European Journal of Operational Research 215 (2011) 524–531

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

Single machine batch scheduling with two competing agents to minimize total flowtime

Baruch Mor, Gur Mosheiov *

School of Business Administration, The Hebrew University, Jerusalem, Israel

ARTICLE INFO

Article history:
Received 15 September 2010
Accepted 28 June 2011
Available online 5 July 2011

Keywords:
Two-agent scheduling
Batch scheduling
Single machine
Flowtime

ABSTRACT

We study a single machine scheduling problem, where two agents compete on the use of a single processor. Each of the agents needs to process a set of jobs in order to optimize his objective function. We focus on a two-agent problem in the context of batch scheduling. We assume identical jobs and identical (agent-dependent) setup times. The objective function is minimizing the flowtime of one agent subject to an upper bound on the flowtime of the second agent. As in many real-life applications, we restrict ourselves to settings where the batches of the second agent must be processed continuously. Thus, the batch sizes are partitioned into three parts, starting with a sequence of the first agent, followed by a sequence of the second agent, and ending by another sequence of the first agent. In an optimal schedule, all three are shown to be decreasing arithmetic sequences. We introduce an efficient $O(n^{\frac{3}{2}})$ solution algorithm (where n is the total number of jobs).

1. Introduction

The topic of "scheduling with two competing agents" has become popular among researchers in recent years. The general setting consists of two agents, denoted by X and Y, who need to schedule two sets of jobs on a single machine. The set of agent X contains n_X jobs, and the set of agent Y contains n_Y jobs. Each one of the two agents has his own objective function. The problem to be solved is minimizing the value of the objective function of agent X, subject to an upper bound on the value of the objective function of agent Y.

The relevant reference list contains the following papers: Baker and Smith (2003) focused on the criteria of makespan, maximum lateness and total weighted completion time, and proved that all three problems are NP-hard. Agnetis et al. (2004) considered the same and additional measures (such as the number of tardy jobs and maximum regular functions), and focused on a single machine and on shops. Cheng et al. (2006) solved the problem with the objective of minimum number of tardy jobs for each agent. They proved that the problem is strongly NP-hard and introduced a polynomial time solution for the case of unit time jobs. Ng et al. (2006) studied minimum weighted completion time for one agent subject to a given permitted number of tardy jobs for the second agent. Cheng et al. (2007) focused on total tardiness. They considered release dates and allowed preemption. Agnetis et al. (2007) solved multi-agent problems with some of the above mentioned objectives. Their main result is that when one agent wants to minimize the sum of weighted completion times and the second minimizes makespan, the total number of non-dominated solutions is exponential. Cheng et al. (2008) considered multi-agent problems with precedence constraints. Liu and Tang (2008) and Liu et al. (2009) provided polynomial time solutions for single machine two agent problems with deteriorating jobs. Agnetis et al. (2009) introduced branch-and-bound algorithms for the problems of minimum total weighted completion time of agent X subject to an upper bound on several measures of agent Y. Lee et al. (2009) provided fully polynomial approximation schemes and an approximation algorithm with a reasonable worst case bound for multi-agent scheduling where the objective function is total weighted completion time. Mor and Mosheiov (2010) focused on two agent problems with various earliness measures. Leung et al. (2010) extended some of the above problems to include preemption, release dates, and parallel identical machine. Wan et al. (2010) studied two agent problems with controllable processing times, and Liu et al. (2010) considered two agent single machine scheduling problems with position-dependent processing times, where the objective is to minimize the first agent's total flow time, subject to an upper bound on the second agent's maximum cost.

In the specific setting studied in this paper, the objective functions of both agents X and Y are minimum flowtime. We focus here on *batch scheduling*, where jobs may be grouped and processed in batches. Batching is generally based on the existence of some similarity between jobs belonging to the same class or family. A batch processing time is identical to the total processing times of the jobs

* Corresponding author. Address: The Hebrew University of Jerusalem, Mount Scopus, Jerusalem 91905, Israel.

E-mail addresses: msomer@mscc.huji.ac.il, baruch.mor@mail.huji.ac.il (G. Mosheiov).

0377-2217/$ - see front matter
doi:10.1016/j.ejor.2011.06.037

contained in the batch. A *setup time* is incurred when initiating the processing of a new batch. We focus on the classical batch scheduling problem introduced by Santos and Magazine (1985), who assumed both identical job processing times, and identical (batch-independent) setup times. They also assumed *batch availability*, i.e., jobs become available only upon the completion of their entire batch. The objective function is minimum total flowtime. Thus, there is a trade-off between creating a few large batches leading to long waiting times of the jobs, and having many small batches causing many setups. Santos and Magazine (1985) and Naddef and Santos (1988) introduced an explicit expression for the optimal number of batches, and showed that batch sizes follow a decreasing arithmetic sequence with a difference identical to the setup time. We note that Santos and Magazine (1985) studied the "relaxed version" of the problem, where batch sizes are not required to be integers. The integer version was solved later by Shallcross (1992) and Mosheiov et al. (2005). For the many extensions and generalizations of the model of Santos and Magazine (1985), we refer the reader to the survey paper on batch scheduling models of Allahverdi et al. (2008).

This paper appears to be the first to combine multi-agent and batch scheduling. In addition to its theoretical importance, this combination appears to be very practical. [Clearly, there are numerous applications to batch scheduling and to group technology manufacturing techniques in various contexts; we refer the reader again to the introduction of Allahverdi et al. (2008) for a list of real-life examples. On the other hand, there are many applications to multi-agent scheduling; see e.g. Agnetis et al. (2007).] For the specific problem studied here, we extend the application given in Cheng et al. (2008). They consider telecommunication services, where the problem is to satisfy the service requirements of individual agents who compete for the use of commercial satellite to transfer voice, image and text files for their clients. The different agents buy communications slots of a single common transponder. Each type of data is partitioned into identical sized packets, and transmitted in a time multiplexed technique. One relevant objective function in this context (also considered by Cheng et al., 2008) is minimum flowtime, which is the objective considered in this paper.

