Applied Mathematical Modelling 38 (2014) 3975–3986

Contents lists available at ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

Minimizing the total weighted late work in scheduling of identical parallel processors with communication delays

Foroogh Abasian [a], Mohammad Ranjbar [a,*], Majid Salari [a], Morteza Davari [b], Seyed Morteza Khatami [a]

[a] Department of Industrial Engineering, Faculty of Engineering, Ferdowsi University of Mashhad, Mashhad, Iran
[b] Research Group for Operations Management, Faculty of Business and Economics, KU Leuven, Leuven, Belgium

ARTICLE INFO

Article history:
Received 26 August 2013
Accepted 23 January 2014
Available online 4 February 2014

Keywords:
Scheduling
Parallel processors
Communication delay
Branch-and-bound algorithm

ABSTRACT

This paper addresses a certain type of scheduling problem that arises when a parallel computation is to be executed on a set of identical parallel processors. It is assumed that if two precedence-related tasks are processed on two different processors, due to the information transferring, there will be a task-dependent communication delay between them. For each task, a processing time, a due date and a weight is given while the goal is to minimize the total weighted late work. An integer linear mathematical programming model and a branch-and-bound algorithm have been developed for the proposed problem. Comparing the results obtained by the proposed branch-and-bound algorithm with those obtained by CPLEX, indicates the effectiveness of the method.

1. Introduction

During the last two decades, the parallel processing has improved the performance of computing in many systems like real-time signal processing (Tokhi and Hossain [1]), image processing (Prajapati and Vij [2]) and robotic control (Jadud et al. [3]). Nevertheless, Ariosy et al. [4] show that the time restriction is usually an important factor in robotic control systems. In such systems the data, including a set of computational tasks, are collected using sensing devices and are processed in predefined time windows. Some tasks are precedence related using communication messages where each message carries certain amount of information. Usually, robots must react to particular programs on given due dates where each due date corresponds to a task. If the required information for a suitable reaction is not processed completely before or at a given time moment, the robot must react based on incomplete information. Obviously, the amount of gathered and processed data affects the accuracy of the control process, and all the information exposed after the given due date (called the information loss) is useless. The information lost is modeled as *late work* and should be minimized to increase the accuracy of the control systems.

In this paper, we consider the problem of scheduling a set of precedence related tasks on a target parallel system (Sinnen [5]), consisting of a set of identical processors connected by a communication network. In this system, each processor can execute only one task at a time and the execution is not preemptive. Also, the cost of communication between tasks executed

* Corresponding author. Address: Department of Industrial Engineering, Faculty of Engineering, Ferdowsi University of Mashhad, P.O. Box: 91775-1111, Mashhad, Iran. Tel.: +98 511 8805092.
E-mail addresses: foroogh.abasian@stu-um.ac.ir (F. Abasian), m_ranjbar@um.ac.ir (M. Ranjbar), msalari@um.ac.ir (M. Salari), morteza.davari@econ.kuleuven.be (M. Davari), sm_khatami@stu.um.ac.ir (S.M. Khatami).

http://dx.doi.org/10.1016/j.apm.2014.01.006
0307-904X/

on the same processor, local communication, is negligible and therefore considered zero. This assumption is based on the observation that for many parallel systems remote communication (i.e. interprocessor communication) is one or more orders of magnitude more expensive than local communication (i.e. intraprocessor communication). In addition, intraprocessor communication is performed by a dedicated communication subsystem. Interprocessor communication, referred to as *communication* hereafter, in the system is performed concurrently; there is no contention for communication resources. As an example the Uniform Memory Access (UMA), a shared memory architecture used in parallel computers, represents this model. For scheduling, suppose we are given a directed acyclic graph which represents the precedence relations among the tasks. The assumption is that for each task, a processing time, a due date and a weight corresponding to its relative importance, are given. The goal is to determine the best schedule, the processor and the start time corresponding to each task, in which the total weighted late work is minimized.

Using standard notification, which was originally presented by Graham et al. [6] and later extended by Veltman et al. [7], this problem can be shown as $P_m|prec,\ comu|Y_w$.

