

МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

УДК 004.41/42

В. Я. НЕЧАЙ^{1*}, Д. О. ВОЛОШИН^{2*}, О. І. НЕЖУМІРА^{3*}

^{1*}Каф. «Комп'ютерні інформаційні технології», Дніпропетровський національний університет залізничного транспорту імені академіка В. Лазаряна, вул. Лазаряна, 2, Дніпро, Україна, 49010, тел. +38 (099) 904 29 64, ел. пошта vnetchay@i.ua, ORCID 0000-0001-5358-4335

^{2*}Каф. «Комп'ютерні інформаційні технології», Дніпропетровський національний університет залізничного транспорту імені академіка В. Лазаряна, вул. Лазаряна, 2, Дніпро, Україна, 49010, тел. +38 (093) 311 37 22, ел. пошта dmtro.voloshyn@gmail.com, ORCID 0000-0003-1250-3206

^{3*}Каф. «Комп'ютерні інформаційні технології», Дніпропетровський національний університет залізничного транспорту імені академіка В. Лазаряна, вул. Лазаряна, 2, Дніпро, Україна, 49010, тел. +38 (066) 572 48 18, ел. пошта nezhumira.ok@gmail.com, ORCID 0000-0002-3905-2591

ДОСЛІДЖЕННЯ ГРАНИЧНИХ ЧАСОВИХ ПОКАЗНИКІВ ПРОГРАМНИХ СЕРЕДОВИЩ ОПЕРАЦІЙНИХ СИСТЕМ РЕАЛЬНОГО ЧАСУ

Мета. Більш ніж двадцятирічний термін дослідження оцінки найгіршого часу виконання програм (Worst Case Execution Time, WCET) призвів до розробки багатьох методів. До цього часу немає остаточних висновків щодо застосування того чи іншого методу. Тому метою нашої роботи є визначення можливості використання гібридного методу для оцінки WCET у системах реального часу. **Методика.** З'ясування оцінки WCET для гібридного методу полягає в тому, щоб провести синтаксичний аналіз вхідного коду мовою програмування C++ та, побудувавши граф потоку керування, отримати час виконання його базових блоків. Знайшовши найдовший шлях графа, оцінити час виконання цього шляху й отримати оцінку WCET. Для пошуку найдовшого за часом виконання шляху був обраний зворотний алгоритм Дейкстри. Після чого було виконане порівняння граничних часових оцінок, отриманих статичним і гібридним методами, а також аналіз розбіжності цих результатів. **Результати.** Визначення найгіршого часу виконання програм найбільш важливе для задач «жорсткого реального часу». Недооцінка цього показника може призвести до катастрофічних наслідків. А переоцінка – до суттєвої перевитрати ресурсів. Тому була проведена оцінка WCET за допомогою статичного й динамічного методів, яка дає змогу визначити, що результати, отримані двома методами, непогано корелюються. Для розглянутого класу задач час виконання WCET може бути визначений за допомогою гібридного методу. Додавання додаткового часу в розмірі 15–20 % до результатів, отриманих гібридним методом, дозволяє отримати безпечну оцінку WCET у системах реального часу. **Наукова новизна.** Гібридний метод існує вже певний час, але надійність його використання ще не була достатньо вивчена. Дане дослідження робить крок у визначенні практичної застосованості гібридного методу для часової оцінки задач реального часу. **Практична значимість.** Результати дослідження дозволяють дійти висновків, що гібридний метод може бути використаний для отримання оцінки WCET у задачах «жорсткого реального часу». А деякий невідомий на даний момент вплив середовища операційної системи реального часу (ОС РЧ) може бути врахований додаванням додаткового часу.

