УДК 004.052.42
DOI https://doi.org/10.32838/2663-5941/2020.5/24

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

ОЦІНЮВАННЯ ПРОСТОРОВОЇ СКЛАДНОСТІ ЗАДАЧІ ФОРМАЛЬНОЇ ВЕРИФІКАЦІЇ, ВИРІШУВАНОЇ МЕТОДОМ ПЕРЕВІРКИ НА МОДЕЛІ[1]

Актуальний стан розвитку процесу розроблення комп'ютерних систем можна охарактеризувати як такий, за якого має місце активне залучення різноманітних формальних методів та засобів. Названі методи та засоби, як правило, знаходять застосування з метою підвищення впевненості колективу розробників у коректності одержуваних артефактів процесу розроблення, зокрема проектних рішень. Коректність при цьому розглядається з позиції відповідності формалізованим вимогам до характеристик розроблюваної системи. Особливої уваги в даному аспекті заслуговують системи критичного призначення – системи, незаплановані сценарії функціонування яких потенційно можуть призвести до значних критичних наслідків. З цієї позиції усунення підстав для виникнення таких сценаріїв є важливим фактором підвищення рівня довіри до результатів процесу розроблення. Досягти цього можливо за рахунок залучення формальних методів уже на етапі проектування названих систем. Беручи до уваги складність програмної складової частини таких систем, першорядної важливості набуває питання автоматизації застосування формальних методів. Цьому критерію задовольняють методи перевірки на моделі, які, проте, висувають значні вимоги до апаратної складової частини обчислювальної платформи, на якій зазначені методи реалізуються.

Дану роботу присвячено експериментальному дослідженню реалізацій поширеного методу перевірки на моделі TLC (TLA Checker) з позиції просторової складності вирішуваної при цьому задачі формальної верифікації. Як рису, що зумовлює відмінність реалізацій методу, розглянуто характер обходу простору станів системи переходів, заданої формальною специфікацією характеристик системи: на основі методів обходу в ширину (BFS, Breadth-first Search) і глибину (DFS, Depth-first Search) теорії графів. Результати проведених досліджень показали, що застосування реалізації методу TLC на основі обходу в глибину висуває більші вимоги до обсягу наявної оперативної пам'яті обчислювальної системи, на якій виконується реалізація методу.

Ключові слова: *BFS, DFS, TLC, перевірка на моделі, формальна специфікація, верифікація, система критичного призначення.*

Постановка проблеми. Актуальний рівень розвитку формальних методів можна охарактеризувати таким чином: значного поширення набуло прикладне застосування методів перевірки на моделі (Model Checkers) через можливість забезпечення автоматизації такого застосування. Названі методи є дієвими засобами реалізації процедури формальної верифікації (далі – ФВ) вимог до розроблюваної системи. Вагомим рушієм у цьому напрямі стали праці таких наукових діячів, як Едмунд Кларк (Edmund M. Clarke), Алан Емерсон (E. Allen Emerson), Джозеф Сіфакіс (Joseph Sifakis), що здобули премію Тьюринга у 2007 р. за розвинення техніки перевірки на моделі до рівня ефективної технології верифікації, що знайшла широке прикладне застосування у сфері розроблення як апаратних, так і програмних систем [1]. Особливої актуальності застосування зазначених методів набуває під час розроблення систем критичного призначення (далі – СКП) – систем, незаплановані (непередбачені) сценарії роботи яких потенційно можуть призвести до критичних наслідків значного масштабу.

Показовими прикладами названих методів є, зокрема, B-метод і його модифікація – Event-B,

[1] Дослідження виконано в 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.

прикладне застосування яких сягає більше 25 років [2]. Демонстративною предметною областю такого застосування є залізничний сектор, представлений системами керування відповідними процесами.

