Coordinated scheduling on single serial batching machine with transportation considerations

Cunchang Gu[1, 2*], Xiaoyan Xu[2]

[1]*School of Management, Qufu Normal University, Rizhao 276826, China*

[2]*College of Science, Henan Universities of Technology, Zhengzhou 450001, China*

Received 15 May 2014, www.cmnt.lv

Abstract

The coordination of production scheduling and transportation has recently received a lot of attention in logistics and supply chain management. We study a coordinated scheduling problem, in which each job is transported to a single serial batching machine for further processing, each batch to be processed occurs a processing cost, and the objective is minimizing the sum of the makespan and the total processing cost. Under the condition of the jobs' processing times are equal, if the job assignment to the vehicles is predetermined, we provide a polynomial time dynamic programming algorithm, for the general problem, we prove it is NP-hard. When the returning time of vehicle is zero, we present the approximation algorithm and prove that the worst case ratio of the algorithm is not greater than $2-\frac{1}{m}$, and the bound is tight.

Keywords: supply chain scheduling, dynamic programming algorithm, complexity, worst case analysis

1 Introduction

The classical scheduling problems usually assume that there is an infinite number of facilities for processing jobs, and the transportation between the warehouse and the facility can be done instantaneously. As a result, the job delivery and the machine scheduling are separately considered without effective coordination between the two. Coordination between job delivery and machine scheduling becomes more practical.

The coordination of production scheduling and transportation has recently received a lot of attention in logistics and supply chain management. Semi-finished jobs are transported from a holding area to a manufacturing facility for further processing by transporters in many manufacturing systems. Another motivation arises in many industries where the coordination of production and transportation can help to save energy and reduce fuel consumption. This is particularly true in the iron and steel industry.

In this paper, motivated by applications in the iron and steel industry, we study a coordinated scheduling problem of transportation and production. The jobs located at a holding area need to be transported by m vehicles to a serial batching machine for further processing. Each vehicle can transported one job at a time, and the serial batching machine can process several jobs at a time. Each batch of jobs to be processed on the serial batching machine occurs a processing cost; assume that batch processing cost is proportional to the batch number. The problem is to find a joint schedule of transportation and production such that the objective is to minimize the sum of makespan and total processing cost.

In the last decade, the coordination of transportation and scheduling has become one of the most important topics in production and operations management research. Chungyee L, Zhilong C [1] considers two types of transportation. The first type is intermediate transportation in a flow shop where jobs are transported from one machine to another for further processing. The second type is the transportation necessary to deliver finished jobs to the customer.

The batching machine scheduling is an important research topic. Recent reviews of batch scheduling research are provided by Potts C N and Kovalyov M Y [2] and Brucker P et al. [3]. The scheduling problems on the batching machine can be divided into two categories of parallel batch and serial batch according to batch processing pattern. In parallel batching scheduling problems, many jobs can be processed on the machines at any time; the processing time of batch is equal to the maximum of all jobs' processing times of this batch. Yuzhong Z, et al. [4] consider single parallel machine scheduling problem which the jobs have the sizes and only two different arrival time, present the approximation algorithm that the worst case ratio is not greater than $\frac{33}{14}$ when the processing times and the sizes are agreeable.

In serial batching scheduling problems, only one job can be processed on the machines at any time, and the

* ***Corresponding author*** e-mail gucunchang@163.com

jobs are processed one by one simultaneously, the processing time of batch is equal to the setup time plus the total processing times of the jobs in this batch. Coffman E G et al. [5] study minimizing the total completion times scheduling problem on single serial batch machine. Liji S et al. [6] consider the serial batch scheduling problem embedded in a job shop environment to minimize makespan, propose a tabu search algorithm which consists of various neighbourhood functions, multiple tabu lists and a sophisticated diversification structure. Glass C A et al. [7] study a problem of scheduling and batching on two machines in a flow-shop and open-shop environment, the aim is to make batching and sequencing decisions, which specify a partition of the jobs into batches on each machine, and a processing order of the batches on each machine, respectively, so that the makespan is minimized. Webster S and Baker K R [8] consider scheduling groups of jobs on a single machine on three basic models known as family scheduling with item availability, family scheduling with batch availability, and batch processing. Baptiste P [9] analyses the single parallel batching and serial batching scheduling problems that all jobs are arrived at time 0, present the dynamic programming algorithms respectively. But, the attention is relatively less for the coordination of transportation and serial batching scheduling at present.

