AIN SHAMS UNIVERSITY
FACULTY OF COMPUTER & INFORMATION SCIENCES
DEPARTMENT OF SCIENTIFIC COMPUTING

Performance-Based Restructuring of Distributed Object-Oriented Computations for a Cluster of Multiprocessors

A Thesis Submitted to the Department of Scientific Computing, Faculty of Computer and Information Sciences, Ain Shams University, in Partial Fulfillment of the Requirements for the Doctorate Degree of Computer and Information Sciences

By

Safwat Helmy Mohammed Hamad
M.Sc. Degree in Scientific Computing (2004).
Teaching Assistant, Department of Scientific Computing,
Faculty of Computer and Information Sciences, Ain Shams University.

Supervised by

Prof. Dr. MOHAMMED ESSAM KHALIFA
Dean of Faculty of Computer and Information Sciences,
Ain Shams University

Prof. Dr. REDA AMMAR
Head of Computer Science and Engineering Department,
University of Connecticut - USA

Prof. Dr. ElSayed Abdel Aziz Soleit
Professor, Department of Scientific Computing,
Faculty of Computer and Information Sciences,
Ain Shams University

2008

ACKNOWLEDGMENTS

All praise and thanks to Allah, who gave me the guidance, the patience and the support to attain this achievement.

I want to express my deep gratitude to Prof. Dr. Mohammed Essam Khalifa for his continuous encouragement, understanding and support. He stood by me throughout all the difficulties I faced during the last few years. Thank you Prof. Dr. Esaam.

I am deeply indebted to, Prof. Reda Ammar, for his endless help, and support. His advice and insight have been invaluable throughout my entire time of research. Our weekly meetings gave me academic guidance and provided the milestones that helped to develop my research skills.

I would like to extend my thanks to my associate advisor Prof. Dr. ElSayed Soleit, for his support and helpful suggestions. In addition, I would like to thank Prof. Tahany Fergany, who gave me a lot of her time and effort. I cannot possibly thank her enough. I won't forget to give my deep thanks to Prof. Dr. Mohammed Said Abdel-Wahab for his great effort in accelerating the formal procedures of dissertation submission.

Finally, this dissertation would have never been possible without the support of my wife. I am really grateful to her for her patience; caring, understanding and encouragement made my mission much easier and supported me during the ups and downs of this work.

Abstract

Over the last few years, designers and engineers utilized the Object-Oriented (OO) approach in developing distributed software systems for solving complex problems in various scientific fields. The initial design of the Distributed OO (DOO) application does not necessarily have the best class distribution. In such class of problems, the solution is possible through two approaches: either to reconfigure the hardware to match the software components (hardware reconfiguration), or to reconfigure the software structure to match the available hardware (software restructuring).

Software restructuring is the process of re-organizing the logical structure of existing software systems in order to improve particular quality attributes of software products. Previous restructuring techniques have not been considered DOO software. In this research, we introduce a methodology for efficiently restructuring the DOO software classes on a specific distributed system. The presented process is achieved in a set of consecutive steps. In the first step, the Distributed Object Oriented Software (DOO) is analyzed to evaluate relationships and interactions among different system classes and then modeled as a class dependency graph. The second step is concerned with identifying clusters of a dense community of classes within the DOO system that have low coupling and are suitable for distribution. Next, the generated clusters are merged into grains. Those grains are then mapped onto the nodes of the multiprocessor system such that the amount of communication among classes is minimized.

Classical techniques and methodologies of performance analysis are unsuitable to capture performance behavior of Object-Oriented (OO) systems. In this thesis, we recommended to use the Distributed Object-Oriented Performance (DOOP) model as a powerful analytical technique for the assessment of relationships between system classes. Once the DOOP model is used to evaluate the inter-class communication costs all over the DOO application under study, these values can be used to generate the Class Dependency Graph (CDG) of the given OO application. This modeling step should be done very carefully and precisely since this CDG will be the basis of all the coming phases in our proposed restructuring approach.

The next restructuring step involves decomposing the OO system into subsystems that have low coupling and are more suitable for distribution. A clustering technique based on recursive spectral graph bi-partitioning was proposed to create a suggested grouping of subsystems that are convenient for guiding the allocation of the subsystems to the set of available machines in a distributed environment. The most interesting thing about this technique is that it does not stop when a certain predefined number of clusters are composed. Instead, it identifies the set of subsystems with the most communication density within each cluster and has less coupling among each other no matter how many clusters have been formed.

The resultant system modules may not be ready for the mapping step. In fact, we may be faced by one of two cases. The first case happens when the number of candidate clusters are less than or equal to the number of the available machines in the target distributed architecture. In this case the mapping process can be simply achieved. The problem occurs in the second case, when the number of the generated clusters exceeds the number of available nodes. Actually, this is a more realistic view since there will always be huge software systems and limited hardware resources. So, in this case; we need an extra step to solve this mismatch problem in order to be ready for the upcoming mapping step.

