KU LEUVEN

FACULTY OF ECONOMICS AND BUSINESS

New results on the coordination of transportation and batching scheduling

Zhu H, Leus R, Zhou H.

KBI_1416

New results on the coordination of transportation and batching scheduling

Hongli Zhu[a], Roel Leus[b,*], Hong Zhou[a]

[a]*School of Economics and Management, Beihang University, Beijing 100191, PR China*
[b]*ORSTAT, Faculty of Economics and Business, KU Leuven, Leuven 3000, Belgium*

Abstract

We study a coordinated scheduling problem of production and transportation, where a set of jobs needs to be transported from a holding area to a single batch machine for further processing. A number of results for this combined transportation and scheduling environment have recently been published, looking into the complexity status of the minimization of the sum of total processing time and processing cost, and of the sum of makespan and processing cost, for a fixed number of transporters. In this paper, we add to these results in that (1) we show that the earlier complexity results are still valid when the processing cost is removed from the objective, thus reducing to more "classic" scheduling objectives; (2) we assess the complexity status of the relevant problem variants with free number of transporters; (3) we prove that the weighted-completion-time objective leads to an intractable problem even with a single transporter, contrary to the unweighted case. We also establish a link with the so-called serial batching problem.

Keywords: scheduling, transportation, batching, NP-hardness, strong NP-hardness

1. Introduction

Tang and Gong [1] address a coordination scheduling problem of transportation and batching processing in the iron and steel industry. A set

*Corresponding author.
Email addresses: honglizhu@sem.buaa.edu.cn (Hongli Zhu), Roel.Leus@kuleuven.be (Roel Leus)

Preprint submitted to Elsevier *July 22, 2014*

$N = \{1, 2, \dots, n\}$ of jobs is initially located at a holding area, and each of the jobs needs to be transported by one of m available vehicles before it can be processed by a single batch machine. Each vehicle can transport one job at a time. We let t_j denote the transportation time for job $j \in N$ from the holding area to the machine, and t the (empty) vehicle return time from the machine back to the holding area (to pick up a new job). All vehicles are assumed to be located in the holding area at the start of the planning horizon. In the production stage, up to c jobs can be processed as one batch on the batching machine; the processing time of (all jobs of) each batch is equal to p, which is a constant. A processing cost $\alpha(b)$ is also incurred, which depends on the number b of batches processed on the batching machine. Following [1], the resulting optimization problem is denoted as TBS, short for *transportation and batching scheduling problem* (in which we do not include the specification of the objective function, for ease of notation infra).

Tang and Gong [1] investigate TBS to minimize the sum of total completion times $\sum C_j$ and processing cost. They prove that it is NP-hard even if $m = 2$ and present a pseudo-polynomial-time algorithm and FPTAS for any fixed m. Therefore, the TBS problem is NP-hard in the ordinary sense from their viewpoint. For $m = 1$, the problem turns out to be polynomially solvable. Zhu [2] shows that TBS to minimize the sum of makespan $C_{\max}$ and processing cost is NP-hard even if $m = 2$. Using a method similar to [1], he provides a pseudo-polynomial-time algorithm and FPTAS for any fixed m, thus again concluding NP-hardness in the ordinary sense; he also describes a polynomial-time algorithm for TBS to minimize $C_{\max} + \alpha(b)$ when $m = 1$. In a general application, however, one would typically expect m to be free, or at least $O(n)$, in which case the pseudo-polynomial-time algorithms in [1, 2] become exponential-time. In the next sections, we show that both the problem variants are strongly NP-hard when m is free, excluding the existence of a pseudo-polynomial-time algorithm (unless P = NP).

In the next few lines, we briefly introduce some related works about the coordinated scheduling of transportation and batching processing. Tang et al. [3] provide an elaborate description of the practical relevance of this type of scheduling problems in the context of ingot processing in the steel industry. They investigate TBS with deterioration considerations to minimize the makespan and the processing cost in a bi-criteria setting when $m = 1$; the authors show that different variants of the problem are all strongly NP-hard and present heuristic algorithms. Tang and Gong [4] study scheduling with batch processing and two-stage transportation, including an extra transportation stage after the stages of TBS. They prove the resulting problem to be strongly NP-hard for objective $C_{\max} + \alpha(b)$. Tang and Liu [5] investigate two-machine flowshop scheduling where a single machine is followed by a batching machine and a transporter carries jobs between the processing stages; they prove its strong NP-hardness for the makespan objective. Tang and Liu [6] also consider a two-machine flow-shop setup, but here the batching machine constitutes the first stage and the second stage is a regular machine. They show that makespan minimization is strongly NP-hard and describe a heuristic algorithm.