In this paper, we investigate the problem of coordinated scheduling on single serial batching machine with transportation. At time 0, the n jobs $J_1, J_2, \ldots, J_n$ located at a holding area are transported by the m vehicles to a serial batching machine for further processing, the transportation times are $t_1, t_2, \ldots, t_n$, respectively. In the transportation stage, each vehicle can deliver only one job at a time, the transportation time of one job transported by different vehicles is identical, and the empty moving time of each vehicle from the batching machine back to the holding area is t. In the production stage, the processing time of the job J_j is p_j, $j = 1, 2 \ldots, n$, the serial batching machine can process several jobs simultaneously as a batch, the maximum number of jobs that can be processed simultaneously in the serial batching machine is called capacity c of that batching machine, the completion time C_j of any job J_j in each batch is the fixed set-up time s plus the total processing times of the jobs in this batch. Once processing of a batch is initiated, it cannot be interrupted and other jobs cannot be introduced into the machine until processing is completed. Each batch to be processed on the batching machine occurs a processing cost, b is the number of batches to be processed on the batching machine, $\alpha(b)$ is the linear non-decreasing function of b, and denote the processing cost function. The objective is minimizing the sum of the makespan and the total processing cost $C_{\max} + \alpha(b)$. This problem is denoted as $D \to 1 \mid m \geq 1, c \geq 1, s - batch \mid C_{\max} + \alpha(b)$.

The paper is organized as follows. In Section 2, we give some preliminaries and a useful lemma. Section 3 studies a special case, and provides a polynomial time dynamic programming algorithm. In Section 4, we study the complexity of the problem; propose the TSPT-DP algorithm to solve the problem. Finally, some conclusions are made in Section 5.

2 Preliminaries

Let π and π^* be a feasible solution and an optimal solution, respectively. We will use the following notations and definitions frequently in the reminder of this paper.

$b^\pi(b^*)$: the total number of batches in $\pi(\pi^*)$.

$B_l^\pi(B_l^*)$: the l-th processing batch in $\pi(\pi^*)$, $l = 1, 2, \ldots, b^\pi(b^*)$.

$n_l^\pi(n_l^*)$: the number of jobs processed up to the l-th batch, clearly, $n_{k^\pi}^\pi = n_{k^*}^* = n$.

$|B_l^\pi|(|B_l^*|)$: the number of jobs processed in $B_l^\pi(B_l^*)$, clearly, $|B_l^\pi| = n_l^\pi - n_{l-1}^\pi (|B_l^*| = n_l^* - n_{l-1}^*)$.

$r_j^\pi(r_j^*)$: the arrival time of the job J_j, i.e., the time when J_j arrives at the serial batching machine.

$r_{[j]}^\pi(r_{[j]}^*)$: the arrival time of the j-th arrived job.

$S_l^\pi(S_l^*)$: the starting time of the batch $B_l^\pi(B_l^*)$.

$C_j^\pi(C_j^*)$: the completion time of the job J_j.

$C^\pi(l)(C^*(l))$: the completion time of the batch $B_l^\pi(B_l^*)$, clearly, $C_j^\pi = C^\pi(l)(C_j^* = C^*(l))$, for $J_j \in B_l^\pi(B_l^*)$.

