УДК 004.052.42
DOI https://doi.org/10.32838/2663-5941/2021.1-1/27

Шкарупило В.В.
Інститут проблем моделювання в енергетиці імені Г.Є. Пухова
Національної академії наук України

Чемерис О.А.
Інститут проблем моделювання в енергетиці імені Г.Є. Пухова
Національної академії наук України

Душеба В.В.
Інститут проблем моделювання в енергетиці імені Г.Є. Пухова
Національної академії наук України

Кудерметов Р.К.
Національний університет «Запорізька політехніка»

Польська О.В.
Національний університет «Запорізька політехніка»

МОДЕЛЬНО-ОРІЄНТОВАНИЙ ПІДХІД ДО КОНТРОЛЮ ПОКАЗНИКІВ НЕФУНКЦІОНАЛЬНИХ ХАРАКТЕРИСТИК ПІД ЧАС ПРОЄКТУВАННЯ[1]

Вирішення завдання виявлення помилок проєктних рішень на ранніх етапах процесу розроблення системи набуває особливої актуальності, коли йдеться про системи критичного призначення, функціональна безпека яких є запорукою уникнення критичних наслідків значного соціально-економічного характеру. Передумовою забезпечення функціональної безпеки є, зокрема, дотримання заданих вимог до функціональних і нефункціональних характеристик розроблюваної системи. Це досягається за рахунок комплексного застосування методів і засобів контролю названих характеристик упродовж усього життєвого циклу останньої. Відповідними представниками є формальні методи, методи імітаційного моделювання.

У межах представленої роботи система розглядається з позиції програмної складової частини. Акцент ставиться саме на етапі проєктування процесу розроблення, що має на меті своєчасне виявлення й усунення архітектурних помилок.

У роботі вирішується завдання розроблення підходу до контролю показників нефункціональних характеристик системи критичного призначення при проєктуванні. Підхід характеризується безшовністю його інтеграції до складу поширеного комплексу засобів контролю функціональних характеристик, а також розвинутим механізмом варіювання рівня деталізації відповідної імітаційної моделі як засобу контролю. Запропонований підхід будується на залученні математичного апарату дискретно-подійного імітаційного моделювання DEVS. Підхід базується на оперуванні концептами «атомарної» та «складеної» DEVS-моделей. Як предметна сфера розглядається космічна галузь. Підходом передбачається, що його застосуванню передує контроль функціональних характеристик системи шляхом формальної верифікації методом перевірки на моделі специфікації, побудованої на основі темпоральної логіки дій TLA. Вихідними даними виступає блок-схема алгоритму.

Ключові слова: *DEVS, TLA, верифікація, система критичного призначення, нефункціональні характеристики, функціональна безпека.*

Постановка проблеми. Актуальний етап розвитку прикладного застосування методів і засобів формальної верифікації (далі – ФВ) систем критичного призначення (далі – СКП) можна охарактеризувати таким чином: має місце активне залучення названих методів і засобів у автоматизованому режимі упродовж усього життєвого циклу розроблюваної системи – у широкому спектрі сфер застосування [1]. Показовою є, зокрема, космічна галузь [2]. Іншим прикладом є перевірка і підтвердження коректності реалізації протоколу Zab (ZooKeeper

[1] Дослідження виконано у межах науково-дослідної роботи № 0120U102683 «Розроблення спеціалізованих комп'ютерних технологій моделювання та опрацювання оперативної інформації в задачах енергетики», що проводиться відділом математичного та комп'ютерного моделювання Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України.

Atomic Broadcast protocol) – засобу забезпечення високоефективного обміну повідомленнями на основі програмної платформи Apache Kafka, поширеної у корпоративній сфері [3]. Як дієві інструменти варто відзначити темпоральну логіку дій TLA (Temporal Logic of Actions), відповідний формалізм TLA+, метод перевірки на моделі TLC (TLA Checker) і засіб автоматизації TLA Toolbox [4; 5]. У зазначених і подібних інструментах акцент, однак, ставиться саме на функціональних характеристиках (далі – ФХ) розроблюваної системи.

