dr hab. inż. Paweł Czarnul
Katedra Architektury Systemów Komputerowych
Wydział Elektroniki, Telekomunikacji i Informatyki
Politechnika Gdańska

Gdańsk, 17 maja 2019 roku

RECENZJA ROZPRAWY DOKTORSKIEJ
DLA
WYDZIAŁU INFORMATYKI, ELEKTRONIKI I TELEKOMUNIKACJI
AKADEMII GÓRNICZO-HUTNICZEJ
IM. STANISŁAWA STASZICA W KRAKOWIE

Autor rozprawy: mgr inż. Kamil Figiela
Wydział Informatyki, Elektroniki i Telekomunikacji
Akademii Górniczo-Hutniczej im. Stanisława Staszica w Krakowie

Tytuł rozprawy: Optimization of scientific workflow execution in the cloud

Wstęp

Praca została napisana w języku angielskim i dotyczy zagadnienia optymalizacji wykonania aplikacji w środowiskach chmurowych. Rozprawa bazuje na wynikach pięciu publikacji naukowych, których współautorem jest mgr inż. Kamil Figiela. Publikacje te wymienione zostały w pracy na stronie 99 i obejmują cztery pozycje z czasopism z listy JCR oraz jeden artykuł konferencyjny. W odpowiednich rozdziałach wymieniono publikacje, których fragmenty składają się na opisy odpowiednich algorytmów oraz wyników. Praca składa się z 8 rozdziałów, listy publikacji autora, bibliografii, listy rysunków, listy tabel oraz dodatków w postaci treści ww. pięciu publikacji. Kolejne rozdziały zawierają opisy odnoszące się do motywacji, tezy pracy; modeli chmurowych (usług, wdrożenia, rozliczania); klas aplikacji (zagadnienia problemowe pracy skupione są wokół scenariuszy usługowych – ang. *workflows*); aktualnego stanu wiedzy z zakresu wykonania scenariuszy naukowych w środowiskach chmurowych, funkcji jako usług; algorytmów szeregowania scenariuszy z wykorzystaniem mieszanego modelu liniowego i całkowitoliczbowego, techniki dziel-i-rządź do optymalizacji wykonania aplikacji w środowisku równoległym i

wygenerowania odpowiedniego scenariusza wykonania dla aplikacji; zagadnienia wykorzystania funkcji jako usług w chmurze, w tym aspektów wdrożeniowo-technicznych, analizy kosztów wersji serverless względem infrastruktury IaaS; wreszcie dyskusji i podsumowania. Główna część pracy wraz z literaturą, listą rysunków oraz tabel i wyjaśnieniem skrótów liczy 116 stron. Zgodnie z wymaganiami odpowiedniej ustawy, praca zawiera streszczenie w języku polskim.

Jaki jest problem naukowy (teza) rozprawy i czy został on trafnie i jasno sformułowany?

Praca zorientowana została na optymalizację wykonania aplikacji na rozproszonych zasobach chmurowych, w modelu scenariusza reprezentowanym przez graf, którego wierzchołki oznaczają zadania do wykonania. Teza pracy umieszczona została w rozdziale 1.2 i stanowi, że wykorzystanie wiedzy o infrastrukturze, strukturze grafu scenariusza aplikacji, modelu wydajnościowym, umożliwia wydajne zaplanowanie i wykonanie naukowych scenariuszy aplikacyjnych w chmurze.

Z tezą skojarzone zostały 3 cele szczegółowe, o następującej treści:

1. Wykazanie, że metody bazujące na mieszanym programowaniu liniowym i całkowitoliczbowym mogą być efektywnie wykorzystane do szeregowania scenariuszy aplikacyjnych w środowisku chmurowym.
2. Zaprezentowanie, że analiza struktury obliczeniowej aplikacji oraz agregacja zadań pozwala na efektywne wykonanie wybranych aplikacji wysokiej wydajności w środowisku chmurowym.
3. Przeprowadzenie badań charakterystyk wydajnościowych platform typu serverless. Określenie czy rozwiązanie serverless może być efektywnym kosztowo rozwiązaniem dla aplikacji naukowych.