The contributions of this paper are the following: we show that the complexity results in [1, 2] are still valid when the processing cost is removed from the objective, thus reducing to more "classic" scheduling objectives; we assess the complexity status of the relevant problem variants with free m; and we establish that the weighted-completion-time objective $\sum w_j C_j$ leads to an intractable problem even with a single transporter, contrary to the unweighted case.

The remainder of this paper is structured as follows: the total-completion-time objective is studied in Section 2, the makespan objective is the subject of Section 3 and we look into weighted completion times in Section 4. We establish a link with the so-called serial batching problem in Section 5 and we provide some concluding remarks in Section 6.

2. Total completion times

In this section, we first present the complexity result for free m. Consider the following decision problem:

3-PARTITION
Input: $3h$ elements with integer sizes $a_1, a_2, \ldots, a_{3h}$, where $\sum_{j=1}^{3h} a_j = ha$ and $\frac{a}{4} < a_j < \frac{a}{2}$ for $j = 1, \ldots, 3h$.
Question: does there exist a partition $I_1, I_2, \ldots, I_h$ of the index set $\{1, \ldots, 3h\}$ such that $|I_i| = 3$ and $\sum_{j \in I_i} a_j = a$ for $i = 1, \ldots, h$?

Problem 3-PARTITION is well known to be strongly NP-complete, see for instance Garey and Johnson [7].

Theorem 1. *TBS to minimize $\sum C_j$ is strongly NP-hard when m is free.*

Proof: The proof is based on a reduction from 3-PARTITION to the decision variant of TBS, which is to decide whether or not a solution exists with objective function value less than or equal to a threshold value y. With a given instance of 3-PARTITION, we associate the following TBS-instance. There are two types and a total of $6h$ jobs: the P-jobs denoted by P_j and X-jobs denoted by $X_j, j = 1, 2, \ldots, 3h$. Furthermore,

number of vehicles $m = h$;

transportation time $t_{P_j} = a_j, t_{X_j} = 0$, for $j = 1, \ldots, 3h$;

batch processing time $p = a$;

vehicle return time $t = 0$;

batch machine capacity $c = 3h$;

threshold $y = 9ha$.

We show that there exists a schedule for this TBS-instance with $\sum C_j \leq y$ if and only if the 3-PARTITION-instance is a yes-instance.

$\Leftarrow$ Consider any yes-instance for 3-PARTITION, with $I_1, I_2, \ldots, I_h$ a complying partition. We construct a schedule for the TBS-instance as follows: the vehicle i transports the three jobs in I_i one by one, for $i = 1, 2, \ldots, h$. The total transportation time of each vehicle is thus $\sum_{j \in I_i} a_j = a$. Since $c = 3h$ and $t_{X_j} = 0$, the batching machine can process all X-jobs as the first batch from time zero and all the P-jobs in the second batch from time a onwards. The sum of completion times is then $3ma + 3m \cdot 2a = 9ma = y$.

$\Rightarrow$ Suppose conversely that there is a schedule satisfying $\sum C_j \leq y$, containing q batches. Since $n = 6h$ and $c = 3h$, we have $q \geq 2$. If $q > 2$, then $q \geq 3$ and then $\sum C_j$ will not be less than $3m \cdot a + (3m - 1) \cdot 2a + 3a = (9m + 1)a > y$. Consequently, if $q > 2$ then the objective value will exceed y, and so $q = 2$ (exactly two batches in the schedule). Let S_i denote the starting time of the i-th batch, $i = 1, 2$. Since $3m \cdot (S_1 + a) + 3m \cdot (S_2 + a) \leq 9ma$ and $S_1 + a \leq S_2$, we find that $S_1 = 0, S_2 = a$. Therefore, all the P-jobs must processed in the second batch at time a. Let $I_i \subset N$ denote the set of jobs transported by vehicle i ($i = 1, 2, \ldots, h$). Since $\sum_{j \in I_i} t_j = \sum_{j \in I_i} a_j \leq a$ and $\sum_{j=1}^{3h} a_j = ha$, it is easy to check that $\sum_{j \in I_i} a_j = a$ for $i = 1, 2, \ldots, h$. From the definition of 3-PARTITION, each I_i must then contain exactly three jobs, and so $I_1, I_2, \ldots, I_h$ is a satisfying solution to the 3-PARTITION-instance. This completes the proof. □

Theorem 1 applies for the classic total-completion-time objective without processing cost, and so including a processing-cost component in the objective, as was the case in the initial reference [1], will not make the problem easier:

Corollary 1. *TBS to minimize $\sum C_j + \alpha(b)$ is strongly NP-hard when m is free.*

In detail, one can simply add $\alpha(b) = 0$ to the proof of Theorem 1.