У свою чергу, згідно зі стандартом IEC 61508 забезпечення заданого рівня функціональної безпеки (ФБ) охоплює питання контролю як ФХ, так і нефункціональних характеристик (НФХ) СКП [6]. Відповідні засоби (у т. ч. формальні методи) рекомендується застосовувати упродовж усього життєвого циклу розроблюваної системи. У нашій роботі розглядається етап проєктування процесу розроблення системи, а саме застосовувані методи та засоби контролю НФХ, оскільки своєчасне виявлення критичних помилок проєктних рішень (ПР) дозволить як забезпечити дотримання заданих вимог до ФБ, так і скороти результуючі часові та матеріальні витрати на розроблення та супроводження СКП.

Методи та засоби контролю НФХ, які фігурують у публікаціях, можна охарактеризувати як такі, що є розрізненими і потребують доопрацювання з позиції їх «безшовної» інтеграції до складу комплексу засобів забезпечення ФБ з позицій як ФХ, так і НФХ. У зв'язку із цим у роботі вирішується завдання розроблення підходу до контролю НФХ СКП на етапі проєктування процесу розроблення системи, що характеризується «безшовністю» інтеграції до складу комплексу засобів контролю ФХ, які вже добре себе зарекомендували з погляду прикладного застосування. Такими є вищезазначені метод TLC і засоби TLA, TLA+ і TLA Toolbox [7; 8]. Під «безшовністю» розуміється відсутність необхідності залучення додаткових засобів перетворення формальної специфікації (ФС) для перевірки ФХ до моделі, придатної до перевірки НФХ. Цей аспект розглядається як фактор сприяння адекватності останньої та, як результат, достовірності одержуваних на основі такої моделі даних.

Аналіз останніх досліджень і публікацій. Зазначимо, що у контексті нашої роботи як СКП розглядається відповідна програмна складова частина.

Наявні рішення у напрямі ФВ СКП мають широкий спектр спрямування. Це, зокрема, комплексні рішення, що охоплюють засоби перевірки як ФХ, так і НФХ системи на етапах аналізу, проєктування та реалізації процесу розроблення СКП. Відповідним прикладом є інструментарій S3 (Systerel Smart Solver), який було залучено для перевірки програмної складової частини вбудованої системи ARP (Automatic Rover Protection) [9]. Засіб охоплює реалізації таких відомих технік, як: індуктивне виведення, перевірка на моделі, генерування тестових послідовностей, перевірка еквівалентності. При перевірці на моделі вирішується задача виконуваності булевої формули. Вагомим недоліком названого засобу, проте, є відсутність прозорого механізму варіювання рівня деталізації використовуваних моделей, що важливо у контексті програмної системи.

Альтернативні засоби є зазвичай вузькоспеціалізованими. У них акцент робиться, зокрема, на реалізації індуктивного підходу до перевірки НФХ. Для цього залучається, наприклад, математичний апарат ланцюгів Маркова, що реалізується у складі комплексу засобів SBIP (Stochastic Behavior-Interaction-Priority) [10]. Більше того, стохастичні методи перевірки та відповідні моделі вже зарекомендували себе як дієві засоби контролю НФХ розроблюваної системи вже на етапі проєктування – як засоби одержання кількісних оцінок [11]. Як альтернативний математичний апарат поширилися також мережі Петрі, де показовою сферою прикладного застосування є програмні системи контролю роботи атомної електростанції [12].

Окрім того, згідно з положеннями стандарту ISO 26262, що регламентує аспекти забезпечення ФБ систем керування транспортними засобами, дотримання заданого рівня ФБ можливе за рахунок контролю як ФХ, так і НФХ упродовж усього життєвого циклу системи [13]. Відповідно до цього контроль названих характеристик має здійснюватися на різних рівнях деталізації одержуваних рішень. Дієвим засобом для цього є інструментарій UML/MARTE, де верифікація ФХ і НФХ реалізується шляхом імітаційного моделювання [14]. Більше того, наголошується на важливості «безшовної» інтеграції засобів верифікації до складу комплексу засобів підтримки процесу проєктування СКП. Для цього було, зокрема, запропоновано модельно-орієнтований інструментарій ФВ, що будується на поданні ФС ПР на основі мови EAST-ADL [15]. Згідно з модельно-орієнтованим підходом верифікацію ФС ПР пропонується здійснювати також шляхом поступальної низхідної модифікації ФС [16]. У цьому контексті ФВ реалізується саме щодо

архітектурної складової частини розроблюваної системи. Як початкова використовується модель на основі текстового AADL-формалізму (Architecture Analysis & Design Language), на виході – система переходів (СП), що задовольняє умови, задані темпоральною формулою мовою LTL (Linear Temporal Logic). Зазначається обмеженість виразних можливостей LTL – зокрема, через неможливість формалізації темпоральних розгалужень.