Poszczególne cele wykazane zostały odpowiednio w rozdziałach 5, 6 oraz 7.

Problem został jasno sformułowany.

Czy autor rozwiązał postawiony problem i czy użył do tego właściwych metod dowodząc, że posiadł umiejętności związane z metodyką i metodologią prowadzenia badań naukowych?

Autor rozwiązał postawiony problem poprzez wykazanie celów wymienionych powyżej, a konkretnie:

1. Sformułowanie problemu optymalizacji wykonania scenariusza z wykorzystaniem mieszanego modelu całkowitoliczbowego i liniowego i minimalizacji funkcji kosztu uwzględniającej czasy wykonania zadań, transferu danych, cen etc. Model uwzględnia grupy zadań oraz etapy wykonania scenariusza. Autor zaproponował dwa modele, w zależności od wartości ograniczeń czasowych względem okresu rozliczeniowego. Model dla zadań mniejszych uwzględnia dodatkowo opóźnienia komunikacyjne. Od strony praktycznej, wykazano wykorzystanie modeli dla szeregowania i uruchomienia konkretnych aplikacji w środowiskach wykorzystujących Amazon S3 oraz Rackspace Cloud Files z wykazaniem zależności kosztu wykonania od narzuconego ograniczenia czasowego dla aplikacji Epigenomics (różne liczby zadań i rozmiaru danych), Cybershake, LIGO, Montage oraz SIPHT. Autor wykazał różnice w ww. funkcjach dla scenariuszy o różnych wartościach czasów wykonania zadań. W przypadku dużych wartości koszt wykonania scenariusza znacząco rośnie dla niewielkich ograniczeń czasowych (rozdział 5.4.5.1). Przedstawiono zależności zarówno dla scenariuszy o ograniczeniach czasowych kilkunastu do kilkudziesięciu godzin jak i wykorzystując drugi wspomniany model, scenariusz Montage, chmurę Amazon EC2 wraz z przestrzenią danych S3 bądź lokalną i ograniczenia rzędu do 60 minut, z podobną charakterystyką (rys. 5.13) do poprzednich.

2. Transformację solwera wielofrontalnego dla obliczeń metodą elementów skończonych do uruchomienia w środowisku chmurowym. Ideą rozwiązania jest wykorzystanie schematu obliczeń i odpowiedniego partycjonowania oraz aglomeracji drzewa przetwarzania tak, aby z jednej strony zapewnić równoległość przetwarzania, z drugiej możliwie minimalizować stosunek czasu komunikacji do czasu obliczeń. Mechanizmy te pozwalają na wygenerowanie struktury scenariusza, który jest następnie uruchamiany. Autor zademonstrował wyniki przetwarzania dla problemu generującego 31-poziomowe drzewo pokazując wymagania pamięciowe implementacji, a przede wszystkim demonstrując dobrą skalowalność rozwiązania tj. przyspieszenie obliczeń 26.3-36.9 dla 64 maszyn w środowisku chmurowym AWS ze współdzieloną przestrzenią NFS i silnikiem Hyperflow oraz RabbitMQ, dla rozpatrywanego problemu z różnymi progami dla aglomeracji.

3. Określenie czasu działania funkcji jako usług w chmurze względem rozmiaru dla generatora liczb losowych uruchomionego w czterech środowiskach: AWS Lambda, Google Cloud Functions, IBM Cloud Functions, Azure Functions, określenie narzutów związanych z wywołaniami tych funkcji dla ww. środowisk. Autor zbadał częstotliwości ponownego wykorzystania instancji w ww. środowiskach chmurowych oraz określił monetarny koszt

wywołania funkcji w zależności od rozmiaru. Określił względne koszty wywołania aplikacji składającej się z określonej liczby zadań w środowisku serverless oraz wykorzystującym EC2. Wyniki badań pozwalają na oszacowanie progu liczby zadań oraz ograniczenia czasowe preferujące jedno bądź drugie środowisko.