Альтернативним поширеним засобом є метод TLC (TLA Checker), орієнтований на автоматизовану перевірку специфікацій вимог до розроблюваної системи, формалізованих виразними засобами TLA+ темпоральної логіки дій TLA (Temporal Logic of Actions) [3]. Відмінною рисою зазначеного формалізму є можливість подавати задану характеристику досліджуваної системи однією темпоральною формулою. Ця специфіка надає зручний механізм композиційного оперування вже створеними артефактами процесу розроблення системи, з метою побудови більш комплексних рішень на основі існуючих напрацювань. Під «артефактом» при цьому розуміється сутність, що характеризується структурою і змістом [4], а саме результат виконання певної складової частини етапу проектування процесу розроблення СКП. У даній роботі як артефакт розглядається формальна специфікація (ФС) функціональних вимог до СКП.

Аналіз останніх досліджень і публікацій. Результати попередніх досліджень показали, що методи перевірки на моделі є дієвими засобами підвищення рівня довіри розробників до артефактів, одержуваних на етапі проектування СКП [5].

З урахуванням того, що прикладне застосування методів перевірки на моделі супроводжується експоненційним зростанням простору станів системи переходів (далі – СП) у залежності від числа змінних станів, що фігурують у ФС, актуальності набувають як аспекти мультипоточної реалізації методів, так і аспекти покращення ефективності застосування названих методів з позицій обчислювальної і просторової складностей алгоритмічного складника.

Було показано, що метод TLC характеризується придатністю до мультипоточної реалізації – для обчислювальної системи із 384 процесорами було одержано коефіцієнт прискорення, близький до 328 [6].

Попередні дослідження методу TLC було проведено з позиції обчислювальної складності вирішення задачі ФВ [7]. Здобуті результати показали, що ефективність застосування тієї чи іншої реалізації методу TLC залежить як від числа змінних станів СП, заданої ФС, так і від структури ФС. Більше того, було встановлено, що обмежуючим чинником такого застосування є обсяг доступної оперативної пам'яті (далі – ОП) обчислювальної системи, ресурси якої залучаються для автоматизованого застосування методу [8]. У зв'язку із цим дану роботу присвячено оцінюванню просторової складності задачі ФВ, вирішуваної методом TLC.

Постановка завдання. У роботі ставиться завдання оцінювання просторової складності задачі ФВ, вирішуваної методом перевірки на моделі TLC, а саме кожною із двох альтернативних реалізацій методу – на основі обходу простору станів СП, заданої ФС, методами обходу в ширину (BFS, Breadth-first Search) і в глибину (DFS, Depth-first Search) теорії графів.

На основі результатів проведених досліджень потрібно сформулювати рекомендації до прикладного застосування реалізацій методу TLC.

Для виконання сформульованого завдання формалізуємо задачу ФВ, вирішувану методом TLC. Для цього залучимо математичний апарат структури Кріпке, заданої поверх множини атомарних висловлювань AP [1]:

$$M = \langle S, S_0, R, L \rangle, \qquad (1)$$

де S – кінцева множина станів; $S_0 \subseteq S\ (S_0 \neq \varnothing)$ – 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. При цьому $AP = V \times D$, де $V = \{v_j \,|\, j = 1, 2, ..., n \in N\}$ – множина змінних станів СП, $D = \{0, 1, 2\}$ – множина значень змінних станів; $s : V \to D$, де $s \in S$.

Задача ФВ, вирішувана методом TLC, формалізується так:

$$M, \sigma \models \phi, \qquad (2)$$

де M – структура (1), $\sigma = s_0, s_1, ...$ – обчислення як послідовність станів СП, що задається засобами структури (1) [9, с. 70]: $s_0 \in S_0$, $s_1 = R(s_0)$; ϕ – темпоральна формула, що має приймати істинні значення $\forall s \in S$: s є елементом σ. При цьому як темпоральні застосовуються такі оператори: X (neXt) і G (Globally).