Альтернативним підходом до перевірки НФХ СКП при проєктуванні є застосування техніки статистичної перевірки на моделі SMC (Statistical Model Checking), що на прикладі інструментарію UPPAAL-SMC дозволяє формалізувати НФХ як часові обмеження на переходи між локаціями СП, побудованої на основі математичного апарату часових автоматів [17].

Зазначимо, що при використанні певного засобу контролю НФХ СКП окрему увагу варто приділити специфіці сфери прикладного застосування. З погляду програмної складової частини значної ваги набувають аспекти модульності та гнучкості варіювання рівня деталізації відповідних ФС та/або імітаційних моделей. У цьому контексті виокремлюється формалізм DEVS (Discrete-event System Specification), що, завдяки оперуванню концептами «атомарної» (АМ) і «складеної» (СМ) DEVS-моделей, надає гнучкий механізм урахування вищезазначених аспектів [18].

Результати попередніх досліджень показали, що інструментарій DEVS є дієвим засобом валідації розподілених програмних систем – шляхом проведення дискретно-подійного імітаційного моделювання [19]. Більше того, було показано, що валідація шляхом моделювання потребує істотно менших часових витрат порівняно з альтернативою у вигляді тестування [20]. На відміну від валідації, за якої перевіряється придатність вже розробленої системи до цільового використання, за верифікації контролюється відповідність одержуваних артефактів процесу проєктування заданим вимогам до ФХ і НФХ СКП [21]. Під «артефактом», у свою чергу, розуміється ПР (блок-схема алгоритму), відповідна ФС та/або модель, що характеризується структурою і змістом [22].

У контексті процесу розроблення СКП етап проєктування набуває особливої ваги – у розрізі розгляду відсутності помилок ПР як запоруки забезпечення ФБ системи [23]. Отже, актуальним стає розроблення підходу до контролю НФХ СКП при проєктуванні, що забезпечує як безшовність інтеграції до складу вже застосовуваних засобів контролю ФХ СКП, так і гнучкість варіювання рівня деталізації відповідних артефактів. Останній аспект, на нашу думку, задовольняється специфікою формалізму DEVS. Аспект безшовності інтеграції, у свою чергу, розкривається прийомами та засобами, викладеними нижче.

Виклад основного матеріалу дослідження. Запропонований підхід базується на маніпулюванні концептами АМ і СМ математичного апарату DEVS. Його застосування передбачається на етапі проєктування процесу розроблення СКП – за умови, що контроль ФХ, поданих у ПР, вже було попередньо виконано методом перевірки на моделі TLC на базі ФС ФХ, синтезованої на основі вихідного артефакту – блок-схеми алгоритму.

У межах роботи оперуватимемо такими артефактами:

– блок-схема алгоритму як графічне подання контексту досліджуваного сценарію предметної сфери;

– ФС ФХ СКП, синтезована на основі формалізму TLA+, коректність якої підтверджена методом перевірки на моделі TLC – розглядається як вихідні дані для запропонованого методу. Зміст артефакту визначається множиною змінних станів, а структура – темпоральними формулами на основі відповідних атомарних висловлювань [24];

– АМ як базові структурні засоби побудови складених конструкцій, зокрема результуючої моделі системи, до якої застосовується метод дискретно-подійного імітаційного моделювання. Структура і зміст відповідних артефактів визначаються математичним апаратом DEVS;

– СМ системи як засіб реалізації запропонованого підходу. Зміст артефакту визначається складом залучених АМ компонентів, а структура – встановленими між ними зв'язками.

Підхід полягає у виконанні нижченаведених кроків.