Uważam, że autor posiadł odpowiednie umiejętności w zakresie prowadzenia badań naukowych, potrafiąc wyrazić problem szeregowania scenariuszy w środowisku chmurowym jako problem mieszanego programowania całkowitoliczbowego i liniowego wskazując dwie wersje w zależności od wartości ograniczeń czasowych i różniące się rozdzielczością rozwiązania, następnie proponując rozwiązanie, implementację oraz dyskusję wyników testów z rzeczywistych środowisk chmurowych (określenie wkładu autora dzięki deklaracji wkładu w publikacje na s. 33). Podobnie wykazał te umiejętności identyfikując wąskie gardła komunikacyjne przy portowaniu aplikacji solwera wielofrontalnego do uruchomienia w środowisku chmurowym, proponując rozwiązanie skalowalne, ale jednocześnie odnosząc je do lepszych czasowo wyników środowiska HPC (określenie wkładu autora dzięki deklaracji wkładu w publikacje na s. 56). Wreszcie, umiejętności praktyczne widać w identyfikacji i przetestowaniu wspomnianych i kluczowych aspektów wydajnościowych architektur serverless i porównaniu do infrastruktury IaaS (określenie wkładu autora dzięki deklaracji wkładu w publikacje na s. 72).

Wyniki zamieszczone w pracy powiązane są ściśle z wysokiej klasy publikacjami, których współautorem jest autor rozprawy.

Czy tematyka rozprawy jest aktualna lub dostatecznie ważna?

Tematyka rozprawy tj. uruchamianie aplikacji naukowych w środowiskach chmurowych jest niezwykle aktualna. Historycznie rzecz biorąc, algorytmy optymalizacji wykonania scenariuszy były uprzednio intensywnie eksploatowanym tematem badawczym w kontekście przetwarzania zorientowanego na usługi (SOA – Service Oriented Architecture) i systemów gridowych. Systemy chmurowe wniosły nowe możliwości, w szczególności w kwestii optymalizacji wydajnościowo-kosztowej. Jednocześnie szeregowanie i uruchamianie tego typu złożonych aplikacji w systemach chmurowych wiąże się z aspektami charakterystycznymi dla takich środowisk jak modele płatności, zarządzanie maszynami wirtualnymi, opóźnienia komunikacyjne, mechanizmy zapewniania bezpieczeństwa, zaufania względem dostawców, API. Wystarczy nadmienić, że ściśle określone hasła jak "workflow scheduling in cloud" lub „cloud workflow scheduling" w wyszukiwarce Google zwracają kilkadziesiąt tysięcy rekordów potwierdzając ważność tematyki. Autor podejmuje

tutaj głównie temat zależności wydajnościowo-kosztowej pozostawiając kwestie bezpieczeństwa, zaufania bądź innych metryk jakościowych. Ważne w tym kontekście jest uwzględnienie charakterystyk działania rozwiązań różnych dostawców chmury takich jak Amazon, Google, IBM oraz Microsoft, co jest cenne dla potencjalnego użytkownika. Należy nadmienić, że scenariusze, które rozważa autor mogą reprezentować nie tylko aplikacje naukowe (jak zaznaczono w tytule pracy), ale potencjalnie także biznesowe oraz mieszane poprzez integrację usług różnych typów, oferowanych w różnych środowiskach, przez różnych dostawców.

Czy rozprawa świadczy o dostatecznej wiedzy autora, wiedzy na zaawansowanym poziomie, o charakterze podstawowym dla dziedziny nauk technicznych oraz o charakterze szczegółowym, odpowiadającym obszarowi badań naukowych?

Czy rozprawa obejmuje najnowsze osiągnięcia nauki i świadczy o znajomości współczesnej literatury z dyscypliny naukowej, której dotyczy?

