Олександру Чемерису

Україна, 03164, Київ-164,
вул. Генерала Наумова, 15

Щодо результатів наукового проєкту

Шановний Олександре Анатолійовичу!

За результатами розгляду наукового проєкту «Розроблення методів та засобів верифікації артефактів процесу проєктування систем критичного призначення» (шифр: Артефакт, реєстраційний номер: 0121U110615, науковий керівник роботи: к.т.н., доцент, с.н.с. ІПМЕ ім. Г.Є. Пухова НАН України Шкарупило В.В.), що виконувалась за Програмно-цільовою та конкурсною тематикою Національної академії наук України (далі – НАН України) відповідно до розпорядження Президії НАН України від 29.03.2021 № 181 «Про результати конкурсу 2020 р. на здобуття грантів НАН України дослідницькими лабораторіями/групами молодих вчених НАН України для проведення досліджень за пріоритетними напрямами розвитку науки і техніки», враховуючи актуальність, новизну, обґрунтованість висновків, а також практичне значення результатів проведеної роботи, повідомляємо, що за підсумками реалізації проєкту досягнуто низку важливих науково-практичних результатів, що сприяло підвищенню ефективності процесу проєктування програмно-алгоритмічної складової систем критичного призначення з позиції рівня функційної безпечності за рахунок розроблення відповідних методів та засобів. Окремо варто відзначити розвиток поширеного методу формальної верифікації TLC з позиції зниження, супутніх його застосуванню, часових витрат за ітеративного підходу до організації процесу проєктування.

З повагою

Директор Ігор Шевченко

Дмитро Чорний
422 49 62

ДОДАТОК К
СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ ТА ВІДОМОСТІ ПРО АПРОБАЦІЮ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЇ

Наукові праці, у яких опубліковано основні наукові результати.

Монографії (6 праць), з яких 2 – опубліковані у закордонних виданнях, серед яких 1 – з індексацією у міжнародній наукометричній базі Scopus:

1. Shkarupylo V.V., Timenko A.V. On the interoperability and consistency aspects with respect to the Internet of Things domain. Engineering sciences: development prospects in countries of Europe at the beginning of the third millennium: Collective monograph. Vol. 2. Riga: Izdevnieciba “Baltija Publishing”, 2018. P. 466–485. ISBN 978-9934-571-63-3 (**розділ колективної монографії**)

2. Блінов І.В., Парус Є.В., Шкарупило В.В. Структура та моделі інформаційної взаємодії учасників ринку електричної енергії: монографія. Вінниця: ГО «Європейська наукова платформа», 2021. 114 с. ISBN 978-617-8037-31-4. DOI: https://doi.org/10.36074/stmivyree-monograph.2021 (**колективна монографія**)

3. Шкарупило В.В., Блінов І.В. Сценарії, методи та засоби формальної верифікації артефактів процесу проєктування систем критичного призначення: монографія. Вінниця : ГО «Європейська наукова платформа», 2021. 104 с. ISBN 978-617-8037-55-0. DOI: https://doi.org/10.36074/smtzfvappskp-monograph.2021 (**колективна монографія**)

4. Shkarupylo V.V., Blinov I.V., Chemeris A.A., Dusheba V.V., Alsayaydeh J.A.J. On Applicability of Model Checking Technique in Power Systems and Electric Power Industry. In: Zaporozhets A. (eds) Systems, Decision and Control in Energy III. Studies in Systems, Decision and Control, 2022, Vol. 399. Springer, Cham. ISBN 978-3-030-87675-3. DOI: https://doi.org/10.1007/978-3-030-87675-3_1

(**Scopus:** https://www.scopus.com/record/display.uri?eid=2-s2.0-85120868312&origin=resultslist&sort=plf-f **; розділ колективної монографії**)

5. Борукаєв З.Х., Блінов І.В., Остапченко К.Б., Чемерис О.А., Шкарупило В.В. Моделі та засоби автоматизації систем організаційного управління енергоринком: монографія / за заг. ред. З.Х. Борукаєва. — Вінниця: ГО «Європейська наукова платформа», 2022. 122 с. ISBN 978-617-8037-82-6 DOI: https://doi.org/10.36074/mtzasoye-monograph.2022 (**колективна монографія**)

6. Шкарупило В., Блінов І., Кучанський В., Давидюк А., Дімітрієва Д. Методи і засоби контролю артефактів процесу проєктування програмно-алгоритмічної складової систем критичного призначення: монографія / за заг. ред. В.В. Шкарупила. Publishing House «European Scientific Platform», 2023. 120 с. ISBN 978-617-8126-22-3 DOI: https://doi.org/10.36074/mzkapppasskp-monograph.2023 (**колективна монографія**)

Статті у фахових періодичних виданнях (22 праці), серед яких 7 – у виданнях, що індексуються у міжнародних наукометричних базах Scopus та Web of Science Core Collection, 3 – у фахових виданнях категорії А:

