

OPINIA

o rozprawie doktorskiej mgr inż. Damiana Giebasa

Pan mgr inż. Damian Giebas od 2017 roku prowadzi badania związane z rozprawą doktorską pt.: „**Metoda wykrywania konfliktów zasobowych w aplikacjach wielowątkowych**”.

Celem rozprawy jest opracowanie oryginalnej metody lokalizowania błędów aplikacji wielowątkowych (ang. Races and Deadlocks – Locating (RD-L)) skutkujących możliwością wystąpienia tzw. konfliktów zasobowych. Tematyka związana z przetwarzaniem współbieżnym bazującym na współlistnieniu wielu wątków (znana i rozwijana od ponad 30 lat), jest inspirowana intensywnym rozwojem środowisk obliczeń rozproszonych implementowanych w technologiach gridowych umożliwiających implementację współbieżnie realizowanych programów równoległych. Pomimo upływu trzech dekad bez mała nic nie traci ze swojej aktualności. Wiele problemów z jej obszaru ciągle oczekuje na swoje, efektywne czasowo metody rozwiązania. O niesłabnącym zainteresowaniu tą tematyką świadczy chociażby liczba publikacji, która w ostatniej dekadzie przekroczyła 1500 pozycji (dane wg bazy Scopus).

Współczesne urządzenia elektroniczne coraz częściej wykorzystują procesory wielordzeniowe, które w sposób naturalny umożliwiają implementowanie aplikacji wielowątkowych. Implementacja ta jest realizowana przy użyciu języków programowania, które powstały zanim upowszechniły się procesory wielordzeniowe. Ich wykorzystanie w aplikacjach wielowątkowych niesie za sobą liczne zagrożenia będące konsekwencją występowania szeregu niekorzystnych zjawisk takich jak: szkodliwa rywalizacja (ang. race condition), zakleszczenie (ang. deadlock), naruszenie niepodzielności (ang. atomicity violation), czy też naruszenie porządku (ang. order violation). Zjawiska te składają się na wcześniej wspomniane konflikty zasobowe i są niezależne od użytego języka programowania. Ich wystąpienie może prowadzić między innymi do:

- niepoprawnego działania aplikacji - wynikającego z błędnych wyników przeprowadzonych obliczeń,
- zawieszenia pracy aplikacji - objawiającego się brakiem reakcji na akcje użytkownika,
- nieoczekiwanym zakończeniem pracy aplikacji.

Działania ukierunkowane na unikanie, zapobieganie oraz identyfikowanie skutków konfliktów zasobowych stanowią główny zakres badań w obszarach modelowania i projektowania metod przetwarzania współbieżnego. Należy zauważyć, że najczęściej wykorzystywane metody (bazujące na reprezentacjach: Grafu Przepływu Sterowania, Sieci Petriego, itp.) sprowadzają się głównie do testów (symulacji) działania aplikacji w różnych scenariuszach ich użycia - tzw. analizy dynamicznej. Z uwagi na fakt, że problemy związane z wykrywaniem błędów prowadzących do konfliktów zasobowych należą do klasy problemów NP-trudnych metody te ograniczają się do przeglądu arbitralnie wybranej reprezentatywnej grupy scenariuszy. Oznacza to, że w praktyce wielokrotnie dochodzi do sytuacji, w których błędy występujące w aplikacjach są niewykrywane przez wiele lat. Jednocześnie badania podejmujące problematykę identyfikacji konfliktów zasobowych w aplikacjach wielowątkowych implementujących metody analizy kodu źródłowego programu (tzw. analizy statycznej) należą do rzadkości tworząc swoją lukę badawczą. Nieliczne publikacje podejmujące tego typu tematykę noszą często wstępny, koncepcyjny charakter.

Możliwość wsparcia programistów w zakresie identyfikacji błędów prowadzących do konfliktów zasobowych na podstawie kodu źródłowego aplikacji wielowątkowej stanowi motywację prowadzonych przez Doktoranta badań. Przyjmuje się, że prowadzone badania ograniczone są do aplikacji napisanych w języku C i zastosowaniu biblioteki *pthread* (dostarczającej mechanizmy umożliwiające pisanie aplikacji wielowątkowych).

W tym kontekście, proponowana metoda RD-L pozwala wesprzeć programistów w zakresie kwestii związanych z pytaniem:

Czy dana aplikacja wielowątkowa jest wolna od błędów prowadzących do konfliktów zasobowych tj.: szkodliwej rywalizacji, zakleszczenia, naruszenia niepodzielności oraz naruszenia porządku?

