

Dr hab. inż. Ireneusz Józwiak, prof. PWr.
Wydział Informatyki i Telekomunikacji
Politechnika Wrocławska
Wybrzeże Wyspiańskiego 27
50-370 Wrocław

Recenzja rozprawy doktorskiej mgra inż. Damiana Giebasa pt. „Metoda wykrywania konfliktów zasobowych w aplikacjach wielowątkowych”

Promotor: dr hab. inż. Grzegorz Bocewicz, prof. PK

Promotor pomocniczy: dr inż. Rafał Wojszczyk

Zlecniodawcą wykonania recenzji rozprawy doktorskiej mgra inż. Damiana Giebasa jest dr hab. inż. Błażej Bałas, prof. PK, Prorektor ds. Nauki Politechniki Koszalińskiej na podstawie Uchwały nr 38/2022 Senatu Politechniki Koszalińskiej z dnia 27 kwietnia 2022 roku w sprawie powołania na recenzenta rozprawy doktorskiej mgra inż. Damiana Giebasa – pismo o sygn.. akt PK/WEiI/11/04/2022 z dnia 29 kwietnia 2022 roku

Uwagi wstępne

Przedstawiona do recenzji rozprawa doktorska Pana mgra inż. Damiana Giebasa pt. :
Metoda wykrywania konfliktów zasobowych w aplikacjach wielowątkowych
obejmuje w swej części merytorycznej wstęp, pięć rozdziałów oraz podsumowanie i zawiera się na 108 stronach. Pozostałe strony obejmują literaturę oraz spis rysunków, tabel, wykaz skrótów, symboli, oznaczeń i załączników. Literatura obejmuje 114 pozycji. W znacznej większości są to publikacje zagraniczne oraz źródła internetowe. Całość rozprawy doktorskiej zawarta jest na 152 stronach. Do rozprawy została dołączona płyta CD.

Zgodnie z art. 13.1. Ustawy z dnia 14 marca 2003 r. o stopniach naukowych i tytule naukowym oraz o stopniach i tytule w zakresie sztuki (Dz. U. z 2003, nr 65, poz. 595, z późn. zm.), rozprawa doktorska powinna „stanowić oryginalne rozwiązanie problemu naukowego (...) oraz wykazywać ogólną wiedzę teoretyczną kandydata w danej dyscyplinie naukowej”. Mając na uwadze ww. ustawę przyjęto następujące kryteria oceny: znaczenie tematyki doktoratu, poprawność struktury i języka pracy, jakość przeglądu literatury, poprawność w

sformułowaniu celów i hipotez badawczych, ocena oryginalności rozwiązania oraz poprawności wnioskowania.

Znaczenie podjętej tematyki

Autor zajął się tematem poszukiwania modelu, którego właściwości umożliwiłyby budowę efektywnej metody lokalizacji błędów aplikacji wielowątkowych. Autor przez efektywność rozumie uzyskanie lepszych od znanych w literaturze rezultatów zlokalizowania wszystkich błędów prowadzących do omawianych konfliktów zasobowych. Jednym z ważnych kryteriów było zmniejszenie kosztów analizy. Zaprezentowana metoda i jej implementacja jest jednym z argumentów na duże znaczenie podjętej tematyki badawczej z zakresu informatyki, gdyż efektywnie będzie wspierać programistów w zakresie lokalizacji błędów w programach wielowątkowych. Rozpatrywana tematyka badawcza jest zatem bardzo aktualna pod względem naukowym.

Poprawność struktury

W nauce przyjęto standardy poprawnej struktury publikacji prezentującej nowe metody. Prace z dziedziny nauk technicznych powinny być realizowane z następującą strukturą:

1. Wprowadzenie do problematyki wraz z streszczeniem pracy.
2. Przegląd literatury, ze wskazaniem braków i wad istniejących rozwiązań.
3. Wskazanie celów i hipotez badawczych.
4. Prezentacja nowej metody.
5. Przedstawienie badań udowadniających spełnienie celów i hipotez.
6. Podsumowanie.

Praca Doktoranta w pełni odzwierciedla tę strukturę. Autor zachował wszelkie wymogi zwyczajowe dotyczące struktury pracy. Na godne uwagi i pochwały zasługują szczególnie przejrzysty spis treści, zrozumiałe tytuły rozdziałów i ich rozsądna liczba w odniesieniu do objętości pracy.

Język pracy i opisy

Język pracy oceniam jako poprawny. Rozprawa posiada jednak drobne niedociągnięcia polonistyczne. Dla przykładu Autor powinien używać jednolitych nazw. W tekście pracy powołuje się na tabelę, a opis tabeli nad nią określa słowami tablica. Należy także zwrócić uwagę, że dla rzeczy przeliczalnych mówi się o liczbie, np. programów, a nie o ilości. Nie ma to jednak wpływu na wartość merytoryczną rozprawy.