Ключові слова: граничний час виконання програм; WCET; операційні системи реального часу; задачі реального часу; гібридний метод; статичний метод; динамічний метод

Вступ

Останнім часом із поширенням автоматизації керування фізичними процесами все більше

уваги приділяється операційним системам реального часу. Такі системи можуть керувати пристроями як зовні, так і бути вбудованими. При цьому всі розв'язувані задачі можна поді-

МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

лити на три категорії: задачі не критичні до часу їх розв'язання, задачі м'якого реального часу (коли перевищення часу розв'язання небажане, але припустиме) і задачі жорсткого реального часу (коли перевищення часу розв'язання може призвести до катастрофічних наслідків). Тому оцінка граничних часових характеристик WCET під час розробки програм реального часу є обов'язковою процедурою [6].

WCET є однією з цілей процесу верифікації сертифікованого авіаційного програмного забезпечення кваліфікаційних вимог DO-178B/C, які використовують у всьому світі. Розв'язання цієї задачі дуже складне, а іноді й неможливе. Здійснені спроби призвели до появи методів оцінки WCET. До всіх методів ставлять дві вимоги: оцінка WCET не може бути меншою за відведений час, а її суттєве перевищення може призвести до збільшення затрат на розробку програмного забезпечення [5].

Усі методи оцінки WCET поділяють на ста-

тичні й динамічні. Час виконання програм можна знайти з використанням моделей обчислювальних середовищ (статичні методи) або за допомогою вимірів, які проводять під час виконання програм (динамічні). Недосконалість цих підходів призвела до появи гібридних методів, що поєднують у собі елементи двох попередніх. Пошук методів оцінки WCET триває понад двадцять років. У роботі [2] наведені результати огляду наявних методів, які відображені на рис. 1.

Традиційним методом оцінки часових характеристик є динамічний метод (статистичний аналіз результатів багатьох випробувань роботи програми за різних вхідних даних). Цей метод не завжди гарантує отримання граничних оцінок часу, оскільки всі можливі комбінації вхідних даних можуть бути невідомі [9]. Також він є досить затратним у разі спроби забезпечення достатнього покриття коду.



Рис. 1. Методи визначення WCET
Fig. 1. Methods for determining WCET

Статичний метод дозволяє оцінити найгірший час виконання за допомогою аналізу вихідного коду програми й оцінки всіх шляхів її виконання. За ним можна оцінити часові характеристики без виконання самої програми, але йому потрібні для цього комплексні абстрактні моделі апаратних і програмних засобів. Недоліки методу наступні:

- неадекватні моделі можуть давати небезпечні оцінки WCET;
- моделі не враховують усіх властивостей сучасних процесорів;
- ускладнена переносимість цих інструментів у різні середовища [2].

Гібридний метод поєднує переваги динамічного й статичного методів і дозволяє отримати

МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

наближену часову оцінку без використання складних абстрактних моделей.

Специфіка гібридного методу оцінки граничних часових характеристик полягає у застосуванні таких технік: динамічні техніки використовують для визначення часу виконання окремих сегментів, з яких складається програма, а техніки статичного аналізу – для пошуку найдовшого шляху виконання, враховуючи оцінки окремих сегментів шляху, які були отримані динамічними техніками [8]. Головною перевагою гібридного методу над динамічним є те, що гібридний метод дозволяє отримати найдовший шлях виконання програми, а динамічний не дає такої гарантії. Перевага ж гібридного методу над статичним полягає у відсутності потреби використовувати абстрактні моделі, що дозволяє не прив'язуватися до конкретної апаратно-програмної конфігурації.

Основною проблемою під час використання гібридного методу є нестатична природа програмного середовища операційної системи, в якій виконується програма [11]. Оскільки гібридний метод застосовує динамічні техніки для визначення часу виконання окремих сегментів, він не може гарантувати, що на момент виконання замірів стан системи був найгірший з усіх можливих [3]. Однак на практиці, для отримання безпечної граничної оцінки, вплив цих динамічних характеристик середовища може бути нівельований додаванням до отриманого часу виконання додаткового часу погіршеності.