Z przeprowadzonych badań Doktoranta wynika, że tego typu wsparcie uzależnione jest od tzw. warunków, spełnienie których pozwala zidentyfikować błędy występujące w kodzie źródłowym aplikacji (tzn. nieprawidłowo zapisanych fragmentów kodu źródłowego). Oznacza to, że osiągnięcie celu rozprawy wymaga opracowania modelu aplikacji wielowątkowych opi-

sującego strukturę wątków i realizowanych operacji; umożliwiającego identyfikację występujących w niej błędów. Istotą proponowanego podejścia jest wykorzystanie w opracowanym modelu informacji o przedziałach czasu współdzielonych jednocześnie przez operacje wielu wątków aplikacji. Zastosowanie tego typu „strukturalno-czasowej” reprezentacji pozwoliło na identyfikację błędów prowadzących do wybranych konfliktów zasobowych już na etapie analizy kodu programu.

Wyniki przeprowadzonych eksperymentów obliczeniowych weryfikujących efektywność opracowanej metody, potwierdzają zasadność podjętego kierunku badań oraz możliwość prowadzenia ww. wsparcia w trybie online. Opracowany prototyp aplikacji *rdao detector* (implementujący metodę RD-L), umożliwia lokalizację błędów aplikacji wielowątkowych o skali spotykanej w praktyce. Aplikację *rdao detector* zweryfikowano eksperymentalnie dla 19 aplikacji wielowątkowych pochodzących z otwartych źródeł (np. projekt aplikacji s-mptcp)

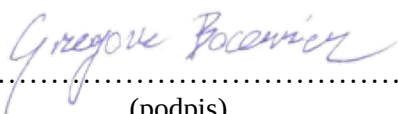
Do najważniejszych rezultatów przeprowadzonych badań należy zaliczyć:

- Opracowanie autorskiego modelu C_P kodu źródłowego aplikacji wielowątkowej.
- Sformułowanie i wykazanie czterech Twierdzeń, spełnienie których może prowadzić do konfliktów zasobowych typu: szkodliwa rywalizacja; zakleszczenie; naruszenie niepodzielności; naruszenie porządku.
- Opracowanie autorskiej metody RD-L umożliwiającej lokalizowanie błędów prowadzących do konfliktów w aplikacjach modelowanych w reprezentacji C_P implementującej wyprowadzone warunki lokalizowania konfliktów zasobowych.
- Opracowanie prototypu aplikacji *rdao detector* wspierającej programistę w zakresie wykrywania błędów w wytwarzanych aplikacjach wielowątkowych.
- Przeprowadzenie serii eksperymentów weryfikujących poprawność i przydatność metody RD-L w procesie wytwarzania aplikacji wielowątkowych.

Doktorant jest absolwentem Wydziału Elektroniki i Informatyki Politechniki Koszalińskiej, na którym obronił pracę magisterską (specjalność: Zastosowanie Systemów Baz Danych). Posiada 8-letni staż pracy jako programista języków Ruby, Python, JavaScript, C, C++ w firmach realizujących projekty informatyczne branży automotive i telekomunikacji. Jego czynna praca zawodowa umożliwia identyfikację potrzeb i niedoskonałości wykorzystywanych w praktyce metod wykrywania konfliktów zasobowych w aplikacjach wielowątkowych. Jego dorobek naukowy obejmuje 10 pozycji literaturowych wydanych w polsko- i anglojęzycz-

nych czasopismach naukowych oraz w materiałach krajowych i międzynarodowych konferencji naukowych. Warto zwrócić uwagę, że jest współautorem 3-ch prac wydanych w czasopismach z tzw. „listy filadelfijskiej”: *IEEE Access*, *Applied Sciences*. Wynik prowadzonych prac były przedstawiane na forum publicznym m.in. na konferencjach krajowych i międzynarodowych tj. „*International Conference on Distributed Computing and Artificial Intelligence*”, „*Krajowej Konferencji Studentów i Młodych Pracowników Nauki*”.

Uważam, że ww. wyniki badań wpisują się w zakres dyscypliny Informatyka Techniczna i Telekomunikacja i spełniają obowiązujące warunki umożliwiające dopuszczenie przedłożonej rozprawy doktorskiej do dalszych etapów postępowania.

.....

(podpis)