The notations can be abbreviated to $b, n_l, r_j, r_{[j]}, B_l$, $|B_l|$, et al. Let $n_0 = 0$, then, $C(i) = S_i + s + \sum_{j \in B_i} p_j$, $C_{\max} = C(b) = S_b + s + \sum_{j \in B_i} p_j$. In situation where the dimension on the measurements $C_{\max}$ and $\alpha(b)$ is difficult to unify, we may adjust cost function $\alpha(b)$ to uniform dimension with the maximum completion time. For ease of presentation, denote our problem as TSBSM.

Although the problem considered in this paper is different from that in Hall N G and Potts C N [10], some properties can be found in the same way.

Lemma 1. For problem $D \to 1 \mid m \geq 1, c \geq 1, s - batch,$ $p_j = p \mid C_{\max} + \alpha(b)$, there exists an optimal solution π^* in which:

(i) There is no idle time between the jobs transported on each vehicle in the transportation part.

(ii) All jobs assigned to the same vehicle are scheduled in the non-decreasing order of transportation times.

(iii) The starting time of each batch on the serial batching machine is made either an arrival time of some job on the machine or immediately at a time when the machine becomes available.

(iv) earlier arrived jobs are processed no later than those arrived later.

Usually, the quality of an approximation algorithm (denoted by H) is measured by the worst case ratio of the algorithm, which is defined as the smallest number ρ such that $F^H \le \rho F^*$ for all instances, where F^H and F^* denote the objectives of solution produced by H and the optimal algorithm, respectively.

3 A polynomial time algorithm for a special case

In this section, we consider a special case where the job assignment to the vehicles is predetermined. It is evident that the problem reduces to an optimal serial batching problem in this case. This special case characterizes the practical situation where the arrival times of the jobs are known. Now, we can provide a dynamic programming algorithm to solve the optimal serial batching problem in a polynomial time as follow.

Re-index all jobs $J_j, j=1,2,\dots,n$, in accordance with the job arrival time on the serial batching machine, i.e., $r_1 \le r_2 \le \dots \le r_n$, the processing time p_j of any job J_j on the serial batching machine is p. It suffices to consider one job sequence and apply it to the processing of jobs on the serial machine. So the starting time of each batch on the machine need to be decided, and this can be done by dynamic programming.

By Lemma 1(iii), the starting time S_l of the batch B_l on the serial batching machine is either the completion time $C(l-1)$ of the batch B_{l-1} or the arrival time r_{n_l} of the last arrived job J_{n_l} in this batch. In the first case, $S_l = C(l-1) = S_{l-1} + s + |B_{l-1}|p$, let B_j denote the earliest processing batch which the jobs are processed consecutively until the batch B_l, the starting time of the batch B_j must be the arrival time r_{n_j} of the last arrived job J_{n_j} in this batch, so,

$$S_l = r_{n_j} + (l-j)s + (n_l - n_{j-1})p,$$
$$= r_{n_j} + (l-j)s + (|B_j| + \dots + |B_{l-1}|)p. \quad (1)$$

Denote $n_l - n_{j-1} = q_l$, satisfy $q_l + j \le n$. In the second case, $S_l = r_{n_l}$.

Hence, the possible starting time of the bath B_l on the serial batching machine can be $r_j, r_j + s + xp, \dots, r_j + zs + yp$, where $\lceil n/c \rceil \le z \le l-1 \le n,\ x \le y \le n-j$, and $S_0 = 0$.

Define $f(k, j, S_l)$ as the minimal makespan to schedule the first k jobs $J_1, J_2, \dots, J_k$, provided that the current last batch B_l contains jobs $J_j, J_{j+1}, \dots, J_k$, and starts to be processed at time S_l , where $k-j+1 \le c,\ S_l \ge r_k$. If we know the available time $f(j-1, i, S_{l-1})$ of the batching machine before process jobs $J_j, J_{j+1}, \dots, J_k$, then the starting time of the batch B_l is actually fixed, i.e., $S_l = \max\{f(j-1, i, S_{l-1}), r_k\}$.