Tang and Gong [1] study TBS with objective $\sum C_j + \alpha(b)$ and show that it is (ordinarily) NP-hard for $m = 2$. Also here we can remove the processing cost component $\alpha(b)$ and maintain the complexity status. This can be shown via reduction from (2-)PARTITION, analogously to the proof of Theorem 1 in [1], requiring only minor changes in the parameter choices, namely setting the processing cost $\alpha(b) = 0$ and the threshold value $y = 3ah$. Due to the number of jobs $n = 2h$ and the machine capacity $c = h$, the number of batches $q \geq 2$. The minimal total completion time is $3ah = y$ when $q = 2$ and otherwise it exceeds y, so that $q = 2$. The remainder of the proof can proceed identically to [1], leading to the following result:

Proposition 1. *TBS to minimize total completion times is NP-hard even if $m = 2$.*

We find this format for this negative result important enough to be stated in its own right because it pertains to a "classic" objective function, without processing cost.

3. Makespan

We obtain the following straightforward negative result for the case where the number of transporters is not fixed.

Theorem 2. *TBS with objective $C_{\max}$ is strongly NP-hard when m is free.*

Proof: Consider problem $P||C_{\max}$, the classic parallel machine scheduling problem with makespan objective and free number of machines, which is strongly NP-hard [7]. In this problem, we denote the machine set by $\{1, \ldots, m'\}$ and the job set by $\{1, \ldots, n'\}$, where each job j has a processing time p_j. For an arbitrary instance of $P||C_{\max}$ we construct a TBS-instance, as follows: number of jobs $n = n'$, number of vehicles $m = m'$, transportation time $t_j = p_j$, for $j = 1, \ldots, n$, processing time $p = 0$, return time $t = 0$, machine capacity $c = 1$. Clearly, a minimum-makespan TBS-schedule will also yield an optimal schedule for $P||C_{\max}$, and vice versa. □

Note that the 'trick' of setting p equal to zero in the proof could not be used in Theorem 1 because $P||\sum C_j$ is polynomially solvable, via a variant of the SPT (shortest processing time) rule [8].

Corollary 2. *Problem TBS to minimize $C_{\max} + \alpha(b)$ is strongly NP-hard when m is free.*

Zhu [2] considers the problem TBS to minimize the sum of makespan and processing cost and proves that it is NP-hard via reduction from (2-)PARTITION for $m = 2$. Applying similar modifications as in the proof of Theorem 2, it is easy to check that TBS with makespan objective only when $m = 2$ is equivalent to $P_2||C_{\max}$, which is NP-hard: set $m = m' = 2$. We formalize this result as follows:

Proposition 2. *Problem TBS to minimize $C_{\max}$ is NP-hard even if $m = 2$.*

Tang et al. [3] consider the problem TBS with deterioration considerations. In this model, they define the exposure time of a job as the length of the time interval from the departure from the holding area to the start of the processing on the machine. A job is said to be a 'hot' job if its exposure time does not exceed a threshold value E, otherwise it is called a 'cold' job. A batch that contains only hot jobs has processing time p_h, otherwise the processing time is $p_c > p_h$. It is shown in [3] that TBS with deterioration to minimize $C_{\max} + \alpha(b)$ when $m = 1$ is strongly NP-hard. Below, we establish the complexity of this problem with makespan objective only ($\alpha(b)$ is constant).

Theorem 3. *TBS with deterioration to minimize $C_{\max}$ is strongly NP-hard when $m = 1$.*

Proof: For a given instance of 3-PARTITION we construct an instance of TBS, as follows:

number of jobs $n = 3h$;

number of vehicles: $m = 1$;

transportation time $t_j = a_j$ for $j = 1, \ldots, 3h$;

processing time $p_h = a, p_c = (h + 1)a$;

return time $t = 0$;

exposure threshold value $E = a$;

machine capacity $c = 3$;

threshold value $y = (h+1)a$.