Крок 1. Рівень деталізації цільової дискретно-подійної моделі системи задається на основі концепту атомарної DEVS-моделі [25]:

$$AM_i = \langle X, S, Y, \delta_{\text{int}}, \delta_{ext}, \lambda, ta \rangle, \qquad (1)$$

де X – множина зовнішніх щодо моделі подій, на які вона реагує зміною поточного стану $s \in S$; Y – множина подій, які модель продукує; $\delta_{\text{int}} : S \to S$ – внутрішня функція переходу, що переводить атомарну модель із поточного стану $s \in S$ у наступний стан $\delta_{\text{int}}(s) \in S$, без урахування зовнішніх подій; $\delta_{ext} : Q \times X \to S$ – зовнішня функція переходу: $Q = \{(s,e) | s \in S, 0 \le e \le ta(s)\}$, де e – модельний час, що минув від моменту останнього переходу, $ta : S \to R^{+}_{0,\infty}$ – функція про-

сування модельного часу; $\lambda: S \to Y$ – функція виходу – функція, яка продукує елементи множини Y на основі поточного стану $s \in S$ атомарної моделі; $i = 1,2,...,m$ – порядковий номер атомарної моделі, $m \in N$ – загальне число атомарних моделей, залучених до синтезу результуючої складеної DEVS-моделі, є числом змінних станів ФС ФХ СКП мовою TLA+.

Крок 2. Оперуючи концептом складеної DEVS-моделі, на основі виокремлених на попередньому кроці атомарних моделей будуємо архітектуру досліджуваної системи [26]:

$$CM = \langle INP, OUTP, A, set \rangle, \qquad (2)$$

де INP ($OUTP$) – множина усіх вхідних (вихідних) портів – засобів зв'язку – залучених АМ (1) – елементів множини $A = \{AM_i\}$; $set: A \times OUTP \to A \times INP$ – функція встановлення зав'язків між елементами множини A. «Порт» – засіб сприйняття подій відповідного типу, тобто порт регламентує тип подій, що надходять з/до нього.

Крок 3. Проводимо контроль НФХ СКП шляхом дискретно-подійного імітаційного DEVS-моделювання на основі результуючої структури (2).

Як узагальнення вищесказаному зауважимо, що на рівні АМ (1) оперуємо поняттям «подія», а на рівні СМ – поняттям «порт». Викладена ієрархічна структура є дворівневою. За необхідності проміжні ієрархічні рівні вводяться на основі структури (2).

Для демонстрації підходу як сценарій предметної сфери розглянемо сценарій космічної галузі – фрагмент блок-схеми алгоритму роботи блоку управління конфігурацією (БУК) бортового цифрового обчислювального комплексу (БЦОК) космічного апарату (КА) (рис. 1).

Поданий фрагмент алгоритму має такий зміст:

– значення показника стану контрольно-перевірочної апаратури (КПА) визначає подальший сценарій роботи алгоритму;

– якщо значення становить 0, виконується запуск ПСЕП (підсистеми електропостачання);

– якщо 1 – спочатку виконується запуск БУК, а вже потім – ПСЕП.

Згідно з поданим алгоритмом можливі два сценарії виконання:

– послідовне виконання блоків 1, 3;

– послідовність вигляду (1, 2), 3.

Як показник НФХ розглянемо часову затримку. У разі останнього сценарію можливе перевищення заданих обмежень на спрацювання блоку 3 – як наслідок накопичення значення затримки на блоках 1 і 2.

