ISSN 1064-2307, Journal of Computer and Systems Sciences International, 2006, Vol. 45, No. 3, pp. 450–458. © Pleiades Publishing, Inc, 2006.
Published in Russian in Izvestiya Akademii Nauk. Teoriya i Sistemy Upravleniya, 2006, No. 3, pp. 120–128.

SYSTEMS ANALYSIS AND OPERATIONS RESEARCH

A Special Case of the Single-Machine Total Tardiness Problem is NP-Hard[1]

E. R. Gafarov and A. A. Lazarev

Computing Center, Russian Academy of Sciences,
ul. Vavilova 40, GSP-1, Moscow, 117967 Russia

Received December 15, 2005

Abstract—In this paper, it is shown that the special case B-1 of the single-machine total tardiness problem $1 \parallel \sum T_j$ is NP-hard in the ordinary sense. For this case, there exists a pseudo-polynomial algorithm with run time $O(n\sum p_j)$.

DOI: 10.1134/S1064230706030117

INTRODUCTION

The NP-complete even–odd partition problem (EOP) [1] and the single-machine total tardiness problem NP-hard in the ordinary sense [2] are considered. A pseudo-polynomial time $O(n^4\sum p_j)$ dynamic programming algorithm was proposed by Lawler [3]. The state-of-the-art algorithms proposed by Szwarc et al. [4, 5] handle special instances [6] of the problem for $n < 600$.

We show that the special case **B-1** [7] is NP-hard in the ordinary sense. Note that there exists a pseudo-polynomial algorithm with run time $O(n\sum p_j)$ for the case **B-1** [8]. We propose a polynomial scheme of reduction from the NP-complete *even–odd partition problem* to the special case **B-1** of the problem $1 \parallel \sum T_j$.

1. STATEMENT OF THE PROBLEMS

1.1. The Single-Machine Total Tardiness Problem $1 \parallel \sum T_j$

Given a set N of n independent jobs that need to be processed on a single machine. Preemptions of jobs are not allowed. The single machine can handle only one job at a time. The jobs are available for processing at time 0. For each job $j \in N = \{1, 2, \ldots, n\}$, a processing time $p_j > 0$ and a due date π are given. A schedule d_j is uniquely determined by a permutation of elements of N. We need to construct an optimal schedule π^* that minimizes the total tardiness value

$$F(\pi) = \sum_{j=1}^{n} \max\{0, c_j(\pi) - d_j\},$$

where $c_j(\pi)$ is the completion time of job j in schedule π.

1.2. Even–Odd Partition Problem (EOP)

Given a set of $2n$ positive integers $B = \{b_1, b_2, \ldots, b_{2n}\}$, $b_i > b_{i+1}$, $1 \le i \le 2n-1$, is there a partition of B into two subsets B_1 and B_2 such that $\sum_{b_i \in B_1} b_i = \sum_{b_i \in B_2} b_i$ and such that, for each $i = 1, 2, \ldots, n$, B_1 (and, hence, B_2) contains exactly one number of $\{b_{2i-1}, b_{2i}\}$? The EOP problem is a well-known NP-complete problem.

Let $\delta_i = b_{2i-1} - b_{2i}$, $i = 1, \ldots, n$, $\delta = \sum_{i=1}^{n} \delta_i$. Now, we construct a modified even–odd partition problem. There is the following set of integers

$$\begin{cases} a_{2n} = M + b \\ a_{2i} = a_{2i+2} + b, \quad i = n-1, \ldots, 1 \\ a_{2i-1} = a_{2i} + \delta_i, \quad i = n, \ldots, 1, \end{cases}$$

where $b \gg n\delta$, $M \ge n^3 b$. Obviously, we have $a_i > a_{i+1}$, $\forall i = 1, 2, \ldots, 2n-1$, $\delta_i = b_{2i-1} - b_{2i} = a_{2i-1} - a_{2i}$, $i = 1, \ldots, n$.

Lemma 1. The original EOP problem has a solution if and only if the modified EOP problem does.

Proof. Let, for the original problem, there exist two subsets B_1 and B_2 such that $\sum_{b_i \in B_1} b_i = \sum_{b_i \in B_2} b_i$. We

[1] Article was translated by the authors.

denote $A_1 = \{a_i \mid b_i \in B_1\}$, $A_2 = \{a_i \mid b_i \in B_2\}$. Then, we have $\sum_{a_i \in A_1} a_i = \sum_{a_i \in A_2} a_i$.

Let, for the modified problem, there exist two subsets A_1 and A_2 such that $\sum_{a_i \in A_1} a_i = \sum_{a_i \in A_2} a_i$. Let us denote $B_1 = \{b_i \mid a_i \in A_1\}$, $B_2 = \{b_i \mid a_i \in A_2\}$. We have $\sum_{b_i \in B_1} b_i = \sum_{b_i \in B_2} b_i$.

2. SPECIAL CASES OF THE $1 \parallel \sum T_j$ PROBLEM

The following case B-1 of the problem $1 \parallel \sum T_j$ was considered [7]:

$$\begin{cases} p_1 \ge p_2 \ge \ldots \ge p_n \\ d_1 \le d_2 \le \ldots \le d_n \\ d_n - d_1 \le p_n. \end{cases} \tag{2.1}$$

This case is referred to the so-called "hard" instances in paper [7]. The research of known algorithms [4, 8, 9] has shown that, for the case B-1, the number of branchings in the search tree is the greatest [8]. Let us introduce the necessary definitions.

The sequence $\pi = (j_1, j_2, \ldots, j_n)$ is an *SPT-schedule* (shortest processing time) if $p_{j_k} \le p_{j_{i+k}}$, and for $p_{j_k} = p_{j_{k+1}}$, we have $d_{j_k} \le d_{j_{k+1}}$, $k = 1, 2, \ldots, n-1$. The sequence $\pi = (j_1, j_2, \ldots, j_n)$ is an *EDD-schedule* (earliest due date) if $d_{j_k} \le d_{j_{k+1}}$ and, for $d_{j_k} = d_{j_{k+1}}$, we have $p_{j_i} \ge p_{j_{k+1}}$, $k = 1, 2, \ldots, n-1$.

For case (2.1), the sequence $\pi = (1, 2, \ldots, n)$ is an EDD-schedule. The sequence $\pi = (n, n-1, \ldots, 1)$ is an SPT-schedule.