Autor wykazał znajomość zagadnień związanych z tematyką rozprawy w następującym zakresie:

1. Podstawowym w zakresie tematyki rozprawy tj. założeń, mechanizmów i technologii przetwarzania w chmurze, w tym modeli usługowych (IaaS, PaaS, SaaS), przetwarzania typu serverless, typów chmur, wirtualizacji oraz rozwiązań związanych z przechowywaniem danych i rozliczeń.
2. Szczegółowym w zakresie tematyki rozprawy tj. zagadnieniom scenariuszy naukowych, konkretnie modelu scenariusza, środowiska wykonawczego dla scenariuszy naukowych, szeregowania scenariuszy z uwzględnieniem podejścia statycznego i dynamicznego, metryk optymalizacji, kosztów obliczeń i komunikacji, wykorzystania wielu dostawców oraz środowisk typu serverless.
3. Szczegółowym w zakresie znajomości przez autora:
    1. istotnych aspektów wydajnościowych determinujących celowość uruchomienia aplikacji w środowiskach chmurowych bądź HPC - porównanie chmury do HPC na s. 67-68,
    2. ważności określenia przydatności rozwiązania serverless lub IaaS do wykonania danych aplikacji – porównanie na s. 89-90.

Treść pracy została opatrzona cytowaniami do 112 pozycji literatury, w tym kilku z roku 2018, a także dokumentacji online rozwiązań chmurowych. Świadczy to o znajomości przez autora literatury w zakresie poruszanym w pracy. Pozycje literatury zawierają m.in. artykuły opublikowane na uznanych konferencjach międzynarodowych a także w czasopismach takich jak Future Generation Computer Systems, Journal of Parallel and Distributed Systems, Cluster Computing, Concurrency and Computation: Practice and Experience, Parallel Computing etc.

Na czym polega oryginalny dorobek autora i jakie jest jego znaczenie poznawcze? Jakie są najważniejsze osiągnięcia rozprawy?

Oryginalny dorobek autora i osiągnięcia przedstawione w pracy postrzegam w:

1. Zamodelowaniu problemu szeregowania scenariuszy aplikacyjnych dla środowisk chmurowych w postaci problemu mieszanego programowania całkowitoliczbowego i liniowego. Tutaj kluczowe i nietrywialne są wybór zmiennych oraz sformułowanie ograniczeń takich jakie autor przedstawił w pracy na s. 43-46. Warto zaznaczyć, że rozwiązanie to ma charakter uniwersalny i może być zastosowane do wielu aplikacji, które mogą być reprezentowane zgodnie z modelem.
2. Projekcie i implementacji algorytmu dla solwera wielofrontalnego oraz demonstracji skalowalności w środowisku chmurowym bazującym na Amazon Web Services. Ciekawą zaletą tego typu podejścia może być wykorzystanie potencjalnie bardzo dużej sumarycznej pamięci środowiska. Argumentacja autora o potencjalnie niższym sumarycznym czasie wykonania w środowisku chmurowym ze względu na konieczność uwzględnienia opóźnień w systemach kolejkowych klastrów w środowisku HPC stanowi ciekawą, praktyczną obserwację, wartościową z punktu widzenia przynajmniej niektórych użytkowników rozpatrujących takie możliwości uruchomienia.
3. Praktycznym porównaniu charakterystyk wydajnościowych (czas działania, narzuty komunikacyjne) oraz wdrożeniowych (wykorzystanie instancji w chmurze, koszty) dla rozwiązań funkcji jako usług różnych dostawców chmur, a konkretnie firm Amazon, Google, IBM, Microsoft. Wynikiem są tutaj praktyczne rezultaty, możliwe do wykorzystania przez innych użytkowników uruchamiających aplikacje obliczeniowe w środowiskach tych dostawców. Konkretnym przykładem wyników doświadczalnych jest analiza np. konkretnych procesorów wykorzystanych przez wykorzystane platformy, przedstawiona na

stronie 88. Pozytywnie należy ocenić analizę przypadków będących motywacją tych badań, przedstawioną w rozdziale 7.1.1.