$f(k, j, S_l)$ satisfies the following three properties:

(i) $0 < k-j+1 \le c$;

(ii) $S_l = r_j, r_j + s + xp, \dots, r_j + zs + yp$, and $S_l - S_{l-1} \ge s + p, l = 2,3,\dots,b$, where $x \le y \le n$, $\lceil n/c \rceil \le z \le l-1$;

(iii) $\lceil k/c \rceil \le l \le k$.

Otherwise, $f(k, j, S_l) = \infty$.

Dynamic Programming Algorithm *DP*

Initial condition:

$f(0,0,0) = 0$;

Recursive relation:

$f(k, j, S_l) = \min\{\max\{f(j-1, i, S_{l-1}), r_k\} + (s + (k-j+1)p) \mid$ all possible states (i, S_{l-1}) },where $j-1, i, S_{l-1}$ satisfy the three condition described above.

Optimal solution:

$F(n) = \min\{f(n, j, S_b) + \alpha(b) \mid$ all possible states $(j, S_b)\}$.

By recording all the necessary information in the above process, an optimal schedule can be calculated. It is not difficult to see that the time complexity of the algorithm DP is $O(cn^4)$. The following theorem can be obtained.

Theorem 1. Algorithm DP finds an optimal schedule for problem $D \to 1 | m \ge 1, c \ge 1, s-batch, p_j = p | C_{\max} +$ $\alpha(b)$ in $O(cn^4)$ time when the job assignment to the vehicles is predetermined.

We now demonstrate the above solution method with a numerical example.

Example. Job set $J = \{J_1, J_2, J_3, J_4\}$, $m = 2, t = 1, c = 3,$ $t_1 = 1, t_2 = 4, t_3 = 2, t_4 = 1, s = 1, p = 2, \alpha(b) = 5b,$ assume that J_1 and J_2 are transported by one vehicle, J_3 and J_4 are transported by another vehicle. From the above method, we know $r_1 = 1, r_2 = 2, r_3 = 4, r_4 = 6$.

We have the following results by the dynamic programming algorithm DP :

$f(1,1,S_1) = \max\{f(0,0,0), r_1\} + (s+p) = 4$;

$f(2,1,S_1) = \max\{f(0,0,0), r_2\} + (s+2p) = 7$;

$f(2,2,S_2) = \max\{f(1,1,S_1), r_2\} + (s+p) = 7$;

$f(3,1,S_1) = \max\{f(0,0,0), r_3\} + (s+3p) = 11$;

$f(3,2,S_2) = \max\{f(1,1,S_1), r_3\} + (s+2p) = 9$;

$f(3,3,S_2) = \max\{f(2,1,S_1), r_3\} + (s+p) = 10$;
$f(3,3,S_3) = \max\{f(2,2,S_2), r_3\} + (s+p) = 10$;
$f(4,2,S_2) = \max\{f(1,1,S_1), r_4\} + (s+3p) = 13$;
$f(4,3,S_2) = \max\{f(2,1,S_1), r_4\} + (s+2p) = 12$;
$f(4,4,S_2) = \max\{f(3,1,S_1), r_4\} + (s+p) = 14$;
$f(4,3,S_3) = \max\{f(2,2,S_2), r_4\} + (s+2p) = 12$;

$$f(4,4,S_3) = \min\{\max\{f(3,2,S_2), r_4\} + (s+p) = 12, \max\{f(3,3,S_2), r_4\} + (s+p) = 13\} = 12;$$

$f(4,4,S_4) = \max\{f(3,3,S_3), r_4\} + (s+p) = 13$;

$$F(4) = \min\{f(4,j,S_b) + \alpha(b) \mid \text{all possible states } (j,S_b)\} = f(4,3,S_2) + \alpha(2) = 22.$$

The optimal schedule on the serial batching machine is finally found as $\pi^* = \{\{J_1,J_3\},\{J_4,J_2\}\}$ with the optimal objective value of 22.

