

**Akademia Górniczo-Hutnicza  
im. Stanisława Staszica w Krakowie**

---

Katedra Informatyki



**PRACA MAGISTERSKA**

**PIOTR KOZŁOWSKI**

**PRZENOŚNY KLASTER OBLICZENIOWY OPARTY NA  
URZĄDZENIACH SOC**

PROMOTOR:  
dr Aleksander Byrski

Kraków 2015

### **OŚWIADCZENIE AUTORA PRACY**

OŚWIADCZAM, ŚWIADOMY ODPOWIEDZIALNOŚCI KARNEJ ZA POŚWIADCZENIE NIEPRAWDY, ŻE NINIEJSZĄ PRACĘ DYPLOMOWĄ WYKONAŁEM OSOBIŚCIE I SAMODZIELNIE, I NIE KORZYSTAŁEM ZE ŹRÓDEŁ INNYCH NIŻ WYMNIENIONE W PRACY.

.....

PODPIS

**AGH**  
**University of Science and Technology in Krakow**

---

Department of Computer Science



**MASTER OF SCIENCE THESIS**

**PIOTR KOZŁOWSKI**

**MOBILE COMPUTING CLUSTER BASED ON SoC DEVICES**

SUPERVISOR:  
Aleksander Byrski Ph.D

Krakow 2015

Podziękowania...

## Spis treści

<b>1. Wprowadzenie</b>	6
1.1. Wstęp	6
1.2. Aktualny stan wiedzy	6
1.3. Cel pracy	7
1.4. Zawartość pracy	7
<b>2. Charakterystyka rozpatrywanego problemu</b>	9
2.1. Urządzenia SoC	9
2.2. Cluster Computing	9
<b>3. Wykorzystane algorytmy</b>	10
3.1. Algorytmy ewolucyjne	10
3.2. Master/slave	10
<b>4. Opis rozwiązania</b>	11
4.1. Platforma	11
4.2. Użyte technologie	11
<b>5. Podsumowanie</b>	12
5.1. Wyniki testów	12
5.2. Możliwości rozwoju	12
1. Instrukcja uruchomienia platformy	14

# 1. Wprowadzenie

## 1.1. Wstęp

Tematyka niniejszej pracy jest związana z przetwarzaniem równoległym w środowisku rozproszonym oraz algorytmami ewolucyjnymi. Obliczenia równoległe, systemy wysokiej dostępności oraz wielkiej skali są tematem bardzo aktualnym i wciąż powszechnie rozwijanym. Powstaje wiele urządzeń oraz systemów wieloprocessorowych zbudowanych w celu połączenia zasobów w jedną całość oraz wykorzystujących w maksymalnym stopniu ich moc obliczeniową. Zasoby te mogą być zlokalizowane lokalnie lub być rozproszone w różnych rejonach świata, czyli np. w sieci Internet. Przykładem mogą być takie projekty jak Hadoop lub Condor. Połączone zasoby mogą zostać wykorzystane do obliczeń o dużej skali (np. symulacji) i przyspieszenia krytycznych obliczeń (systemy obronne, loty kosmiczne, prognozy). Jak również są przydatne w przypadku przetwarzania dużej liczby rozproszonych danych i rozwiązywania trudnych problemów optymalizacji (przeszukiwania dużych przestrzeni rozwiązań np. w celu poszukiwania ekstremum). Do głównych obszarów zastosowań można zaliczyć też obliczenia naukowe prowadzone przez różne uniwersytety i inne ośrodki badawcze, które w dużym stopniu przyczyniają się do rośnięcia zapotrzebowania na takie rozwiązania oraz powstawanie nowych. We wcześniejszych latach ta tematyka napotykała różne bariery w rozwoju, takie jak zbyt mały rynek zbytu, spowodowany wysokimi kosztami utrzymania oraz realizacji architektury. Było również mniej rozwiązań tego typu oraz mniejsze zapotrzebowanie na nie. Klastry są bardziej elastycznym rozwiązaniem a co za tym idzie bardziej opłacalnym. Stały się zatem nową perspektywą, powszechnie już wykorzystywaną w dzisiejszych systemach. Możliwość użycia tego typu rozwiązań przez zwykłych użytkowników i mniejsze firmy, komercjalizacja, wykorzystanie w przemyśle i elastyczne formy handlu, np. sprzedaż zużytej mocy obliczeniowej w bardzo dużym stopniu przyczyniły się do popularyzacji tejże tematyki. Trwa ciągły rozwój technologii z tym związanych, mając na uwadze takie rzeczy jak wydajność i zapotrzebowanie na energię oraz łatwa dostępność. Wiele problemów nie zostało jeszcze rozwiązanych lub w wystarczającym stopniu zbadanych. Argumenty stojące za rozwojem tych technologii to m.in. rosnące zapotrzebowanie na moc obliczeniową, stosunek ceny do mocy obliczeniowej, bariery technologiczne rozwoju procesorów sekwencyjnych, dzielenie zasobów, skalowalność, komunikacja.