We prove that there exists a schedule for this TBS-instance with $C_{\max} \leq y$ if and only if the 3-PARTITION-instance is a yes-instance.

$\Leftarrow$ Suppose that $I_1, I_2, \ldots, I_h$ is a solution to 3-PARTITION. We construct a schedule for the TBS-instance as follows: the vehicle consecutively transports the jobs of each set $I_1, I_2, \ldots, I_h$, one by one. The batching machine can process all jobs in I_i as the i-th batch from time a onwards. The makespan is then $(h+1)a = y$.

$\Rightarrow$ Suppose conversely that there is a schedule satisfying $C_{\max} \leq y$, containing q batches. Since $n = 3h$ and $c = 3$, we have $q \geq h$. Let S_1 denote the starting time of the first batch; we have $S_1 + qa \leq y = (h+1)a$. Knowing that $S_1 \geq 0$, we obtain that either $q = h$ or $q = h+1$. If $q = h+1$ then $S_1 = 0$, which is impossible because each $t_j = a_j > 0$, and therefore $q = h$. Let I_i denote the set of jobs that are processed in the i-th batch of the schedule $(i = 1, 2, \ldots, h)$. We can see that I_i contains exactly three jobs. Denote the total transportation time $\sum_{j \in I_i} t_j$ of batch i as x_i for $i = 1, 2, \ldots, h$. Due to $p_c = (h+1)a$, we have $x_i \leq E = a$. Since $\sum_{i=1}^{h} x_i = ha$, x_i must be exactly a for $i = 1, \ldots, h$. Therefore, $I_1, I_2, \ldots, I_h$ is a satisfying solution to the 3-PARTITION-instance. This completes the proof. □

4. Weighted completion times

In practical operational scheduling environments, the jobs to be processed often have different weights. The weight w_j of job $j \in N$ is basically a priority factor, denoting the importance of job j relative to the other jobs in the system [8]. This weight can for instance represent the actual cost of keeping a job in the system, or the importance of early processing because the job is part of a larger order, or a measure of the importance of the client who placed the corresponding order. The following theorem establishes the computational complexity of TBS for minimizing the sum of weighted completion times and processing cost with a single transporter.

Theorem 4. *TBS with objective $\sum w_j C_j + \alpha(b)$ is strongly NP-hard even if $m = 1$.*

Proof: Consider an arbitrary instance of 3-PARTITION, for which we construct a TBS-instance in the following manner:

number of jobs $n = 3h$;

only one vehicle: $m = 1$;

transportation time $t_j = a_j$ for $j = 1, \ldots, 3h$;

weight $w_j = a_j$, $j = 1, \ldots, 3h$;

processing time $p = a$;

return time $t = 0$;

processing cost $\alpha(b) = \frac{3h+h^2}{2} a^2 b$;

machine capacity $c = 3$;

threshold value $y = \frac{(h+1)(3h+h^2)}{2} a^2$.

The time it takes to construct the instance is obviously polynomial. We prove that there exists a schedule for this TBS-instance with $\sum w_j C_j + \alpha(b) \leq y$ if and only if the answer is 'yes' for the 3-PARTITION-instance.

$\Leftarrow$ Suppose that $I_1, I_2, \ldots, I_h$ is a solution to 3-PARTITION-instance. We construct a schedule for TBS as follows: the vehicle consecutively transports the jobs of each set $I_1, I_2, \ldots, I_h$, one by one. The total transportation time of each set I_i is $\sum_{j \in I_i} t_j = \sum_{j \in I_i} a_j = a$. Since $p = a$ and $c = 3$, the batching machine can process the jobs I_i as the i-th batch at time ia. Then

$$\begin{aligned}
\sum_{j=1}^{3h} w_j C_j &= \sum_{j \in I_1} w_j \cdot (a + \sum_{l \in I_1} t_l) + \sum_{j \in I_2} w_j \cdot (a + \sum_{l \in I_1 \bigcup I_2} t_l) + \ldots \\
&\quad + \sum_{j \in I_h} w_j \cdot (a + \sum_{l \in I_1 \bigcup I_2 \bigcup \ldots \bigcup I_h} t_l) \\
&= a \cdot 2a + a \cdot 3a + \ldots + a \cdot (h+1)a \\
&= \frac{3h + h^2}{2} a^2.
\end{aligned}$$

We also obtain that $\alpha(b) = \frac{h(3h+h^2)}{2} a^2$, which leads to $\sum w_j C_j + \alpha(b) = \frac{(h+1)(3h+h^2)}{2} a^2 = y$.