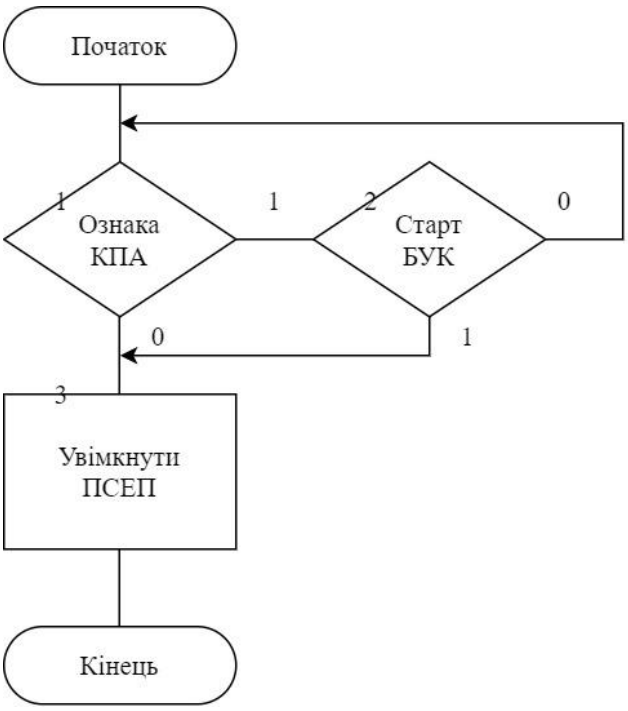

Рис. 1. Фрагмент блок-схеми алгоритму роботи БУК БЦОК КА

Згідно з рис. 1, досліджуваний фрагмент алгоритму відтворює взаємодію таких трьох компонентів СКП: КПА, БУК, ПСЕП. Відповідно, ФС ФХ СКП на основі TLA+ містить три змінні стани СП, які подамо множиною $V = \{v_1, v_2, v_3\}$. За запропонованим підходом, для кожної $v_i \in V\,(i = 1,2,3)$ синтезуємо відповідну $AM_i \in A$ (1). Результуючий артефакт подано на рис. 2.

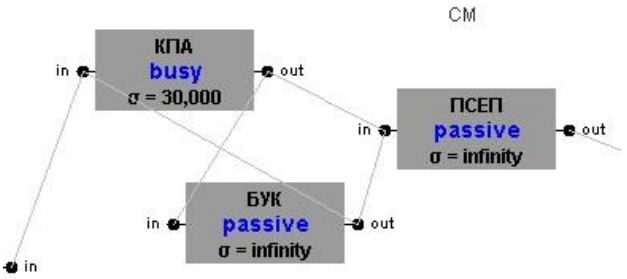

Рис. 2. Графічне подання результуючої СМ

На рис. 2 фігурують три АМ (оскільки $|V| = 3$), що разом формують СМ, на основі якої здійснюється контроль НФХ шляхом дискретно-подійного імітаційного DEVS-моделювання. Контроль полягає у відстеженні накопиченого значення часової затримки внаслідок взаємодії АМ згідно з алгоритмом рис. 1 та порівнянні цього значення із табличним, яке у цьому разі становить 300 мс. Згідно з рис. 2 $S = \{"busy","passive"\}$. Значення σ задає затримку на спрацювання відповідної $AM_i \in A$.

Отже, безшовність інтеграції підходу полягає у синтезі один до одного елементів множини A (2) на основі елементів множини V. Розвинутий механізм варіювання рівня деталізації результуючого артефакту СМ, у свою чергу, базується на оперуванні концептами (1) і (2) згідно з викладеним підходом.

Висновки. Таким чином, було запропоновано підхід до контролю показників нефункціональних характеристик розроблюваної системи критичного призначення, що дозволяє виконувати перевірку названих характеристик вже на етапі проєктування процесу розроблення.

Було здобуто такі результати:

1) запропонований підхід характеризується розвинутим механізмом варіювання рівня деталізації одержуваних на його основі артефактів, а також забезпечує безшовність інтеграції до складу комплексу із методу та засобів, застосовуваних на етапі проєктування для контролю функціональних характеристик розроблюваної системи, а саме широко використовуваних формального методу перевірки на моделі TLC, темпоральної логіки дій TLA, відповідного формалізму TLA+, а також засобу автоматизації TLA Toolbox;

2) застосування підходу продемонстровано на прикладі фрагменту блок-схеми алгоритму роботи блоку управління конфігурацією бортового цифрового обчислювального комплексу космічного апарата.

Подальші дослідження орієнтовано на розвиток комплексу засобів автоматизації процесу синтезу артефактів згідно із запропонованим підходом.

Список літератури:

1. 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. Т. 30 (69). Ч. 1. № 6. С. 188–193. DOI https://doi.org/10.32838/2663-5941/2019.6-1/34.

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* / 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.

3. 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. № 6. P. 1312–1323. DOI: https://doi.org/10.1007/s11390-020-0538-7.

