prof. dr hab. inż. Krzysztof Goczyła
Politechnika Gdańska
Wydział Elektroniki, Telekomunikacji i Informatyki
kris@eti.pg.edu.pl

Gdańsk, 12.09.2022

Recenzja rozprawy doktorskiej
mgr. inż. Bartosza Żurkowskiego
zatytułowanej

"Root Cause Analysis for Large-Scale Cloud-native Applications"
(„*Analiza źródeł defektów działania aplikacji dużej skali w chmurze obliczeniowej"*)

1. Wstęp

Niniejsza recenzja została sporządzona na podstawie pisma Przewodniczącego Rady Dyscypliny Informatyka Techniczna i Telekomunikacja Akademii Górniczo-Hutniczej im. Stanisława Staszica w Krakowie prof. dr hab. inż. Marka Kisiela-Dorohinickiego z dn. 13.06.2022 r., w którym zawarta jest informacja o powołaniu mnie na recenzenta ww. rozprawy doktorskiej. Do pisma dołączono tekst rozprawy.

2. Problem badawczy i jego znaczenie

Tematyka rozprawy dotyczy analizy źródeł defektów (ang. Root Cause Analysis – RCA) w chmurowych aplikacjach rozproszonych opartych na mikroserwisach. Rozprawa dotyczy jednego z istotnych zagadnień szeroko rozumianej współczesnej inżynierii oprogramowania, aktualnego z uwagi na silną tendencję do rozpraszania współczesnych aplikacji w środowiskach chmurowych. Jest to spowodowane potencjalnie dużym obciążeniem aplikacji, wielkim rozmiarem przetwarzanych danych, koniecznością zapewnienia ewolucji i innymi aspektami wymuszającymi rozproszenie i korzystanie z wirtualizacji. Praca koncentruje się na zagadnieniu wykrywania przyczyn defektów występujących nieuchronnie w pozornie prostej, bo hierarchicznej, a faktycznie skomplikowanej poprzez rozmiar i heterogeniczność architekturze. Analiza defektów, a w szczególności detekcja ich przyczyn źródłowych, jest zagadnieniem zasadniczym dla administrowania takimi aplikacjami, gdyż błąd powstały w jednym komponencie zwykle przenosi się poprzez warstwy systemu na inne komponenty, i to w sposób niekontrolowany, kaskadowy, trudny do przeanalizowania i wykrycia jego pierwotnej przyczyny j.

W tym kontekście Autor rozprawy podjął się zadania niełatwego i ambitnego, polegającego na opracowaniu, weryfikacji i walidacji całościowej metody polegającej na systematycznym analizowaniu przebiegu symptomów błędów i generowaniu najbardziej prawdopodobnych trajektorii błędów od ich pierwotnego źródła do efektów widocznych na interfejsie aplikacji. Metoda zaproponowana przez Autora jest wieloaspektowa. Wiąże ona ze sobą aspekty statystyczne, takie jak analiza współwystępowania symptomów i analiza odstępów czasowych pomiędzy ich występowaniem, z aspektami strukturalnymi, czyli z budową aplikacji. W efekcie zastosowania tej metody administrator analizowanego systemu otrzymuje istotne wskazówki co do źródeł błędów i może podjąć działania zaradcze. Warto podkreślić, że bez tego rodzaju wskazówek ręczna analiza źródeł błędów jest praktycznie niemożliwa z uwagi na poziom komplikacji architektury i funkcji

systemu oraz na przypadkowy charakter błędów i wynikającą z tego trudność w ich śledzeniu i odtworzeniu.

Podjęcie takiego wyzwania w rozprawie doktorskiej uważam za zadanie ambitne, ważne i aktualne z punktu widzenia współczesnej inżynierii oprogramowania, w szczególności inżynierii chmurowych systemów rozproszonych.