Мета

На підставі порівняльного аналізу результатів гібридного і статичного методів на тестовій вибірці програм у конкретній операційній системі на заданому апаратному забезпеченні визначити можливість застосування гібридного методу в середовищі операційних систем реального часу (ОС РЧ) для оцінки WCET.

Об'єктом даного дослідження є гібридний метод, який використовують для оцінки граничного часу виконання програм, орієнтованих на операційні системи реального часу.

Предметом дослідження є часові показники виконання програм, отримані в середовищі ОС РЧ різними засобами.

Методика

Отримання часової оцінки передбачає розробку власного програмного продукту. Базовий алгоритм роботи такої програми має наступні кроки:

- провести парсинг вихідного C++ коду;
- побудувати граф потоку керування;
- отримати час виконання базових блоків графа потоку керування;
- присвоїти час виконання блоків як вагу дуг у графі;
- знайти найдовший шлях графа, використовуючи відомі алгоритми;
- оцінити час виконання цього шляху й отримати оцінку WCET.

Результати

Дослідження рівня впливу програмного середовища операційних систем реального часу і застосованості гібридного методу для оцінки граничного часу виконання програм зводиться до порівняння розбіжності результатів оцінки WCET статичним і гібридним методами на тестовій вибірці програм.

Статичний метод використовують для отримання безпечної оцінки WCET, яка враховує вплив апаратно-програмного забезпечення і відображає реальний максимально можливий час виконання програми. Для отримання оцінки статичним методом використовують інструмент aiT від компанії AbsInt [12]. Оскільки статичний метод потребує наявності абстрактної моделі апаратного забезпечення, в якості тестового стенда була обрана конфігурація з процесором, який є в списку змодельованих, а саме ARM7.

Парсинг вихідного коду C++ належить до задач текстової обробки інформації. Одним із найбільш гнучких і зручних засобів обробки й розпізнавання текстової інформації є регулярні вирази, але для синтаксичного аналізу програмного коду їх можливостей недостатньо. Для таких задач, як розпізнавання програмного коду, використовують парсер-генератори [10]. Тому для реалізації розпізнавання вихідного коду C++ була обрана бібліотека Boost.Xpressive, яка поєднує гнучкість і зручність регулярних виразів і потужність парсер-генераторів, що дозволить побудувати зручну і негроміздку граматику для розпізнавання коду C++.

МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

Пошук найдовшого шляху за допомогою статичного методу може бути виконаний за алгоритмом, вказаним у [1].

Побудова графа потоку керування відбувається після розпізнавання коду з використанням розпізнаних токенів (елементів мови) і контексту переходів для визначення базових блоків як вершин та їх подальшого зв'язку між собою дугами (рис. 2).

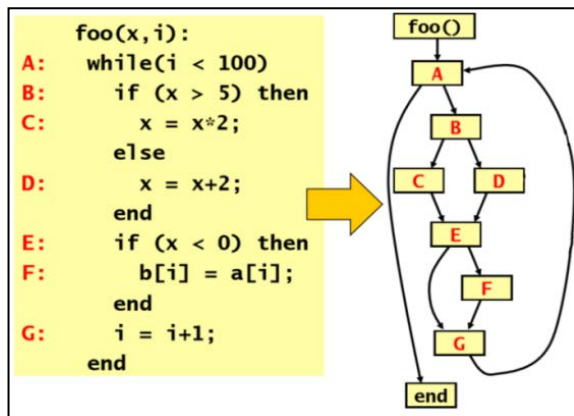


Рис. 2. Побудова графа потоку керування

Fig. 2. Constructing a control flow graph

Задача оцінки часу виконання базових блоків графа потоку керування в гібридному методі може бути вирішена за допомогою замірів часу на старті і в кінці цього блока або через визначення кількості процесорних тактів, які необхідні для виконання цього блока [8].

Під час реалізації гібридного методу надана перевага другому способу, оскільки він є більш гнучким і краще підходить для врахування погрішності часової оцінки (рис. 3).