7. Shkarupylo V.V., Tomičić I., Kasian K.M. The investigation of TLC model checker properties. *Journal of Information and Organizational Sciences*, 2016. Vol. 40, No. 1. P. 145–152. ISSN: 1846-9418 (Online), 1846-3312 (Print). DOI: https://doi.org/10.31341/jios.40.1.7 (**Web of Science Core Collection:** https://www.webofscience.com/wos/woscc/full-record/WOS:000409240900008 ; **Scopus, Q4:** https://www.scopus.com/record/display.uri?eid=2-s2.0-84975057117&origin=resultslist)

8. Shkarupylo V., Skrupsky S., Oliinyk A., Kolpakova T. Development of stratified approach to software defined networks simulation. *Eastern-European Journal of Enterprise Technologies. Information and controlling systems*, 2017. Vol. 5, No. 9 (89). P. 67–73. ISSN: 1729-3774 (Print), 1729-4061 (Online). DOI:

https://doi.org/10.15587/1729-4061.2017.110142 (**Scopus, Q3**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85031750626&origin=resultslist; **фахове видання категорії А**)

9. Alsayaydeh J.A.J., Shkarupylo V., Hamid M. S. B., Skrupsky S., Oliinyk A. Stratified model of the Internet of Things infrastructure. *Journal of Engineering and Applied Sciences*, 2018. Vol. 13, No. 20. P. 8634–8638. ISSN: 1816-949x (Print), 1818-7803 (Online). DOI: https://medwelljournals.com/abstract/?doi=jeasci.2018.8634.8638 (**Scopus, Q3**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85056326734&origin=resultslist)

10. Timenko A.V., Shkarupylo V.V., Oliinyk A.O., Hrushko S.S. Formal Model for Checking the Interoperability Between the Components of the IoT system. *Problemele Energeticii Regionale*, 2019. Vol. 40, No. 1-1. P. 69–78. ISSN: 1857-0070. DOI: https://zenodo.org/record/3239196 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000472596400007)

11. Shkarupylo V., Alsayaydeh J.A.J, Tomičić I., Chemeris A., Dusheba V. A technique for checking the adequacy of formal model. *ARPN Journal of Engineering and Applied Sciences*, August 2021. Vol. 16, No. 16. P. 1707–1719. ISSN: 1819-6608. URL: http://www.arpnjournals.org/jeas/research_papers/rp_2021/jeas_0821_8670.pdf (дата звернення: 06.08.2023) (**Scopus, Q3**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85118181893&origin=resultslist)

12. Polska O.V., Kudermetov R.K., Shkarupylo V.V. An approach web service selection by quality criteria based on sensitivity analysis of MCDM methods. *Radio Electronics, Computer Science, Control*, 2021. No. 2. P. 133–143. ISSN: 1607-3274 (Online), 2313-688X (Print). DOI: https://doi.org/10.15588/1607-3274-2021-2-14 (**Web of Science Core Collection:**

https://www.webofscience.com/wos/woscc/full-record/WOS:000673377700014 **; фахове видання категорії А**)

13. Shkarupylo V., Blinov I., Dusheba V., Alsayaydeh J. A. J. Case Driven TLC Model Checker Analysis in Energy Scenario. *CEUR Workshop Proceedings*, 2023. Vol. 3392. P. 65–75. ISSN 1613-0073. DOI: https://doi.org/10.32782/cmis/3392-6 (**Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85160296577&origin=resultslist&sort=plf-f)

14. Шкарупило В.В., Кудерметов Р.К., Польська О.В. DEVS-орієнтована методика валідації композитних веб-сервісів. *Радіоелектроніка, інформатика, управління*, 2015. № 4. С. 79–86. ISSN: 1607-3274 (Print), 2313-688X (Online). DOI: 10.15588/1607-3274-2015-4-12 URL: http://ric.zntu.edu.ua/article/view/60404 (дата звернення: 06.08.2023) (**фахове видання категорії А**)

15. Polska O.V., Kudermetov R.K., Shkarupylo.V.V. Discovery and selection of web-services. *Electrotechnic and computer systems*, 2015. No. 19(95). P. 169–173. ISSN: 2221-3937 (Print), 2221-3805 (Online). URL: http://nbuv.gov.ua/UJRN/etks_2015_19_39 (**фахове видання**)

16. Kudermetov R., Polska O., Shkarupylo V., Shcherbak N. Quality of services in scientific workflows. *Electrotechnic and Computer Systems*, 2018. Vol. 28, No. 104. P. 170–177. ISSN: 2221-3805 (Print), 2221-3937 (Online). DOI: 10.15276/eltecs.28.104.2018.20 URL: https://eltecs.op.edu.ua/index.php/journal/article/view/155/42 (**фахове видання**)