The sequence π' is a *partial schedule* if it contains only jobs from the subset $N' \subset N$. Let $P(N') = \sum_{i \in N'} p_i$ be the subset $N' \subset N$ of jobs processed in $\{\pi'\} = N'$, and we denote $P(\pi') = \sum_{i \in \{\pi'\}} p_i$.

Lemma 2 [8]. For case (2.1), there exists an optimal sequence $\pi^* = (\pi_{EDD}, l, \pi_{SPT})$, where π_{EDD} and π_{SPT} are partial sequences constructed according to *EDD* and *SPT* rules.

Corollary. For case (2.1), late jobs for all optimal schedules are processed according to the SPT-order, except, perhaps, the first one.

Now, we present the polynomial reduction from the modified *EOP* problem to the special subcase (2.1) of the problem $\mathbf{1} \parallel \sum \boldsymbol{T_j}$. The number of jobs is $2n + 1$.

We denote the jobs by $V_1, V_2, V_3, V_4, \ldots, V_{2i-1}, V_{2i}, \ldots, V_{2n-1}, V_{2n}, V_{2n+1}$, $N = \{1, 2, \ldots, 2n, 2n+1\}$. To simplify the notation, we introduce $p_{V_i} = p_i$, $d_{V_i} = d_i$, $T_{V_i} = T_i$, and $C_{V_i} = C_i$, $i = 1, \ldots, 2n + 1$. The case that satisfies the following constraints is called a *canonical LG-instance.*

$$\begin{cases} p_1 > p_2 > \ldots > p_{2n+1} \\ d_1 < d_2 < \ldots < d_{2n+1} \\ d_{2n+1} - d_1 < p_{2n+1} \\ p_{2n+1} = M = n^3 b \\ p_{2n} = p_{2n+1} + b = a_{2n} \\ p_{2i} = p_{2i+2} + b = a_{2i}, \quad i = n-1, \ldots, 1 \\ p_{2i-1} = p_{2i} + \delta_i = a_{2i-1}, \quad i = n, \ldots, 1 \\ d_{2n+1} = \sum_{i=1}^{n} p_{2i} + p_{2n+1} + \frac{1}{2}\delta \\ d_{2n} = d_{2n+1} - \delta \\ d_{2i} = d_{2i+2} - (n-i)b + \delta, \quad i = n-1, \ldots, 1 \\ d_{2i-1} = d_{2i} - (n-i)\delta_i - \varepsilon\delta_i, \quad i = n, \ldots, 1, \end{cases} \tag{2.2}$$

where $b = n^2\delta$, $0 < \varepsilon < \dfrac{\min_i \delta_i}{\max_i \delta_i}$.

The due dates pattern of the canonical LG instance is presented in Fig. 1.

Let

$$L = \frac{1}{2}\sum_{i=1}^{2n} p_i,$$

then, we have $d_{2n+1} = L + p_{2n+1}$, because $\frac{1}{2}\sum_{i=1}^{2n} p_i = \sum_{i=1}^{n} p_{2i} + \frac{1}{2}\delta$. It is worth noting that canonical DL-instances from paper [2] do not satisfy case (2.2). The first two inequalities show that (2.1) is not a subcase.

3. PROPERTIES OF THE SPECIAL CASE (2.2) OF THE PROBLEM $1 \parallel \sum T_j$

Let us formulate the following lemma.

Lemma 3. For case (2.2), for all sequences, the number of tardy jobs equals n or $n + 1$.

Proof. We split the proof into two states. (1) Consider the set N' of $n + 2$ jobs with the smallest processing times and arrange them at the beginning of the schedule. Obviously, we have

$$\sum_{i \in N'} p_i > (n+2)p_{\min} = (n+2)n^3 b,$$

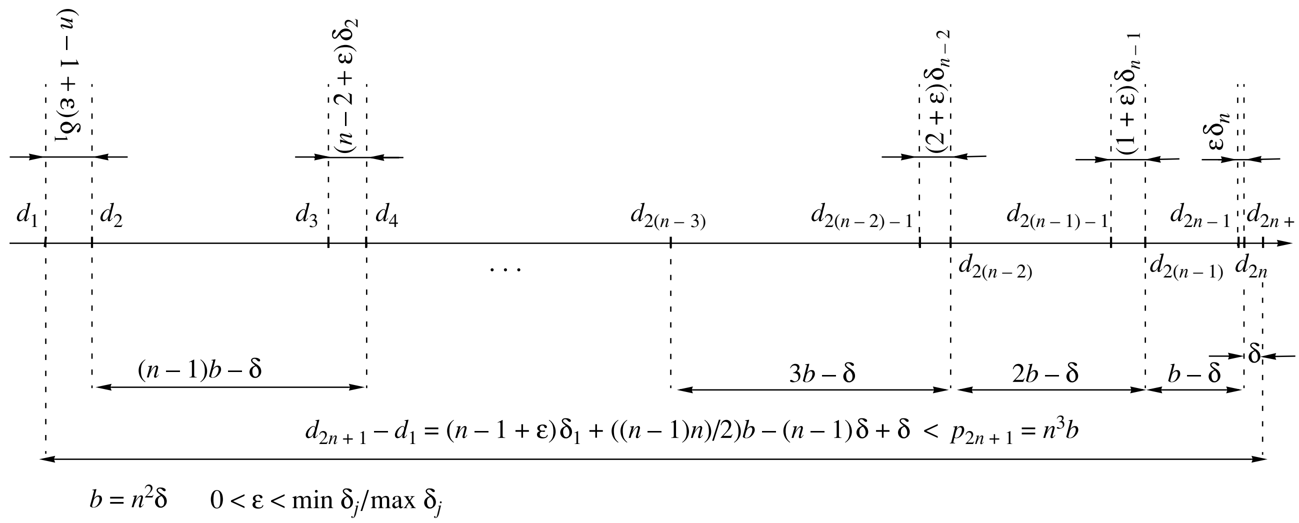

Fig. 1. Due date pattern of the canonical LG instance.

where $p_{\min} = \min_{j \in N}\{p_j\} = p_{2n+1}$. According to the fourth to eighth equations from (2.2), we have

$$d_{\max} = \max_{j \in N}\{d_j\} = d_{2n+1}$$
$$= (n+1)n^3b + (b + 2b + \ldots + nb) + \frac{1}{2}\delta,$$

therefore,

$$d_{\max} = d_{2n+1}$$
$$= (n+1)n^3b + \frac{n(n+1)}{2}b + \frac{1}{2}\delta < (n+2)n^3b < \sum_{i \in N'} p_i.$$

Thus, the job processed on the $(n+2)$th position is tardy in all schedules. The subsequent jobs are also tardy by the third equation from (2.2) since the difference between due dates of any two jobs is less than the processing time for each job. Thus, for every schedule π, the number of tardy jobs is greater than or equal to $n+1$.