$\Rightarrow$ Suppose conversely that there is a schedule satisfying $\sum w_j C_j + \alpha(b) \leq y$, containing q batches. Since $n = 3h$ and $c = 3$, we have $q \geq h$. If $q > h$, then $q \geq h + 1$ and the processing cost alone will be no less than

$\frac{(h+1)(3h+h^2)}{2}a^2 = y$, so the total objective value will be higher than y. Consequently, there are exactly h batches in the schedule. This implies the following upper bound on the total weighted completion time:

$$\sum w_j C_j \leq \frac{3h+h^2}{2}a^2. \tag{1}$$

Let I_i denote the set of jobs that are processed in the i-th batch of the schedule ($i = 1, 2, \ldots, h$). We see that I_i contains exactly three jobs. Denote the total transportation time $\sum_{j \in I_i} t_j$ of batch i as x_i for $i = 1, 2, \ldots, h$. We have the following inequalities:

$$\begin{aligned}
\sum_{j=1}^{3h} w_j C_j &\geq x_1(x_1+a) + x_2(x_1+x_2+a) + \ldots + x_h(x_1+x_2+\ldots+x_h+a) \\
&= (x_1^2 + x_2^2 + \ldots + x_h^2) + a(x_1+x_2+\ldots+x_h) \\
&\quad +(x_2x_1 + x_3x_1 + x_3x_2 + \ldots + x_hx_{h-1}) \\
&= \frac{1}{2}\sum_{i=1}^{h} x_i^2 + a(\sum_{i=1}^{h} x_i) + \frac{1}{2}(\sum_{i=1}^{h} x_i)^2 \\
&= \frac{1}{2}\sum_{i=1}^{h} x_i^2 + \frac{2h+h^2}{2}a^2. \qquad (2)
\end{aligned}$$

According to the Cauchy-Schwarz inequality,

$$\left(\sum_{i=1}^{h} x_i^2\right)\left(\sum_{i=1}^{h} 1^2\right) \geq \left(\sum_{i=1}^{h} x_i \cdot 1\right)^2 = (ha)^2,$$

so we have $\sum_{i=1}^{h} x_i^2 \geq ha^2$, where equality holds if and only if $x_1 = x_2 = \ldots = x_h = a$. Combining with (1) and (2), the value $\sum_{j=1}^{3h} w_j C_j$ is exactly $\frac{3h+h^2}{2}a^2$. Therefore, the overall objective value of the schedule is $\frac{(h+1)(3h+h^2)}{2}a^2 = y$ and $x_1 = x_2 = \ldots = x_h = a$.

Since $\sum_{j \in I_i} t_j = x_i = a$ and $|I_i| = 3$ for $i = 1, 2, \ldots, h$, the sets $I_1, I_2, \ldots, I_h$ form a solution for 3-PARTITION, which completes this proof. □

The foregoing result should be contrasted with the unweighted case (all $w_j = 1$), for which Tang and Gong [1] show that it can be solved in polynomial time. This negative result is somewhat less satisfactory than those in the previous sections, however, because we still need the processing cost in the objective; we have not immediately found a suitable reduction for the case with $\alpha(b) = 0$.

5. Link between TBS and the serial batching problem

In this section, we investigate the link between TBS and the serial batching problem $1|s\text{-batch}|\sum w_jC_j$, which pertains to a single machine that processes jobs in batches. The jobs in one batch are serially processed, their completion time is defined to be equal to the finishing time of the last job in the batch, and the processing time of a batch is equal to the sum of the processing times of all the jobs in the batch. The number of jobs in a batch is called the batch size, which is an arbitrary number. There is a setup time for the production of a batch.