## 1.2. Aktualny stan wiedzy

Do tej pory zostało opracowanych kilka klastrów obliczeniowych z wykorzystaniem urządzeń SoC takich jak Raspberry Pi. Przykładem może być praca opisana w artykule Iridis-pi: a low-cost, compact demonstration cluster [3]. Powstało również kilka podobnych rozwiązań o których informacje można znaleźć na Internecie. Długo dawno ukazało się wydanie czasopisma Informatica opisujące klastr obliczeniowy wykorzystujący telefony komórkowe, oraz rolę komunikacji i kilka różnych podejść bazujących na różnych architekturach systemu [1]. Dwa kolejne artykuły opisują klastry w których skład wchodzi urządzenia przenośne podłączone do sieci Internet lub połączone w szybkiej sieci lokalnej ukazały się na łamach ... [2, 5]. Poruszono tam także problem dynamicznego load-balancingu oraz zrównoleglania obliczeń. W artykule Adaptive clustering for mobile wireless networks [4] opisano samoorganizującą się, odporną na uszkodzenia składowych węzłów mobilną sieć oraz algorytmy z

nią związane, które są dosyć istotne w kontekście niniejszej pracy. W pracy Cluster Computing: A Mobile Code Approach [6] opisano klastery wysokiej wydajności wraz z ich architekturą oraz implementacją. Natomiast praca pt. DroidCluster: Towards Smartphone Cluster Computing - The Streets are Paved with Potential Computer Clusters [7] opisuje nowoczesne smartfony wchodzące w skład klastra obliczeniowego oraz temat Grid-Computing. Jak widać temat i problemy z nim związane znany jest od dosyć dawna. Natomiast wciąż jest aktualny i nadal brakuje skutecznych rozwiązań wszystkich problemów. Same urządzenia SoC zyskały swoją popularność dopiero w ostatnich latach, głównie za sprawą taniego mini komputera jakim jest Raspberry Pi. Większość powstałych rozwiązań bazuje na takich technologiach jak MPI oraz Hadoop, natomiast w niniejszej pracy autor skupi się na technologiach pracujących w wirtualnej maszynie Javy, co jeszcze bardziej zwiększy mobilność powstałej platformy.