(2) Let us consider set N'' of n jobs that are the longest in the processing time and process it at the beginning of the schedule. Two cases are considered.

(a) Let $n = 2k$; then $N'' = \{V_1, V_2, \ldots, V_{2k-1}, V_{2k}\}$. We have

$$P(N'') = nn^3b + 2(nb + (n-1)b + \ldots$$
$$+ (n-k+1)b) + \sum_{i=1}^{k}\delta_i,$$

$$P(N'') = nn^3b$$
$$+ 2\left(\frac{n(n+1)}{2} - \frac{(n-k)(n-k+1)}{2}\right)b + \sum_{i=1}^{k}\delta_i.$$

According to equations eight to eleven from (2.2), we have

$$d_{\min} = \min_{j \in N}\{d_j\} = d_1 = d_{2n+1}$$
$$-\left(\sum_{i=1}^{n-1}((n-i)b - \delta) + \delta + (n-1)\delta_1 - \varepsilon\delta_1\right)$$
$$= (n+1)n^3b + (b + 2b + \ldots + nb) + \frac{1}{2}\delta$$
$$-\left(\sum_{i=1}^{n-1}((n-i)b - \delta) + \delta + (n-1)\delta_1 + \varepsilon\delta_1\right) > P(N'');$$

(b) Let $n = 2k+1$; then, $N'' = \{V_1, V_2, \ldots, V_{2k-1}, V_{2k}, V_{2(k+1)-1}\}$ and

$$P(N'') = nn^3b + 2(nb + (n-1)b + \ldots$$
$$+ (n-k+1)b) + (n-k)b + \sum_{i=1}^{k+1}\delta_i,$$

$$P(N'') = nn^3b + 2\left(\frac{n(n+1)}{2}\right.$$
$$\left.- \frac{(n-k)(n-k+1)}{2}\right)b + (n-k)b + \sum_{i=1}^{k+1}\delta_i,$$

$$d_{\min} = d_1 = d_{2n+1}$$
$$-\left(\sum_{i=1}^{n-1}((n-i)b - \delta) + \delta + (n-1)\delta_1 - \varepsilon\delta_1\right)$$

$$= (n+1)n^3b + (b + 2b + \ldots + nb) + \frac{1}{2}\delta$$

$$-\left(\sum_{i=1}^{n-1}((n-i)b-\delta) + \delta + (n-1)\delta_1 + \varepsilon\delta_1\right) > P(N'').$$

This means that the first n jobs are not tardy. Hence, for any schedule π, the number of tardy jobs is less than or equal to n.

Thus, for case (2.2) in all sequences, the number of tardy jobs equals n or $n + 1$.

Lemma 4. For case (2.2), for all schedules $\pi = (\pi_1, \pi_2)$, there exists a schedule $\pi' = (\pi_{EDD}, \pi_{SPT})$, where $\{\pi_1\} = \{\pi_{EDD}\}$, $\{\pi_2\} = \{\pi_{SPT}\}$, $|\{\pi_1\}| = n + 1$, $|\{\pi_2\}| = n$, and, what is more, $F(\pi) \le F(\pi')$ holds.

Proof. The partial sequence π_1 is considered. The first n jobs in π_1 are not tardy, only the last job may be tardy, so the EDD order is optimal for the set of jobs $\{\pi_1\}$. In this case, on the $(n + 1)$th position, job $j = \arg\max\{d_i : i \in \{\pi_1\}\}$ is processed.

Now, we consider the sequence π_2. The EDD-order is optimal for set of jobs $\{\pi_2\}$, because all n jobs are tardy.

The schedule $((V_{1,1}, V_{2,1}, \ldots, V_{i,1}, \ldots, V_{n,1}, V_{2n+1}, V_{n,2}, \ldots, V_{i,2}, \ldots, V_{2,2}, V_{1,2})$ is called a canonical LG schedule, where $\{V_{i,1}, V_{i,2}\} = \{V_{2i-1}, V_{2i}\}$, $i = 1, 2, \ldots, n$.

Lemma 5. If the sequence $\pi = (\pi_1, \pi_2)$, $|\{\pi_1\}| = n + 1$, $|\{\pi_2\}| = n$ is not a canonical LG schedule or we cannot reduce it to a canonical LG schedule by the EDD and SPT rules to π_1 and π_2 sets, then, in the schedule π, two jobs $\{V_{2i-1}, V_{2i}\}$, $i < n$ are on-time processed or π have the structure

$$\begin{gathered}(V_{1,1}, V_{2,1}, \ldots, V_{i,1}, \ldots, V_{n-1,1}, V_{2n-1}, V_{2n}, V_{2n+1},\\ V_{n-1,2}, \ldots, V_{i,2}, \ldots, V_{2,2}, V_{1,2}),\end{gathered} \tag{3.1}$$

i.e., the pair of jobs $\{V_{2n-1}, V_{2n}\}$ is processed before V_{2n+1}, one of the jobs of each pair $\{V_{2i-1}, V_{2i}\}$, $i = 1, \ldots, n-1$ is processed before V_{2n+1}, the other job of the pair is processed after V_{2n+1}, and the job V_{2n+1} is processed at the $(n + 2)$th step.

Proof. Let $\pi = (\pi_1, \pi_2)$, where $|\{\pi_1\}| = n + 1$, $|\{\pi_2\}| = n$. Consider the following cases.

(1) If $\{\pi_2\} = \{V_{1,2}, \ldots, V_{n,2}\}$, then π_2 consists of n jobs and only one job from each pair $\{V_{2i-1}, V_{2i}\}$ for all $i = 1, \ldots, n$ belongs to it. Jobs from π_2 are sequenced by the SPT-rule. We have a new canonical schedule π'. By Lemma 4, we have $F(\pi') \le F(\pi)$.