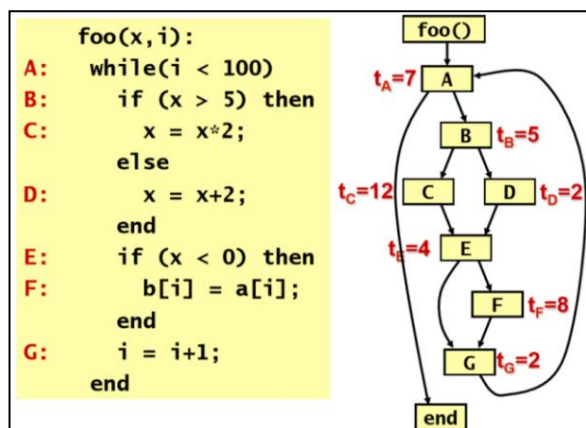


Рис. 3. CFG з розрахованими оцінками

Fig. 3. CFG with calculated estimates

Отриманий граф потоку керування (CFG) і часові оцінки базових блоків дозволяють знайти найдовший за часом шлях виконання [4]. Для цього можна використати будь-який зворотний алгоритм пошуку найкоротшого шляху у зваженому орієнтованому графі. Для вирішення задачі був обраний зворотний алгоритм Дейкстри (рис. 4).

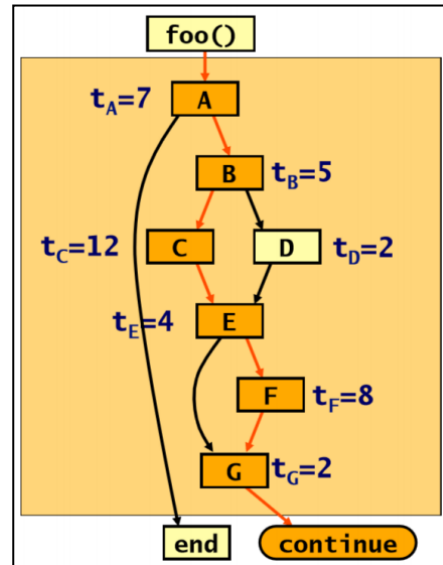


Рис. 4. Пошук найдовшого шляху в CFG

Fig. 4. Searching for the longest way in CFG

Після знаходження найдовшого шляху, можна оцінити граничний час виконання програми. Для цього визначають входні параметри, які сприяють проходженню найдовшого шляху в графі, а потім проводять багаторазові заміри часу виконання цього шляху на цільовому обладнанні. З усіх замірів вибирають найбільший, до нього додають додатковий час, й отриманий сумарний час буде відображати розраховану гібридним методом оцінку WCET для заданої програми.

Дані для дослідження будуть являти собою набір розрахованих статичним і гібридним методами WCET оцінок для тестової вибірки програм. Тестові програми та умови, в яких їх будуть виконувати, підібрані так, щоб максимально виділити різницю між теоретичним і фактичним граничним часом на вибірці програм за різними сценаріями виконання. Проаналізувавши рівень відхилення вимірюваного часу від теоретичного та визначивши кореляцію між ними, можна зробити висновок щодо застосованості

МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

гібридного методу для часової оцінки й ефективності використання додаткового часу погрішності для покриття впливу програмного середовища ОС РЧ на час виконання програм.

Отримані статичним і гібридним методами оцінки WCET для тестованих програм [7] наведені в табл. 1.

Порівняльний аналіз теоретичного й фактичного часу виконання програм дозволив оцінити рівень впливу програмного середовища операційних систем реального часу на час виконання програм. За результатами можна побачити, що рівень розбіжності часових оцінок у статичному й гібридному методах майже не

залежить від складності тестової програми й загального часу її виконання. Для розглянутих прикладів програм рівень розбіжності не перевищує 15 %. Тому вплив програмного середовища ОС РЧ, хоч і має динамічну природу, може бути приблизно нівельований додатковим часом у розмірі 15–20 % від отриманої гібридним методом оцінки. Для деяких програм такий показник виявиться доволі песимістичним і призведе до зайвого резервування ресурсів, але це є виправданою платою за безпечну часову оцінку в разі неможливості використання статичних методів.