3. Cele i tezy rozprawy

We wstępie (rozdział 1. rozprawy) Autor przedstawia motywację leżącą u podstaw pracy oraz definiuje tezę rozprawy. Zgodnie z tą tezą, **zaproponowana w pracy metoda obejmująca wykrywanie korelacji pomiędzy symptomami błędów oraz ciągłe zbieranie i analizowanie wiedzy o funkcjonowaniu i strukturze systemu umożliwia automatyczną identyfikację źródeł skomplikowanych defektów występujących w dużych aplikacjach chmurowych** (*tłum. moje*). Autor precyzyjnie określa też założenia, przy których będzie dowodził słuszności tezy:

1) celem rozprawy nie są same metody detekcji błędów, gdyż zakłada się korzystanie z istniejących narzędzi obserwacyjnych,
2) proponowana metoda dotyczy aplikacji chmurowych zbudowanych według paradygmatu hierarchii mikroserwisów,
3) elementy aplikacji są traktowane jak „szare skrzynki" – nie ma możliwości ingerencji w kod aplikacji, można natomiast korzystać z wewnętrznych interfejsów komponentów aplikacji.

Są to założenia uzasadnione w kontekście głównego celu rozprawy, jakim jest opracowanie metody wykrywania źródeł defektów, powodujących zakłócenia w funkcjonowaniu aplikacji chmurowej dużej skali.

4. Układ i zawartość pracy

Rozprawa napisana jest w bardzo dobrym języku angielskim. W całym tekście natrafiłem na jedynie kilka błędów literowych. Praca zawiera 9 rozdziałów, w tym wstęp i podsumowanie, dodatek prezentujący aplikację testową oraz bibliografię obejmującą 83 pozycje, w tym jedną pozycję, której Autor rozprawy jest współautorem. W rozdziale 2. przedstawiony jest w sposób wyczerpujący aktualny stan wiedzy w tematyce rozprawy. Zasadniczą teoretyczną część rozprawy stanowią rozdziały od 3. do 6. W rozdziale 3. Autor przedstawia ogólną koncepcję swojej metody analizy źródeł defektów, uzasadniając zastosowanie podejścia opartego na badaniu korelacji pomiędzy wartościami wybranych parametrów działania i struktury sytemu, a następnie wywodzeniu z tego badania wniosków co do przebiegu trajektorii defektów. Rozdziały od 4. do 6. poświęcone są szczegółowej realizacji tego podejścia. Rozdział 4. prezentuje statystyczną analizę współwystępowania symptomów defektów, statystyczną analizę odstępów czasowych pomiędzy tymi symptomami, statystyczną analizę generowanych przez defekty serii czasowych oraz wpływ struktury systemu na wnioskowanie o przyczynach defektów. W rozdziale 5. opisano sposób tworzenia struktur pomocniczych – odpowiednich grafów i macierzy, a w rozdziale 6. przedstawiono szczegóły i przykładowe efekty działania algorytmu wnioskującego. Kolejne dwa rozdziały poświęcono szczegółowemu opisowi prototypowej implementacji proponowanej metody oraz testom weryfikacyjnym i walidacyjnym. Rozprawę kończy podsumowanie i wskazanie obszarów dalszych prac.

Ogólnie układ rozprawy jest logiczny i konsekwentny, służący realizacji celów rozprawy i uzasadnieniu prawdziwości postawionej tezy. Mam jednak kilka uwag:

1) W pracy brakuje wykazu stosowanych skrótów i oznaczeń. Takie wykazy umieszcza się zazwyczaj na początku rozprawy, tak by czytelnik mógł łatwo odnaleźć znaczenie danego skrótu lub symbolu w sytuacji, gdy jest ich sporo w tekście (a tak jest w recenzowanej rozprawie).
2) Szkoda, że w punkcie 2.2.1 Autor nie pokusił się na graficzną reprezentację prezentowanej taksonomii (ontologii) używanych terminów i zależności pomiędzy nimi, np. w formie diagramu UML lub innej metody graficznej reprezentacji wiedzy.
3) Praca jest bardzo obszerna – zawiera 259 stron, z czego 230 stanowi tekst zasadniczy. W szczególności rozdziały 7. i 8. zawierają mnóstwo szczegółów technicznych, które można by pominąć lub zamieścić w dodatku. Utrudnia to nieco uchwycenie kwestii zasadniczych, takich jak uzasadnienie przyjętej metody weryfikacji i walidacji, a także możliwości zastosowania proponowanej metody w innych środowiskach produkcyjnych.