(2) If $\{\pi_2\} \ne \{V_{1,2}, \ldots, V_{n,2}\}$, the following cases are possible:

(a) $V_{2n+1} \in \{\pi_2\}$;

(b) there exists a pair of jobs $\{V_{2j-1}, V_{2j}\} \subset \{\pi_2\}$.

Then, for some i, we have $\{V_{2i-1}, V_{2i}\} \subset \{\pi_1\}$, because $|\{\pi_2\}| = n$.

In what follows, we show in Theorem 1 that, for case (2.2), all optimal schedules are canonical LG-schedules. We will prove that a schedule π can be transformed to a canonical LG-schedule π' and $F(\pi) > F(\pi')$. In the proof of Theorem 1, Lemmas 6–9 are used.

Lemma 6. Let the schedule π has the form (3.1), where the job V_{2n+1} is processed on the $(n + 2)$th position. For schedule $\pi' = (V_{1,1}, V_{2,1}, \ldots, V_{i,1}, \ldots, V_{n-1,1}, V_{2n-1}, V_{2n+1}, V_{2n}, V_{n-1,2}, \ldots, V_{i,2}, \ldots, V_{2,2}, V_{1,2})$, we will have $F(\pi) > F(\pi')$.

Proof. In schedule π, the job V_{2n-1} on the nth position are processed. According to Lemma 3, the job V_{2n-1} is not tardy. The job V_{2n+1} on the $(n + 2)$th position is processed, so it is a tardy job.

For jobs $\{V_2, V_4, \ldots, V_{2i}, \ldots, V_{2n-2}, V_{2n-1}\}$, we have

$$P(\{V_2, V_4, \ldots, V_{2i}, \ldots, V_{2n-2}, V_{2n-1}\})$$

$$= nn^3b + \sum_{k=1}^{n} kb + \delta_n = d_{V_{2n+1}} - n^3b - \frac{1}{2}\delta + \delta_n,$$

by the eighth equation from (2.2). Obviously,

$$P(\{V_{1,1}, V_{2,1}, \ldots, V_{i,1}, \ldots, V_{n-1,1}, V_{2n-1}\}) + p_{2n}$$

$$\ge P(\{V_2, V_4, \ldots, V_{2i}, \ldots, V_{2n-2}, V_{2n-1}\}) + p_{2n},$$

holds; thus,

$$C_{2n}(\pi) \ge d_{2n+1} + b - \frac{1}{2}\delta + \delta_n > d_{2n}.$$

Therefore, the job V_{2n}, which is processed on the $(n + 1)$th place in schedule π is tardy. Let $\pi = (\pi_{11}, V_{2n}, V_{2n+1}, \pi_{21})$. Consider the canonical LG schedule $\pi' = (\pi_{11}, V_{2n+1}, V_{2n}, \pi_{21})$. Let us show that $F(\pi) > F(\pi')$.

(a) Let, in the schedule π', the job V_{2n+1} be not tardy. According to (2.2), $d_{2n+1} - C_{2n+1}(\pi') \le \frac{1}{2}\delta$ holds, because the schedule π' is a canonical LG.

From Fig. 2, we can see that the equation

$$F(\pi) - F(\pi') = T_{2n}(\pi) + T_{2n+1}(\pi)$$

$$-(T_{2n}(\pi') + T_{2n+1}(\pi')) = (T_{2n+1}(\pi) - T_{2n+1}(\pi'))$$

$$-(T_{2n}(\pi') - T_{2n}(\pi)) \ge \left(p_{2n} - \frac{1}{2}\delta\right) - p_{2n+1}$$

$$= p_{2n+1} + b - \frac{1}{2}\delta - p_{2n+1} > 0$$

holds.

(b) Let, in the schedule π', the job V_{2n+1} is tardy, then we have

$$F(\pi) - F(\pi') = T_{2n}(\pi) + T_{2n+1}(\pi)$$

$$-(T_{2n}(\pi') + T_{2n+1}(\pi')) = p_{2n} - p_{2n+1} = b > 0.$$

Lemma 7. Assume that, in the schedule $\pi = (\pi_{11}, V_{2i-1}, V_{2i}, \pi_{12}, \pi_{21}, X, \pi_{22})$, a pair of jobs $\{V_{2i-1}, V_{2i}\}$,

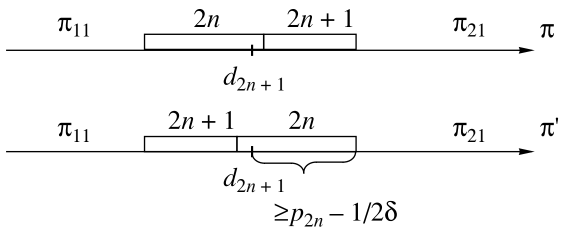

Fig. 2. The permutation of V_{2n} and V_{2n+1}.

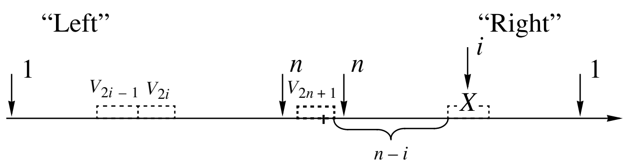

Fig. 3. The permutation.

$i < n$, are not tardy, and, on the position i ("right"), the job $X \in \{V_{2j-1}, V_{2j}\}, j \geq i+1$. Then, for the schedule $\pi' = (\pi_{11}, V_{2i-1}, X, \pi_{12}, \pi_{21}, V_{2i}, \pi_{22})$, we have $F(\pi) > F(\pi')$.

Proof. Let, in the schedule π, only the jobs from $\{\pi_{21}, X, \pi_{22}\}$ be tardy, where $|\{\pi_{22}\}| = i - 1$. The job X is processed on the position i ("right") (Fig. 3). In the canonical LG schedule, the job $V_{i,2} \in \{V_{2i-1}, V_{2i}\}$ is processed on the position i ("right").

Construct a schedule $\pi' = (\pi_{11}, V_{2i-1}, X, \pi_{12}, \pi_{21}, V_{2i}, \pi_{22})$. By Lemma 3, in both schedules, the number of tardy jobs is greater than or equal to n. Therefore, the number of tardy jobs preceding V_{2i} in the schedule π' is greater than or equal to $n - i$ and no greater than $n - i + 1$. Thus,

$$F(\pi) - F(\pi') \geq (p_{2i} - p_X)(n-i) - (d_X - d_{2i}).$$

The following situations are possible.