Таблиця 1

Оцінки WCET, отримані статичним і гібридним методами

Table 1

WCET estimates obtained by static and hybrid methods

Тестова програма	Кількість вузлів	WCET _{static}	WCET _{hybrid}	Розбіжність
Constant Loop-1	3	221	194	+12 %
Constant Loop-2	3	88	79	+10 %
Branches-1	4	41	39	+3 %
Branches-2	6	51	48	+4 %
ContextLoop-1	5	686	634	+7 %
ContextLoop-2	5	695	640	+8 %
CntSumMat	8	26 572	23 118	+13 %
MatMult	7	523 599	466 003	+11 %

Наукова новизна та практична значимість

Необхідність використання абстрактних моделей і наявна велика різноманітність апаратних засобів та операційних систем реального часу інколи роблять майже неможливим використання статичних методів для оцінки найгіршого часу виконання програм. Тому на практиці для оцінки використовують менш точні динамічні методи з замірами часу. В якості альтернативи статичному методу був розроблений гібридний метод оцінки, але його застосованість для оцінки критичних до часу виконання систем недостатньо визначена.

Наукова й практична значимість цього дослідження полягає в тому, що його результати дозволяють отримати узагальнене уявлення щодо ступеня впливу програмного середовища

операційних систем на час виконання програм і дають відповідь щодо можливості та надійності використання гібридного методу з додаванням додаткового часу для отримання безпечної оцінки найгіршого часу виконання програм для ОС РЧ.

Висновки

Проведене дослідження дало рекомендації щодо застосування гібридного методу часової оцінки WCET. Було визначено, що вплив програмного середовища ОС РЧ, хоч і має динамічну природу, може бути компенсований додатковим часом погрішності в розмірі 15–20 % від оцінки, отриманої гібридним методом. Таким чином, гібридний метод із додаванням додаткового часу дозволяє отримати безпечну оцінку WCET, і його можна застосовувати для часової верифікації задач реального часу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Балашов, В. В. Встроенные информационно-управляющие системы реального времени. Лекция 3: Оценка наихудшего времени выполнения программ (WCET) [Электронный ресурс] / В. В. Балашов ; каф. АСВК, Лаборатория Вычислительных Комплексов, МГУ им. В. М. Ломоносова. – Режим доступа: http://lvk.cs.msu.su/~sveta/Балашов_3_Оценка_WCET.pdf. – Назва з екрана. – Перевірено : 31.05.2018.
2. Козырев, В. П. Методы оценки времени выполнения в системах реального времени / В. П. Козырев // Программирование. – 2016. – Т. 42, № 1. – С. 39–50.
3. Betts, A. Hybrid Measurement-Based WCET Analysis using Instrumentation Point Graphs : PhD. Thesis / A. Betts ; University of York. – Heslington, 2010. – 208 p.
4. Corti, M. Approximation of Worst-Case Execution Time for Preemptive Multitasking Systems / Matteo Corti, Roberto Brega, Thomas Gross // Languages, Compilers, and Tools for Embedded Systems. – Berlin, 2001. – P. 178–198. doi: 10.1007/3-540-45245-1_12
5. Damschen, M. Extending the WCET Problem to Optimize for Runtime-Reconfigurable Processors / Marvin Damschen, Lars Bauer, Jörg Henkel // ACM Transactions on Architecture and Code Optimization. – 2016. – Vol. 13. – Iss. 4. – P. 1–24. doi: 10.1145/3014059
6. Damschen, M. Timing Analysis of Tasks on Runtime Reconfigurable Processors / Marvin Damschen, Lars Bauer, Jorg Henkel // IEEE Transactions on Very Large Scale Integration (VLSI) Systems. – 2017. – Vol. 25. – Iss. 1. – P. 294–307. doi: 10.1109/TVLSI.2016.2572304
7. Fellger, W. Comparison of WCET Tools / W. Fellger, S. Gepperth, F. Krause ; Institute of Software Technology University of Stuttgart. – Stuttgart, 2011. – 45 p.
8. Lokuciejewski, P. Worst-Case Execution Time Aware Compilation Techniques for Real-Time Systems / P. Lokuciejewski, P. Marwedel. – Dordrecht : Springer, 2011. – 262 p. doi: 10.1007/978-90-481-9929-7
9. Puschner, P. Testing the results of static worst-case execution-time analysis / P. Puschner, R. Nossal // Real-Time Systems Symposium : Proc. of the 19th Conf. (Washington, December 02–04, 1998). – Washington, 1999. – P. 134–143. doi: 10.1109/real.1998.739738
10. Sewell, T. High-assurance timing analysis for a high-assurance real-time operating system / T. Sewell, F. Kam, G. Heiser // Real-Time Systems. – 2017. – Vol. 53. – Iss. 5. – P. 812–853. doi: 10.1007/s11241-017-9286-3
11. Stappert, F. Complete worst-case execution time analysis of straight-line hard real-time programs / F. Stappert, P. Altenbernd // Journal of Systems Architecture. – 2000. – Vol. 46. – Iss. 4. – P. 339–355. doi: 10.1016/S1383-7621(99)00010-7
12. Wilhelm, R. The worst-case execution time problem – overview of methods and survey of tools / R. Wilhelm, J. Engblom, A. Ermedahl // Transaction on Embedded Computing Systems. – 2008. – Vol. 7. – Iss. 3. P. 1–53. doi: 10.1145/1347375.1347389

