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

Gdańsk, 4 marca 2019 roku

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

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

Tytuł rozprawy: Agent model based on stream processing in simulations and computational applications

Wstęp

Praca została napisana w języku angielskim i zawiera wprowadzenie (z omówieniem celu pracy, zdefiniowaniem tezy i wyjaśnieniem kluczowych występujących w niej terminów oraz listą 11 prac powiązanych z rozprawą), rozdział zawierający definicję agenta, opis systemów wieloagentowych oraz agentowych platform wspomagających symulacje oraz obliczenia, rozdział opisujący modele wykonania w ww. systemach, modele komunikacji pomiędzy agentami, migrację agentów, kwestie dotyczące rozproszenego wykonania, rozdział opisujący platformę AgE, w szczególności AgE 3 (w tym model strumieniowy z podstawowymi pojęciami oraz krokami wykonania wraz z przepływem danych i synchronizacją). W dalszej kolejności autor przedstawił elementy implementacji platformy agentowej wspierającej zdefiniowany model a także implementację komponentów niezbędnych do wykonywania rozproszonych obliczeń. Kolejne dwa rozdziały stanowią omówienie przypadków użycia i demonstrują możliwość definicji rzeczywistych problemów w zaproponowanym modelu i późniejszej implementacji. Rozdział 5 prezentuje sposób wyrażenia hybrydowego modelu symulacji z wykorzystaniem nanorobotów w krwi,

w szczególności połączenia metod symulacji agentowej ABS oraz symulacji przepływu cząstek (PBS). Z kolei rozdział 6 demonstruje wyrażenie implementacji EMAS z wykorzystaniem zaproponowanego strumieniowego modelu agentowego. Oba rozdziały zawierają dyskusję z porównaniem do reprezentacji w modelu FLAME GPU. Pracę kończy podsumowanie, skrócony opis modelu FLAME GPU oraz spis literatury. Praca zawiera spis rysunków, tabel, listingów oraz algorytmów. Główna część pracy wraz z literaturą liczy 138 stron.

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

Praca zorientowana została na stworzenie agentowego modelu wykorzystującego przetwarzanie strumieniowe, jak również demonstrację przydatności i implementacji modelu do reprezentacji oraz możliwości wykonania przykładowych rozwiązań symulacyjnych i obliczeniowych.

Teza pracy umieszczona została w rozdziale 1.2 i stanowi, że przetwarzanie strumieniowe jako podstawa do konstrukcji modelu agentowego zapewnia mechanizmy do implementacji efektywnej i wszechstronnej platformy dla symulacji i aplikacji obliczeniowych.

Autor wyjaśnia pokrótce w jaki sposób rozumie kluczowe terminy w kontekście tezy i pracy w tym: przetwarzanie strumieniowe (wymieniając strumienie agentów oraz populacji), model agentowy, efektywną i wszechstronną platformę umożliwiającą szerokie spektrum zastosowań i skalowalność oraz symulacje i aplikacje obliczeniowe.

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 (tj. wykazanie możliwości zastosowania agentowego modelu wykorzystującego przetwarzanie strumieniowe) poprzez:

1. Omówienie założeń i wymagań względem nowego (proponowanego) modelu (w rozdziale 4.2) w kontekście wcześniej używanego modelu agentowego na platformie AgE 2 (rozdział 4.1).
2. Propozycję modelu agentowego z przetwarzaniem strumieniowym wraz z omówieniem poszczególnych komponentów, takich jak agent, populacja, miejsce „workplace", funkcje krokowe „step functions".

