Available online at www.sciencedirect.com
ScienceDirect
European Journal of Operational Research 183 (2007) 944–948
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/ejor

Short Communication

Customer order scheduling to minimize the number of late jobs ☆

B.M.T. Lin [a,*], A.V. Kononov [b]

[a] Department of Finance and Information Management, Institute of Information Management, National Chiao Tung University, Taiwan
[b] Sobolev Institute of Mathematics, Novosibirsk, Russia

Received 12 January 2006; accepted 13 October 2006
Available online 13 December 2006

Abstract

In the order scheduling problem, every job (order) consists of several tasks (product items), each of which will be processed on a dedicated machine. The completion time of a job is defined as the time at which all its tasks are finished. Minimizing the number of late jobs was known to be strongly *NP*-hard. In this note, we show that no FPTAS exists for the two-machine, common due date case, unless $P = NP$. We design a heuristic algorithm and analyze its performance ratio for the unweighted case. An LP-based approximation algorithm is presented for the general multicover problem. The algorithm can be applied to the weighted version of the order scheduling problem with a common due date.

Keywords: Order scheduling; Number of late jobs; Approximability; Approximation algorithm; Multicover problem

1. Introduction

Order management is one of the crucial issues in the manufacturing industry. In this study, we consider order scheduling to minimize the number of late orders. For consistency with the scheduling literature we use *jobs* instead of *orders* hereafter. Consider a set of jobs $N = \{J_1, J_2, \ldots, J_n\}$ available from time zero onwards for processing. Each job $J_i \in N$ consists of m operations O_{ik}, $1 \leqslant k \leqslant m$, to be processed on m independent dedicated machines $M_1, M_2, \ldots, M_m$ for which operation O_{ik},$1 \leqslant k \leqslant m$, can be processed on machine M_k only and has a processing time p_{ik}. The m operations of a job are independent and therefore can be processed simultaneously by their specific machines. At any time, each machine can process at most one operation. No preemption is allowed. The completion time of operation O_{ik} on machine M_k is denoted by C_{ik}. A job is completed only if its operations are all finished. Therefore, the completion time of job J_i is defined as $C_i = \max_{1 \leqslant k \leqslant m}\{C_{ik}\}$. Each job $J_i \in N$ is associated with a due date d_i that specifies the time it is expected to be completed. Binary variable U_i dictates whether job J_i is late or not, i.e.

☆ This research is supported by Taiwan–Russia joint grant under contract numbers NSC94-2416-H-009-013 and RP05H01(05-06-90606-HHCa).
* Corresponding author. Tel.: +886 3 5131472; fax: +886 3 5729915.
E-mail addresses: bmtlin@iim.nctu.edu.tw (B.M.T. Lin), alvenko@math.nsc.ru (A.V. Kononov).

0377-2217/$ - see front matter
doi:10.1016/j.ejor.2006.10.021

$U_i = 1$ if $C_i > d_i$; 0, otherwise. In this paper, we want to schedule the jobs so as to minimize the number of late jobs. Following the three-field notation used in Leung et al. (2006), the studied problem is denoted by $PD\|\sum U_i$.

As indicated by Roemer (2006), the first paper concerning order scheduling may be due to Ahmadi and Bagchi (1990). Order scheduling can also be regarded as a degenerate case of the two-stage assembly-type flowshop (Lee et al., 1993) by ignoring the second-stage assembly operation. Wagneur and Sriskandarajah (1993) later studied the order scheduling problem from a different aspect "open shop scheduling with job overlaps". They investigated several standard objectives, including makespan, total completion time, maximum lateness, total tardiness, and number of late jobs. Roemer (2006) summarized several different but independent lines of order scheduling research that have appeared in the literature. Because this note focuses on only the objective of number of late jobs $\sum U_i$, we do not review the known results on other objectives. The reader is referred to Roemer (2006) for a thorough classification. To minimize the number of late jobs, Wagneur and Sriskandarajah (1993) showed that the problem is NP-hard even if there are only two machines. Ahmadi and Bagchi (1997) and Cheng et al. (2006) independently developed a pseudo-polynomial time algorithm for the weighted case $PDm\|\sum w_iU_i$, where m is a fixed number of machines. Leung et al. (2006) showed that the $PD\|\sum U_i$ problem is strongly NP-hard, even if a common due date is assumed. They introduced a Revised Hodgson–Moore (RHM) algorithm to solve the case with agreeable conditions. Ng et al. (2003) presented a negative approximability result that the $PD|d_i = d, p_{ik} \in \{0, 1\}|\sum U_i$ problem has no $c \cdot \ln m$-approximation algorithm for some constant $c > 0$, unless $P = NP$. Further, an LP-rounding algorithm with performance ratio $d + 1$ was designed for the weighted case $PD|d_i = d, p_{ik} \in \{0, 1\}|\sum w_iU_i$.