17. Shkarupylo V.V., Tomičić I., Kasian K.M., Alsayaydeh J.A.J. An Approach to increase the Effectiveness of TLC Verification with Respect to the Concurrent Structure of TLA+ Specification. *International Journal of Software Engineering and Computer Systems*, 2018. Vol. 4, No. 1. P. 48–60. ISSN: 2289-8522. DOI: https://doi.org/10.15282/ijsecs.4.1.2018.4.0037 (**фахове видання**)

18. Shkarupylo V.V., Kudermetov R.K., Polska O.V. On the approaches to cyber-physical systems simulation. *Advances in Cyber-Physical Systems (ACPS)*, 2018. Vol. 3, No. 1. P. 51–54. ISSN: 2524-0382 (Print), 2707-0069 (Online). DOI: https://doi.org/10.23939/acps2018.01.051 (**фахове видання**)

19. Шкарупило В.В., Євдокимов В.Ф., Душеба В.В. Застосування формальних методів для перевірки систем критичного призначення. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2019. Том 30 (69), Ч. 1, № 6. С. 188–193. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI https://doi.org/10.32838/2663-5941/2019.6-1/34 (**фахове видання**)

20. Тіменко А.В., Шкарупило В.В., Скрупський С.Ю., Смолій В.В. Дослідження шляхів підвищення пропускної спроможності підсистеми пам'яті сучасної обчислювальної системи. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2020. Том 31 (70), Ч. 1, № 2. С. 208–212. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/2663-5941/2020.2-1/32 (**фахове видання**)

21. Шкарупило В.В., Чемерис О.А., Душеба В.В., Кудерметов Р.К., Польська О.В. Метод синтезу формальних специфікацій на основі трійок Хоара. *Наукові праці ДонНТУ, Серія "Інформатика, кібернетика та обчислювальна техніка"*, 2020. № 1(30). С. 49–57. ISSN: 1996-1588. DOI: 10.31474/1996-1588-2020-1-30-49-57 (**фахове видання**)

22. Шкарупило В.В., Чемерис О.А., Душеба В.В. Оцінювання просторової складності задачі формальної верифікації, вирішуваної методом перевірки на моделі. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2020. Том 31 (70), № 5. С. 147–151. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/2663-5941/2020.5/24 (**фахове видання**)

23. Шкарупило В.В., Чемерис О.А., Душеба В.В., Кудерметов Р.К. Дослідження мультипоточної реалізації методу перевірки на моделі для темпоральної логіки дій. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2020. Том 31 (70), № 6, Ч. 1. С. 173–177. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/TNU-2663-5941/2020.6-1/28 (**фахове видання**)

24. Polska O.V., Kudermetov R.K., Shkarupylo V.V. The approach for QoS based web service selection with user's preferences. *Наукові праці Донецького національного технічного університету, серія: «Проблеми моделювання та автоматизації проектування»*, 2020. №2 (16). С. 19–27. ISSN: 2074-7888. DOI: 10.31474/2074-7888-2020-2-19-27 URL: http://pmap.donntu.edu.ua/sites/upload/articles/pmap_2020_19-27.pdf (дата звернення: 06.08.2023) (**фахове видання**)

25. Polska O.V., Kudermetov R.K., Zolotukhina O.A., Shkarupylo V.V. A UML profile for quality-based web service selection using logic scoring of preference method. *Telecommunication and information technologies*, 2021. No. 1 (2021). P. 65–78. ISSN: 2412-4338. DOI: https://doi.org/10.31673/2412-4338.2021.016578 (**фахове видання**)

26. Шкарупило В.В., Чемерис О.А., Душеба В.В., Кудерметов Р.К., Польська О.В. Модельно-орієнтований підхід до контролю показників нефункціональних характеристик під час проєктування. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2021. Том 32 (71), Ч. 1, № 1. С. 166–171. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32838/2663-5941/2021.1-1/27 (**фахове видання**)

27. Шкарупило В.В., Душеба В.В., Скрупський С.Ю., Блінов І.В. Стратифікована модель подання нефункціональних характеристик системи критичного призначення при проєктуванні. *Електронне моделювання,* 2022.

Т. 44, № 2. С. 90–106. ISSN: 0204-3572. DOI: https://doi.org/10.15407/emodel.44.02.090 (**фахове видання**)

28. Куликовська Н.А., Руденко В.В., Тіменко А.В., Шкарупило В.В. Дослідження часу збирання додатків, побудованих на основі сучасних стратегій розроблення. *Вчені записки Таврійського національного університету імені В.І.Вернадського, серія «Технічні науки»*, 2023. Том 34 (73), № 4. С. 65–70. ISSN: 2663-5941 (Print), 2663-595X (Online). DOI: https://doi.org/10.32782/2663-5941/2023.4/11 (**фахове видання**)

Праці апробаційного характеру (40 праць), серед яких 7 – з індексацією у міжнародних наукометричних базах Scopus та Web of Science Core Collection:

29. Shkarupylo V. A Technique of DEVS-Driven Validation. *Proc. XIIIth Int. Conf. on Modern Problems of Radio Engineering, Telecommunications, and Computer Science, TCSET'2016* (Lviv-Slavske, Ukraine, February 23–26, 2016). P. 495–497. DOI: https://doi.org/10.1109/TCSET.2016.7452097 (**Web of Science Core Collection:** https://www.webofscience.com/wos/woscc/full-record/WOS:000381804300127 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-84969263650&origin=resultslist)

30. Shkarupylo V. A Simulation-driven Approach for Composite Web Services Validation. *Proc. 27th Int. Central European Conference on Information and Intelligent Systems, CECIIS 2016* (Varazdin, Croatia, September 21–23, 2016). P. 227–231. URL: http://archive.ceciis.foi.hr/app/public/conferences/1/ceciis2016/papers/QoS-1.pdf (дата звернення: 06.08.2023) (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000595003500030)

31. Shkarupylo V., Polska O. The Approach to SDN Network Topology Verification on a Basis of Temporal Logic of Actions. *Proc. 14th Int. Conf. on*

Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET'2018 (Lviv-Slavske, Ukraine, February 20–24, 2018). P. 183–186. DOI: https://doi.org/10.1109/TCSET.2018.8336182 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000465121700033 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85047524592&origin=resultslist)

32. Shkarupylo V., Kudermetov R., Golub T., Polska O., Tiahunova M. Towards Model Checking of the Internet of Things Solutions Interoperability. *Problems of Infocommunications. Science and Technology: proc. 2018 IEEE International Scientific and Practical Conference, PIC S&T-2018* (Kharkiv, Ukraine, October 9–12, 2018). P. 465–468. DOI: https://doi.org/10.1109/INFOCOMMST.2018.8632037 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000458659100087 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85062879597&origin=resultslist)

33. Shkarupylo V., Kudermetov R., Timenko A., Polska O. On the Aspects of IoT Protocols Specification and Verification. *Problems of Infocommunications. Science and Technology: 2019 International Scientific-Practical Conference, PIC S&T'2019* (Kyiv, Ukraine, October 8–11, 2019). P. 93–96. DOI: https://doi.org/10.1109/PICST47496.2019.9061406 (**Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85083637232&origin=resultslist)

34. Shkarupylo V., Chemerys O., Dusheba V., Kudermetov R., Oliinyk A. On Hoare triples applicability to dependable system specification synthesis. *Dependable Systems, Services and Technologies, DESSERT'2020: The 11th International Conference* (Kyiv, Ukraine, May 14–18, 2020). Kyiv, 2020. P. 371–

375. DOI: https://doi.org/10.1109/DESSERT50317.2020.9125074 (**Web of Science Core Collection**: https://www.webofscience.com/wos/woscc/full-record/WOS:000619228000064 ; **Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85087906543&origin=resultslist)

35. Shkarupylo V., Blinov I., Chemeris A., Dusheba V., Alsayaydeh J., Oliinyk A. Iterative Approach to TLC Model Checker Application. *Proc. 2021 IEEE KhPI Week on Advanced Technology* (Kharkiv, Ukraine, September 13–17, 2021). P. 283–287. DOI: https://doi.org/10.1109/KhPIWeek53812.2021.9570055 (**Scopus**: https://www.scopus.com/record/display.uri?eid=2-s2.0-85118943601&origin=resultslist)

36. Шкарупило В.В. Концепція формальної верифікації UML-діаграм методами Model Checking. *Моделювання: XXXIV науково-технічна конференція*, 13–14 січня 2015 р.: тези доп. К.: ІПМЕ ім. Г. Є. Пухова НАН України, 2015. С. 13.

37. Шкарупило В.В. Особливості використання методу формальної верифікації TLC. *Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*: тези доп., м. Київ, 12 січня 2016 р. С. 31. DOI: http://dx.doi.org/10.5281/zenodo.2545399

38. Shkarupylo V.V. An in-depth look at TLC model checker. *Тиждень науки-2016*: зб. тез доп. науково-практ. конф., 18–22 квітня 2016 р. Запоріжжя: ЗНТУ, 2016. С. 523–524. URL: https://zp.edu.ua/uploads/conference/2016/TN2016_T1.pdf (дата звернення: 06.08.2023)

39. Shkarupylo V.V., Tomičić I., Arapin D.V. The concurrency representation in TLA+ specification. Proc. VIIIth Int. scientific-practical conf. on Modern problems and achievements of radio engineering (electronics),

telecommunications and information technology (Zaporizhzhya, Ukraine, September 21–23, 2016). P. 118–119. URL: http://rtt.zntu.edu.ua/data/Tezy_ZNTU_2016.pdf (дата звернення: 06.08.2023)