3. Omówienie implementacji zaproponowanego modelu wraz z opisem platformy AgE jako środowiska obliczeniowego, pozwalającego na wdrożenie aplikacji wykorzystujących model z przetwarzaniem strumieniowym.
4. Demonstrację możliwości zamodelowania złożonej, hybrydowej symulacji uwzględniającej nanoroboty i przepływ krwi w organizmie, z mechanizmami synchronizacji symulacji agentów oraz przepływu cząstek oraz zrównoleglaniem.
5. Demonstrację możliwości zamodelowania i odzworowania modelu EMAS (Evolutionary Multi-Agent System) na agentowy model z przetwarzaniem strumieniowym, implementację, przypadek testowy weryfikujący przydatność modelu w postaci aplikacji rozwiązującej problem LABS (Low Autocorrelation Binary Sequence) i demonstrację wykonania w środowisku wykorzystującym kartę NVIDIA GeForce GTX (tutaj implementacja pozostawia pole do optymalizacji w przyszłości, jak przyznaje także autor).

Uważam, że autor posiadł odpowiednie umiejętności w zakresie prowadzenia badań naukowych, wskazując ograniczenia istniejącego modelu (rozdział 4.1 – 'main issues'), następnie proponując nowy model i dowodząc, na wybranych przykładach, możliwość wykorzystania powyższego w aplikacjach obliczeniowych i symulacjach.

Wyniki zamieszczone w pracy powiązane są, chociaż dosyć luźno, także z publikacjami, których współautorem jest autor rozprawy, i których lista została zamieszczona w rozdziale 1.3. W kontekście wyników pracy za najbardziej powiązane uważam pozycje [6] (hybrydowy model symulacji z rozdziału 5) oraz [10] (odwołanie do strumieniowej implementacji AgE 3).

Czy tematyka rozprawy jest aktualna lub dostatecznie ważna?

Tematyka rozprawy tj. elastyczne modele wspierające symulacje i wymagające obliczeniowo aplikacje oraz mechanizmy ich wydajnego uruchomienia w środowiskach rozproszonych ze wsparciem dla akceleratorów jak GPU jest bardzo aktualna. Z jednej strony, do programowania aplikacji równoległych wykorzystuje się obecnie niskopoziomowe interfejsy programistyczne jak OpenMP, CUDA, OpenCL, OpenACC, MPI. Wynikowe aplikacje uruchamiane są następnie na maszynach z wielordzeniowymi procesorami i GPU, które autor wykorzystywał do testów. Z drugiej strony, frameworki implementujące przetwarzanie bazujące na populacjach i przepływach danych stanowią łatwiejszą w użyciu alternatywę, w szczególności jeśli ukrywają szczegóły implementacyjne przed mniej doświadczonymi programistami. Kluczem staje się dostarczenie tego typu rozwiązań z możliwie niską stratą wydajności w porównaniu do rozwiązań pierwszego rodzaju.

Warto zauważyć, że obecne systemy komputerowe, zarówno typu serwerowego (w szczególności dotyczy to najbardziej wydajnych klastrów z listy top500.org) jak i domowego, wykorzystują powszechnie zarówno wielordzeniowe procesory jak i karty GPU. Konstrukcja modeli i późniejsza demonstracja wydajności ich implementacji, zwłaszcza w kierunku optymalizacji przetwarzania hybrydowego, są bardzo istotne dla wydajnego przetwarzania w różnych dziedzinach. W tym kontekście znaczenie pracy jest duże, gdyż pozwala na reprezentację i wykonanie obliczeń dla różnych aplikacji zgodnych z zaproponowanym modelem.

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. Zagadnień rozwiązań agentowych oraz przetwarzania na platformach agentowych zorientowanych na symulacje i obliczenia, wraz z mechanizmami technologicznymi (rozdział 2). W kontekście rozpatrywanego problemu można potraktować je jako zagadnienia o charakterze podstawowym.
2. Zagadnień modeli wykonania z użyciem agentów, mechanizmów komunikacji pomiędzy agentami w środowisku, migracji agentów, mechanizmów wymiany danych w systemach rozproszonych (RPC, wymiana wiadomości, rozproszone magazyny danych), kwestie rozproszenia i scentralizowania wykonania, przezroczystości rozproszenia oraz działania agentów w środowiskach heterogenicznych (rozdział 3). Z punktu widzenia tematyki rozprawy, dowodzi to znajomości zagadnień na poziomie szczegółowym.