Following Agnetis et al. (2004), we minimize the flowtime of agent X, subject to a pre-specified upper bound on the flowtime of agent Y . Each agent needs to process a set of identical jobs on the same machine. Each agent may process his jobs in batches, where the completion time of each job is identical to the completion time of its batch. Identical (batch-independent) but *agent-dependent* setup times are assumed. Job preemption is not permitted. As in many real-life applications, we restrict ourselves to settings where the batches of agent Y must be processed continuously. We thus focus here on a schedule consisting of a sequence of batches of agent X, followed by a continuous sequence of batches of agent Y, followed by a sequence of batches of agent X. Indeed, a continuous processing of the Y jobs is a requirement in many applications. For example, in the above mentioned application of telecommunication services, a relevant setting is when agent X needs to transmit relatively static text files, while agent Y transmits a live-event (e.g. an important speech of the US president, a space shuttle launch, the finals of the world cup, etc.), which clearly cannot be interrupted. Another general possible setting is that of a manufacturing firm with excessive production capability. The firm (agent X) offers its production line to an external company (agent Y). Restricted by a contract with agent Y that his flowtime cannot exceed a prespecified value, agent X schedules his jobs so as to optimize his own objective.

The relaxed version of the problem is formulated as a quadratic programming with linear constraints, and is shown to be solved in linear time in the number of jobs. The optimal schedule is shown to have a unique and surprising form: the batch sizes of agent X are divided into two decreasing arithmetic sequences, and the batch sizes of agent Y, scheduled as late as possible, form a decreasing arithmetic sequence as well. We introduce a simple rounding procedure to obtain integer batch sizes with no additional computational effort.

The paper is organized as follows: in Section 2 we provide the problem formulation. In Sections 3 and 4 we solve the relaxed versions for agent Y and agent X, respectively. Section 5 contains the formal algorithm. Section 6 presents the rounding procedure; Section 7 contains numerical examples. In Section 8 we report the results of our numerical tests.

2. Formulation

Formally, two agents X and Y need to process n_X and n_Y jobs, respectively, on a single machine. All the jobs are available at time zero, and preemption is not allowed. We assume unit processing time for all the jobs. Jobs may be processed in batches. A setup time is required when starting a new batch. We assume *agent-dependent* setup time denoted by S_X and S_Y. For a given job allocation to batches, let m_X and m_Y denote the number of batches of agents X and Y, respectively. Let $B_{Z,j}$, $Z = X,Y$, $j = 1,\dots,m_Z$, denote batch j of agent Z, and $n_{Z,j} = |B_{Z,j}|$, $Z = X,Y$, $j = 1,\dots,m_Z$, denote the number of jobs assigned to batch j of agent Z. Clearly $n_Z = \sum_{j=1}^{m_Z} n_{Z,j}$.

For a given allocation of jobs to batches, let $C_{Z,j}$ denote the completion time of batch $B_{Z,j}$, $Z = X,Y$, $j = 1,\dots,m_Z$. As mentioned, we assume *batch availability*, i.e., the completion time of a job is the completion time of the batch to which it is assigned. The contribution of batch $B_{Z,j}$ to the flowtime is $n_{Z,j}C_{Z,j}$, thus the flowtime of agent Z (assuming a single-agent problem) is given by $(\sum C)_Z = \sum_{j=1}^{m_Z} n_{Z,j}C_{Z,j}$. The maximum allowed flowtime of agent Y (an upper bound) is denoted by U_Y.

The problem to be solved is minimum flowtime of agent X subject to an upper bound on the flowtime of agent Y, i.e., $min(\sum C)_X$ s.t. $(\sum C)_Y \leqslant U_Y$. As mentioned above, we restrict ourselves to settings where the batches of agent Y must be processed continuously, i.e. the batches of agent X are partitioned into two sequences, scheduled prior to and after the batches of agent Y, respectively (and thus forming schedules of X-Y-X structure). Using the conventional notation, the problem studied in this paper is $1/batch,\ p = 1,\ S_X,\ S_Y/(\sum C)_X : (\sum C)_Y$.

[For convenience, let $n = n_X + n_Y$. The solution procedure introduced in this paper requires no more than an effort of $O(n)$ time. We note that the input for the problem contains the number of unit jobs n, the setup times S_X and S_Y, and the upper bound U_Y. It follows that the complexity of our proposed procedure is *not* polynomial in the input size.]

3. An optimal solution for agent *Y* (the relaxed version)

In this section we focus on solving the relaxed version of the problem of agent Y, where batch sizes are not necessarily integers. Thus, the (optimal) solution obtained here is a lower bound on the true optimal flowtime for the original problem, where batch sizes are forced to be integers. (As mentioned, a simple rounding procedure is presented later.)

In order to guaranty a feasible solution we assume that $(\sum C)_Y \leqslant U_Y$ (otherwise, obviously no schedule is feasible). In the following, an upper script will reflect the solution of Santos and Magazine model (1985) for agent Z, e.g., $(\sum C)_Z^{(SM)}$ denotes the optimal flowtime of agent Z, $Z = X,Y$. Thus the solution for agent Z according to Santos and Magazine (1985) contains the following:

The optimal number of batches:

$$m_Z^{(SM)} = \left\lfloor \sqrt{\frac{1}{4} + \frac{2n_Z}{s_Z}} + \frac{1}{2} \right\rfloor. \quad (1)$$