(a) If $X = V_{2j}$, then $p_{2i} - p_X = (j-i)b$,

$$d_X - d_{2i} = \sum_{k=i}^{j-1}(n-k)b - (j-i)\delta$$

$$= n(j-i)b - \sum_{k=i}^{j-1} kb - (j-i)\delta$$

$$= n(j-i)b - i(j-i)b - \sum_{k=0}^{j-1-i} kb - (j-i)\delta.$$

Hence,

$$F(\pi) - F(\pi') \geq (j-i)b(n-i) - (n(j-i)b - i(j-i)b$$

$$- \sum_{k=0}^{j-1-i} kb - (j-i)\delta) = \sum_{k=0}^{j-1-i} kb + (j-i)\delta > 0.$$

(b) If $X = V_{2j-1}$, then

$$p_{2i} - p_X = ((j-i)b - \delta_j),$$

$$d_X - d_{2i} = \sum_{k=i}^{j-1}(n-k)b - (j-i)\delta - (n-j)\delta_j - \varepsilon\delta_j$$

$$= n(j-i)b - \sum_{k=i}^{j-1} kb - (j-i)\delta - (n-j)\delta_j - \varepsilon\delta_j$$

$$= n(j-i)b - i(j-i)b$$

$$- \sum_{k=0}^{j-1-i} kb - (j-i)\delta - (n-j)\delta_j - \varepsilon\delta_j.$$

Hence,

$$F(\pi) - F(\pi') \geq ((j-i)b - \delta_j)(n-i) - (n(j-i)b$$

$$- i(j-i)b - \sum_{k=0}^{j-1-i} kb - (j-i)\delta - (n-j)\delta_j - \varepsilon\delta_j)$$

$$= \sum_{k=0}^{j-1-i} kb + (j-i)\delta - (j-i)\delta_j + \varepsilon\delta_j > 0.$$

Lemma 8. Assume that, in the schedule $\pi = (\pi_{11}, V_{2i-1}, V_{2i}, \pi_{12}, \pi_{21}, X, \pi_{22})$, a pair of jobs $\{V_{2i-1}, V_{2i}\}$, $i < n$, are not tardy, and, on the position i ("right"), the job $X \in \{V_{2j-1}, V_{2j}\}$, $j < i - 1$. Then, for the schedule $\pi' = (\pi_{11}, V_{2i-1}, X, \pi_{12}, \pi_{21}, V_{2i}, \pi_{22})$, we have $F(\pi) > F(\pi')$.

Proof. Suppose that, in the schedule π, only the jobs from $\{\pi_{21}, X, \pi_{22}\}$ are tardy, where $|\{\pi_{22}\}| = i - 1$. The job X is processed on the position i ("right") (Fig. 3). In the canonical LG-schedule, the job $V_{i,2} \in \{V_{2i-1}, V_{2i}\}$ is processed on the position i ("right").

Construct a schedule $\pi' = (\pi_{11}, V_{2i-1}, X, \pi_{12}, \pi_{21}, V_{2i}, \pi_{22})$. By Lemma 3, in both schedules, the number of tardy jobs is greater than or equal to n. Therefore, by Lemma 4, the number of tardy jobs following V_{2i} in the schedule π' is greater than or equal to $n - i$ and no greater than $n - i + 1$. Thus, we have

$$F(\pi) - F(\pi') \geq (d_{2i} - d_X) - (p_X - p_{2i})(n-i+1).$$

(a) If $X = V_{2j}$, then $p_X - p_{2i} = (i-j)b$,

$$d_{2i} - d_{2j} = \sum_{k=j}^{i-1}(n-k)b - (i-j)\delta$$

$$= n(i-j)b - \sum_{k=j}^{i-1} kb - (i-j)\delta$$

$$= n(i-j)b - (i-1)(i-j)b + \sum_{k=0}^{i-1-j} kb - (i-j)\delta.$$

Therefore,

$$F(\pi) - F(\pi') \geq n(i-j)b - (i-1)(i-j)b + \sum_{k=0}^{i-1-j} kb$$

$$-(i-j)\delta - (i-j)b(n-i+1) = \sum_{k=0}^{i-1-j} kb - (i-j)\delta > 0.$$

(b) If $X = V_{2j-1}$, then $p_X - p_{2i} = (i-j)b + \delta_j$,

$$d_{2i} - d_{2j-1} = \sum_{k=j}^{i-1} (n-k)b - (i-j)\delta + (n-j)\delta_j + \varepsilon\delta_j$$

$$= n(i-j)b - \sum_{k=j}^{i-1} kb - (i-j)\delta + (n-j)\delta_j + \varepsilon\delta_j$$

$$= n(i-j)b - (i-1)(i-j)b$$

$$+ \sum_{k=0}^{i-1-j} kb - (i-j)\delta + (n-j)\delta_j + \varepsilon\delta_j.$$

Hence,

$$F(\pi) - F(\pi') \geq n(i-j)b - (i-1)(i-j)b + \sum_{k=0}^{i-1-j} kb$$

$$-(i-j)\delta + (n-j)\delta_j + \varepsilon\delta_j - ((i-j)b + \delta_j)(n-i+1)$$

$$= \sum_{k=0}^{i-1-j} kb - (i-j)\delta - \delta_j + \varepsilon\delta_j > 0.$$

Lemma 9. Assume that, in the schedule $\pi = (\pi_{11}, V_{2i-1}, V_{2i}, \pi_{12}, \pi_{21}, X, \pi_{22})$, a pair of jobs $\{V_{2i-1}, V_{2i}\}$, $i < n$, are not tardy, and, on the position i ("right"), the job $X \in \{V_{2(i-1)-1}, V_{2(i-1)}\}$. Let, in the schedule $\pi' = (\pi_{11}, V_{2i-1}, X, \pi_{12}, \pi_{21}, V_{2i}, \pi_{22})$, the job Y be processed on the position $n+1$ and $T_Y(\pi') < 2\delta$. Then, we will have $F(\pi) > F(\pi')$.

Proof. Let, in the schedule π, only the jobs from $\{\pi_{21}, X, \pi_{22}\}$ be tardy, where $|\{\pi_{22}\}| = i-1$. The job X is processed on the position i ("right") (Fig. 3). In the canonical LG-schedule, the job $V_{i,2} \in \{V_{2i-1}, V_{2i}\}$ is processed on the position i ("right").