4 Complexity and approximation algorithm

In this section, we firstly show that for TSBSM is NP-hard by reducing the 2-Partition problem to the decision version of this problem when the job assignment to the vehicles is not predetermined, secondly propose an approximation algorithm for the problem TSBSM.

4.1 COMPLEXITY

2-Partition problem: Given $h+1$ positive integers a_j, $j=1,2,\dots,h$, and a, such that $\sum_{i=1}^{h} a_j = 2a$. The question asks if the set $H=\{1,2,\dots,h\}$ can be divided into two disjoint subsets G_1 and G_2, such that $G_1+G_2=H$, $\sum_{j\in G_1} a_j = \sum_{j\in G_2} a_j = a$.

Theorem 2. The problem $D \to 1 \mid m\geq 1, c\geq 1, s-batch, p_j = p \mid C_{\max}+\alpha(b)$ is NP-hard even if $m=2$.

Proof. To any instance of the 2-Partition problem, we construct an instance of TSBSM as follow. There are $n=2h$ jobs split into two job sets: the J'-jobs (partition jobs) denoted by J_j', $j=1,2,\dots,h$, the J''-jobs (auxiliary jobs) denoted by J_j'', $j=1,2,\dots,h$. Their transportation times and other parameters are given as follows:

Transportation times: $t_{J_j'} = 2a_j, t_{J_j''} = 0, j=1,2,\dots,h$;

Returning time: $t=0$;

Set-up time of the serial batch: $s=a$;

Processing time: $p_j = \frac{a}{h}, j=1,2,\dots,2h$;

Processing cost: $\alpha(b)=3ba$;

Machine capacity: $c=h$;

Threshold value: $y=10a$.

We are going to show that for the constructed scheduling problem instance, a schedule π with $C_{\max}+\alpha(b)\leq y$ exists if and only if the 2-Partition problem has a solution.

If there is a solution to the 2-Partition problem instance, we show that there is a schedule π with $C_{\max}+\alpha(b)\leq y$ for the scheduling instance. Suppose that the 2-Partition problem instance has a solution G_1 and G_2. Now construct the following schedule π:

Since the transportation times of J''-jobs and the returning time of the vehicles are equal to 0, J''-jobs are transported to the serial batching machine, and two vehicles are available at time 0, the machine can first process these jobs. Due to $c=h$, the J''-jobs are processed as the first batch at time 0, and the completion time $C(1)$ of this batch is equal to $S_1+s+\sum_{j\in J''} p_j = 0+a+h\frac{a}{h} = 2a$. Next, vehicle 1 transports the jobs of G_1 one by one, and vehicle 2 transports the jobs of G_2. Let T_i denotes the total running time of vehicle i, for $i=1,2$, we can see that $T_1 = \sum_{j\in G_1} t_{J_j'} = 2a, T_2 = \sum_{j\in G_2} t_{J_j'} = 2a$. The J'-jobs are processed as the second batch at time $2a$, and the completion time $C(2)$ of this batch is equal to $S_2+s+\sum_{j\in J'} p_j = 2a+a+h\frac{a}{h} = 4a$. It is easy to check that $C_{\max}+\alpha(b)\leq y$. (See Figure 1)

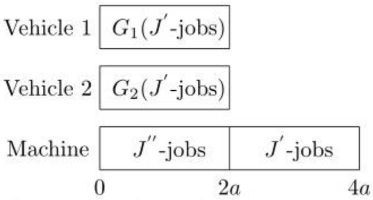


Figure 1. A schedule for the instance of problem TSBSM

If there exists a schedule π with $C_{\max}+\alpha(b)\leq y$ to the instance of TSBSM, we show that the 2-Partition problem has a solution.

First, the following properties hold in schedule π:

(i) Schedule π exactly contains two batches, i.e., each batch contains exactly h jobs.