Відомо, що в залежності від способу подання графу $G = \langle S, R \rangle$ у складі структури (1) – матрицею чи списком суміжності – витрати ОП на обхід елементів множини S оцінюються, відповідно, як $O(|S^2|)$ чи $O(|S| + |R|)$ [10, с. 591]. При цьому в залежності як від структури ФС, так і від специфіки 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.

Експериментальна складова частина даної роботи будується на вирішенні задачі (2) кожною із двох зазначених реалізацій методу TLC для $n = 2^k$, де $k = 1, 2, ..., 8$. При цьому ФС для вказаних n змінних станів синтезовано згідно з послідовним шаблоном [7]. Застосовуваний підхід до синтезу ФС передбачає використання формалізму TLA+ темпоральної логіки TLA [11].

Виклад основного матеріалу дослідження. Показники для оцінювання просторової складності задачі ФВ, вирішуваної методом TLC, будуватимемо на основі таких складників:

– число станів СП, виявлених у процесі ФВ ФС методом TLC, в яких формула ϕ приймає істинне значення;

– загальне число станів СП, згенерованих у процесі ФВ.

Значення названих складників фіксуватимемо для кожної з реалізацій методу – на основі BFS- та DFS-обходів (табл. 1).

Експериментальні дослідження проведено на програмно-апаратній платформі такої 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, тактова частота – 3,8 ГГц; обсяг ОП – 16 ГБ.

У табл. 1 $|S|$ – загальне число станів СП, виявлених у процесі ФВ методом TLC; $|S^*_{BFS}|$ – загальне число станів, згенерованих при ФВ на основі BFS-обходу, $|S^*_{DFS}|$ – DFS-обходу; $|S|/|S^*_{BFS}|$, $|S|/|S^*_{DFS}|$ – відносна частка виявлених станів від згенерованих – для BFS- і DFS-реалізацій методу, відповідно. Відношення $|S|/|S^*_{BFS}|$ і $|S|/|S^*_{DFS}|$ розглядатимемо як показники ефективності вирішення задачі ФВ, відповідно, на основі BFS- і DFS-обходів; z – розмір файлу ФС.

Із табл. 1 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 таке: згідно з відношенням $|S^*_{DFS}| / |S^*_{BFS}|$ 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 таке співвідношення: $|S^*_{DFS}| / |S^*_{BFS}| \approx 2^k = n$. Отже, характер залежності $|S^*_{DFS}| / |S^*_{BFS}|$ від k є експоненційним (рис. 1). Це означає, що залучення DFS-реалізації методу TLC висуває у n разів вищі вимоги до обсягу наявної ОП обчислювальної системи, на базі якої вирішується задача ФВ. Для побудови графіку застосовано кусочно-лінійну інтерполяцію.

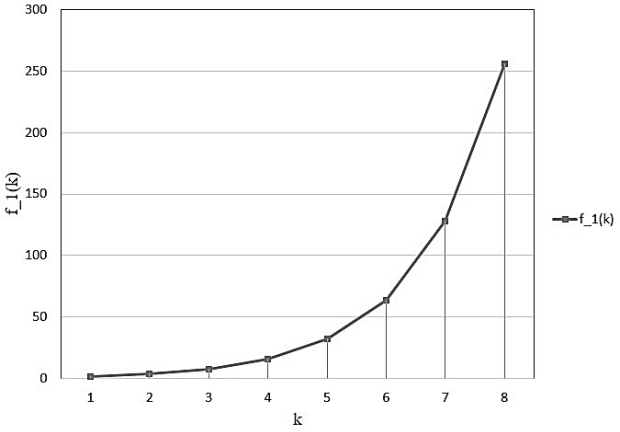

Рис. 1. Графік залежності показника $|S^*_{DFS}| / |S^*_{BFS}|$ від k

На рис. 1 графік функції $f_1(k)$ демонструє характер залежності значення показника $|S^*_{DFS}| / |S^*_{BFS}|$ від значення k (табл. 1). При цьому