Construct a schedule $\pi' = (\pi_{11}, V_{2i-1}, X, \pi_{12}, \pi_{21}, V_{2i}, \pi_{22})$. By Lemma 3, in all schedules, the number of tardy jobs is greater than or equal to n. Hence, the number of tardy jobs preceding V_{2i} in π is greater than or equal to $n-i$. Thus,

$$F(\pi) - F(\pi') > (d_{2i} - d_X) - (p_X - p_{2i})(n-i)$$

$$-(T_Y(\pi') - T_Y(\pi)) > (d_{2i} - d_X) - (p_X - p_{2i})(n-i) - 2\delta.$$

(a) If $X = V_{2(i-1)}$, then $p_X - p_{2i} = b$, $d_{2i} - d_{2i-2} = (n-i+1)b - \delta$. Thus,

$$F(\pi) - F(\pi') > (n-i+1)b - \delta - (n-i)b - 2\delta = b - 3\delta > 0.$$

(b) If $X = V_{2(i-1)-1}$, then $p_X - p_{2i} = b + \delta_{i-1}$, $d_{2i} - d_{2i-2} = (n-i+1)b - \delta + (n-i+1)\delta_{i-1} + \varepsilon\delta_{i-1}$. Therefore,

$$F(\pi) - F(\pi') > (n-i+1)b - \delta + (n-i+1)\delta_{i-1}$$

$$+ \varepsilon\delta_{i-1} - (n-i)(b + \delta_{i-1}) - 2\delta$$

$$= b - 3\delta + \delta_{i-1} + \varepsilon\delta_{i-1} > 0,$$

because $b = n^2\delta$.

The results of Lemma 9 are employed in the proof of Theorem 1. Note that the inequality $T_Y(\pi) < 2\delta$ takes place. The situation $T_Y(\pi') \geq 2\delta$ is not considered since it does not occur. Based on the lemmas obtained, we prove the following theorem.

Theorem 1. For case (2.2), all optimal schedules are canonical LG schedules or can be reduced to canonical LG-schedules by the application of EDD-rule to the first $n+1$ jobs.

Proof. Assume that π is an arbitrary schedule. By Lemma 4, we can consider only schedules of the form $\pi = (\pi_{EDD}, \pi_{SPT})$, where $|\{\pi_{EDD}\}| = n+1$. Note that the job V_{2n+1} is processed on the position $n+1$ or $n+2$. Suppose that the schedule π is not a canonical LG schedule.

Then, in π, two jobs $\{V_{2i-1}, V_{2i}\}$, $i < n$, are not tardy or π has structure (3.1) (see Lemma 6). Therefore, by Lemma 6, there exists a canonical LG schedule $\pi' = (V_{1,1}, V_{2,1}, \ldots, V_{i,1}, \ldots, V_{n-1,1}, V_{2n-1}, V_{2n+1}, V_{2n}, V_{n-1,2}, \ldots, V_{i,2}, \ldots, V_{2,2}, V_{1,2})$, so that $F(\pi) > F(\pi')$. Redenote $\pi = \pi'$.

The following algorithm transforms a schedule π to a canonical LG-schedule. The algorithm consists of two cycles.

Cycle 1. WHILE, in the next schedule π, there exists i such that, on the position i ("right"), a job $X \notin \{V_{2(i-1)-1}, V_{2(i-1)}\}$, $X \neq V_{2n+1}$ is processed AND jobs V_{2i-1}, V_{2i} are not tardy DO

We apply a permutation for V_{2i}, and X are denoted in Lemmas 7 and 8. We have a new schedule π'. The total tardiness is decreased.

End of cycle 1.

Denote $\pi = \pi'$. Obviously, the number of steps of cycle 1 is fewer than n. Then, apply the **EDD**-rule for the first $n+1$ jobs in π.

The job V_{2n+1} is processed on the position $n+1$ or $n+2$ in the schedule π. If the job V_{2n+1} is processed on the position $n+2$ ("left"), then the job V_{2n-1} has the position n and V_{2n} has the position $n+1$ according to cycle 1 and the **EDD**-rule.

The following cases are possible.

I. Let the job V_{2n+1} be processed on the position $n+2$.

We consider the schedule $\pi = (\pi_1, V_{2n-1}, V_{2n}, V_{2n+1}, \pi_2)$, where V_{2n} is processed on the $(n+1)$th position. Here, we have $|\{\pi_1\}| = n-1 = |\{\pi_2\}|$.

According to cycle 1, only the situations described in Lemma 9 are probable. So, $P(\pi_1) + 2qb + \delta > P(\pi_2) > P(\pi_1) + 2qb - \delta$, where q is the number of situations in the schedule π.

For example, $\{\pi_1\} = \{V_{2i-1}, V_{2i}\} \cup \{V_{1,1}, V_{2,1}, \ldots, V_{i-2,1}, V_{i+1,1}, \ldots, V_{n-1,1}\}$; $\{\pi_2\} = \{V_{2(i-1)-1}, V_{2(i-1)}\} \cup \{V_{1,2}, V_{2,2}, \ldots, V_{i-2,2}, V_{i+1,2}, \ldots, V_{n-1,2}\}$.

Then, $q = 1$ and $P(\pi_1) + 2b + \delta > P(\pi_2) > P(\pi_1) + 2b - \delta$ holds, because $-(\delta - \delta_{i-1} - \delta_i - \delta_n) < P(\{V_{1,1}, V_{2,1}, \ldots, V_{i-2,1}, V_{i+1,1}, \ldots, V_{n-1,1}\}) - P(\{V_{1,2}, V_{2,2}, \ldots, V_{i-2,2}, V_{i+1,2}, \ldots, V_{n-1,2}\}) < \delta - \delta_{i-1} - \delta_i - \delta_n$ and $P(\{V_{2(i-1)-1}, V_{2(i-1)}\}) - P(\{V_{2i-1}, V_{2i}\}) = 2b + \delta_{i-1} - \delta_i$.

Consider two cases when $q = 1$ and $q > 1$.

In the case $q = 0$, we have (3.1) (see Lemma 6).

(a) Let $q = 1$.