(ii) All J''-jobs are processed as the first batch at time 0.

(iii) All J'-jobs are processed as the second batch at time $2a$.

Next, we prove three properties as follows:

(i) Suppose that there are b batches in schedule π. Due to $c=h, \alpha(b)=3ba$, we have $2\leq b\leq 3$. If $b=3$, the sum of the makespan and the processing cost

$C_{\max}+\alpha(b)$ is more than $s+\sum_{j\in J''\cup J'}p_j+9a=12a$, which is a contradiction.

Hence, schedule π exactly contains two batches and each batch contains h jobs.

(ii) Denote S_1 and S_2 as the starting time of the first batch and the second batch on the serial batching machine, respectively. Note that $S_1+s+h\dfrac{a}{h}=S_1+2a\le S_2$. Since the processing time of each batch on the machine is $2a$, we have $C_{\max}+\alpha(b)=S_2+2a+6a\le 10a$. Hence, we obtain that $S_1=0$ and $S_2=2a$. At time 0, there are only J''-jobs available. Thus, all J''-jobs are processed as the first batch at time 0.

(iii) From (i) and (ii), we know that all J'-jobs are processed as the second batch at time $2a$.

Let G_1 and G_2 be a partition of J'-jobs. We assume that vehicle 1 transports the jobs of G_1 one by one, and vehicle 2 transports the jobs of G_2 one by one. Let T_i denotes the total running time of vehicle i, for $i=1,2$. Based on the above discussion, the starting time of the second batch satisfies $S_2=2a\ge\max\{T_1,T_2,2a\}$, where $T_1=\sum_{j\in G_1}2a_j\le 2a, T_2=\sum_{j\in G_2}2a_j\le 2a$. Due to $T_1+T_2=\sum_{j\in H}2a_j=4a$, we have $\sum_{j\in G_1}a_j=\sum_{j\in G_2}a_j=a$. Then it is easy to see that G_1 and G_2 form a solution to the 2-Partition problem instance. Theorem 2 follows.

4.2 APPROXIMATION ALGORITHM

The proof of theorem 2 indicates the problem $D\to 1|m\ge 1,c\ge 1,s-batch,p_j=p|C_{\max}+\alpha(b)$ is NP-hard even if $t=0$. When the returning time of vehicle is 0, we present the following approximation algorithm and analysis its worst case ratio.

TSPT-DP Algorithm

Step 1: In the transportation part, assign jobs to vehicles by the SPT rule of the transportation times, i.e., always assign the shortest transportation time job to the available vehicle ;

Step 2: Renumber the jobs in the non-decreasing arrival time order, i.e., $r_1\le r_2\le\cdots\le r_n$;

Step 3: In the processing part, use the dynamic programming algorithm *DP* for batching and processing jobs.

Let π be the solution generated by the TSPT-DP algorithm for the problem $D\to 1|m\ge 1,t=0,c\ge 1,$ $s-batch,p_j=p|C_{\max}+\alpha(b)$. We denote by π' the solution generated by assigning jobs to vehicles by the SPT rule of the transportation times, and processing the same number of jobs in each batch coincide with that of the optimal solution π^* as early as possible (maybe wait).

Therefore, π' transports jobs the same as π and uses the same batching policy as π^*. Consequently, we have $r_j^{\pi'}=r_j^{\pi}, j=1,2,\ldots,n$, and $n_i^{\pi'}=n_i^*, 1\le i\le b^*=b^{\pi'}$, $\alpha(b^*)=\alpha(b^{\pi'})$. By Theorem 1,

$$C_{\max}^{\pi}+\alpha(b^{\pi})\le C_{\max}^{\pi'}+\alpha(b^{\pi'}). \quad (2)$$

Lemma 2. $r_j^{\pi'}\le(2-\dfrac{1}{m})r_{[j]}^*, 1\le j\le n$.