Opisy zagadnień umieszczone w pracy opatrzone zostały cytowaniami do pozycji literatury aż do roku 2018 i świadczą o znajomości literatury w zakresie dyscypliny informatyka, powiązanej tematycznie z rozprawą. Bibliografia liczy 110 pozycji.

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. Opracowaniu propozycji agentowego modelu z przetwarzaniem strumieniowym (rozdział 3), w szczególności koncepcji przetwarzania wykorzystującej populacje agentów, ich transformacje w środowisku. Na uwagę zasługuje referencyjna implementacja modelu z określeniem sposobu równoległego wykonania z wykorzystaniem wielu wątków oraz GPU.
2. Zamodelowaniu aplikacji hybrydowej do symulacji z wykorzystaniem nanorobotów we krwi, w szczególności połączeniu metod symulacji agentowej ABS oraz symulacji przepływu cząstek PBS. Autor wykorzystał zaproponowany model agentowy dowodząc jego użyteczności.
3. Zamodelowaniu i wykonaniu testów implementacji EMAS (z wykorzystaniem zaproponowanego modelu agentowego) wraz z rozwiązaniem problemu LABS, z demonstracją wykonania z użyciem systemu z wielordzeniowym procesorem oraz akceleratorem GPU demonstrując zysk z równoległości na podstawowym poziomie (pozostawiając jednak pole dla głębszego sprofilowania wykonania i dalszych optymalizacji).

Powyższe stanowią według mnie rozwiązanie problemu naukowego rozprawy i dają podstawy do wnioskowania o przydatności modelu dla innych podobnych problemów symulacyjnych, wymagających dużej mocy obliczeniowej.

Jakie są wady i słabe strony rozprawy?

1. Autor mógłby lepiej i szerzej opisać zagadnienie modelu przetwarzania strumieniowego w aktualnych systemach obliczeniowych. W szczególności, istnieje wiele frameworków i rozwiązań do wydajnego przetwarzania strumieniowego jak: Storm, Spark, Flink, PicCo, etc., także wykorzystujących GPU, w tym np. GStream, G-Storm. Ponadto, przetwarzanie strumieniowe używane jest jako mechanizm optymalizacji wykonania równoległego (w technologii NVIDIA CUDA) do nakładania obliczeń na GPU i/lub CPU z komunikacją pomiędzy stroną hosta (CPU) oraz urządzenia (GPU). Autor zamieścił wzmiankę odnośnie strumieni w GPGPU na stronach 25 oraz 61, ale opis nie wyczerpuje tematu. Warto byłoby

przedstawić rozpatrywany przez autora model strumieniowego przetwarzania także na ww. tle.

2. Przy opisie implementacji (strona 61) brakuje szczegółów implementacji na GPU. W jaki sposób można zrównoleglić zaproponowane przez autora przetwarzanie populacji na poziomie GPU z wykorzystaniem wielu strumieni CUDA (do nakładania obliczeń i komunikacji)? Czy nowoczesne mechanizmy jak Unified Memory w CUDA (który także można połączyć ze strumieniami CUDA) mogłyby ułatwić implementację przetwarzania w systemie?

3. W pracy brakuje, moim zdaniem, oceny wydajnościowej podstawowych operacji implementacji zaproponowanej platformy agentowej (opisanej w rozdziale 4.5) w środowisku rozproszonym, w szczególności w porównaniu do niskopoziomowych mechanizmów komunikacyjnych jak np. implementacji MPI. Pozwoliłoby to na ocenę wydajności względem elastyczności różnych możliwych implementacji.

4. Rozdział 5 demonstruje możliwość zamodelowania złożonego przetwarzania w zaproponowanym modelu. Przydałyby się tutaj także wyniki wydajnościowe dla wersji równoległej, w szczególności przyspieszenia obliczeń. Pożądane byłoby określenie wpływu warstwy translacji pomiędzy stronami symulacji na przyspieszenie obliczeń.