The rest of this study proceeds as follows. In Section 2, we introduce a proof for the non-existence of fully polynomial time approximation schemes for the two-machine, common due date case. Section 3 is dedicated to the development of a heuristic algorithm that exhibits a worst-case performance of m. In Section 4, we design and analyze an LP-based approximation algorithm for the multicover problem. The result is applied to the common due date case of minimizing the weighted number of late jobs. Section 5 gives some concluding remarks.

2. Non-approximability about FPTAS

In this section, we prove that there is no fully polynomial time approximation scheme (FPTAS) unless $P = NP$ for the $PD\|\sum U_i$ problem even when there are only two machines and a common due date.

Theorem 1. *No FPTAS exists for* $PD2|d_i = d|\sum U_i$ *unless* $P = NP$.

Proof. To prove the theorem, we use the NP-hard Equal-Size-Partition problem [SP12] (Garey and Johnson, 1979): The input instance consists of integer W and $2t$ positive integers $S = \{x_1, \ldots, x_{2t}\}$ that sum up to $2W$. The problem is to determine if there is a subset of items S' with $|S'| = t$ and $\sum_{x_i \in S'} x_i = \sum_{x_i \notin S'} x_i = W$. Let $X = 1 + W$. Consider the following instance of the $PD2|d_i = d|\sum U_i$ problem with $2t$ jobs: For $i = 1, \ldots, 2t$, job J_i is created with $p_{i1} = x_i + X$ and $p_{i2} = \frac{2W}{t} - x_i + x_{\max}$, where $x_{\max} = \max_{x_i \in S}\{x_i\}$. The common due date is $d = W + tX$.

Suppose that $PD2|d_i = d|\sum U_i$ does permit the existence of an FPTAS. We set $\epsilon = \frac{1}{2t}$. Assume that the FPTAS returns a schedule with at most t late jobs. So we have at least t jobs completed before time d. Note that for any $t + 1$ jobs, the sum of their processing times on machine one is greater than $(t + 1)X > W + tX$. It follows that there must be exactly t non-tardy jobs. Denote the set of non-tardy jobs by N'. With this definition, we must have $\sum_{J_i \in N'} p_{i1} \leqslant d$. It follows that $\sum_{J_i \in N'}(x_i + X) \leqslant W + tX$. Therefore,

$$\sum_{J_i \in N'} x_j \leqslant W. \tag{1}$$

Also, we have $\sum_{J_i \in N'} p_{i2} \leqslant d$. It follows that $\sum_{J_i \in N'}\left(\frac{2W}{t} - x_i + X\right) \leqslant W + tX$. Therefore,

$$W \leqslant \sum_{J_i \in N'} x_i. \tag{2}$$

Inequality (1) and Inequality (2) together imply that $\sum_{J_i \in N'} x_i = W$. Consequently, the elements corresponding to the jobs in set N' constitute a solution to Equal-Size-Partition.

Next we assume that the Equal-Size-Partition problem has a solution S' with $|S'| = t$ and $\sum_{i \in S'} x_i = W$. We schedule the jobs corresponding to the elements in S' first. Then, we have t jobs that are early or on-time. By the choice of ϵ, the approximate objective value reported by the FPTAS must be at most $(1 + \epsilon)t < t + 1$. Since the objective function takes only integer values, the approximate objective value must be at most t

Therefore, we come up with the fact that the FPTAS of $PD2|d_i = d|\sum U_i$ would find in polynomial time a solution with an objective value at most t if and only if Equal-Size-Partition has a solution. □

3. Approximation algorithm for the unweighted case

In this section, we develop a lower bound and a heuristic with performance analysis. We start from the development of a lower bound of the general problem. An algorithm due to Hodgson and Moore finds an optimal solution of the single machine problem $1\|\sum U_i$ in $O(n \log n)$ time. We can decompose the m-machine order scheduling problem to m independent single machine problems. Let G_k be the number of late jobs in the optimal solution of the single machine problem on machine M_k. It is clear that $G_k \leqslant \text{OPT}$, where OPT stands for the optimal solution value in the order scheduling problem. Thus we have the following lower bound by combining all these independent bounds: $\text{LB} = \max_{k=1,\dots,m}\{G_k\} \leqslant \text{OPT}$. Note that the lower bound can be obtained in $O(mn \log n)$ time.