Proof. Since $r_{[j]}^*$ is the arrival time of the j-th arrived job in π^*, it follows $r_{[1]}^*\le r_{[2]}^*\le\cdots\le r_{[n]}^*$, and $t_1,t_2,\ldots,t_j$ are the j minimum transportation times. Thus, we have

$$r_{[j]}^*\ge\max\{t_{[1]},t_{[2]},\cdots,t_{[j]},\frac{1}{m}\sum_{l=1}^{j}t_{[l]}\}\ge\max\{t_j,\frac{1}{m}\sum_{l=1}^{j}t_l\}. \quad (3)$$

On the other hand, as π' uses the SPT rule of the transportation times to assigning jobs to the vehicles,

$$r_j^{\pi'}\le(1-\frac{1}{m})t_j+\frac{1}{m}\sum_{l=1}^{j}t_l\le(2-\frac{1}{m})r_{[j]}^*, 1\le j\le n. \quad (4)$$

Lemma 3. $C^{\pi'}(b^{\pi'})\le(2-\dfrac{1}{m})C^*(b^*)$.

Proof. Since π' and π^* have the same number of jobs in each batch, we denote $n_i^{\pi'}=n_i^*=n_i$, $|B_i^{\pi'}|=|B_i^*|=|B_i|=n_i-n_{i-1}, 1\le i\le b^*=b^{\pi'}$. $r_{n_i}^{\pi'}$ and $r_{[n_i]}^*$ are denoted as the arrival time of the last arrived job $J_{n_i}^{\pi'}$ and $J_{[n_i]}^*$ in the i-th batch in π' and π^*, respectively.

By Lemma 2 and (4), we have

$$C^{\pi'}(1)=r_{n_1}^{\pi'}+s+|B_1|p\le(2-\frac{1}{m})r_{[n_1]}^*+s+|B_1|p$$
$$\le(2-\frac{1}{m})(r_{[n_1]}^*+s+|B_1|p)=(2-\frac{1}{m})C^*(1). \quad (5)$$

By Lemma 1(iii) and (5),

$$C^{\pi'}(2)=\max\{r_{n_2}^{\pi'},C^{\pi'}(1)\}+s+|B_2|p$$
$$\le(2-\frac{1}{m})\max\{r_{[n_2]}^*,C^*(1)\}+s+|B_2|p$$
$$\le(2-\frac{1}{m})(\max\{r_{[n_2]}^*,C^*(1)\}+s+|B_2|p)$$
$$=(2-\frac{1}{m})C^*(2). \quad (6)$$

Repeat the above process, we get the result

$$C^{\pi'}(b^{\pi'})=\max\{r_{n_{b^{\pi'}}}^{\pi'},C^{\pi'}(b^{\pi'}-1)\}+s+|B_{b^{\pi'}}|p$$
$$\le(2-\frac{1}{m})\max\{r_{[n_{b^*}]}^*,C^*(b^*-1)\}+s+|B_{b^*}|p$$

$$\leq (2-\frac{1}{m})C^*(b^*).$$

Theorem 3. For the problem $D \to 1 | m \geq 1, t=0, c\geq 1, \quad s-batch, p_j = p | C_{\max} + \alpha(b)$, the worst case ratio of TSPT-DP algorithm is not greater than $2-\frac{1}{m}$, and the bound is tight.