It is known that

$$\sum_{i=1}^{2n+1} p_i = 2L + p_{2n+1} = 2L + n^3 b.$$

We denote $\Delta = P(\pi_2) - (P(\pi_1) + 2b)$, where $-\delta < \Delta < \delta$. Let $S = P(\pi_1)$. Then, $2S + 2b + \Delta + p_{2n-1} + p_{2n} + p_{2n+1} = 2S + \Delta + 2b + 3n^3b + 2b + \delta_n = 2L + n^3b$. Thus,

$$L = S + \frac{1}{2}\Delta + 2b + n^3 b + \frac{1}{2}\delta_n,$$

then,

$$\begin{aligned} C_{2n}(\pi) &= P(\pi_1) + p_{2n-1} + p_{2n} \\ &= S + 2n^3 b + 2b + \delta_n = L + n^3 b + \frac{1}{2}\delta_n - \frac{1}{2}\Delta. \end{aligned}$$

It is known that $L + n^3b = d_{2n+1}$; then, $-\delta < C_{2n}(\pi) - d_{2n+1} < \delta$.

There exist two subcases when $C_{2n}(\pi) \geq d_{2n+1}$ and $C_{2n}(\pi) < d_{2n+1}$.

(1) $C_{2n}(\pi) \geq d_{2n+1}$. For the schedule $\pi' = (\pi_1, V_{2n-1}, V_{2n+1}, V_{2n}, \pi_2)$, we have

$$\begin{aligned} F(\pi) - F(\pi') &= T_{2n}(\pi) + T_{2n+1}(\pi) - (T_{2n}(\pi') \\ &+ T_{2n+1}(\pi')) = (T_{2n+1}(\pi) - T_{2n+1}(\pi')) \\ &- (T_{2n}(\pi') - T_{2n}(\pi)) = (p_{2n+1} + (C_{2n}(\pi) - d_{2n+1})) \\ &- p_{2n+1} = C_{2n}(\pi) - d_{2n+1} \geq 0. \end{aligned}$$

(2) $C_{2n}(\pi) < d_{2n+1}$, and $C_{2n}(\pi) > d_{2n}$ holds, because $d_{2n+1} - d_{2n} = \delta$ and $d_{2n+1} - C_{2n}(\pi) < \delta$.

Let us describe the schedule π:

$$\begin{aligned} \pi = (&\pi_{11}, V_{2i-1}, V_{2i}, \pi_{12}, V_{2n-1}, \\ &V_{2n}, V_{2n+1}, \pi_{21}, X, \pi_{22}), \end{aligned}$$

where $|\{\pi_{22}\}| = i - 1$, $X \in \{V_{2(i-1)-1}, V_{2(i-1)}\}$. If $X = V_{2(i-1)-1}$, then the permutation of neighboring jobs $V_{2(i-1)-1}$ and $V_{2(i-1)}$, according to the *SPT-rule*, does not increase the total tardiness.

Let $X = V_{2(i-1)}$. In π, $n + 1$ jobs are late. We construct the schedule

$$\begin{aligned} \pi' = (&\pi_{11}, V_{2i-1}, X, \pi_{12}, V_{2n-1}, \\ &V_{2n}, V_{2n+1}, \pi_{21}, V_{2i}, \pi_{22}). \end{aligned}$$

Here, we have $F(\pi) - F(\pi') = (d_{2i} - d_{2(i-1)}) - (n - i + 1)(p_{2(i-1)} - p_{2i}) = (n - i + 1)b - \delta - (n - i + 1)b = -\delta$, so the total tardiness is increased by δ.

Then, we have $C_{2n}(\pi') - d_{2n+1} > b - \delta$. We construct the schedule

$$\begin{aligned} \pi'' = (&\pi_{11}, V_{2i-1}, X, \pi_{12}, V_{2n-1}, \\ &V_{2n+1}, V_{2n}, \pi_{21}, V_{2i}, \pi_{22}). \end{aligned}$$

We have $F(\pi') - F(\pi'') > (p_{2n+1} + b - \delta) - p_{2n+1} > b - \delta$.

Then, $F(\pi) - F(\pi'') = b - \delta - \delta > 0$.

(b) Let $q > 1$. Then, $d_{2n} - C_{2n}(\pi) > b - 2\delta$.

If $q = 2$, then, in the schedule π' considered in Lemma 9, for job $Y = V_{2n}$, we have $T_Y(\pi') < 2\delta$. Therefore, we can use the permutation described in Lemma 9.

If $q > 2$, then, in the schedule π', n jobs are late, and, by Lemma 9, we have $F(\pi) > F(\pi')$.

II. Let the job V_{2n+1} be processed on the position $n + 1$. Then, from Lemma 10, we have $T_Y(\pi') = T_{2n+1}(\pi') < \frac{1}{2}\delta$. Therefore, we can use the permutation described in Lemma 10.

Cycle 2. WHILE, in the next schedule π', there exist two jobs V_{2i-1}, V_{2i}, so that, on the position i (“right”), a job $X \in \{V_{2(i-1)-1}, V_{2(i-1)}\}\}$ is processed AND jobs V_{2i-1}, V_{2i} are not tardy DO.

We apply the permutation for V_{2i} and X described in cases I and II. We have a new schedule π'. The total tardiness decreases.

End of cycle 2.

End of algorithm.

Therefore, we can transform a schedule π to a canonical LG-schedule π^* in $O(n)$ time, and the inequality $F(\pi) > F(\pi^*)$ holds.

Theorem 2. The modified EOP problem has a solution if and only if in an optimal canonical LG-schedule $C_{2n+1}(\pi) = d_{2n+1}$.