To generate a feasible schedule, it is sufficient to find a set S of jobs which are early or on-time. We will construct set S by the following steps. Add the jobs to S one by one in nondecreasing order of due dates. If job J_j is completed after d_j on some machine M_k when added into S, we call J_j a *critical job*. A job with the largest processing time on this machine is marked to be late and removed from S. Let t_k denote the current schedule time on machine M_k in the running session of the algorithm. The algorithm is outlined as:

Algorithm A

1. Re-index the jobs such that $d_1 \leqslant d_2 \leqslant \cdots \leqslant d_n$.
2. $S := \emptyset$; $t_k := 0$ for all $k = 1,\dots,m$.
3. **For** $j := 1$ **to** n **do:**
   $S := S \cup \{J_j\}$;
   For $k := 1$ **to** m **do:** $t_k := t_k + p_{jk}$;
   For $k := 1$ **to** m **do:**
   If $t_k > d_j$ **then**
      Find job J_i in S with the largest p_{ik};
      $S := S \backslash \{J_i\}$;
      For $k := 1$ **to** m **do:** $t_k := t_k - p_{ik}$.
4. Stop.

Denote the number of late jobs in the schedule constructed by Algorithm A by $\sum U_i(\mathbf{A})$.

Theorem 2. *For the $PD\|\sum U_i$ problem, the performance ratio $\frac{\sum U_i(\mathbf{A})}{LB} \leqslant m$.*

Proof. Let N_k be the set of jobs that were removed from S when the algorithm encountered due-date violation on machine M_k. Then $\sum U_i(\mathbf{A}) = \sum_{k=1}^{m}|N_k|$. Consider machine M_k that has the largest value of $|N_k|$, i.e. $|N_k| = \max_{1 \leqslant j \leqslant m}\{|N_j|\}$. Let $N' = S \cup N_k$. On machine M_k, consider the classical $1\|\sum U_i$ problem on the set of jobs N' with processing times p_{jk} of jobs $J_j \in N'$. Let G'_k be the optimal solution value of the set N'. Since $N' \subseteq N$, we know that $G'_k \leqslant G_k$, the optimal value of $1\|\sum U_i$ on set N. But Algorithm A on set N' constructs the same schedule as Hodgson–Moore algorithm, i.e our algorithm produces an optimal solution to the $1\|\sum U_i$ problem. It follows that $\sum U_i(\mathbf{A}) = \sum_{k=1}^{m}|N_k| \leqslant mG'_k \leqslant mG_k \leqslant m\text{LB}$ and that Algorithm A is an m-approximation algorithm for $PDm\|\sum U_i$. □

To establish the tightness of the performance ratio m, we consider the following instance of $m+1$ jobs. For jobs $J_i, 1 \leqslant i \leqslant m$, $p_{ik} = 2$ if $k = i$; 0, otherwise. Job J_{m+1} has $p_{m+1,k} = 1$ on all k. All jobs have a common due date $d = 2$. An optimal solution to the instance successfully schedules jobs $J_1, J_2, \dots, J_m$. Algorithm A reports a schedule with only job J_{m+1} scheduled early. Therefore, $\frac{\sum U_i(\mathbf{A})}{\text{LB}} = m$.

The agreeable conditions specify that for any $J_i, J_j \in N$, either $p_{ik} \leqslant p_{jk}$ for all $k, 1 \leqslant k \leqslant m$ or $p_{ik} \geqslant p_{jk}$ for all $k, 1 \leqslant k \leqslant m$. The RHM algorihtm proposed by Leung et al. (2006) works in the following way to optimally solve the agreeable case: If an order is scheduled and the maximum of its completion time over all m machines is larger than its due date, then the order in the partial schedule that has the longest processing time on each one of the machine is deleted. Although Algorithm A provides approximate solutions to the general case, it can solve the agreeable case to optimality. The RHM however cannot deal with the general case.