### 1.3. Cel pracy

Do celów niniejszej pracy należy nakreślenie możliwości zastosowania przenośnych klastrów obliczeniowych, przedstawienie problemów związanych z ich wykorzystaniem, zidentyfikowanie obszarów potencjalnych usprawnień oraz wykorzystanie przygotowanego klastra w celu rozwiązania trudnego problemu obliczeniowego. Jako część programistyczna niniejszej pracy, zostanie wykonana mobilna platforma łącząca możliwości urządzeń SoC, nadzorująca ich pracę oraz zarządzająca dostępną energią. Pod uwagę zostanie również wzięta odporność na uszkodzenia oraz łatwa zastępowalność urządzeń. Zostaną wykorzystane takie paradygmaty jak programowanie funkcjonalne oraz reaktywne, z wykorzystaniem technologii Scala oraz Akka wraz z podejściem równoległym. Jako część persystencji danych zostanie wykorzystana baza oparta na grafach w technologii Neo4J. Część kliencka zostanie zrealizowana w technologii REST/HTTP z użyciem frameworka Spray. Zaprojektowana platforma będzie w stanie monitorować pracę klastra, "łączyć" komponenty w jedną całość czyli zarządzać komunikacją, być odporna na sytuację uszkodzenia jednego z elementów klastra oraz częściową utratę danych, maksymalnie wykorzystywać moc obliczeniową elementów klastra oraz zarządzać load balancingiem. Niniejsza praca będzie próbą wykorzystania urządzeń SoC do wykonywania wydajnych obliczeń prowadzonych w wirtualnej maszynie Javy oraz próbą odpowiedzi na pytanie czy jest możliwe zaprogramowanie komunikacji między elementami klastra w sieci Ethernet nie powodującej zbyt wielkiego narzutu przesyłu danych w stosunku do szybkości obliczeń. Praca będzie też próbą wykorzystania wirtualnej maszyny Javy oraz technologii wysokiego poziomu do wykonywania obliczeń dużej skali i sprawdzenie wydajności takich rozwiązań. Stworzenie samoorganizującej się oraz inteligentnej platformy będzie bardzo dużym wyzwaniem wkraczającym trochę w tematykę AI oraz Systemów Agentowych. Jeśli rozwiązanie wykorzystujące technologie przedstawione w niniejszej pracy będzie wystarczająco wydajne i odpowiednio skalowalne, ułatwi to w znacznym stopniu mobilność tego rozwiązania oraz rozwiąże potrzebę tworzenia nowych bibliotek z myślą o konkretnej platformie. Co za tym idzie większą część czasu i nakładów finansowych będzie można poświęcać na implementację rozwiązania samego problemu aniżeli na uruchomienie tej implementacji lub tworzenie rozwiązań pasujących do konkretnego sprzętu (architektury) lub systemu.

### 1.4. Zawartość pracy

W pierwszym rozdziale został zawarty wstęp, aktualny stan wiedzy, cel pracy oraz hipoteza badawcza.

W drugim rozdziale został opisany kontekst rozpatrywanego problemu oraz została przedstawiona charakterystyka rozwiązania proponowanego w niniejszej pracy.

W trzecim rozdziale zostały opisane urządzenia SoC oraz Cluster Computing.

W czwartym rozdziale zostaną przedstawione sposoby zrównoleglenia algorytmów ewolucyjnych.

W piątym rozdziale zostały opisane rozwiązania implementacyjne zastosowane przy realizacji niniejszej pracy.

W szóstym rozdziale zostało zawarte podsumowanie w którym zostały omówione wyniki pracy oraz naświetlone możliwości rozwoju.

Ostatnim rozdziałem pracy jest bibliografia zawierająca przegląd literatury, artykułów oraz innych pozycji wykorzystanych w niniejszej pracy.

Został także stworzony listing wszystkich wykresów, schematów oraz rysunków, jak również tabel oraz wzorów. Do pracy została dołączona również instrukcja uruchomienia stworzonego rozwiązania.



## **2. Charakterystyka rozpatrywanego problemu**

### **2.1. Urządzenia SoC**

### **2.2. Cluster Computing**

### **3. Wykorzystane algorytmy**

#### **3.1. Algorytmy ewolucyjne**

#### **3.2. Master/slave**

## **4. Opis rozwiązania**

### **4.1. Platforma**

### **4.2. Użyte technologie**

## **5. Podsumowanie**

### **5.1. Wyniki testów**

### **5.2. Możliwości rozwoju**

## Bibliografia

- [1] An international journal of computing and informatics. *Informatica*, 23, 1999.
- [2] K. B. An introduction to cluster computing using mobile nodes. *Emerging Trends in Engineering and Technology (ICETET)*, 2009.
- [3] S. J. Cox. Iridis-pi: a low-cost, compact demonstration cluster. *Cluster Computing*, June 2014.
- [4] C. R. Lin. Adaptive clustering for mobile wireless networks. *IEEE Journal on Selected Areas in Communications*, 2006.
- [5] M. A. M. Mohamed. Cluster computing with mobile nodes: A case study. *Distributed and Object Systems Lab, Department of CS & E, Indian Institute of Technology Madras, Chennai, India*.
- [6] R.B.Patel and M. Singh. Cluster computing: A mobile code approach. *Department of Computer Engineering, M.M. Engineering College, Mullana-133203, Haryana, India, 2006*.
- [7] L. Wolf. Droidcluster: Towards smartphone cluster computing - the streets are paved with potential computer clusters. *Technische Universitat Braunschweig, Institute of Operating Systems and Computer Networks*.

## **Dodatek A**

### **.1. Instrukcja uruchomienia platformy**