In the previous sections, TBS was associated with a limited batch size c on the batching machine. We now find that

Theorem 5. *TBS with unlimited batch size and with objective $\sum w_jC_j$ is strongly NP-hard even if $m = 1$.*

Proof: We use a reduction from $1|s\text{-batch}|\sum w_jC_j$, which is known to be NP-hard in the strong sense [9]. Consider an instance of serial batching with a set of jobs $N = 1, 2, \ldots, n'$, processing time p_j, weight w'_j for each job $j \in N$. We now construct an instance of TBS as follows:

number of jobs $n = n'$;

only one vehicle $m = 1$;

transportation time $t_j = p_j$ for $j = 1, \ldots, n$;

weight $w_j = w'_j$, $j = 1, \ldots, n$;

processing time $p = s$;

return time $t = 0$;

machine capacity $c = n$;

It is easy to check that the weighted sum of completion times is the same for the two instances. □

We can also recognize that TBS with objective $\sum w_jC_j$ and with the additional constraint that each batch must contain exactly three jobs is equivalent to the serial batch scheduling problem $1|3\text{-in-}1, s\text{-batch}|\sum w_jC_j$ [10], in which each batch also consists of exactly three jobs. Problem $1|3\text{-in-}1, s\text{-batch}|\sum w_jC_j$ is strongly NP-hard [10]. Consider now an instance of this serial batching problem with n jobs, processing times p_j and job weights w_j. We construct an equivalent TBS-instance with n jobs, $m = 1$ vehicle, travel time $t_j = p_j$, return time $t = 0$, batch processing time $p = 0$, machine capacity $c = 3$ and with the constraint that each batch must contain exactly three jobs. It can be easily verified that the weighted sum of completion times is the same for the two instances. Thus, we arrive at the following result.

Proposition 3. *When each batch must contain exactly three jobs, TBS to minimize weighted completion times is strongly NP-hard even if* $m = 1$.

6. Conclusion and final remarks

In this paper, we have investigated a number of variants of a coordinated scheduling problem of production and transportation, in line with the initial paper by Tang and Gong [1] in 2009, and which was also studied in a number of follow-up papers. We show that the earlier complexity results are still valid when the processing cost is removed from the objective, thus reducing to more "classic" scheduling objectives. We also assess the complexity status of the relevant problem variants with free number of transporters, we establish that the weighted-completion-time objective leads to an intractable problem even with a single transporter, and we recognize a link with the serial batching problem.

A number of open problems remain, with the most prominent one being the complexity status of the scheduling problem with weighted completion times, one vehicle and without processing cost. For future work, it might be interesting also to investigate this scheduling environment with other common scheduling objectives, for instance relating to due-date performance. From a more practical perspective, bringing the stylized problem statement closer to practical applications, for instance by incorporating deterioration effects, has already been undertaken in some recent articles and might be pursued further, dependent on the needs of individual industrial partners.

Acknowledgements

This work is partly supported by the Natural Science Foundation of China (Grant No. 71071008 and Grant No. 91224007), China Scholarship Council and the Innovation Foundation of BUAA for PhD Graduates.

References

[1] L. X. Tang, H. Gong, The coordination of transportation and batching scheduling, Applied Mathematical Modelling 33 (10) (2009) 3854–3862.

[2] H. L. Zhu, A two stage scheduling with transportation and batching, Information Processing Letters 112 (2012) 728–731.

[3] L. X. Tang, H. Gong, J. Y. Liu, F. Li, Bicriteria scheduling on a single batching machine with job transportation and deterioration considerations, Naval Research Logistics 61 (2014) 269–285.

[4] L. X. Tang, H. Gong, A hybrid two-stage transportation and batch scheduling problem, Applied Mathematical Modelling 32 (2008) 2467–2479.

[5] L. X. Tang, P. Liu, Two-machine flowshop scheduling problems involving a batching machine with transportation or deterioration consideration, Applied Mathematical Modelling 33 (2009) 1187–1199.

[6] L. X. Tang, P. Liu, Flowshop scheduling problems with transportation or deterioration between the batching and single machines, Computers and Industrial Engineering 56 (2009) 1289–1295.

[7] M. R. Garey, D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman, 1979.

[8] M. L. Pinedo, Scheduling: Theory, Algorithms, and Systems, Springer, New York, 2008.

[9] S. Albers, P. Brucker, The complexity of one-machine batching problems, Discrete Applied Mathematics 47 (1993) 87–107.

[10] J. J. Yuan, Y. X. Lin, T. C. E. Cheng, C. T. Ng, Single machine serial-batching scheduling problem with a common batch size to minimize total weighted completion time, International Journal of Production Economics 105 (2007) 402–406.

FACULTY OF ECONOMICS AND BUSINESS
Naamsestraat 69 bus 3500
3000 LEUVEN, BELGIË
tel. + 32 16 32 66 12
fax + 32 16 32 67 91
info@econ.kuleuven.be
www.econ.kuleuven.be