4. Approximation algorithm for the weighted case

In this section, we consider the order scheduling problem of minimizing the weighted number of late jobs, $PD\|\sum w_iU_i$. Now, we assume that job J_i has weight $w_i \geqslant 0$. Let $P_k = \sum_{i=1}^{n}p_{ik}$ for $k = 1, 2, \dots,$

m, and $P_{\max} = \max_{1 \leqslant k \leqslant m}\{P_k\}$. As noted in Ng et al. (2003) the weighted version with common due date d can be formulated as the following integer linear program:

$$\text{ILP}_{PD}\text{Minimize} \sum_{i=1}^{n} w_i x_i \tag{3}$$

subject to

$$\sum_{i=1}^{n} p_{ik} x_i \geqslant P_k - d, \quad 1 \leqslant k \leqslant m; \tag{4}$$

$$x_i \in \{0,1\}, \quad 1 \leqslant i \leqslant n. \tag{5}$$

Here, job J_i is late if $x_i = 1$.

Note that ILP_{PD} (3)–(5) is a special case of the multicover problem (Hochbaum, 1997).

$$\text{ILP}_{\text{MC}}\text{Minimize} \sum_{i=1}^{n} w_i x_i \tag{6}$$

subject to

$$\sum_{i=1}^{n} p_{ik} x_i \geqslant b_k, \quad 1 \leqslant k \leqslant m; \tag{7}$$

$$x_i \in \{0,1\}, \quad 1 \leqslant i \leqslant n. \tag{8}$$

For the multicover problem, three approximation algorithms were proposed: LP-algorithm, rounding algorithm (Hochbaum, 1997), and dual-feasible algorithm (Hall and Hochbaum, 1986). The three approaches are all $P_{\max}$-approximate algorithms, but the heuristic solution value of the dual-feasible algorithm does not exceed $\max_{k=1,\ldots,m}\sum_{J_i \in N_{\text{late}}} p_{ik}$ times the value of optimum, where N_{late} is the set of late jobs. For instances with large due dates, which correspond to small values of b_i, $i = 1,\ldots,n$, this quantity could be much smaller than $P_{\max}$.

Another approach is considered by Ng et al. (2003) for the case of common due date and 0–1 processing times, $PD|d_i = d, p_{ik} \in \{0,1\}|\sum w_i U_i$. They presented a $d+1$-approximate algorithm where d is the common due date. In the following, we show that a similar algorithm can be used for the multicover problem and thereby for the problem $PD|d_i = d|\sum w_i U_i$. Let LP be a linear program obtained from ILP (6)–(8) by relaxing the integer constraints (8), i.e. $0 \leqslant x_{ik} \leqslant 1$ for $i = 1,\ldots,n$. Denote the value of optimal solution of LP by LB_{LP}. It is obvious that LB_{LP} is a lower bound of the multicover problem. Let $\delta_k = P_k - b_k$, $1 \leqslant k \leqslant m$ and $\delta_{\max} = \max_{k=1,\ldots,m}\{\delta_k\}$. The following LP-based algorithm can provide new approximation results for the multicover problem.

Algorithm LP:

1. Solve the LP and obtain an optimal solution $x^* = (x_1^*, x_2^*, \ldots, x_n^*)$.
2. Output a subset $\{i | x_i^* \geqslant 1/(\delta_{\max}+1)\}$.

Theorem 3. *For the multicover problem,* **Algorithm LP** *has a performance ratio* $\frac{\sum w_i U_i(\mathbf{LP})}{LB_{LP}} \leqslant \delta_{\max} + 1$.

Proof. It is clear that after the rounding step, $\sum w_i x_i$ is no greater than $(\delta_{\max}+1)\text{LB}_{\text{LP}}$. So we have to prove that the derived solution is feasible for ILP (6)–(8). Let constraints (7) fail for some k, i.e. $\sum_{\{i|x_i^* \geqslant \frac{1}{\delta_{\max}+1}\}} p_{ik} \leqslant P_k - \delta_k - 1$. It follows that

$$\sum_{\{i|x_i^* < \frac{1}{\delta_{\max}+1}\}} p_{ik} \geqslant \delta_k + 1. \tag{9}$$