40. Shkarupylo V. TLC model checking and the concurrency in specification. *Proc. Tenth International Scientific-Practical Conference "INTERNET-EDUCATION-SCIENCE-2016", IES-2016* (Vinnytsia, Ukraine, October 11–14, 2016). P. 89–91. URL: http://ir.lib.vntu.edu.ua/handle/123456789/13390 (дата звернення: 06.08.2023)

41. Shkarupylo V.V. An Approach to DEVS-driven Simulation of Software-defined Networks. *Тиждень науки-2017*: науково-практ. конф., 18–21 квітня 2017 р.: тези доп. Запоріжжя: ЗНТУ, 2017. С. 652. URL: https://zp.edu.ua/uploads/dept_s&r/2017/conf/1/TN2017.pdf (дата звернення: 06.08.2023)

42. Shkarupylo V.V., Timenko A.V. An approach to the Internet of things simulation on a basis of discrete event system specification. *Proc. Int. research and practice conference on Modern methods, innovations, and experience of practical application in the field of technical sciences* (Radom, Republic of Poland, Dec. 27–28, 2017). P. 32–34.

43. Shkarupylo V.V. On the applicability of model checking techniques in the Internet of Things domain. *Тиждень науки-2018*: науково-практ. конф., 16–20 квітня 2018 р.: тези доп. Запоріжжя: ЗНТУ, 2018. С. 967–968. URL: https://zp.edu.ua/uploads/dept_s&r/2018/conf/1/TN2018.pdf (дата звернення: 06.08.2023)

44. Shkarupylo V.V., Timenko A.V. On the expediency of stratification to foster the reconfigurability of formal specifications. *Тенденції та вектор розвитку науки в сучасному світі: VI Міжнародна науково-практична інтернет-конференція*: тези доповідей, Дніпро, 30 квітня 2018 р. Ч. 1. Дніпро: НБК, 2018.

С. 46–49. URL: https://ispic.ngo-seb.com/assets/files/6_conf_30.04.18_P.1.pdf (дата звернення: 06.08.2023)

45. Shkarupylo V., Kudermetov R. On the aspects of cyber-physical systems modeling with UPPAAL. *Simulation-2018: 6th Int. conference*, September 12–14, 2018: theses. Kyiv: Pukhov Institute for Modelling in Energy Engineering, 2018. P. 267–269. URL: https://ipme.kiev.ua/en/conference/simulation-2018/ (дата звернення: 06.08.2023)

46. Shkarupylo V., Polska O., Shcherbak N. On the classification of model checking methods for the Internet of Things. *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: IX Міжнародна науково-практична конференція*, 3–5 жовтня 2018 р.: тези доп. Запоріжжя: ЗНТУ, 2018. С. 77–78.

47. Шкарупило В.В., Кудерметов Р.К., Польська О.В., Тіменко А.В. Щодо доцільності перевірки протоколів взаємодії компонентів систем інтернету речей. *Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2019*: матеріали VII Міжнародної науково-практичної конференції, 15–16 травня 2019 р. Київ: НУБіП України, 2019. С. 63–65. URL: https://lib.lntu.edu.ua/sites/default/files/2021-03/Київ_Конференція_НУБІП_2019_UA.pdf (дата звернення: 06.08.2023)

48. Шкарупило В.В., Євдокимов В.Ф., Душеба В.В. Аспекти застосування методів перевірки на моделі при проектуванні систем критичного призначення. *Безпека енергетики в епоху цифрової трансформації: науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України* : програма та матеріали, 20 грудня 2019 р. Київ : ІПМЕ ім. Г.Є.Пухова НАН України, 2019. С. 94–96. URL: https://ipme.kiev.ua/wp-content/uploads/2019/12/Програма-КБЕЕЦ-2019.pdf (дата звернення: 06.08.2023)

49. Шкарупило В.В. Про застосування правила композиції при синтезі формальних специфікацій. *Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*, 15 травня 2020 р. Київ : ІПМЕ ім. Г.Є. Пухова НАН України, 2020. С. 21–22. URL: https://zenodo.org/record/3813710 (дата звернення: 06.08.2023)

50. Шкарупило В.В. Дослідження методу перевірки на моделі TLC. *Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2020 : VIII Міжнародна науково-практична Інтернет-конференція* (м. Київ, Україна, 14–15 травня, 2020). 2020. Київ: НУБіП України. С. 84–86. URL: http://econference.nubip.edu.ua/index.php/grpi/grpi20/paper/view/2306/317 (дата звернення: 06.08.2023)

51. Шкарупило В.В., Скрупський С.Ю. Комбінований підхід до застосування методу перевірки на моделі TLC. *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: X Ювілейна міжнародна науково-практична конференція, присвячена 120-річчю з дня заснування Національного університету «Запорізька політехніка»* (Запоріжжя, Україна, 7–9 жовтня, 2020). Запоріжжя: НУ «ЗП». С. 95–97. URL: http://rtt.zntu.edu.ua/data/Tezy_NUZP_2020.pdf (дата звернення: 06.08.2023)