Proof. By (2) and Lemma 3, we have

$$C^{\pi}_{\max} + \alpha(b^{\pi}) \leq C^{\pi'}_{\max} + \alpha(b^{\pi'}) = C^{\pi'}(b^{\pi'}) + \alpha(b^{\pi'})$$
$$\leq (2-\frac{1}{m})C^*(b^*) + \alpha(b^*)$$
$$\leq (2-\frac{1}{m})(C^*_{\max} + \alpha(b^*)). \quad (7)$$

We can conclude that the worst case ratio of TSPT-DP algorithm is at most $2-\frac{1}{m}$. To show the tightness, let us consider the following instance: $n = m^2 - m + 1$, $t = 0$, $t_1 = t_2 = \cdots = t_{m^2-m} = 1, t_{m^2-m+1} = m$, $s = p = 0$, $c = 1$, and $\alpha(b) = 0$. Clearly, both TSPT-DP and the optimal algorithm must process jobs in $m^2 - m + 1$ batches. Consequently, we have $C^{\pi}_{\max} = 2m-1$, $C^*_{\max} = m$, and $\alpha(b^{\pi}) = 0$, $\alpha(b^*) = 0$. Thus, the desired result follows.

5 Conclusion

In this paper, we studied the problem of coordinated scheduling on a serial batching machine with transportation. Under the condition of the job processing times are equal, if the job assignment to the vehicles is predetermined, we provide a polynomial time dynamic programming algorithm, for the general problem, we prove it is NP-hard. When the returning time of vehicle is 0, we present the approximation algorithm TSPT-DP and prove that the worst case ratio of the algorithm is not greater than $2-\frac{1}{m}$, and the bound is tight.

Several possible extension to this research can be considered, such as, minimizing maximum job tardiness, developing effective heuristics to solve the general problem, et al.

Acknowledgments

This research is partly supported by the projects as follows:

(i) Specialized Research Fund for the Doctoral Program of Higher Education (Grant No. 20123705110003);

(ii) National Natural Science Foundation of China (Grant No. 11071142, 11201259, 11201121, 1271338);

(iii) Natural Science Foundation of Education Department of Henan Province (Grant No. 2011B110007, 2011B110008);

(iv) Natural Science Foundation of Shandong Province (Grant No. ZR2010AM034).

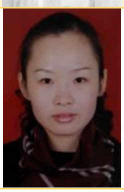
References

[1] Chungyee L, Zhilong C 2001 *J. Sched.* **4** 3-24
[2] Potts C N, Kovalyov M Y 2000 *Eur. J. Oper. Res.* **120** 228-49
[3] Brucker P, Gladky A, Hoogeveen H, Kovalyov M Y, Potts C N, Tautenhahn T, Van de Velde S L 1998 *J. Sched.* **1** 31-54
[4] Yuzhong Z, Qingguo B, Jianteng X 2006 *Oper. Res. Trans.* **10** 99-105
[5] Coffman E G, Yannakakis M, Magazine J, Santos C 1990 *Ann. Oper. Res.* **26** 135-47
[6] Liji S, Buscher U 2012 *Eur. J. Oper. Res.* **221** 14-26
[7] Glass C A, Potts C N, Strusevich V A 2001 *INFORMS J. Comput.* **13** 120-37
[8] Webster S, Baker K R 1995 *Oper. Res.* **43** 692-703
[9] Baptiste P 2000 *Math. Methods Oper. Res.* **52** 355-67
[10] Hall N G, Potts C N 2005 *Ann. Oper. Res.* **135** 41-64

Authors

Cunchang Gu, born in 1980

Current position and grades: a Ph. D. candidate at Qufu Normal University, and an assistant professor at Henan University of Technology in China.
University studies: BSc and MSc degrees from Department of Mathematics, Zhengzhou University, Zhengzhou City, China, in 2002 and 2009, respectively. He is currently working towards Ph.D. degree at School of Management, Qufu Normal University, Rizhao City, China.
Research interests: scheduling, combinatorial optimization, and supply chain management.



Xiaoyan Xu, born in 1979

Current position and grades: a lecturer at Henan University of Technology in China.
University studies: BSc and MSc degrees from Department of Mathematics, Henan University, Kaifeng, City, China, in 2002 and 2010, respectively. She is currently working as a teacher at a College of Science, Henan University of Technology, Zhengzhou City, China.
Research interests: combinatorial optimization, linear programming and statistics theory.