Then, we have

$$\begin{aligned}\sum_{k=1}^{n} p_{ik} x_i^* &= \sum_{\{i|x_i^* \geqslant \frac{1}{\delta_{\max}+1}\}} p_{ik} x_i^* + \sum_{\{i|x_i^* < \frac{1}{\delta_{\max}+1}\}} p_{ik} x_i^* \\ &< \sum_{\{i|x_i^* \geqslant \frac{1}{\delta_{\max}+1}\}} p_{ik} + \frac{1}{\delta_{\max}+1} \sum_{\{i|x_i^* < \frac{1}{\delta_{\max}+1}\}} p_{ik} \\ &= P_k - \frac{\delta_{\max}}{\delta_{\max}+1} \sum_{\{i|x_i^* < \frac{1}{\delta_{\max}+1}\}} p_{ik}.\end{aligned}$$

Inequality (9) further implies

$$\sum_{i=1}^{n} p_{ik} x_i^* < P_k - \frac{\delta_{\max}(\delta_k+1)}{\delta_{\max}+1} \leqslant P_k - \delta_k.$$

The last inequality follows from $\delta_k \leqslant \delta_{\max}$. Therefore, we come to a contradiction to the assumption $\sum_{i=1}^{n} p_{ik} x_i^* \geqslant P_k - \delta_k$. □

Assume $\delta_k = d$ for all $1 \leqslant k \leqslant m$. The following corollary readily follows.

Corrollary 1. *For the* $PD|d_i = d|\sum w_i U_i$ *problem, Algorithm LP has a performance ratio* $\frac{\sum w_i U_i(\mathbf{LP})}{LB_{LP}} \leqslant d+1$.

The performance ratio is the same as that given in Ng et al. (2003) for $PD|d_i = d, p_{ik} \in \{0,1\}|\sum w_i U_i$. However, our result is given for a more general case $PD|d_i = d|\sum w_i U_i$, in which no constraint is assumed for processing times. Note that in contrast to the dual-feasible algorithm, Algorithm **LP** will produce good approximate solutions for the case with small due dates.

5. Conclusion

This study has investigated the order scheduling problem of minimizing the (weighted) number of late jobs. A reduction from Equal-Size-Partition was conducted to establish the fact that there cannot exist a fully polynomial time approximation scheme unless $P = NP$. A heuristic algorithm with performance ratio m was designed. We have also examined the general multicover problem and proposed a new approximation algorithm. The results in the mean time lead to a $(d + 1)$-approximation algorithm for the order scheduling problem with a common due date.

As aforementioned, order scheduling problems are not yet extensively studied. There is considerable room for further research involving such constraints as release dates or precedence relationships.

Acknowledgements

The authors are grateful to the anonymous referees for indicating typos and providing information on some papers that have addressed the studied problem.

References

Ahmadi, R.H., Bagchi, U., 1990. Scheduling of multi-job customer orders in multi-machine environments. ORSA/TIMS, Philadelphia.

Ahmadi, R.H., Bagchi, U., 1997. Coordinated scheduling of customer orders, Updated Working Paper, Anderson School at UCLA.

Cheng, T.C.E., Wang, Q., Yuan, J., 2006. Customer order scheduling on multiple facilities. Private Communication.

Garey, M.R., Johnson, D.S., 1979. Computers and Intractability: A Guide to the Theory of NP-Completeness. Freedman, San Francisco, CA.

Hall, N.G., Hochbaum, D.S., 1986. A fast approximation algorithm for the multicovering problem. Discrete Applied Mathematics 15, 35–40.

Hochbaum, D.S., 1997. Approximating covering and packing problems: set cover, vertex cover, independent set, and related problems. In: Dorit Hochbaum (Ed.), Chapter 3 in Approximation Algorithms for NP-hard Problems, PWS Publishing Company, University of California, Berkeley, USA.

Lee, C.Y., Cheng, T.C.E., Lin, B.M.T., 1993. Minimizing the makespan in three-machine assembly type flow shop problem. Management Science 39, 616–625.

Leung, J.Y.-T., Li, H., Pinedo, M., 2006. Scheduling orders for multiple product types with due date related objectives. European Journal of Operational Research 168 (2), 370–389.

Ng, C.T., Cheng, T.C.E., Yuan, J.J., 2003. Concurrent open shop scheduling to minimize the weighted number of tardy jobs. Journal of Scheduling 6, 405–412.

Roemer, T.A., 2006. A note on the complexity of the concurrent open shop problem. Journal of Scheduling 9 (4), 389–396.

Wagneur, E., Sriskandarajah, C., 1993. Open shops with jobs overlap. European Journal of Operational Research 71, 366–378.