В. Я. НЕЧАЙ^{1*}, Д. А. ВОЛОШИН^{2*}, А. И. НЕЖУМИРА^{3*}

^{1*}Каф. «Компьютерные информационные технологии», Днепропетровский национальный университет железнодорожного транспорта имени академика В. Лазаряна, ул. Лазаряна, 2, Днипро, Украина, 49010, тел. +38 (099) 904 29 64, эл. почта vnetchay@i.ua, ORCID 0000-0001-5358-4335

^{2*}Каф. «Компьютерные информационные технологии», Днепропетровский национальный университет железнодорожного транспорта имени академика В. Лазаряна, ул. Лазаряна, 2, Днипро, Украина, 49010, тел. +38 (093) 311 37 22, эл. почта dmtro.voloshyn@gmail.com, ORCID 0000-0003-1250-3206

^{3*}Каф. «Компьютерные информационные технологии», Днепропетровский национальный университет железнодорожного транспорта имени академика В. Лазаряна, ул. Лазаряна, 2, Днипро, Украина, 49010, тел. +38 (066) 572 48 18, эл. почта nezhumira.ok@gmail.com, ORCID 0000-0002-3905-2591

ИССЛЕДОВАНИЕ ПРЕДЕЛЬНЫХ ВРЕМЕННЫХ ПОКАЗАТЕЛЕЙ ПРОГРАММНЫХ СРЕД ОПЕРАЦИОННЫХ СИСТЕМ РЕАЛЬНОГО ВРЕМЕНИ

Цель. Более чем двадцатилетний срок исследования оценки наихудшего времени выполнения программ (Worst Case Execution Time, WCET) привел к разработке многих методов. До сих пор нет окончательных выводов относительно применения того или иного метода. Поэтому целью нашей работы является определение возможности использования гибридного метода для оценки WCET в системах реального времени.

МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

Методика. Выяснение оценки WCET для гибридного метода заключается в том, чтобы провести синтаксический анализ на языке программирования C++ и, построив граф потока управления, получить время выполнения его базовых блоков. Найдя самый длинный путь графа, оценить время выполнения этого пути и получить оценку WCET. Для поиска самого длинного по времени выполнения пути был выбран обратный алгоритм Дейкстры. После чего было выполнено сравнение предельных временных оценок, полученных статическим и гибридным методами, а также анализ расхождения этих результатов. **Результаты.** Определение худшего времени выполнения программ наиболее важно для задач «жесткого реального времени». Недооценка этого показателя может привести к катастрофическим последствиям. А переоценка – к существенному перерасходу ресурсов. Поэтому была проведена оценка WCET с помощью статического и динамического методов, которая дала возможность определить, что результаты, полученные двумя методами, неплохо коррелируют. Для рассматриваемого класса задач время выполнения WCET может быть определено с помощью гибридного метода. Добавление дополнительного времени в размере 15–20 % к результатам, полученным гибридным методом, позволяет получить безопасную оценку WCET в системах реального времени. **Научная новизна.** Гибридный метод существует уже некоторое время, но надежность его использования еще не была достаточно изучена. Данное исследование делает шаг в определении практической применимости гибридного метода для временной оценки задач реального времени. **Практическая значимость.** Результаты исследования позволяют сделать выводы, что гибридный метод может быть использован для получения WCET в задачах «жесткого реального времени». А некоторое неизвестное на данный момент влияние среды операционных систем реального времени (ОС РВ) может быть учтено добавлением дополнительного времени.

Ключевые слова: предельное время выполнения программ; WCET; операционные системы реального времени; задачи реального времени; гибридный метод; статический метод; динамический метод

V. Y. NECHAY¹*, D. O. VOLOSHYN²*, O. I. NEZHUMIRA³*

¹*Dep. «Computer Information Technologies», Dnipropetrovsk National University of Railway Transport named after Academician V. Lazaryan, Lazaryan St., 2, Dnipro, Ukraine, 49010, tel. +38 (099) 904 29 64, e-mail vnechay@i.ua, ORCID 0000-0001-5358-4335

²*Dep. «Computer Information Technologies», Dnipropetrovsk National University of Railway Transport named after Academician V. Lazaryan, Lazaryan St., 2, Dnipro, Ukraine, 49010, tel. +38 (093) 311 37 22, e-mail dmtro.voloshyn@gmail.com, ORCID 0000-0003-1250-3206

³*Dep. «Computer Information Technologies», Dnipropetrovsk National University of Railway Transport named after Academician V. Lazaryan, Lazaryan St., 2, Dnipro, Ukraine, 49010, tel. +38 (066) 572 48 18, e-mail nezhumira.ok@gmail.com, ORCID 0000-0002-3905-2591

RESEARCH OF LIMIT TIME INDICES OF SOFTWARE ENVIRONMENTS OF REAL TIME OPERATING SYSTEMS

Purpose. More than 20 years of the Worst Case Execution Time (WCET) studies have led to the development of many methods for its evaluation. So far, there are no definitive conclusions about usage of these methods. Therefore, the purpose of this paper is to determine the possibility of using a hybrid method for estimating WCET in real-time systems. **Methodology.** The approach for evaluating WCET for a hybrid method is to parse the input code in the C++ programming language and, after constructing the control flow graph, get the execution time of its base blocks. After finding the longest way of the graph, to estimate the time of execution of this way and get the WCET evaluation. To find the longest-running way, the reverse Dijkstra algorithm was chosen. After that, there was made a comparison of the limiting time estimations that were obtained by static and hybrid methods, as well as an analysis of the discrepancy between these results. **Findings.** Determining the worst execution time of programs is most important for «hard real-time» tasks. Underestimation of this indicator can lead to catastrophic consequences. An overestimation – to a significant overexpenditure of resources. Therefore, WCET was evaluated using static and dynamic methods. It was determined that the results obtained by the two methods correlate well. For the class of tasks that are under consideration, WCET execution time can be determined using a hybrid method. Adding an extra time of 15–20 % to the results, obtained by the hybrid method, allows getting a foolproof estimate of WCET in real-time systems. **Originality.** The hybrid method has been existing for some time, but the reliability of its use has not been sufficiently studied yet. The given study makes a step in determining the practical applicability of the hybrid method for time estimation of real-time tasks. **Practical value.** The results of this research allow us to conclude that a hybrid

МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

method could be used for obtaining WCET in "hard real-time" tasks. Furthermore, some unknown at this moment influence of the OS RT environment can be taken into account by adding extra time.

Keywords: worst case execution time; WCET; real-time operating systems; real-time tasks; hybrid method; static method; dynamic method

REFERENCES

1. Balashov, V. V. (undated). *Vstroennyye informatsionno-upravlyayushchie sistemy realnogo vremeni. Lektsiya 3: Otsenka naikhudshego vremeni vypolneniya programm (WCET)*. Retrieved from http://lvk.cs.msu.su/~sveta/Балашов_3_Оценка_WCET.pdf (in Russian)
2. Kozyrev, V. P. (2016). Estimation of the execution time in real-time systems. *Programming*, 42(1), 39-50. (in Russian)
3. Betts, A. (2010). *Hybrid Measurement-Based WCET Analysis using Instrumentation Point Graphs. PhD thesis*. University of York, Heslington. (in English)
4. Corti, M., Brega, R., & Gross, T. (2001). Approximation of Worst-Case Execution Time for Preemptive Multi-tasking Systems. In *Languages, Compilers, and Tools for Embedded Systems. Lecture Notes in Computer Science* (pp. 178-198). Berlin. doi: 10.1007/3-540-45245-1_12 (in English)
5. Damschen, M., Bauer, L., & Henkel, J. (2016). Extending the WCET Problem to Optimize for Runtime-Reconfigurable Processors. *ACM Transactions on Architecture and Code Optimization*, 13(4), 1-24. doi: 10.1145/3014059 (in English)
6. Damschen, M., Bauer, L., & Henkel, J. (2017). Timing Analysis of Tasks on Runtime Reconfigurable Processors. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 25(1), 294-307. doi: 10.1109/TVLSI.2016.2572304 (in English)
7. Fellger, W., Gepperth, S., & Krause, F. (2011). *Comparison of WCET Tools: Fachstudie Nr. 142*. Stuttgart: Institute of Software Technology University of Stuttgart. (in English)
8. Lokuciejewski, P., & Marwedel, P. (2011). *Worst-Case Execution Time Aware Compilation Techniques for Real-Time Systems*. New York: Springer (in English)
9. Puschner, P., & Nossal, R. (1998). Testing the results of static worst-case execution-time analysis. *Proceedings of the IEEE Real-Time Systems Symposium, December 02-04*, 134-143 (in English)
10. Sewell, T., Kam, F., & Heiser, G. (2017). High-assurance timing analysis for a high-assurance real-time operating system. *Real-Time Systems*, 53(5), 812-853. doi: 10.1007/s11241-017-9286-3 (in English)
11. Stappert, F., & Altenbernd, P. (2000). Complete worst-case execution time analysis of straight-line hard real-time programs. *Journal of Systems Architecture*, 46(4), 335-339. doi: 10.1016/S1383-7621(99)00010-7 (in English)
12. Wilhelm, R., Engblom, J., & Ermedahl, A. (2008). The worst-case execution time problem – overview of methods and survey of tools. *Transaction on Embedded Computing Systems*, 7(3), 1-53. doi: 10.1145/1347375.1347389 (in English)

Стаття рекомендована до публікації д.т.н., проф. В. І. Шинкаренком (Україна)

Надійшла до редколегії: 22.02.2018

Прийнята до друку: 30.05.2018