UNIVERSITI TEKNOLOGI MARA

DYNAMIC PARTITIONING AND DATA ALLOCATION METHOD ON HETEROGENEOUS ARCHITECTURE

MUHAMMAD HELMI BIN ROSLI

Thesis submitted in fulfillment
of the requirement for the degree of
Master of Science

Faculty of Computer & Mathematical Sciences

September 2015

AUTHOR'S DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Teknologi MARA. It is original and is the result of my work, unless otherwise indicated or acknowledged as referenced work. This thesis has not been submitted to any other academic institution or non-academic institution for any other degree or qualification.

I, hereby, acknowledge that I have been supplied with the Academic Rules and Regulations for Post Graduate, Universiti Teknologi MARA, regulating the conduct of my study and research.

Name of Student	:	Muhammad Hemi Bin Rosli
Student ID No.	:	2011284026
Program	:	Master of Science
Faculty	:	Faculty of Computer & Mathematical Sciences
Thesis Title	:	Dynamic Partitioning and Data Allocation Method on Heterogeneous Architecture
Signature of Student	:	………………………………………………..
Date	:	September 2015

ABSTRACT

In recent years, processing large data set to produce result in a timely manner poses a lot of challenges to ICT researchers. Currently most organization has an elaborate local network system whose computers are underutilized. These network form cluster of computing resources that simulates supercomputer. Processing images are computationally complex due to its data and task intensive nature. This can be solved by parallelizing the process in cluster environment. Most cluster environment have a variety of computer hardware specification namely heterogeneous environment. Optimizing the resources in heterogeneous environment during parallel processing is not a simple task. These involves partitioning and allocating task to each cluster node. The aim of these research is to investigate various method of partitioning and allocating task in cluster environment and produce a dynamic partitioning and allocating method. Initial stage of the research consist of exploring the heuristic performance of cluster and multi-threading involving five experiments; homogeneous architecture with node partitioning; heterogeneous architecture with node partitioning; heterogeneous architecture with node partitioning including multi-threading; heterogeneous architecture with node and core partitioning; heterogeneous architecture with node and core partitioning including multi-threading. The performances use sequential processing speed as a benchmark. Each experiment highlight the advantages and disadvantages of the experimental architecture. The disadvantages from each experiment prompts the design of dynamic parallel partitioning and allocating framework. The case study use for this experiment is Sobel edge detection algorithm. The test data set focuses on processing images of three different sizes; (1K x 1K), (2K x 2K) and (3K x 3K). The performance evaluation is based on the processing speed in second, speedup, and efficiency. In conclusion, it is found that in idle situation heterogeneous architecture with node and core partitioning including multi-threading perform better from other experiment. However, in real working condition where some computer are serving users processes, the dynamic algorithm provides a potential alternative.

ACKNOWLEDGEMENTS

Firstly and foremost, with deep sense of gratitude, I wish to express my sincere thanks to my supervisor Dr. Noor Elaiza Binti Abdul Khalid, Faculty of Computer Science and Mathematics, UiTM Shah Alam, Selangor, Malaysia for her excellent guidance and support throughout the work. Her constant encouragement has helped me acquire and develop some of the skills and intricacies of good independent research. I thank her also for providing me an opportunity to grow as a research student in the unique research environment she creates.

Special thanks are given my co supervisor Noorhayati Mohamed Noor whose encouragement as well as technical and non-technical advices have always been an important support for my research. Special thanks to them for helping me through my difficult time of supervisor change in order to express my deep appreciation and gratitude for providing enormous guidance especially in technical part, critics, motivation and constant encouragement during my studies and throughout the project.

I would also like to express sincere thanks to Assoc. Prof. Dr. Hj. Mazani Hj. Manaf for given me opportunity to participate in their granted.

Finally, I would like to share this moment of happiness with my family members for their encouragement, understanding, patience and endless love during my studies. Without your support, nothing could be achievable.

CHAPTER ONE
INTRODUCTION

1.1 RESEARCH BACKGROUND

The era of big data is upon us. Big data is a collection of data sets so large and complex that it becomes difficult to process using tools such as database management or traditional data processing applications. There are no limits on the size of data sets that require processing within a reasonable amount of time. Scientists often face limitations due to the large data sets in various fields, including meteorology research, genomics, connectomics, complex physics simulations, and the biological and environmental (Baradwaj et al., 2012).

One of the data consider as big data is images. Currently, massive influx of images via digital image capturing devices has created challenges to analyze images in areas such as medical, satellites and content base image retrieval (Sagiroglu et al., 2013). Often, this images need to be process quickly. Images processing method such as image segmentation, image enhancement and image analysis can be complex and slow when processing large images. Traditional sequential algorithm to process big data presents less challenge but suffers from very long processing time. This renders sequential algorithm as ineffective solution for processing large images. Therefore in order to speed up processing, image processing algorithms based on parallel processing which combines multiple processing nodes to emulate high-end servers is an attractive alternative.

Parallel processing is basically an instructions that partitions a program across multiple processors with the objective of carrying out the program in less time. In traditional sequential programming, only one program running at a time (Blaise et. al., 2011). In data-intensive processing, sequential processing may take more than double the time it take to process in parallel platform.