4. Praktycznym porównaniu kosztu rozwiązania serverless względem IaaS dla środowiska firmy Amazon, z oszacowaniem parametrów aplikacji pozwalających na preferowanie jednego bądź drugiego rozwiązania.

Jakie są wady i słabe strony rozprawy?

Praca nie zawiera istotnych słabych stron, biorąc pod uwagę tezę oraz cele szczegółowe i ich wykazanie. Tym niemniej, następujące zagadnienia mogłyby być bardziej dopracowane w pracy integrującej wkład zawarty w publikacjach:

1. Wskazane byłoby podać więcej szczegółów na temat konkretnych liczb zmiennych oraz nierówności w przypadkach rzeczywistych problemów rozpatrywanych w rozdziale 5.
2. Oprócz dwóch modeli problemu bazujących na mieszanym programowaniu całkowitoliczbowym oraz liniowym (rozdziały 5.4.3.1 oraz 5.4.3.3) dobrze byłoby opisać bardziej szczegółowo klasy rzeczywistych problemów, które mogą być w ten sposób zamodelowane. Np. jakie są przykłady rzeczywistych aplikacji z ograniczeniami dla poszczególnych poziomów scenariusza, jak pokazano na rys. 5.4?
3. Poszczególne opisy aspektów optymalizacji związanej z szeregowaniem scenariuszy są nieco pobieżne – rozdziały 4.1.2-4.1.9. Autor cytuje prace, ale nie omawia szczegółowo ich wkładu. Warto by tutaj podkreślić powiązanie z problemem szeregowania zadań, czym różni się szeregowanie scenariuszy w chmurach względem wielu prac dla systemów typu grid?
4. Algorytm i wyniki testów w rozdziale 5 pokazują generalnie oczekiwane zależności tj. odwrotną zależność kosztu od limitu czasu. Pojawia się pytanie odnośnie jakości algorytmu rozwiązania problemu. Z jednej strony autor odniósł się do tej kwestii w ramach rys. 5.12 oraz odpowiedniego komentarza. Z drugiej, warto byłoby odnieść się do innych algorytmów w kontekście tak zdefiniowanego problemu, także ze względu na fakt, że były używane do podobnych problemów dla systemów SOA oraz typu grid. Przykładem podejść były np. algorytmy genetyczne, algorytmy z dystrybucją ograniczeń czasowych dla optymalizacji z ograniczeniem na czas wykonania lub też algorytmy GAIN/LOSS, algorytmy genetyczne dla optymalizacji z ograniczeniem na koszt etc.

Uwagi szczegółowe i redakcyjne

Praca zawiera nieliczne błędy jezykowe, poniżej podano właściwą formę:

1. s. 34 „may be converted to linear ones at the cost of ...",
2. s. 36 „Scheduling algorithms that are presented in the subsequent sections require to provide estimates of task execution time ..."
3. s. 37 „In subsequent sections we discuss two case studies that demonstrate both approaches depending on what data was available."
4. s. 45 „Variables are similar to the ones in the coarse-grained model, but the problem has fewer dimensions, ..."
5. s. 47 „In the infrastructure model we assumed that we had 4 public cloud providers (...) and a private cloud with no cost."
6. s. 50 rys. 5.6 b. - please explain - „legend the same as ???" - raczej nie „above"
7. s. 53 „In this paper a multi-stage adaptive scheduling algorithm that adapts to inaccurate task execution time estimates was presented."
8. s. 59 „The pseudocode of this heuristic is shown in Algorithm 6.1."

Pozostałe uwagi:

1. Temat i teza pracy dotyczą aplikacji w formie scenariuszy. Cele szczegółowe (wykazane odpowiednio w rozdziałach 5, 6 oraz 7) w różnym stopniu dotyczą optymalizacji wykonania tego typu aplikacji: model rozpatrywany w rozdziale 5 – w pełni, zaś model w rozdziale 7 w znacznie mniejszym stopniu. Np. w istotnym rozdziale 7.4.5 do oceny aplikacji rozpatrywana jest aplikacja typu bag-of-tasks, która jest bardzo mocno uproszczonym przykładem scenariusza – jak sam autor wspomina w rozdziale 3.1.2. Stąd też wydaje się, że rozdział 7 wnosi niewiele w kontekście aspektów optymalizacyjnych w tezę rozprawy, sporo zaś w zakresie praktycznym.