Таблиця 1

Показники просторової складності вирішуваної задачі ФВ

№ з/п	n	$\|S\|$	$\|S^*_{BFS}\|$	$\|S^*_{DFS}\|$	$\frac{\|S\|}{\|S^*_{BFS}\|}$	$\frac{\|S\|}{\|S^*_{DFS}\|}$	$\frac{\|S^*_{DFS}\|}{\|S^*_{BFS}\|}$	z, байтів
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	0	31,993	35702
6	2^6	129	16513	1056769	0,008	0	63,996	132896
7	2^7	257	65793	8421377	0,004	0	127,998	527085
8	2^8	513	262657	67239937	0,002	0	255,999	2169856

було зафіксовано, що для $k = 8$ ($n = 256$), для здійснення ФВ DFS-реалізацією методу TLC залучається більше 5 ГБ ОП.

Для демонстрації характеру спадання відносної частки кількостей виявлених станів, у яких виконується формула ϕ, від значення k залучено показники $|S| / |S^*_{BFS}|$ і $|S| / |S^*_{DFS}|$ табл. 1 (рис. 2).

На рис. 2 функція $f_2(k)$ демонструє залежність $|S| / |S^*_{BFS}|$ від k, функція $f_3(k)$ – залежність $|S| / |S^*_{DFS}|$ від k. З рис. 2 видно, що функція $f_3(k)$ убуває приблизно у 2^k швидше за функцію $f_2(k)$. Це свідчить про гіршу просторову складність вирішення задачі ФВ на основі DFS-реалізації методу TLC.

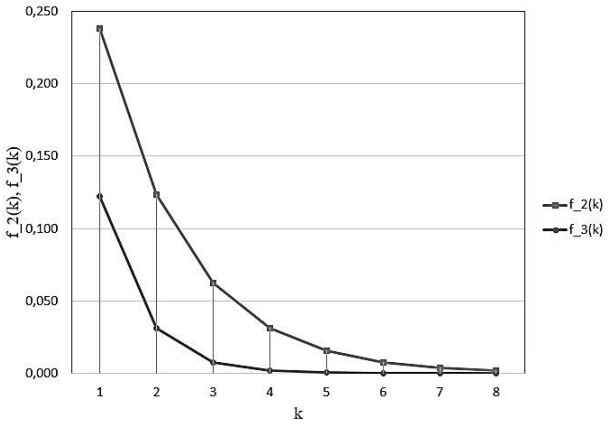

Рис. 2. Графік залежності показників $|S| / |S^*_{BFS}|$ і $|S| / |S^*_{DFS}|$ від k

Отже, за результатами проведених досліджень можна рекомендувати застосовувати BFS-реалізацію методу TLC у випадку обмеженості доступної ОП обчислювальної системи.

Висновки. Таким чином, у роботі було проведено дослідження просторової складності задачі формальної верифікації, вирішуваної методом перевірки на моделі TLC на прикладі двох альтернативних реалізацій методу – BFS і DFS. Було одержано такі результати:

1. Для випадку ФС, синтезованих згідно з послідовним шаблоном, показано, що для досліджуваного набору тестових даних ($n = 2^1, 2^2, ..., 2^8$) просторова складність задачі ФВ, вирішуваної на основі DFS-реалізації методу TLC, є у 2^k $(k = 1, 2...., 8)$ разів гіршою за альтернативну BFS-реалізацію методу. Іншими словами, застосування DFS-реалізації методу TLC є у n разів менш ефективним з позиції просторової складності вирішуваної задачі ФВ, у порівнянні з альтернативною BFS-реалізацією.

2. За результатами проведених досліджень можна рекомендувати застосовувати BFS-реалізацію методу TLC у випадку обмеженості доступної ОП обчислювальної системи. При цьому було зафіксовано, що для $k = 8$ ($n = 256$) для здійснення ФВ DFS-реалізацією методу TLC знадобилося більше 5 ГБ ОП.