Jakość przeglądu literatury

Rozprawa doktorska zawiera przegląd literatury z zakresu istniejących rozwiązań pozwalających na lokalizowanie błędów prowadzących do konfliktów zasobowych typu: szkodliwa rywalizacja, zakleszczenie, naruszenie niepodzielności i naruszenie porządku. Omawiana literatura została oceniona pod względem kryteriów jakościowych i ilościowych. Dużo uwagi poświęcił Doktorant analizie literaturowej metod i narzędzi do lokalizowania błędów powodujących szkodliwą rywalizację. Zwykle są one dedykowane dla określonych zastosowań, np. dla jąder systemów operacyjnych, sterowników urządzeń, czy też dla aplikacji uruchamianych w przestrzeni użytkownika. Autor metody te przeanalizował wyróżniając w nich trzy grupy: statyczne, dynamiczne i mieszane. W zależności od grupy, do której należy metoda, przeanalizował różne warunki dotyczące łatwości użycia, precyzji, wydajności i liczby lokalizowanych błędów. Wśród najczęściej spotykanych metod lokalizacji błędów prowadzących do szkodliwej rywalizacji są metody dedykowane konkretnym językom lub nawet rodzinom języków wykorzystujących wspólne środowisko uruchomieniowe. Autor przedstawił także wnikliwą analizę literaturową tych metod.

Odrębną grupę narzędzi stanowią metody i narzędzia pozwalające zapobiegać występowaniu konfliktów zasobowych. Konflikty szkodliwej rywalizacji zlokalizowane w tych programach także zostały przeanalizowane w pracy z punktu widzenia zastosowanych mechanizmów, które umożliwiały asynchroniczne operacje na wybranych obszarach pamięci.

Autor słusznie zauważył, że w dostępnej literaturze przedmiotu jest trend wzrastającej liczby publikacji opisujących błędy prowadzące do szkodliwej rywalizacji. Liczba tych publikacji systematycznie wzrastała w latach 2002 – 2011. Obecnie kształtuje to się w liczbie kilku prac rocznie. Większość z tych prac dotyczyła metod dynamicznej analizy programów pisanych w językach obiektowych lub sposobów testowania programów.

Doktorant słusznie zauważył, że z przeprowadzonego przeglądu literatury wynika, że wciąż podejmuje się próby opracowania efektywnej metody lokalizowania szkodliwej rywalizacji. Skutkuje to dużą różnorodnością proponowanych metod o różnej skuteczności. Wśród metod największa ich liczba skupia się na językach obiektowych. Żadne ze znanych w literaturze rozwiązań nie pozwala na lokalizowanie szkodliwej rywalizacji w procesie statycznej analizy kodu napisanego w języku C. Literatura tematu nie dostarcza także wielu informacji o modelach stosowanych w opisywanych rozwiązaniach.

Należy w tej recenzji podkreślić, że nie spotkałem wśród dotychczas recenzjowanych rozpraw doktorskich tak głębokiej i wnikliwej analizy literaturowej, jaka jest zawarta w tej rozprawie doktorskiej. Jest to wielką zasługą Autora rozprawy i świadczy o Jego profesjonalizmie.

Poprawność w sformułowaniu celów i hipotez badawczych

Autor w pracy postawił tezę, że istnieją modele kodu źródłowego aplikacji wielowątkowych umożliwiające wyprowadzenie warunków, których spełnienie może skutkować wystąpieniem konfliktów zasobowych typu: szkodliwa rywalizacja, zakleszczenie, naruszenie podzielności lub naruszenie porządku.

Postawiona teza, jak i sformułowany cel pracy pokazuje dużą aktualność podjętej tematyki badawczej w zakresie identyfikacji błędów aplikacji wielowątkowych. Celem przeprowadzonych eksperymentów było potwierdzenie lub zanegowanie postawionej tezy, którą można zawrzeć w pytaniu: czy zaproponowana metoda bazująca na opracowanym modelu aplikacji wielowątkowej umożliwia lokalizowanie błędów, które prowadzą do konfliktów zasobowych? Analiza otrzymanych wyników pozwoliła określić kryteria umożliwiające efektywne wsparcie procesu weryfikacji poprawności aplikacji wielowątkowych. Eksperymenty obejmowały weryfikację 19 aplikacji napisanych w języku C. Dla każdej z nich przeprowadzono proces lokalizowania konfliktów zasobowych. Otrzymane raporty zawierające informacje o zlokalizowanych błędach zostały zweryfikowane przez zespół ekspertów, którymi byli programiści w języku C.

Z powyższego wnioskuję, że nowością w pracy jest zaproponowanie metody bazującej na opracowanym modelu kodu źródłowego aplikacji wielowątkowej pozwalającej lokalizować błędy, które powodują konflikty zasobowe szkodliwej rywalizacji i zakleszczenia, jak też powodują naruszenie niepodzielności i porządku. Eksperymenty potwierdziły, że opracowana metoda wykrywa wszystkie występujące w kodzie błędy prowadzące do szkodliwej rywalizacji, zakleszczenia, naruszenia niepodzielności i porządku, co jest zagadnieniem nowym i innowacyjnym. W tym sensie była to teza badawcza.

Cel pracy jest sformułowany poprawnie, jako cel aplikacyjny. W pracy przedstawiono sposoby eliminacji błędów. Uzyskane wyniki spełniają warunki do wykorzystania zaproponowanej metody, jako narzędzia wspierającego zespół programistów w procesie lokalizacji błędów aplikacji wielowątkowych.