Proof. Consider a canonical LG-schedule

$$\begin{aligned} \pi = (&V_{1,1}, V_{2,1}, \ldots, V_{i,1}, \ldots, V_{n,1}, V_{2n+1}, V_{n,2}, \\ &\ldots, V_{i,2}, \ldots, V_{2,2}, V_{1,2}). \end{aligned}$$

The jobs $V_{n,2}, \ldots, V_{i,2}, \ldots, V_{2,2}, V_{1,2}$ are tardy. The job V_{2n+1} can be tardy; then,

$$F(\pi) = \sum_{i=1}^{n} T_{V_{i,2}}(\pi) + T_{V_{2n+1}}(\pi).$$

We denote $\sum_{i=1}^{2n+1} p_i = C$. Then,

$$\sum_{i=1}^{n} C_{V_{i,2}}(\pi) = nC - \sum_{i=1}^{n-1} (n-i) p_{V_{i,2}}.$$

Denote

$$\phi(i) = \begin{cases} 1, & V_{i,2} = V_{2i-1} \\ 0, & V_{i,2} = V_{2i}, \end{cases}$$

then,

$$d_{V_{i,2}} = d_{2n+1}$$

$$-\left(\sum_{k=i}^{n-1} (n-k)b + (n-i+1)\delta + \phi(i)((n-i)\delta_i + \varepsilon\delta_i)\right).$$

Therefore,

$$\sum_{i=1}^{n} T_{V_{i,2}}(\pi) = nC - \sum_{i=1}^{n-1} (n-i) p_{V_{i,2}} - \sum_{i=1}^{n} \left(d_{2n+1} \right.$$

$$\left. -\left(\sum_{k=i}^{n-1} (n-k)b + (n-i+1)\delta + \phi(i)((n-i)\delta_i + \varepsilon\delta_i)\right)\right).$$

The problem

$$\min_{\pi} F(\pi) = \min\left(\sum_{i=1}^{n} T_{V_{i,2}}(\pi) + T_{V_{2n+1}}(\pi)\right)$$

is reduced to the problem $\max\Phi$, where

$$\Phi = \sum_{i=1}^{n-1} (n-i) p_{V_{i,2}}$$

$$-\sum_{i=1}^{n} \phi(i)((n-i)\delta_i + \varepsilon\delta_i) - T_{V_{2n+1}}(\pi).$$

(1) If $V_{i,2} = V_{2i}$, $i = 1, \ldots, n$, then

$$T_{V_{2n+1}}(\pi) = \frac{1}{2}\delta, \quad \Phi_1 = \sum_{i=1}^{n-1} (n-i) p_{2i} - \frac{1}{2}\delta.$$

(2) If $V_{i,2} = V_{2i-1}$, $i = 1, \ldots, n$, then

$$T_{V_{2n+1}}(\pi) = \max\left\{-\frac{1}{2}\delta, 0\right\} = 0,$$

$$\Phi = \sum_{i=1}^{n-1} (n-i) p_{2i-1} - \sum_{i=1}^{n} ((n-i)\delta_i + \varepsilon\delta_i)$$

$$= \sum_{i=1}^{n-1} (n-i) p_{2i} + \sum_{i=1}^{n-1} (n-i)\delta_i$$

$$-\sum_{i=1}^{n} ((n-i)\delta_i + \varepsilon\delta_i) = \Phi_1 + \frac{1}{2}\delta - \sum_{i=1}^{n} \varepsilon\delta_i.$$

The function Φ has the maximal value

$$\Phi_1 + \frac{1}{2}\delta - \frac{1}{2}\sum_{i=1}^{n} \varepsilon\delta_i,$$

when

$$\sum_{i=1}^{n} \phi(i)(\varepsilon\delta_i) = \frac{1}{2}\sum_{i=1}^{n} \varepsilon\delta_i,$$

therefore,

$$\sum_{i=1}^{n} \phi(i)\delta_i = \frac{1}{2}\sum_{i=1}^{n} \delta_i.$$

Hence, for the modified problem, there exist two subsets A_1 and A_2 such that

$$\sum_{a_i \in A_1} a_i = \sum_{a_i \in A_2} a_i$$

(the modified EOP problem has a solution). Here, we have $C_{2n+1}(\pi) = d_{2n+1}$.

If the modified EOP problem does not have a solution, then

$$\sum_{i=1}^{n} \phi(i)\delta_i = \frac{1}{2}\sum_{i=1}^{n} \delta_i$$

does not hold. Taking into account the value d_{2n+1}, we have $C_{n+1}(\pi) \neq d_{2n+1}$.

If $C_{2n+1}(\pi) = d_{2n+1}$, then

$$\sum_{i=1}^{n} p_{V_{i,1}} = \sum_{i=1}^{n} p_{V_{2i}} + \frac{1}{2}\delta = \sum_{i=1}^{n} p_{V_{i,2}},$$

therefore, the modified EOP problem also has a solution.

4. CONCLUSIONS

In conclusion, we note that there exists a pseudopolynomial algorithm with $O(n\sum p_j)$ run time that solves cases (2.1) and (2.2) [8]. To solve canonical DL-instances [2] and case (2.1), we proposed an exact algorithm **B-1** with $O(n\sum p_j)$ run time. For the special case (2.2), there exists a pseudopolynomial algorithm *B-1 canonical* with $O(n\delta)$ run time. The algorithm *B-1 modified* is able to solve instances when $1 \,\|\, \sum T_j$, so we can find a solution for the noninteger EOP problem.

REFERENCES

1. M. R. Garey and D. S. Johnson, *Computers and Intractability: The Guide to the Theory of NP-Completeness* (Freeman, San Francisco, 1979; Mir, Moscow, 1982).
2. J. Du and J. Y. Leung, "T. Minimizing Total Tardiness on One Processor is NP-Hard," Math. Oper. Res, No. 15 (1990).
3. E. L. Lawler, "A Pseudopolynomial Algorithm for Sequencing Jobs To Minimize Total Tardiness," Ann. Discrete Math, No. 1 (1977).
4. W. Szwarc, F. Della Croce, and A. Grosso, "Solution of the Single Machine Total Tardiness Problem," J. Scheduling, No. 2 (1999).
5. W. Szwarc, A. Grosso, and F. Della Croce, "Algorithmic Paradoxes of the Single-Machine Total Tardiness Problem," J. Scheduling, No. 4 (2001).
6. C. N. Potts and L. N. Van Wassenhove, "A Decomposition Algorithm for the Single-Machine Total Tardiness Problem," Oper. Res. Lett, No. 1 (1982).
7. Croce F. Della, A. Grosso, and V. Paschos, "Lower Bounds on the Approximation Ratios of Leading Heuristics for the Single-Machine Total Tardiness Problem," J. of Scheduling, No. 7 (2004).
8. A. Lazarev, A. Kvaratskhelia, and A. Tchernykh, "Solution Algorithms for the Total Tardiness Scheduling Problem on a Single Machine," in *Workshop Proceedings of the ENC'04 International Conference, Colima, Mexico, 2004*, pp. 474–480.
9. S. Chang, Q. Lu, G. Tang, and W. Yu, "On Decomposition of Total Tardiness Problem," Oper. Res. Lett, **17**, 221–229 (1995).