There are many papers related to the scheduling problems of identical processors in the literature such as Gendreau et al. [8] and Blazewicz et al. [9]. Verrite [10] considered the problem $P_m|prec,\ comu|T_{max}$ and $P_m|prec,\ comu|L_{max}$ where T_{max} and L_{max} indicate the maximum *tardiness* and the maximum *lateness*, respectively. Sinnen [5] reviewed different situations of task scheduling for parallel systems. Similarly, Drozdowski [11] considered scheduling problems for parallel processing.

Many research papers related to the minimization of *late work* are proposed in the literature. In particular, these papers can be classified into the five following categories: *single machine* (Potts and Van Wassenhove [12], Kovalyov et al. [13]), *parallel machine* (Blazewicz [14]), *flow shop* (Blazewicz et al. [15], Pesch and Sterna [16]), *job shop* (Blazewicz et al. [17]) and *open shop* (Blazewicz et al. [18]). For a comprehensive survey, interested readers are referred to (Sterna [19]). However, to the best of our knowledge, the problem $P_m|prec,\ comu|Y_w$ has not been taken into account in the literature.

As shown by Sterna [20], there is a polynomial time algorithm for $P_m|r_i,\ p_i = 1|Y_w$ while $P_2|p_j = 1,\ cains|Y$ is *NP*-hard. Since $P_2|prec,\ comu|Y_w$ is a generalization of $P_2|p_j = 1,\ cains|Y$, it is *NP*-hard as well.

The main contributions of this article are twofold: (1) we provide the first description of $P_m|prec,\ comu|Y_w$ and developed an integer linear formulation for it; (2) we develop a branch-and-bound (B&B) algorithm for the problem and derive upper and lower bounds as well as dominance rules to improve the performance of the B&B algorithm.

The remainder of this paper is organized as follows. Theory of the work including the mathematical statement of the problem and the proposed B&B algorithm are presented in Sections 2 and 3, respectively. Results and discussions are reported in Section 4. Finally, concluding remarks are presented in Section 5.

2. Problem statement

In $P_m|prec,\ comu|Y_w$, a set of tasks $N = \{1,\ldots,n\}$ must be processed on a set of identical parallel processors $M = \{M_1,\ldots,M_m\}$. For each task $i \in N$, we are given a processing time p_i, a due date d_i and a weight w_i where all parameters are supposed to be deterministic and non-negative integer values. Each task $i \in N$ must be processed without preemption on a processor. The output of some tasks constitutes the input of some others; thus, there is finish-to-start precedence relation between some pairs of tasks, represented by set A, i.e. a (strict) partial order on N. If s_i indicates the start time of task i, set A is defined as an irreflexive and transitive relation imposing the constraints $s_i + p_i + \Delta_{ik} \leqslant s_k$ for all $(i,k) \in A$ in which Δ_{ik} shows the communication delay between tasks i and k and is zero if both are processed on the same processor. We establish the directed acyclic graph $G(N,A)$ in which sets N and A correspond to the set of nodes and arcs, respectively. We aim to find a schedule that minimizes the total *weighted late work* where such a schedule can be obtained by employing efficient task partitioning and scheduling strategies. The late work of task i is mathematically defined as $LW_i = \min\{Tr_i,\ p_i\}$ where $Tr_i = \max\{f_i - d_i,\ 0\}$ indicates the tardiness of task i in which f_i shows the finish time of task i.

In order to formulate the problem, in the following we introduce some variables.

$$X_{ijt} = \begin{cases} 1; & \text{if task } i \text{ is completed on processor } M_j \text{ at time instant } t, \\ 0; & \text{Otherwise.} \end{cases}$$

$$Z_i = \begin{cases} 1; & \text{if} Tr_i \leqslant p_i, \\ 0; & \text{if} Tr_i > p_i. \end{cases}$$

The model reads as follows:

$$Min \sum_{i=1}^{n} w_i LW_i. \tag{1}$$

Subject to

$$Tr_i \geqslant tX_{ijt} - d_i; \quad \forall i \in N, \quad \forall j \in M \text{ and } t = 1,\ldots,T, \tag{2}$$

$$LW_i \leqslant Tr_i; \quad \forall i \in N, \quad \forall j \in M \text{ and } t = 1,\ldots,T, \tag{3}$$