بسم الله الرحمن الرحيم

Ain Shams University Information Network
جامعة عين شمس
شبكة المعلومات الجامعية
@ ASUNET

شبكة المعلومات الجامعية

جامعة عين شمس

التوثيق الالكتروني والميكروفيلم

قسم

نقسم بالله العظيم أن المادة التي تم توثيقها وتسجيلها
علي هذه الأفلام قد اعدت دون آية تغيرات

يجب أن

تحفظ هذه الأفلام بعيداً عن الغبار

في درجة حرارة من 15 – 20 مئوية ورطوبة نسبية من 20- 40 %

To be kept away from dust in dry cool place of 15 – 25c and relative humidity 20-40 %

بالرسالة صفحات

لم ترد بالأصل

بعض الوثائق

الأصلية تالفة

Cairo University

Faculty of Economics and Political Science

Task Allocation in Distributed Systems using a Stochastic Programming Approach

By

Amany Yehia Abd-Elrahman

Supervised by

Prof. Dr. Nadia M. Girgis
Professor of Statistics
Dep. of Statistics

Dr. Kamal S. Selim
Assistant Prof. of Statistics
Dep. of Computational Socio- Science

Faculty of Economics and Political Science
Cairo University

A Thesis submitted to the department of Statistics, Faculty of Economics and Political Science, Cairo University in partial fulfillment of the requirements for M.Sc. degree.

1999

Approval sheet

Task Allocation in Distributed Systems using a Stochastic Programming Approach

By

Amany Yehia Abd-Elrahman

This thesis for the degree of master of Science in Statistics has been approved by:

------Nadia Makary------

------Kamal------

To my parents.
To my family.

Acknowledgment

First and foremost, I would like to express my sincere gratitude to Prof. Dr. ***Nadia Makary*** for her support and her valuable suggestions.

I am deeply indebted to Dr. ***Kamal Selim*** for suggesting the point of research and for his guidance, fruitful discussions, and continuous encouragement.

My great thanks to Dr. ***Maged George***, associate professor, department of Economics, American University in Cairo, for his valuable advice.

I would like also to thank lecturer ***Samir Wagih***, department of Statistics, Faculty of Economics and Political Science, Cairo University, for his helpful discussions.

Amany Yehia.

Abstract

The rapid progress in microprocessors and the decreasing cost of hardware have made the distributed processing systems economically attractive for many applications. However, many problems in the distributed systems are still in their development stage.

A distributed processing system consists of a set of processors (processing elements) which are connected via communication channels whose tasks (distributed task) are coordinated by a distributed operating system that controls information processing and flow among processors to achieve the desired requirements.

Task to be processed by the distributed system is decomposed into a set of cooperating tasks (sub-tasks) which are then allocated to the system's processors.

Assuming that the distributed task have been partitioned into sub-tasks, finding the optimal allocation for these sub-tasks on the system's processors using goal programming methodology is the aim of this thesis. Two allocation models are included. The first is the deterministic sub-task allocation model in which all model's parameters are assumed deterministic with known values. The second is more general which is the stochastic sub-task allocation model in which some of the model's parameters are assumed random with known distributions.

This thesis consists of four chapters organized as follows [1] :

Chapter 1 is an introductory chapter which reviews the basic concepts of two related fields to task allocation problem. These two fields are distributed systems, and distributed software design. The first field deals with distributed systems environment,

[1] Through out the thesis, the term task allocation will be used instead of sub-task allocation just for simplicity, but it should be kept in mind that we seek for the total task allocation by allocating its components which are the sub-tasks.

distributed systems architecture, distributed systems application, advantages of distributed systems, and finally, software engineering and life cycle. The second field concentrates on task partitioning and task allocation as two important steps in distributed software design, and states alternative modeling approaches for task allocation, followed by a literature review.

Chapter 2 briefly reviews the basic concepts of multiple objective stochastic programming needed in modeling task allocation problem. Stochastic goal programming is discussed as one of several approaches to multiple objective programming. Stochastic programming approaches are presented in this chapter specially the chance constrained approach. The iterative algorithm to goal programming is also presented since it will be used to solve task allocation problem.