52. Шкарупило В.В., Кудерметов Р.К., Польська О.В. Дослідження просторової складності алгоритмів в основі методу верифікації TLC. *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: X Ювілейна міжнародна науково-практична конференція, присвячена 120-річчю з дня заснування Національного університету «Запорізька політехніка»* (Запоріжжя, Україна, 7–9 жовтня, 2020). Запоріжжя: НУ «ЗП». С. 93–95. URL: http://rtt.zntu.edu.ua/data/Tezy_NUZP_2020.pdf (дата звернення: 06.08.2023)

53. Шкарупило В.В., Чемерис О.А., Душеба В.В. Дослідження впливу мультипоточності на швидкодію методу перевірки на моделі. *Безпека енергетики в епоху цифрової трансформації*: Друга науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України (Київ, Україна, 28–29 грудня, 2020). Київ : ІПМЕ ім. Г.Є. Пухова НАН України, 2020. С. 75–77. URL: https://ipme.kiev.ua/wp-content/uploads/2021/01/Програма-та-матеріали-КБЕЕЦ-2020.pdf (дата звернення: 06.08.2023)

54. Шкарупило В.В., Блінов І.В. Щодо застосування методу перевірки на моделі при проєктуванні інформаційно-технологічних систем суб'єктів ринку електроенергії. *XXXIX науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, присвячена 40-річчю Інституту, Дню науки в Україні та з нагоди відзначення 30-ї річниці незалежності України* (м. Київ, Україна, 12 травня, 2021). Київ: ІПМЕ ім. Г.Є. Пухова НАН України. С. 7–9. URL: https://drive.google.com/file/d/1QOydMJU3nHOjXZ92vcLF2zAclu8w1RcG/view?usp=share_link (дата звернення: 06.08.2023)

55. Шкарупило В.В., Блінов І.В. Модельно-орієнтований підхід до формалізації нефункціональних характеристик систем критичного призначення, зокрема у природокористуванні. *Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні 2021*: IX Міжнародна науково-практична Інтернет-конференція (м. Київ, Україна, 13–14 травня, 2021). Київ: НУБіП України. С. 55–57. URL: https://drive.google.com/file/d/1IPmtWaLu85W3c9CsrYXYqx2_iFYIq4q2/view?usp=sharing (дата звернення: 06.08.2023)

56. Шкарупило В.В., Блінов І.В., Душеба В.В., Тіменко А.В. Дуальний підхід до формалізації функціональних характеристик систем критичного призначення. *European scientific discussions : 9th International scientific and*

practical conference. Potere della ragione Editore (м. Рим, Італія, 18–20 липня, 2021 р.). С. 143–149. URL: https://sci-conf.com.ua/wp-content/uploads/2021/07/EUROPEAN-SCIENTIFIC-DISCUSSIONS-18-20.07.2021.pdf (дата звернення: 06.08.2023)

57. Шкарупило В.В., Блінов І.В., Душеба В.В., Кучанський В.В. Щодо мультипоточного застосування формального методу перевірки на моделі TLC. *Topical issues of modern science, society and education. Proceedings of the 2nd International scientific and practical conference. SPC "Sci-conf.com.ua"*. Kharkiv, Ukraine. 2021. P. 231–236. URL: https://sci-conf.com.ua/wp-content/uploads/2021/09/TOPICAL-ISSUES-OF-MODERN-SCIENCE-SOCIETY-AND-EDUCATION-5-7.09.21.pdf (дата звернення: 06.08.2023)

58. Дімітрієва Д.О., Шкарупило В.В. Огляд інструментів використання формальних методів та засобів при проєктуванні систем критичного призначення. *ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ: ЕКОНОМІКА, ТЕХНІКА, ОСВІТА '2021*: Збірник матеріалів XI Міжнародної науково-практичної конференції молодих вчених, 11–12 листопада 2021 року, НУБіП України, Київ. С. 164–165. URL: https://drive.google.com/file/d/10iRiRUwpXqTY510LzL1j0BeDt-Kpx4Ab/view?usp=sharing (дата звернення: 06.08.2023)

59. Шкарупило В.В., Душеба В.В. Підхід до синтезу формалізованих подань нефункціональних характеристик на етапі проєктування. *Безпека енергетики в епоху цифрової трансформації* : III науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України : матеріали, 22 грудня 2021 р. Київ : ІПМЕ ім. Г.Є.Пухова НАН України, 2021. С. 128–130. URL: https://ipme.kiev.ua/wp-content/uploads/2021/12/Матеріали-КБЕЕЦ-2021-1.pdf (дата звернення: 06.08.2023)

60. Шкарупило В.В., Душеба В.В. Спадковість артефактів у контексті багатовимірної верифікації. *Тиждень науки-2022*: науково-практ. конф., 18–22

квітня 2022 р.: тези доп. Запоріжжя: НУ “Запорізька політехніка”, 2022. С. 789–791. URL: https://zp.edu.ua/uploads/dept_s&r/2022/conf/4.1/TN_2022.pdf (дата звернення: 06.08.2023)