Для розвитку і узагальнення здобутих результатів подальші зусилля спрямовано на дослідження просторової складності задачі ФВ, вирішуваної методом TLC, з позиції просторової складності, по відношенню до ФС, що містять подання паралелізму.

Список літератури:

1. Model checking: 2nd ed. / E.M. Clarke et al. Massachusetts : The MIT Press, 2018.

2. The first twenty-five years of industrial use of the B-method / M. Butler et al. *Formal Methods for Industrial Critical Systems, FMICS 2020* : 25th Int. Conf. / eds. M. ter Beek, D. Ničković. Vienna, Austria, September 2–3, 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

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

4. 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

5. Шкарупило В.В., Євдокимов В.Ф., Душеба В.В. Застосування формальних методів для перевірки систем критичного призначення. *Вчені записки ТНУ імені В.І. Вернадського*. 2019. Том 30 (69). Ч. 1. № 6. С. 188–193.

6. Lamport L. Checking a multithreaded algorithm with + CAL. *Distributed Computing, DISC'06* : Proceedings of the 20th international conference (Stockholm, Sweden, September 18-20, 2006). 2006. P. 151–163. DOI: https://doi.org/10.1007/11864219_11

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. №. 1. P. 145–152.

8. An Approach to increase the Effectiveness of TLC Verification with Respect to the Concurrent Structure of TLA+ Specification / V.V. Shkarupylo et al. *International Journal of Software Engineering and Computer Systems*. 2018. Vol. 4. №. 1. P. 48–60.

9. Карпов Ю.Г. Model Checking. Верификация параллельных и распределенных программных систем. Санкт-Петербург : БХВ-Петербург, 2010. 560 с.

10. Introduction to algorithms: 3rd ed. / T.H. Cormen et al. Cambridge, Massachusetts : The MIT Press, 2009. 1320 p.

11. Метод синтезу формальних специфікацій на основі трійок Хоара / В.В. Шкарупило та ін. *Наукові праці ДонНТУ, Серія «Інформатика, кібернетика та обчислювальна техніка»*. 2020. № 1 (30). С. 49–57.

Shkarupylo V.V., Chemerys O.A., Dusheba V.V. FORMAL VERIFICATION PROBLEM SOLVED WITH THE MODEL CHECKING METHOD SPATIAL COMPLEXITY ESTIMATION

Current state of computer systems engineering process can be described as the one with an active involvement of various formal methods and tools. Named methods and tools are commonly applied to increase the confidence of the developers in the correctness of the resulting artifacts of engineering process, design solutions in particular. Correctness is approached from the standpoint of compliance with formalized requirements to the characteristics of the system under development. Safety-critical systems are of particular attention here. The unplanned scenarios of these systems functioning can potentially lead to the significant critical consequences. From this viewpoint, eliminating the preconditions for such scenarios is an important factor increasing the confidence in the results of engineering process. This can be achieved through formal methods involvement at the design stage of named process. Taking into consideration the complexity of the software constituent of named systems, the issue of formal methods application automation becomes of paramount importance. This criterion is satisfied with the model checking methods (model checkers). However, these methods provoke significant demands to the hardware of computing platform, where the model checkers are supposed to be implemented.

Paper is devoted to an experimental study of the widespread proved model checker – TLC (TLA Checker), with respect to the spatial complexity of model checking task resolved. As a distinctive feature prompting the diversification between method implementations, the algorithm of state space search is considered: BFS (Breadth-first Search) and DFS (Depth-first Search) algorithms of graph theory. The state space is predefined with formal specification. Grounding on the results obtained, it has been shown that application of DFS-based implementation of TLC method is paired with higher requirements to the capacity of random access memory available.

Key words: *BFS, DFS, TLC, model checking, formal specification, verification, safety-critical system.*