5. Odnośnie wyników zamieszczonych w rozdziale 6.5 – w jaki sposób odwzorowano obliczenia na GPU? Jakie były ustawienia konfiguracji gridu (liczba wątków na blok, liczba bloków wątków)? Czy wykorzystano nakładanie obliczeń i komunikacji? Czy autor może dokładniej wytłumaczyć względne wyniki w liczbie ocen dla różnej liczby wysp, poza komentarzem na stronie 116?

6. Opis FLAME GPU (rozdział 2.5 oraz dodatek A) wydaje się nieco zbyt powierzchowny jako baza do porównania modeli. Właściwe byłoby uprzednie umieszczenie prostego przykładu modelu rzeczywistej aplikacji i elementów uruchomienia na platformie wraz z omówieniem.

Uwagi szczegółowe i redakcyjne

Praca zawiera nieliczne błędy tzw. „literówki", poniżej podano właściwą formę:

1. strona 33 „Tanenbaum and Van Steen distinguish two forms of messaging in distributed systems: ...",
2. strona 34 „This model works very well",
3. strona 62 „All methods that change internal values",
4. strona 63 „Although this approach to queries does not introduce ... ",
5. strona 81 – usunąć „[type",
6. strona 87 „At the moment of writing this dissertation, we did not decide which method is better for our needs",
7. strona 104, podpis Figure 6.1 „High energy agents reproduce and create new offspring agents, ...",
8. strona 112 „Other parameters were constants and did not change during tests, ...", „Included charts present various views of the system behaviour over time: ...".

Pozostałe uwagi:

1. Opis mechanizmu „blackboard" (strona 29) mógłby uwzględniać kwestie spójności widoku dla różnych agentów.
2. Opis zagadnienia „distributed data stores" (strona 34) mógłby być szerszy.
3. Odnośnie notatki 18 w stopce na stronie 64 – warto dodać, że pamięć „shared memory" na GPU jest współdzielona tylko pomiędzy wątkami jednego bloku, zaś pamięć globalna „global memory" może być odczytywana/zapisywana przez wątki grida, ale jest relatywnie wolna przy dostępie typu 'non-coalesced' i nie można poza tym polegać na określonej kolejności wykonania bloków wątków. Pełna synchronizacja możliwa jest po wykonaniu kernela. Te elementy znacząco utrudniają wydajną implementację tego typu schematu na GPU.
4. Strona 66 - „Most of the services use some distributed facilities to communicate with other services of the same type on other nodes.". Zdanie jest zbyt ogólne, autor powinien zamieścić więcej szczegółów.
5. Czy przy uruchomieniu autor próbował korzystać z mechanizmu NVIDIA MPS?

6. Podział danych w symulacji cząstek opisany w rozdziale 5.7 i przedstawiony na rysunku 5.9 może nie być optymalny dla niektórych zestawów wymiarów trójwymiarowej przestrzeni, ze względu na czas komunikacji przy wymianie informacji o cząstkach blisko ścian.

Podsumowanie i ocena rozprawy

W rozprawie doktorskiej mgr inż. Łukasz Faber wykazał stosowalność agentowego modelu z przetwarzaniem strumieniowym do reprezentacji dwóch wybranych problemów – hybrydowej symulacji uwzględniającej nanoroboty i przepływ krwi w organizmie oraz EMAS z implementacją rozwiązania problemu LABS, jednocześnie, w moim przekonaniu, dowodząc tezy rozprawy. Tematyka pracy jest istotna w rozpatrywanej dziedzinie i o dużym znaczeniu dla programowania różnych algorytmów, bazujących na populacjach z przepływem danych w nowoczesnych systemach obliczeniowych z akceleratorami.

Podsumowując, w moim przekonaniu, rozprawa spełnia wymagania postawione rozprawom doktorskim przez odpowiednią ustawę, w dziedzinie nauk technicznych, w dyscyplinie informatyka i wnioskuję o dopuszczenie rozprawy do publicznej obrony.

Do której kategorii recenzent zalicza rozprawę?

Rozprawę doktorską Pana magistra inżyniera Łukasza Fabera zaliczam do kategorii:

zadowalająco spełniająca wymagania.

dr hab. inż. Paweł Czarnul