2. Na ile podejście i metoda zaprezentowana w rozdziale 6 jest specyficzna dla konkretnego typu obliczeń w rozpatrywanej aplikacji, na ile, w opinii autora, może być wykorzystana do innych aplikacji, typowo rozpatrywanych w kontekście HPC? Jeśli to ostatnie, czy autor może podać listę takich aplikacji?
3. Opis planów taryfowych rozliczania za wykorzystanie chmury w rozdziale 2.5 mógłby obejmować większą liczbę dostawców.
4. Autor rozpatruje czasy działania algorytmu optymalizacji (rys. 5.11). Czy autor mógłby odnieść się do sytuacji, w której wykonywany scenariusz ulega awarii (np. wykonywane zadanie ulega awarii)? Jak wówczas można zastosować wybrane podejście, jak czas optymalizacji może wpływać na czas wykonania scenariusza (np. bazując na wynikach z ww. wykresów)? Czy autor badał prawdopodobieństwo awarii w środowisku rzeczywistym?
5. Czy można by uprościć model, rezygnując z niektórych zmiennych, w szczególności całkowitoliczbowych?
6. S. 34 „In fact, the tools we use can do this for us." - warto byłoby je tutaj przywołać, zostało to zrobione nieco dalej.
7. Porównanie rzeczywistego czasu wykonania aplikacji w rozdziale 6: w chmurze Amazon oraz w systemie HPC z systemem kolejkowym jest interesujące z praktycznego punktu widzenia. Mogłoby być jednak rozszerzone o analizę finansowego kosztu obu wersji, ew. także przypadku własnego klastra bez systemy kolejkowego, ale z koniecznością ponoszenia całkowitych kosztów utrzymania infrastruktury.
8. Jakie były stosunki czasów obliczeń do komunikacji dla testów opisanych w rozdziale 6? Wskazane byłoby precyzyjne określenie czynników wpływających na uzyskane przyspieszenie obliczeń.

Podsumowanie i ocena rozprawy

W rozprawie doktorskiej mgr inż. Kamil Figiela wykazał tezę rozprawy, jednocześnie prezentując szereg wartościowych wyników demonstrujących praktyczne aspekty wykorzystania różnych sposobów wdrożenia i uruchomienia wymagających obliczeniowo aplikacji w chmurze. Wyniki omówione w pracy bazują na pięciu wartościowych publikacjach (w tym cztery opublikowane w

czasopismach z listy JCR). Biorąc to pod uwagę, a także praktyczne znaczenie zaproponowanego modelu dla różnych aplikacji, które można wyrazić w formie modelu scenariusza, stwierdzam, że rozprawa spełnia wymagania postawione rozprawom doktorskim przez odpowiednią Ustawę z dnia 14 marca 2003 r. o stopniach naukowych i tytule naukowym oraz stopniach i tytule w zakresie sztuki, w dziedzinie nauk technicznych, w dyscyplinie informatyka i wnioskuję o dopuszczenie rozprawy do publicznej obrony. Jednocześnie uwzględniając wagę wyników, jakość publikacji, na których rozprawa bazuje, szerokie zastosowania wyników i wieloaspektowość rozprawy dla przetwarzania w środowisku chmurowym, wnioskuję o wyróżnienie rozprawy.

Do której kategorii recenzent zalicza rozprawę?

Rozprawę doktorską Pana magistra inżyniera Kamila Figieli zaliczam do kategorii:

wyraźnie wykraczająca poza poziom przeciętny (spełniająca wymagania z nadmiarem)

dr hab. inż. Paweł Czarnul