Powyższe uwagi nie wpływają jednak na moją wysoką ocenę poziomu merytorycznego, technicznego i edytorskiego przedstawionego do recenzji tekstu.

5. Oryginalny wkład Autora

Jak już wspomniałem, oryginalny dorobek i wkład w dziedzinę Autor zaprezentował w rozdziałach od 3. do 6. Za najważniejsze elementy tego oryginalnego wkładu uważam:

1. Zdefiniowanie ogólnego modelu wykrywania źródeł defektów systemu (aplikacji) dużej skali działającego w chmurze obliczeniowej, polegającej na konsolidacji różnych aspektów funkcjonowania i budowy systemu prowadzącej do wnioskowania o przyczynach i skutkach defektów.
2. Opracowanie teoretyczne proponowanej metody (modelu RCA) poprzez zdefiniowanie parametrów korelacyjnych dla symptomów defektów i sposobu ich konsolidacji.
3. Zdefiniowanie struktur danych w postaci grafów zależności pomiędzy obiektami występującymi w systemie, macierzy korelacji i innych struktur wynikających z wprowadzonego modelu RCA, z podaniem algorytmów zbierania potrzebnych danych i tworzenia tych struktur.
4. Opracowanie algorytmu, który na podstawie utworzonych struktur estymuje źródłowe przyczyny defektów poprzez tworzenie grafów potencjalnych przepływów defektów, z wyszczególnieniem parametrów wskazujących na prawdopodobieństwo występowania pomiędzy nimi zależności przyczynowo-skutkowych.
5. Zweryfikowanie proponowanej metody na zbiorze sztucznie wygenerowanych danych, a następnie dokonanie jej testowej walidacji w środowisku nietrywialnej aplikacji chmurowej opartej na mikroserwisach, zbudowanej z zastosowaniem znanych i powszechnie używanych narzędzi i komponentów.

Warto podkreślić wysoki poziom merytoryczny oryginalnej części rozprawy (tj. rozdziałów 3.-6.), co dowodzi nie tylko bardzo dobrej znajomości przez Autora problemów ściśle związanych z tematyką rozprawy, ale także bardzo dobrego obeznania z niezbędnym aparatem matematycznym.

Podsumowując tę część recenzji, stwierdzam, że Autor rozprawy w pełni zrealizował postawione w rozprawie cele i uczynił to w sposób kompleksowy i dojrzały z naukowego punktu widzenia. Tezę rozprawy można uznać za w pełni zweryfikowaną poprzez przeprowadzone testy ze sztucznie wygenerowanymi danymi, a proponowaną metodę za zwalidowaną w eksperymentalnym środowisku zbliżonym do środowiska produkcyjnego.

6. Uwagi krytyczne i komentarze

Poniżej zamieszczam uwagi i komentarze, jakie nasunęły mi się w trakcie czytania rozprawy. Mają one charakter dyskusyjny i mam nadzieję na podjęcie ich w trakcie publicznej obrony.