Ocena oryginalności rozwiązania i poprawności wnioskowania

Ocena implementacji metody została przedstawiona w rozdziale 6. Wyniki przeprowadzonych eksperymentów pozwoliły na stwierdzenie, że narzędzie, w którym zaimplementowano metodę może stanowić odstawę do budowy komercyjnego środowiska wspierającego proces lokalizacji błędów aplikacji wielowątkowych, w szczególności zintegrowanego z procesami stosowanymi w metodykach zwinnych.

Do głównych zalet opracowanego narzędzia należy zaliczyć jego skuteczność, co znaczy, że narzędzie zgłosiło wszystkie błędy prowadzące do oczekiwanych konfliktów zasobowych.

Poprawność wnioskowania została przedstawiona w części dotyczącej przeglądu literatury. W późniejszych rozdziałach badawczych i zastosowania metod Autor wykazuje się dobrym wnioskowaniem i poprawną matematyką.

Jednakże chciałbym przedstawić pewne uwagi krytyczne, które się nasunęły przy lekturze pracy. W obszernym przeglądzie literaturowym Autor podał wiele narzędzi do analizy kodu w celu wykrywania konfliktów będących tematem rozprawy. Natomiast już w samej treści badań Autor nie skorzystał z narzędzi, które wymienił do porównania się z nimi, jak np. LOCKSMITH, Współbieżny Graf Przepływu Sterowania, Bauhaus, DR-Frame. Drugą uwagą, którą chciałbym wyartykułować, to podjęcie tematu wykrywania konfliktów zasobowych w aplikacjach wielowątkowych napisanych w języku C z wykorzystaniem biblioteki pthread. Dlaczego Autor rozprawy nie rozpatrzył aplikacji napisanych w innych językach, jak np. Java, Python i in.? Brak tej analizy nie umniejsza wielkiej wartości rozprawy doktorskiej.

W mojej opinii praca jest bardzo dobrze napisana, język jest bardzo profesjonalny, przeprowadzone są logiczne wywody.

Osiągnięte rezultaty stanowią wielki wkład Autora rozprawy doktorskiej w rozwój informatyki w obszarze inżynierii oprogramowania, teorii języków programowania, przetwarzania współbieżnego i równoległego, systemów procesów współbieżnych itp. Ważnym podkreślenia jest fakt praktycznych wyników pracy poprzez opracowanie prototypu narzędzia, bazującego na opracowanej metodzie, wspierającego programistów w zakresie lokalizacji błędów wytwarzanych aplikacji wielowątkowych.

Wykonanie prototypu narzędzia w oparciu o zaproponowaną metodę stanowi w mojej opinii najważniejsze osiągnięcie Autora.

Podsumowanie – wnioski końcowe

Rozprawę doktorską Pana mgra inż. Damiana Giebasa pt. : **Metoda wykrywania konfliktów zasobowych w aplikacjach wielowątkowych** czytałem z wielką przyjemnością. Zaproponowane rozwiązania są ciekawe i stanowią wartość dodaną do nauki. Wyartykułowane w recenzji uwagi pracy nie umniejszają wielkiej wartości rozprawy. Głównymi zaletami pracy są oryginalność proponowanych rozwiązań, wdrożenie w warunkach rzeczywistych, możliwość zastosowania rozwiązania w innych wdrożeniach oraz łatwość obsługi rozwiązania.

W mojej opinii Doktorant wykazał dużą wiedzę z dyscypliny naukowej „Informatyka techniczna i telekomunikacja”, poprawnie przeprowadzał reguły wnioskowania oraz dobrze opisywał algorytmy i rozwiązania. Aplikacja zaproponowana przez Autora stanowi oryginalne rozwiązanie problemu naukowego weryfikacji aplikacji wielowątkowych. Jest ona również istotnym i aktualnym problemem informatycznym.

Działając na podstawie art. 179 ust. 1 Ustawy z dnia 3 lipca 2018 roku Przepisy wprowadzające ustawę – Prawo o szkolnictwie wyższym i nauce (Dz. U. z 2018, poz. 1669) oraz biorąc pod uwagę recenzję przedstawionej rozprawy doktorskiej mgra inż. Damiana Giebasa pt.: „Metoda wykrywania konfliktów zasobowych w aplikacjach wielowątkowych” uważam, że spełnia ona wymagania określone w Ustawie z dnia 14 marca 2003 r. o stopniach naukowych i tytule naukowym oraz o stopniach i tytule w zakresie sztuki (Dz. U. z 2003, nr 65, poz. 595, z późn. zm.), w dziedzinie nauk inżynieryjno – technicznych i dyscyplinie informatyka techniczna i telekomunikacja.

W konsekwencji wnioskuję o przyjęcie rozprawy doktorskiej oraz o dopuszczenie Pana mgra inż. Damiana Giebasa do dalszych etapów przewodu doktorskiego. Jednocześnie wnioskuję o wyróżnienie tej rozprawy doktorskiej.

