Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®
Parallel Computing 30 (2004) 103–120
PARALLEL COMPUTING
www.elsevier.com/locate/parco

Sensitivity analysis of tree scheduling on two machines with communication delays

Frédéric Guinand [a,*], Aziz Moukrim [1], Eric Sanlaville [2]

[a] *Laboratoire d'Informatique du Havre, EA 3219, Université du Havre, 25 rue Philippe Lebon, BP 540, F-76058 Le Havre Cédex, France*

Received 15 May 2002; received in revised form 18 January 2003; accepted 19 June 2003

Abstract

This paper presents a sensitivity analysis for the problem of scheduling trees with communication delays on two identical processors, to minimize the makespan. Tasks are supposed to have unit execution time (UET), and the values associated to communication delays are supposed unknown before the execution.

The paper compares the optimal makespans with and without communication delays. The results are used to obtain sensitivity bounds for algorithms providing optimal schedules for graphs with unit execution and communication times (UECT). The notion of *processor-ordered* schedules, for two-processor systems, is introduced. It describes schedules in which all communications are oriented from one processor to the other. It is shown that these schedules are dominant for unit delays, for zero delays, but not for delays of less than or equal to one. We establish that algorithms building optimal processor-ordered schedules for UECT graphs admit an absolute sensitivity bound equal to the difference between the maximum and the minimum actual communication delays: $\omega - \omega^* \leqslant \bar{c} - \underline{c}$. This bound is tight.

Keywords: Uncertain communication delays; Trees; Unit execution and communication times; Sensitivity analysis; Two-processor scheduling

* Corresponding author.
E-mail addresses: frederic.guinand@univ-lehavre.fr (F. Guinand), aziz.moukrim@hds.utc.fr (A. Moukrim), eric.sanlaville@math.univ-bpclermont.fr (E. Sanlaville).
[1] HeuDiaSyC, UTC Compiègne, (France).
[2] LIMOS, Clermont-Ferrand, (France).

0167-8191/$ - see front matter
doi:10.1016/S0167-8191(03)00091-7

1. Introduction

Last years have seen a growing number of teams involved in the conception, the deployment and the use of infrastructures and applications distributed over a wide geographical area [26,25]. Peer-to-peer computing [24,27] as well as programs running on grids [28] are some examples of such applications. They are made of many tasks processed on computing devices located on various, possibly very distant, places. The underlying interconnection network linking together all computing elements is partially common with the internet. As a consequence, codes running on such environments may be subject to unexpected slowdowns occurring somewhere on the network. Then, it is not surprising that most peer-to-peer applications are merely based on the common master/slaves algorithmic scheme, while resources managers of grids give greater importance to high-throughput rather than to application performances. In order to avoid poor performances, the tasks of the application have to be distributed among the computing elements in order to obtain a good trade-off between load balanding and minimization of the communication overhead. Within previously described environments, both execution times of tasks and communication delays may vary. In addition, the application itself may be subject to structural variations depending on the input data, then, it is unlikely that a pure static scheduling strategy could lead to good performances. However, in our work a careful analysis of the schedule structure shows the importance of the distribution of tasks for minimizing the impact of communication variations.

In this paper, we focus our study on the degradation of schedules when communication durations estimated a priori differ from actual ones. The application model is a tree-structured task graph with unit-estimated communication delays and unit execution time (UECT task graphs). The overlapping of communications by computations is allowed and communications between tasks executed on the same processor are neglected (known as the locality assumption). This set of hypotheses corresponds to the *standard delay model* (SDM) [3,6].

Using this environment execution model, in [8], the authors proved that scheduling complete binary trees on an unlimited number of processors with arbitrary communication delays is an NP-Hard problem. The problem remains NP-Hard for binary trees when communications depend on the size of the tree. When the communication delays are equal to a constant value c, the problem can be solved polynomially for complete k-ary trees. For arbitrary trees, keeping the same assumptions, the NP-Hardness of the problem is proved in [9]. The same complexity result holds for UECT trees on an arbitrary number of processors organised as a full-connected graph [9,10,12]. When the number of processors, say m, is fixed, the problem may be solved using a dynamic programming approach (complexity $O(n^{2(m-1)})$) [20]. Finally, this problem becomes polynomial when the number of processors is 2 [9,11–13] or is not limited [4]. The main results are summed up in the following table using the notation described in [5]. A survey on scheduling problems with communication delays can be found in [14].