1. W pracy brakuje odniesienia proponowanej przez Autora metody do cyklu życia aplikacji chmurowej. Zastosowanie metody ilustrowane jest aplikacją zbudowaną przez Autora. Jednak jak będzie przebiegał proces zastosowana tej metody do innej działającej w chmurze aplikacji? Czy będzie to możliwe i przy jakich założeniach? Czy taka aplikacja powinna być zbudowana w analogiczny sposób, jak aplikacja Online Boutique, z użyciem tych samych komponentów obserwacyjnych, diagnostycznych i raportujących? Jeśli tak, to praca jawnie powinna zawierać taką informację. Jeśli nie (a wydaje się to bardziej pożądane z uwagi na potencjalną uniwersalność metody), to jak powinna być zbudowana taka aplikacja, by można było zastosować metodę Autora?
2. W ogólnym wzorze (4.35) występują 4 ważone składniki zagregowanego współczynnika korelacji pomiędzy symptomami *a* i *b*. W dalszej części rozprawy Autor, po krótkiej dyskusji, przypisuje tym składnikom wagi: 0,5 dla składowej topologicznej oraz po 0,25 dla składowej współwystępowania symptomów i składowej odstępów czasowych. Dla składowej dotyczącej serii czasowych w dalszej części rozprawy przyjęto wagę 0 (czyli jest ona ignorowana). Zachodzi pytanie, czy można przyjąć wagę 0 także dla innych składowych, np. pozostawiając tylko składnik topologiczny lub tylko dwa pozostałe składniki? Czy Autor analizował takie możliwości?
3. Obliczenie składników współwystępowania symptomów i odstępów czasowych, dokonywane w trybie off-line, wymaga wcześniejszego zebrania danych z pewnego okresu funkcjonowania aplikacji. Jak duże powinny być te dane i jak długi może być ten okres?
4. Składnik topologiczny określony wzorem (4.32) nazwany jest współczynnikiem korelacji. Ta nazwa nie jest uzasadniona, gdyż współczynnik ten nie ma charakteru statystycznego, tylko wynika z bieżącej struktury systemu. Można by też oczekiwać pewnego uzasadnienia tego wzoru – dlaczego przyjęto zależność geometryczną? Dlaczego odległość 0 i 1 jest traktowana identycznie?

W tekście dostrzegłem też kilka błędów edytorskich:

5. W kilku pozycjach bibliograficznych (63, 67, 68, 79) nie podano źródła – cytuję: „In: ( )”.
6. W wielu miejscach (np. str. 134, 135, 138, 139, 140, 141) w przypisach dolnych brak jest konkretnych odesłań, prawdopodobnie do stron internetowych.
7. Na stronie 173 podpunkt „Symptom Time-series Analysis” w pierwszym zdaniu odnosi się do „(…) symptom time-lag analysis”.

Podkreślam, że powyższe uwagi krytyczne i komentarze nie stoją w sprzeczności z moją ogólnie bardzo pozytywną opinią o recenzowanej rozprawie.

7. Podsumowanie

Stwierdzam, że mgr inż. Bartosz Żurkowski w swojej rozprawie doktorskiej wykazał się bardzo dobrą znajomością zagadnień funkcjonowania i monitorowania aplikacji chmurowych, w szczególności problematyki analizowania źródeł defektów występujących w aplikacjach dużej skali. Przedstawił w sposób wyczerpujący aktualny stan wiedzy w tym zakresie, a dla osiągnięcia celów

swojej pracy poprawnie wykorzystał istniejące narzędzia informatyczne oraz opracował własne. Swoją bardzo dobrą znajomość obszaru objętego tematyką rozprawy potwierdził adekwatnym doborem literatury przedmiotu. Wynikiem jego pracy jest kompleksowa metoda o charakterze inżynierskim, oparta na solidnych podstawach teoretycznych.

Oceniam, że Autor recenzowanej rozprawy jest badaczem zdolnym do prowadzenia samodzielnych badań w obszarach związanych z szeroko pojętą inżynierią oprogramowania, w szczególności inżynierią systemów rozproszonych. W swojej rozprawie wykazał się wiedzą i umiejętnościami wymaganymi do uzyskania stopnia doktora nauk technicznych w dyscyplinie **informatyka techniczna i telekomunikacja**, zgodnie z obowiązującymi przepisami, i wnoszę o przekazanie recenzowanej rozprawy do dalszych etapów przewodu doktorskiego.