61. Шкарупило В.В., Душеба В.В. Модельно-орієнтований підхід до синтезу формалізованих подань. *XL Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*, 11 травня 2022 р.: тези доп. Київ: ІПМЕ ім. Г.Є. Пухова НАН України, 2022. С. 20–22. URL: https://bit.ly/3cPGyFk (дата звернення: 06.08.2023)

62. Дімітрієва Д.О., Шкарупило В.В. Огляд нефункціональних характеристик систем критичного призначення. *XL Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України*, 11 травня 2022 р.: тези доп. Київ: ІПМЕ ім. Г.Є. Пухова НАН України, 2022. С. 78–79. URL: https://bit.ly/3cPGyFk (дата звернення: 06.08.2023)

63. Шкарупило В.В., Тіменко А.В. Складові методу контролю показників нефункціональних характеристик розроблюваної комп'ютерної системи при проєктуванні. *XLIX Міжнародна науково-практична інтернет-конференція «Проблеми та перспективи розвитку сучасної науки в країнах Європи та Азії»*, 31 серпня 2022 р.: тези доп., Переяслав, 2022. С. 69–71.

64. Шкарупило В.В., Душеба В.В. Щодо аспектів контролю несуперечності програмно-алгоритмічної складової систем критичного призначення. *Продовольча та екологічна безпека в умовах війни та повоєнної відбудови, присвячена 125-річчю заснування Національного університету біоресурсів і природокористування України: виклики для України та світу*: мат. Міжн. наук.-практ. конф., секція 5: Інженерія, енергетика та інформаційні технології в умовах війни та післявоєнній відбудові країни (м. Київ, 25 трав. 2023 р.): тези доп. Київ: НУБіП України, 2023. С. 170–172. URL:

https://nubip.edu.ua/sites/default/files/u381/sekciya_5.pdf (дата звернення: 06.08.2023)

65. Шкарупило В.В., Блінов І.В., Душеба В.В. Дослідження методу верифікації TLC при вирішенні задач енергетики. *XLI Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України* (м. Київ, Україна, 17 травня, 2023 р.). С. 21–22. URL: https://ipme.kiev.ua/wp-content/uploads/2023/05/Матеріали-конференції-2023.pdf (дата звернення: 06.08.2023)

66. Шкарупило В.В., Душеба В.В., Тіменко А.В., Казакова Н.О. Аспекти досягнення функційної безпечності при розробленні систем критичного призначення. *Актуальні проблеми та перспективи розвитку фундаментальних, прикладних, загальнотехнічних та безпекових наук*: III Всеукраїнської науково-практичної конференції пам’яті академіка Академії наук вищої освіти, професора Анатолія Володимировича Касперського (м. Київ, Україна, 21 червня 2023 р.): тези доп. Київ: УДУ ім. Михайла Драгоманова, 2023. С. 394–395. URL: https://drive.google.com/file/d/1nS1d9cf9EUNUZDnY0ZWLkQTaBlb0xoIN/view (дата звернення: 06.08.2023)

67. Шкарупило В.В., Душеба В.В., Тіменко А.В. Огляд рівнів забезпечення резилієнтності у галузі енергетики. *Survivability & Resilience – 2023: collection of materials of the international scientific and practical conference*, Kyiv, October 19, 2023, PIMEE of NAS of Ukraine. 2023. P. 33–34. URL: https://ipme.kiev.ua/konferencii/zhivuchist-ta-rezilyentnist-2023/ (дата звернення: 21.10.2023)

68. Шкарупило В.В., Душеба В.В. Аспекти введення мультипоточності до реалізації методу формальної верифікації TLC. *Безпека енергетики в епоху цифрової трансформації*: П’ята науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук

України, Київ, Україна, 22 листопада, 2023 р. Київ : ІПМЕ ім. Г.Є. Пухова НАН України. С. 121–122. URL: https://ipme.kiev.ua/konferencii/naukovo-praktichna-konferenciya-bevect-2023/ (дата звернення: 23.11.2023)

Відомості про апробацію результатів дисертації:

1. Науково-технічний семінар «Критичні комп'ютерні технології та системи» – КриКТехС-2024/1/186 (м. Харків, 2024 р.).
2. П'ята науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України «Безпека енергетики в епоху цифрової трансформації» (м. Київ, 2023 р.).
3. Міжнародна науково-практична конференція «живучість та резильєнтність – 2023» (Survivability & Resilience – 2023), (м. Київ, 2023 р.).
4. III Всеукраїнська науково-практична конференція пам'яті академіка Академії наук вищої освіти, професора Анатолія Володимировича Касперського «Актуальні проблеми та перспективи розвитку фундаментальних, прикладних, загальнотехнічних та безпекових наук» (м. Київ, 2023 р.).
5. XLI Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України (м. Київ, 2023 р.).
6. Sixth International Workshop on Computer Modeling and Intelligent Systems, CMIS-2023 (Zaporizhzhia, 2023).
7. Міжнародна науково-практична конференція «Продовольча та екологічна безпека в умовах війни та повоєнної відбудови: виклики для України і світу», присвячена 125-річчю заснування Національного університету біоресурсів і природокористування України (м. Київ, 2023 р.).
8. XLIX Міжнародна науково-практична інтернет-конференція «Проблеми та перспективи розвитку сучасної науки в країнах Європи та Азії» (м. Переяслав, 2022 р.).

9. XL Науково-технічна конференцію молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України (м. Київ, 2022 р.).

10. Науково-практична конференція «Тиждень науки-2022» (м. Запоріжжя, 2022 р.).

11. Третя науково-практична конференція «Безпека енергетики в епоху цифрової трансформації» (м. Київ, 2021 р.).

12. XII Міжнародна науково-практична конференція молодих вчених «Інформаційні технології: економіка, техніка, освіта ‘2021» (м. Київ, 2021 р.).

13. 2021 IEEE KhPI Week on Advanced Technology (м. Харків, 2021 р.).

14. 2nd International scientific and practical conference on Topical issues of modern science, society and education – SPC «Sci-conf.com.ua» (м. Харків, 2021 р.).

15. IX Міжнародна науково-практична конференція «European scientific discussions» (м. Рим, Італія, 2021 р.).

16. IX Міжнародна науково-практична Інтернет конференція «Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні 2021» (м. Київ, 2021 р.).

17. XXXIX Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, присвячена 40-річчю Інституту, Дню науки в Україні та з нагоди відзначення 30-ї річниці незалежності України (м. Київ, 2021 р.).

18. Друга науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України «Безпека енергетики в епоху цифрової трансформації» (м. Київ, 2020 р.).

19. X Ювілейна міжнародна науково-практична конференція «Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних

технологій», присвячена 120-річчю з дня заснування Національного університету «Запорізька політехніка» (м. Запоріжжя, 2020 р.).

20. VIII Міжнародна науково-практична Інтернет-конференція «Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2020» (м. Київ, 2020 р.).

21. 11th International Conference on Dependable Systems, Services and Technologies, DESSERT'2020 (м. Київ, 2020 р.; доповідь відзначено сертифікатом за кращу доповідь).

22. Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України (м. Київ, 2020 р.).

23. 2019 International Scientific-Practical Conference on Problems of Infocommunications. Science and Technology, PIC S&T`2019 (м. Київ, 2019 р.).

24. Науково-практична конференція Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України «Безпека енергетики в епоху цифрової трансформації» (м. Київ, 2019 р.).

25. VII Міжнародна науково-практична конференція «Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2019» (м. Київ, 2019 р.).

26. 2018 IEEE International Scientific and Practical Conf. on Problems of Infocommunications. Science and Technology, PIC S&T`2018 (м. Харків, 2018 р.).

27. IX Міжнародна науково-практична конференція «Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій» (м. Запоріжжя, 2018 р.).

28. VI Міжнародна наукова конференція «Моделювання-2018», приурочена до 100-річчя від дня утворення Національної академії наук України та річниці з Дня народження академіка НАН України Пухова Георгія Євгеновича (м. Київ, 2018 р.).

29. VI Міжнародна науково-практична інтернет-конференція «Тенденції та вектор розвитку науки в сучасному світі» (м. Дніпро, 2018 р.).

30. Науково-практична конференція «Тиждень науки-2018» (м. Запоріжжя, 2018 р.).

31. 14th Int. Conf. on Advanced Trends in Radioelectronics, Telecommunications and Comp. Engineering, TCSET'2018 (с. Славське, 2018 р.).

32. Int. research and practice conference on Modern methods, innovations, and experience of practical application in the field of technical sciences (Radom, Republic of Poland, 2017).

33. Науково-практична конференція «Тиждень науки 2017» (м. Запоріжжя, 2017 р.).

34. Tenth International Scientific-Practical Conference «Internet-education-science-2016», IES-2016 (м. Вінниця, 2016 р.).

35. VIIIth Int. scientific-practical conf. on Modern problems and achievements of radio engineering (electronics), telecommunications and information technology (м. Запоріжжя, 2016 р.).

36. 27th Int. Central European Conference on Information and Intelligent Systems, CECIIS 2016 (Varazdin, Croatia, 2016).

37. Науково-практична конференція «Тиждень науки 2016» (м. Запоріжжя, 2016 р.).

38. XIIIth Int. Conf. on Modern Problems of Radio Engineering, Telecommunications, and Computer Science, TCSET'2016 (с. Славське, 2016 р.).

39. Науково-технічна конференція молодих вчених та спеціалістів Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України (м. Київ, 2016 р.).

40. XXXIV науково-технічна конференція «Моделювання» (м. Київ, 2015 р.).