We proposed a solution by merging the generated system clusters into groups or larger grains such that the effect of class dependency and data communication is minimized. It was assumed that the target distributed system consists of a set of homogeneous processors that are fully connected via a communication network. We investigated a number of approaches that use the graph of clusters generated in the previous step and group it into a Merged Cluster Graph (MCG). Three different approaches were implemented and compared under different systems and architectures: The "*K-Partitioning*" algorithm, the "*Cluster Grouping"* Approach, and the "*Double-k Clustering"* Approach. Experimental results showed that the Double-K provides the best performance over the other algorithms since it gives the minimum interclass communication cost.

The last step in the proposed restructuring approach is the physical mapping. Physical mapping is the process of efficient placement of the MCG into real processor network topologies. The general physical mapping problem was proven to be NP-hard, thus allowing only for heuristic approaches. We have developed three algorithms for solving the mapping problem using a randomized approach. These algorithms have proved to be efficient, simple and easy to understand and implement. Furthermore, the performance of the proposed algorithms was tested against some existing deterministic techniques. The experimental results showed an outstanding performance of these algorithms in minimizing the overall mapping cost of the produced assignments.

TABLE OF CONTENTS

List of figures

List of Tables

1

Introduction

1.1 Problem Statement

1.2 Research Objectives

1.3 Thesis Organization

CHAPTER 1
Introduction

1.1 Problem Statement

Advanced scientific computing systems are often Distributed Object-Oriented (DOO) in nature. Pairing distributed systems with object oriented paradigms results in distributed objects. Choosing the most efficient distributed object design is a multi criteria decision process. Object-oriented paradigm [Mey88, Ost02] is based on several concepts such as encapsulation, inheritance, polymorphism, and dynamic binding. Although these features contribute to the reusability and extensibility of systems, they produce complex dependencies between classes. This makes investigating and evaluating the performance of the DOO applications a challenging task. Required techniques should capture the behavior of the system while preserving the OO properties. Furthermore, in many cases the initial OO Software design doesn't have the best class distribution and may need to be restructured.

Software restructuring is the process of re-organizing the logical structure of existing software systems in order to improve particular quality attributes of software products [Arn89]. Commercial software systems are generally far too large and complex to be effectively restructured on an ad hoc basis. No doubt, we need techniques and tools to help analysts extract design and structure information and hence support software restructuring process. In this research, we developed a multi-step approach for restructuring DOO applications to fully utilize system resources and hence improve the overall system performance.

1.2 Research Objectives

Restructuring is a mechanism that aims to improve the system performance and choose the appropriate structure that fit with the user and the system requirements. In this research, we attempt to find an approach for efficiently restructuring the DOO software classes in order to be mapped onto a certain target distributed architecture. We can summarize the objectives of this research work as follows:

1. Study of the state of art in performance modeling of distributed object oriented software system.

2. Utilizing this modeling approach as a base for driving the application time cost.

3. Developing restructuring algorithms while considering architectures with different topologies, system constraints and operating conditions

1.3 Thesis Organization

This thesis consists of four main parts. Each one of them illustrates one step or phase of the restructuring approach. It starts from the DOOP model and how to use it to evaluate the communication activity between different classes in the DOO system. Then it moves to the clustering step and go through the different techniques that can be used to group the generated clusters into larger grains and finally how to find the efficient way to map those grains to the physical architecture of the distributed environment. In general, the rest of this thesis is organized as follows:

Chapter 2, Literature Survey: this chapter provides an overview of the distributed object oriented systems and its applications. It also defines the principles of object oriented restructuring, as well as some of its related practical considerations.

Chapter 3, Clustering OO Systems: In this chapter, we present an approach for decomposing object-oriented systems into subsystems that have low coupling and are suitable for distribution. The developed technique is based on the recursive spectral bi-partitioning. The steps of this algorithm are illustrated and analyzed through a simulated case study

Chapter 4, Unrestricted Clusters Mapping: this chapter introduces a number of methods to solve the mismatch between the hardware and the DOO software components or clusters. We investigated three techniques for merging heavily related clusters into larger grains and mapping them to fully connected nodes such that the dependency communication among classes is minimized.

Chapter 5, Physical Mapping: this is the final part of the research. It defines the main idea of physical mapping as well as the notations and criteria used in finding a solution. In addition, we develop three algorithms to solve the physical mapping problem. Each one of them applies the idea of randomized algorithms on the physical mapping from a different point of view. Next, the performance of these algorithms is compared to a couple of other mapping techniques under different topologies.

Chapter 6, Conclusions and Future Work: this part summarizes the total findings of the conducted research and gives hints that can lead other researchers in their future work.