4. Lamport L. Specifying systems: The TLA+ language and tools for hardware and software engineers. Boston : Addison-Wesley, 2002. 382 p.

5. 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.

6. IEC 61508 Edition 2.0. Functional safety of electrical/electronic/programmable electronic safety-related systems. [Approved: April 2010]. URL: https://www.iec.ch/functionalsafety/standards/page2.htm. (дата звернення: 19.11.2020).

7. 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.

8. 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.

9. 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. № 4. P. 423–440. DOI: https://doi.org/10.1007/s10009-017-0475-0.

10. 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. № 2. P. 171–185. DOI: https://doi.org/10.1007/s10009-014-0313-6.

11. Ghezzi C., Sharifloo A.M. Model-based verification of quantitative non-functional properties for software product lines. *Information and Software Technology*. 2013. Vol. 55. № 3. P. 508–524. DOI: https://doi.org/10.1016/j.infsof.2012.07.017.

12. 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.

13. ISO 26262:2018. Road vehicles. Functional safety. Part 1: Vocabulary. Published: December 2018. URL: https://www.iso.org/standard/68383.html (дата звернення: 05.10.2020).

14. 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.

15. 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.

16. 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.

17. 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.

18. Van Tendeloo Y., Vangheluwe H. An evaluation of DEVS simulation tools, *SIMULATION*. 2017. Vol. 93. № 2. P. 103–121. DOI: https://doi.org/10.1177/0037549716678330.

19. 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.

20. Шкарупило В.В., Кудерметов Р.К., Польська О.В. DEVS-орієнтована методика валідації композитних веб-сервісів. *Радіоелектроніка, інформатика, управління*. 2015. № 4. С. 79–86. DOI: 10.15588/1607-3274-2015-4-12.

21. IEEE 1012-2016. IEEE Standard for system, software, and hardware verification and validation. Approved: 28 September 2017. URL: https://ieeexplore.ieee.org/document/8055462 (дата звернення: 23.07.2020).

22. 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.

23. Myers G.J. Software reliability: principles and practices. New York : Wiley, 1976. 360 p.

24. 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. DOI: 10.31474/1996-1588-2020-1-30-49-57.

25. Concepcion A.I., Zeigler B.P. DEVS formalism: a framework for hierarchical model development. *IEEE Transactions on Software Engineering*. 1988. Vol. 14. № 2. P. 228–241. DOI: https://doi.org/10.1109/32.4640.

26. 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. № 9 (89). P. 67–73. DOI: https://doi.org/10.15587/1729-4061.2017.110142.

Shkarupylo V.V., Chemerys O.A., Dusheba V.V., Kudermetov R.K., Polska O.V. MODEL-DRIVEN APPROACH TO NON-FUNCTIONAL PROPERTIES INDEXES CONTROL AT DESIGN

Resolving the task of design solutions errors discovery at the early stages of system design process becomes of particular significance, when the system under development is the safety-critical one – a system, which functional safety is a precondition for critical outcomes of significant social and economic character prevention. A precondition to functional safety is, in particular, sticking to a given requirements to both functional and non-functional properties of system under development. It is reached by way of complex usage of control methods and tools alongside the whole life cycle of a system. Corresponding representatives are, in particular, the model checking methods and the methods of simulation.

Within the paper, a system is approached with respect to a software plane. The accent is put on the design stage of engineering process. It is devoted to discover the architectural faults on-time.

In given paper, a task of developing the approach to non-functional properties of safety-critical systems control at design is resolved. The approach is characterized as a seamless tool to be integrated into a proven framework for functional properties formal verification, and also as a developed mechanism for varying the atomicity level of corresponding simulation model as a control tool. The approach is based on the mathematical apparatus of the Discrete-event System Specification (DEVS) formalism. It is grounded on the concepts of "atomic" and "coupled" DEVS-models. As a case study, the scenario of the space industry is considered. The approach is supposed to be applied after the functional properties control – by way of formal verification of specification synthesized on the basis of Temporal Logic of Actions (TLA) with a model checking method. As an input data, the algorithm block diagram is utilized.

Key words: *DEVS, TLA, verification, safety-critical system, non-functional properties, functional safety.*