Chapter 3 contains problem formulation and solution for the deterministic task allocation problem. It states variables definitions, constraints , objectives, goals and their priorities followed by the mathematical model which is classified as nonlinear zero-one goal programming model. A linear transformation to the nonlinear model is presented to make it possible to solve the model since no computer code for the nonlinear case was available to the author at the time of preparing the thesis. The last section of this chapter contains a numerical example solved iteratively by GAMS (version 2.25) package to clarify the deterministic task allocation problem presented in this chapter.

Chapter 4 presents the stochastic task allocation problem. Some parameters in the deterministic model presented in chapter 3 are assumed to be random variables rather than deterministic. The deterministic equivalent of the resulting stochastic model is constructed under the suggestions of the exponential distribution and chi square distribution as suitable probability distributions for the random parameters. Two numerical examples solved iteratively by GAMS (version 2.25) package are presented in this chapter to illustrate the stochastic task allocation model. The final section of this chapter contains conclusions and points for further research.

Appendices (A) and (B) contain the GAMS codes of the three numerical examples considered in chapters 3 and 4.

Contents

Chapter 1

Distributed Systems and Distributed Software Design

1.1: Introduction

This chapter is an introductory chapter which reviews the basic concepts of two related fields to task allocation problem. These two fields are distributed systems, and distributed software design. The first field is considered in sections (1.2) and (1.3) as follows: Section (1.2) contains distributed systems environment, distributed systems architecture, distributed systems application, and finally, advantages of distributed systems. Section (1.3) contains software engineering and life cycle. The second field is considered in section (1.4) which contains task partitioning and task allocation. Alternative modeling approaches for task allocation are contained in section (1.5), followed by a literature review in section (1.6).

1.2: Distributed systems environment

Distributed systems have become, for the past two decades, an area of research and development. The overall poor performance of centralized computer systems when dealing with an application which requires a large number of computations to be performed (specially when some of these computations can be processed concurrently) on one hand, and the unreality of them in the case of geographically distributed systems on the other hand, have made a need for a solution essential.

Recent, technological advances in computer hardware (specially in the areas of microprocessor and memory) have made such solution possible. Distributed systems are now in use in a wide range of computer applications. However, some of the principles of such systems are still under investigations and many problems in the distributed computing systems are still in their development stage.

The next subsection outlines the architecture of distributed systems. Subsection (1.2.2) discusses the applications of distributed systems. Subsection (1.2.3) explains the advantages of distributed systems. Section (1.3) defines the basic terms of software engineering and its life cycle.

1.2.1: Distributed systems architecture

From the physical point of view, a distributed system can be viewed as a collection of processing elements (machines) that are connected via communication channels (lines) and are controlled by a distributed operating system [Ziegler67], [Hariri86], [Deitel90]. The processing elements (processors) of the distributed system may be locally or geographically dispersed, so we have two terms "local area network" (LAN) and "Wide Area Network" (WAN). LAN's span a single building and WAN's span countries or continents.

In this subsection, we briefly define the architecture (physical components) of the distributed systems concentrating only on the hardware terms that are used through the rest of the thesis.

From the hardware prospective, the design of the distributed system means how to select the types of each processor and communication line, also to determine the topology of the system.

Processor type means processor capacity which is usually expressed as throughput. For example, in the context of databases environment the throughput of the computer is the maximum number of transaction processed per unit of time (the transaction is based on length of 60 characters and the time unit usually is second). In addition to the capacity, each computer has it's own local memory space (measured in words).

Communication line can be viewed as a virtual bit pipe along which bits travel, starting from one point (computer) called the transmitter or the origin, and arriving at another point (computer) called the receiver or the destination. There are several types of such pipes, each has its capacity which is usually expressed as bandwidth. In other words, the capacity of the communication line is called bandwidth (number of transactions processed per unit of time) [Chen80]. Another definition can be found for the capacity of the communication line which is the maximum rate at which bits can be transmitted over a unit of time (second).

The distributed system topology describes the way by which the processors of the system are going to be connected. Different types of topology can be considered, each