



ELSEVIER

Mathematics and Computers in Simulation 58 (2001) 71–92



MATHEMATICS
AND
COMPUTERS
IN SIMULATION

www.elsevier.com/locate/matcom

Large matrix–vector products on distributed bus networks with communication delays using the divisible load paradigm: performance analysis and simulation

S.K. Chan^a, V. Bharadwaj^{a,*}, D. Ghose^b

^a *Department of Electrical Engineering, National University of Singapore,
10 Kent Ridge Crescent, Singapore 119260, Singapore*

^b *Department of Mechanical and Aerospace Engineering, University of California at Los Angeles,
Los Angeles, CA 90095, USA*

Received 1 October 2000; accepted 2 March 2001

Abstract

We present a performance analysis and experimental simulation results on the problem of scheduling a divisible load on a bus network. In general, the computing requirement of a divisible load is CPU intensive and demands multiple processing nodes for efficient processing. We consider the problem of scheduling a very large matrix–vector product computation on a bus network consisting of a *homogeneous* set of processors. The experiment was conducted on a PC-based networking environment consisting of Pentium II machines arranged in a bus topology. We present a theoretical analysis and verify these findings on the experimental test-bed. We also developed a software support system with flexibility in terms of scalability of the network and the load size. We present a detailed discussion on the experimental results providing directions for possible future extensions of this work. © 2001 IMACS. Published by Elsevier Science B.V. All rights reserved.

Keywords: Divisible load; Matrix–vector product; Communication delay; Computation delay; Bus networks; Processing time minimisation

1. Introduction

The interest in network-based computing has grown considerably in recent times. In this environment, several computers are linked through a communication network to form large loosely-coupled distributed networks. One of the major attributes of such a distributed system is the capability that it offers to the user of any single node to exploit the considerable power of the complete network or a subset of it by partitioning and transferring its own processing load to the other processors in the network. This

* Corresponding author.

E-mail addresses: elebv@nus.edu.sg (V. Bharadwaj), ghose@artegal.seas.ucla.edu (D. Ghose).

capability is useful in handling *large* computational loads. This kind of application is different from that of scheduling and load balancing which is more relevant to a distributed computing system composed of mainframe computers, each handling a large number of jobs.

In this paper, we are concerned with the problem of distributing a single large load that originates at one of the nodes in the network. The load is massive in size compared to the computing capability of the node. So, the processor partitions the load into many fractions and distributes them among the processor nodes in the network. When the processing at each node is complete, the partial solutions are gathered and consolidated at the load originating processor to construct the complete solution. An important objective is to achieve a balance in the load distribution between processors so that computation is completed in the shortest possible time. This kind of processing load has the additional property that there is minimal inter-processor communication during the actual parallel execution of the program. However, there is a considerable communication delay at the beginning or at the end of a computational phase. The generic names of such loads are *divisible loads* [1–3] and the theory of scheduling such divisible loads is commonly referred to as divisible load theory (DLT) [1], in the literature. The theory adopts a linear model for the processor speed and communication link speed parameters. In this model, the communication time over a channel is assumed to be proportional to the amount of load that is transferred over that channel, and the computation time is assumed to be proportional to the amount of load assigned to that processor. Also, the processing load is assumed to be arbitrarily divisible in the sense that, each partitioned portion of the load can be independently processed on any processor on the network. The primary concern in DLT research is to determine the optimal fractions of the total load to be assigned to the processors in such a way that the total processing time of the entire load is a minimum.

The importance of the divisible load paradigm arises from the fact that although there is a well-developed literature on parallel processing algorithms, these algorithms have not been developed for distributed computing systems or network-based computing scenarios where the issues of communication delays and variable computation capability at processing nodes play a crucial role in the computational performance. Divisible load theory attempts to adapt these algorithms for optimised performance under these factors. However, this task is not straightforward since the communication delays are considerably higher in a distributed network environment than in a parallel processing environment. This is especially true for the processing of divisible loads since there is very little interprocessor communication during the actual computation process. So, an implementation in a parallel processor will not be affected much by communication overheads, unlike an implementation in a distributed computing environment where communication overheads have a very significant effect on optimal load distribution strategy. Below, we present a brief survey on some of the significant contributions in the DLT area.

In [4], a heterogeneous linear network of processors was considered and a computational algorithm was developed to obtain the optimal load fractions by assuming that all the processors participating in the computation stop computing at the same time instant. In fact, this has been shown to be a necessary and sufficient condition for obtaining optimal processing time in linear networks by using the concept of processor equivalence [5]. An analytic proof of this assumption in bus networks was presented in [6]. A more general proof of this assertion is also available in [1]. In [7], closed form solutions for optimal schedule for bus and tree networks were derived. The concepts of optimal sequencing and optimal network arrangement were introduced in [8]. Barlas [9] obtained conditions on the optimal sequencing of load distribution in a single-level tree network by including the results collection phase. To find the ultimate speed-up using DLT analysis, Li [10] conducted the asymptotic analysis for various network topologies. In [11], finite-size buffer constraint was imposed on each processor and an efficient

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات