www.czasopisma.pan.pl www.journals.pan.pl

BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES
Vol. 55, No. 1, 2007

Dominance relations for two-machine flow shop problem with late work criterion

M. STERNA*

Institute of Computing Science, Poznań University of Technology, 2 Piotrowo St., 60-965 Poznań, Poland

Abstract. The paper concerns the two-machine non-preemptive flow shop scheduling problem with a total late work criterion and a common due date ($F2|d_j = d|Y$). The late work performance measure estimates the quality of a solution with regard to the duration of late parts of activities performed in the system, not taking into account the quantity of this delay. In the paper, a few theorems are formulated and proven, describing features of an optimal solution for the problem mentioned, which is NP-hard. These theorems can be used in exact exponential algorithms (as dominance relations reducing the number of solutions enumerated explicitly), as well as in heuristic and metaheuristic methods (supporting the construction of sub-optimal schedules of a good quality).

Key words: scheduling, flow shop, late work criterion, dominance relation.

1. Introduction

The performance measure is a crucial component of the definition of any scheduling problem (cf. e.g. [1–3]). The viewpoint from which the quality of a schedule is estimated, i.e. the objective function, usually determines the strategy of searching for this solution. In most practical problems arising in different domains, the problem analysis has to take into account various time restrictions. This makes due date involving criteria, such as the late work one, especially interesting subject of the research.

The late work performance measure, investigated in this paper, evaluates schedules with regard to the number of tardy units of particular activities executed in a system. It can be considered as a special case of the imprecise computation model (cf. e.g. [4]), in which a job is divided in the mandatory and optional part. In a feasible schedule, the mandatory part of a job has to be early, while the optional part can be late, but the more its units is tardy the worse solution is obtained. The late work scheduling corresponds to the situation when the whole processing time is optional.

Such a concept of an objective function finds practical applications, for example, in the process of collecting data in control systems, where the amount of information exposed after a due date influences the accuracy of a steering algorithm. Moreover, the late work criterion can be considered in industrial systems, where late parts of customer orders should be minimized. It supports also production planning in a certain time horizon, since it allows reducing the amount of work not assigned to a particular time slot, which has to be considered in the following one. Furthermore, the late work concept can be applied in agriculture to optimize harvesting and land cultivation processes. In the first case, the amount of wasted crop, in the latter one, the amount of not spread fertilizers or plant protection substances should be minimized.

The late work concept was introduced in the context of the scheduling problem on identical parallel machines [5] and, next, it was applied to uniform [6] and single [7–15] machine(s) cases. Then, the late work criteria were analyzed in the shop environment, i.e. in systems with dedicated machines [16–23], starting from the simplest two-machine models with a common due date. All cases with two machines and the weighted late work criterion appeared to be binary NP-hard [24] (i.e. $O2|d_j = d|Y_w$ [17], $F2|d_j = d|Y_w$ [18] and $J2|d_j = d, n_j \leq 2|Y_w$ [19]). The two-machine open shop without weights ($O2|d_j = d|Y$ [17]) is solvable in polynomial time. The two machine flow shop problem with a common due date and the total late work criterion, $F2|d_j = d|Y$, being the subject of this paper, has been recently classified as binary NP-hard. Its intractability results from the transformation from the partition problem [25]. Moreover, it is possible to apply pseudo-polynomial time dynamic programming developed for the weighted case $F2|d_j = d|Y_w$ [18]. Since the job shop case is a generalization of the flow shop system (cf. e.g. [1–3]), $J2|d_j = d, n_j \leq 2|Y_w$, is also computationally hard.

In the paper, the features of an optimal solution for problem $F2|d_j = d|Y$ are presented, which can be helpful in constructing efficient exact and heuristic approaches solving it. In Section 2, the formal definition of the case under consideration is given, together with the description of a general structure of an optimal solution of this problem. In Section 3, three dominance relations are formulated and proven, which support the process of construct-

*e-mail: malgorzata.sterna@cs.put.poznan.pl

ing an optimal schedule for $F2|d_j = d|Y$. Conclusions and further research directions are given in Section 4.

2. Problem definition

The two machine flow shop problem with the late work criterion and a common due date, $F2|d_j = d|Y$, concerns scheduling of a set of jobs $J = \{J_1, \ldots, J_j, \ldots, J_n\}$ on two dedicated machines M_1 and M_2. Each job J_j has to be performed first on machine M_1 and then on M_2 for p_{1j} and p_{2j} time units, respectively. Hence, one can consider a job as a sequence of two tasks, for which a precedence constraint is defined. Each machine can process only one job at a time and each job can be executed by only one machine at a time. Moreover, a common due date d is defined for all jobs in the system. The goal is to construct a feasible non-preemptive schedule for which the total late work is minimal, i.e. the amount of work executed after a given common due date d is minimal.

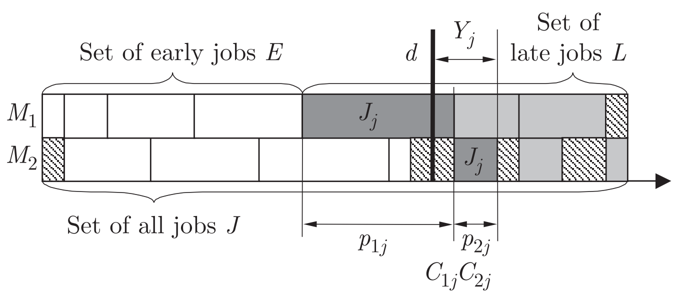

Fig. 1. The late work Y_j for job J_j in an exemplary solution for $F2|d_j = d|Y$

Denoting by C_{ij} the completion time of job J_j on machine M_i, the late work Y_j for this job is determined as (cf. Fig. 1):

$$Y_j = \sum_{i=1,2} \min\{\max\{0, C_{ij} - d\}, p_{ij}\}. \quad (1)$$

The criterion value to be minimized, estimating the quality of a complete schedule for the whole set of n jobs, is calculated as

$$Y = \sum_{j=1}^{n} Y_j. \quad (2)$$

The late work criterion represents slightly different point of view than other classical performance measures. If an activity is scheduled totally late, then the quantity of its delay does not influence the criterion value (on the contrary to the tardiness or lateness for example). Minimizing this performance measure, one tends to execute as many units of work before a common due date as possible, which is equivalent to minimizing idle time before d [18,23]. Such a strategy usually does not result in a schedule of the minimal length. Since early jobs are selected in order to fill the gap between time zero and d in the best way and jobs scheduled after a common due date are not important for the objective function and are executed in an arbitrary order, the whole solution can have quite large makespan. On the contrary, all early activities have to be processed in the order that minimizes the schedule length [18,23]. Once it is decided which jobs are executed before a common due date, the set of such jobs should be executed as soon as possible, i.e. in Johnson's order [26].

Johnson's algorithm, designed for problem $F2||C_{\max}$, divides the set of jobs J into two subsets: J^1 containing jobs $J_j \in J$ with $p_{1j} \le p_{2j}$ and J^2 built by jobs $J_j \in J$ with $p_{1j} > p_{2j}$. In order to minimize the schedule length, jobs from set J^1 should be sequenced in non-decreasing order of p_{1j} and precede jobs from J^2 processed in non-increasing order of p_{2j}.

Summing up, solving problem $F2|d_j = d|Y$, one has to divide the set of all jobs J into two subsets $J = E \cup L$ (cf. Fig. 1), where E contains early jobs, while late jobs (possibly partially) belong to L. Jobs from E have to be processed in Johnson's order, while the first late job in L (job J_j in Fig. 1) has to be chosen in the way minimizing idle time before d. The sequence of the remaining late jobs from set L is arbitrary.

Although the general structure of an optimal solution of problem $F2|d_j = d|Y$, described above, is quite strict, this scheduling case is NP-hard [24, 25]. The crucial decision, difficult from the viewpoint of the computational complexity, is to select the first late job in the system and, first of all, to divide the set of jobs into two subsets of early and late activities. In the paper, some dominance relations are formulated and proven, which support the process of constructing this division of the set of jobs.

3. Dominance relations

Since problem $F2|d_j = d|Y$ is NP-hard, a polynomial time method solving it optimally is rather unlikely to exist [24]. Exact approaches of the exponential time complexity, such as a branch and bound algorithm for example, can be efficiently applied only for small problem instances. Exact methods have to explore the whole solution space in the search for an optimal schedule, but not every possible sequence of jobs is analyzed explicitly. The efficiency of such algorithms depends on their ability to discard partial solutions which do not lead to the optimal one. The careful analysis of the structure of an optimal schedule often results in formulating dominance relations, which justify discarding some parts of the solution space. On the other hand, these relations make it possible to design good heuristic or metaheuristic approaches, concentrating their search in the most promising areas of the problem space.

The optimal solution for problem $F2|d_j = d|Y$ can be represented as a permutation of jobs. Thus, the solution space contains all permutations of the set of jobs. But, with regard to the fact that early jobs have to be executed in Johnson's order, all permutations of these activities which are not consistent with Johnson's order can be discarded. Obviously, the crucial decision is to select jobs to this set of early activities. Three theorems formulated and proven in the paper impose conditions, which may be taken into account in this decision process. They show that selecting jobs in non-decreasing order of processing times on the first machine is usually profitable from the late work point of view.

THEOREM 1. In some optimal solution of problem $F2|d_j = d|Y$, there is no pair of jobs J_k, $J_s \in J^1$ such that $J_s \in E \wedge J_k \in L \wedge p_{1k} \leq p_{1s}$.

Proof. Let assume that there exists an optimal solution π, for which the condition of Theorem 1 does not hold, i.e. there exist two jobs J_s, $J_k \in J^1$ (i.e. $p_{1s} \leq p_{2s}$ and $p_{1k} \leq p_{2k}$) such that $J_s \in E$ and $p_{1k} \leq p_{1s}$, but $J_k \in L$. Moreover, let assume, without the loss of generality, that J_s is the first job in the sequence of early jobs for which the theorem does not hold. There will be shown that processing job J_k early, one obtains a schedule π' satisfying the condition of Theorem 1 with the criterion value not worse than π.

If J_k is processed early, then it has to be executed directly before J_s in π', due to the assumption that J_s is the first job contradicting Theorem 1, as well as to the fact that all early jobs are sequenced in Johnson's order ($p_{1k} \leq p_{1s}$ and all jobs from J^1, including J_s and J_k, are scheduled in non-decreasing order of the processing times of their first tasks). Exemplary solutions π and π' are depicted in Fig. 2 (obviously, some partial schedules π_1, π_2, π_3 can be empty, as well as some idle times can be equal to zero).

Because minimizing late work is equivalent to minimizing idle time before a common due date d, to prove the theorem, it is enough to show that the total idle time before d does not increase as a result of shifting J_k before J_s and that no unit of idle time is shifted to the left (especially before a due date d).

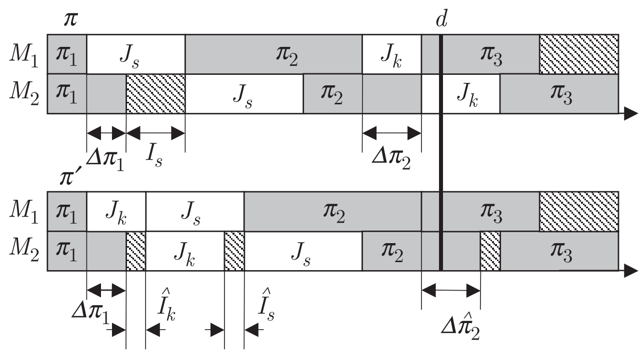

Fig. 2. Exemplary schedules π and π', where $J_k \in J^1$ is shifted before $J_s \in J^1$ ($p_{1k} \leq p_{1s}$)

Let's denote with I_1, I_2, I_3, I_s the amount of idle time within partial schedules π_1, π_2, π_3, and before job J_s in schedule π, respectively (if there is idle time before J_k in π, then it is included in I_2). The values $\hat{I}_1$, $\hat{I}_2$, $\hat{I}_3$, $\hat{I}_k$, $\hat{I}_s$ denote the corresponding idle times in schedule π'. In order to avoid a possible violation of precedence constraints, partial schedules π_2 and π_3 cannot be shifted to the left on M_2 in π' with regard to π (cf. e.g. subschedule π_3 in Fig. 2).

The total idle times for schedules π and π' are equal to $I = I_1 + I_s + I_2 + I_3$ and $\hat{I} = \hat{I}_1 + \hat{I}_k + \hat{I}_s + \hat{I}_2 + \hat{I}_3$. In consequence, the change of the total idle time, representing the change of the criterion value, is equal to

$$\begin{aligned} \Delta I = \hat{I} - I &= (\hat{I}_1 + \hat{I}_2 + \hat{I}_3 + \hat{I}_k + \hat{I}_s) \\ &- (I_1 + I_2 + I_3 + I_s) \\ &= (\hat{I}_1 - I_1) + (\hat{I}_2 - I_2) + (\hat{I}_3 - I_3) \\ &+ (\hat{I}_k + \hat{I}_s - I_s). \end{aligned} \tag{3}$$

Because the partial schedule π_1 is the same in π as in π', therefore $(I_1 = \hat{I}_1) \Rightarrow (\hat{I}_1 - I_1 = 0)$ and

$$\Delta I = (\hat{I}_2 - I_2) + (\hat{I}_3 - I_3) + (\hat{I}_k + \hat{I}_s - I_s) \tag{4}$$

where

$$I_s = \max\{0, p_{1s} - \Delta\pi_1\}, \tag{5}$$

$$\hat{I}_k = \max\{0, p_{1k} - \Delta\pi_1\}, \tag{6}$$

$$\hat{I}_s = \max\{0, p_{1k} + p_{1s} - \max\{\Delta\pi_1, p_{1k}\} - p_{2k}\}. \tag{7}$$

Since it is assumed that π_2 cannot be shifted to the left on M_2, the change of idle time for π_2 is equal to the change of the difference between schedule lengths on machines M_1 and M_2 before π_2 (i.e. the change of the schedule offset) in π' and π, this means that

$$\begin{aligned} (\hat{I}_2 - I_2) &= \{(p_{1k} + p_{1s}) - (\Delta\pi_1 + \hat{I}_k + p_{2k} + \hat{I}_s + p_{2s})\} \\ &- \{(p_{1s}) - (\Delta\pi_1 + I_s + p_{2s})\} \\ &= p_{1k} - \hat{I}_k - p_{2k} - \hat{I}_s + I_s \\ &= (p_{1k} - p_{2k}) - (\hat{I}_k + \hat{I}_s - I_s). \end{aligned} \tag{8}$$

First, jobs J_k, J_s and a partial schedule π_2 will be considered, then a partial schedule π_3 will be analyzed.

Case 1. Let's assume that

$$\hat{I}_k = 0 \Rightarrow p_{1k} - \Delta\pi_1 \leq 0 \Rightarrow p_{1k} \leq \Delta\pi_1. \tag{9}$$

Eq. (7) and (9) result in

$$\begin{aligned} \hat{I}_s &= \max\{0, p_{1k} + p_{1s} - \max\{\Delta\pi_1, p_1k\} - p_2k\} \\ &= \max\{0, p_{1k} + p_{1s} - \Delta\pi_1 - p_2k\} \\ &= \max\{0, (p_{1k} - p_{2k}) + (p_{1s} - \Delta\pi_1)\}. \end{aligned} \tag{10}$$

Subcase 1. Let's assume that

$$I_s = 0 \Rightarrow p_{1s} - \Delta\pi_1 \leq 0. \tag{11}$$

Taking into account (11) and $J_k \in J^1 \Rightarrow p_{1k} - p_{2k} \leq 0$, Eq. (10) reduces to $\hat{I}_s = \max\{0, (p_{1k} - p_{2k}) + (p_{1s} - \Delta\pi_1)\} = 0$. In consequence, taking into account (9) and (11), there is $(\hat{I}_k + \hat{I}_s - \hat{I}_s) = 0$. This means that no new idle time appears in a partial schedule before J_s in π'. Moreover, from (8) results $(\hat{I}_2 - I_2) = (p_{1k} - p_{2k}) - (\hat{I}_k + \hat{I}_s - I_s) = (p_{1k} - p_{2k}) \leq 0$. It denotes that the schedule offset between M_1 and M_2 before π_2 can only increase in π' in comparison to π, possibly reducing the internal idle time within π_2.

Subcase 2. Let's assume that

$$I_s > 0 \Rightarrow p_{1s} - \Delta\pi_1 > 0. \tag{12}$$

Eq. (5) and (12) result in

$$I_s = p_{1s} - \Delta\pi_1, \tag{13}$$

while (10) and (13) lead to

$$\begin{aligned}\hat{I}_s &= \max\{0, (p_{1k} - p_{2k}) + (p_{1s} - \Delta\pi_1)\}\\ &= \max\{0, p_{1k} - p_{2k} + I_s\}.\end{aligned} \tag{14}$$

First, let's assume that

$$\hat{I}_s = 0 \Rightarrow p_{1k} - p_{2k} + I_s \le 0. \tag{15}$$

Equations (9), (12) and (15) lead to $(\hat{I}_k + \hat{I}_s - I_s) = I_s \le 0$ and Eq. (8) reduces to $(\hat{I}_2 - I_2) = (p_{1k} - p_{2k}) - (\hat{I}_k + \hat{I}_s - I_s) = p_{1k} - p_{2k} + I_s \le 0$.

Then, let's assume that

$$\hat{I}_s > 0 \Rightarrow p_{1k} - p_{2k} + I_s > 0. \tag{16}$$

Equations (14) and (16) denote that $\hat{I}_s = p_{1k} - p_{2k} + I_s$. Combining this observation with (9) and then with (8), one obtains $(\hat{I}_k + \hat{I}_s - I_s) = 0 + (p_{1k} - p_{2k} + I_s) - I_s = p_{1k} - p_{2k} \le 0$ and $(\hat{I}_2 - I_2) = (p_{1k} - p_{2k}) - (\hat{I}_k + \hat{I}_s - I_s) = (p_{1k} - p_{2k}) - (p_{1k} - p_{2k}) = 0$, respectively.

Regardless of the value $\hat{I}_s$ in Subcase 2, idle time in a partial schedule before J_s cannot increase, because job J_k may only reduce idle time before J_s $(\hat{I}_k + \hat{I}_s - I_s \le 0)$. Moreover, the schedule offset before π_2 cannot become smaller in π' than in π $((\hat{I}_2 - I_2) \le 0)$, and, in consequence, the internal idle time within π_2 cannot increase (actually, it can be possibly reduced, if $(\hat{I}_2 - I_2) < 0$).

Case 2. Let's assume that

$$\hat{I}_k > 0 \Rightarrow p_{1k} - \Delta\pi_1 > 0 \Rightarrow p_{1k} > \Delta\pi_1. \tag{17}$$

Eq. (6) and (17) imply

$$\hat{I}_k = p_{1k} - \Delta\pi_1. \tag{18}$$

Because $p_{1s} \ge p_{1k}$, from Eq. (17) there is $p_{1s} > \Delta\pi_1$ and from (5)

$$I_s = \max\{0, p_{1s} - \Delta\pi_1\} = p_{1s} - \Delta\pi_1 > 0. \tag{19}$$

From Eq. (7) and (17), there is

$$\begin{aligned}\hat{I}_s &= \max\{0, p_{1k} + p_{1s} - \max\{\Delta\pi_1, p_{1k}\} - p_{2k}\}\\ &= \max\{0, p_{1k} + p_{1s} - p_{1k} - p_{2k}\}\\ &= \max\{0, p_{1s} - p_{2k}\}.\end{aligned} \tag{20}$$

Subcase 1. Let's assume that

$$\hat{I}_s = 0 \Rightarrow p_{1s} - p_{2k} \le 0. \tag{21}$$

Eq. (18), (19) and (21) lead to $(\hat{I}_k + \hat{I}_s - I_s) = (p_{1k} - \Delta\pi_1) - (p_{1s} - \Delta\pi_1) = p_{1k} - p_{1s} \le 0$ and reduce (8) to $(\hat{I}_2 - I_2) = (p_{1k} - p_{2k}) - (\hat{I}_k + \hat{I}_s - I_s) = (p_{1k} - p_{2k}) - (p_{1k} - p_{1s}) = p_{1s} - p_{2k} \le 0$.

Subcase 2. Let's assume that

$$\hat{I}_s > 0 \Rightarrow p_{1s} - p_{2k} > 0 \Rightarrow \hat{I}_s = p_{1s} - p_{2k}. \tag{22}$$

Eq. (18), (19) and (22) lead to $(\hat{I}_k + \hat{I}_s - I_s) = (p_{1k} - \Delta\pi_1) + (p_{1s} - p_{2k}) - (p_{1s} - \Delta\pi_1) = p_{1k} - p_{2k} \le 0$ and reduce (8) to $(\hat{I}_2 - I_2) = (p_{1k} - p_{2k}) - (\hat{I}_k + \hat{I}_s - I_s) = (p_{1k} - p_{2k}) - (p_{1k} - p_{2k}) = 0$.

Again, in both subcases of Case 2, idle time in a schedule before J_s cannot increase $(\hat{I}_k + \hat{I}_s - I_s \le 0)$ and the schedule offset before π_2 cannot decrease in π' with regard to π $((\hat{I}_2 - I_2) \le 0)$.

To complete the proof, a partial schedule π_3 will be considered. From the case study presented above it results that $(\hat{I}_2 - I_2) \le 0$. This means that the schedule offset between M_1 and M_2 before π_2 can only increase in π' with regard to π or it remains unchanged. In the latter case the schedule offset after π_2 does not change too, $\Delta\pi_2 = \Delta\hat{\pi}_2$. In the first case, the larger offset before π_2 may reduce the internal idle time in this subschedule (if any) and, then, increase the offset between machine M_1 and M_2 after π_2, $\Delta\pi_2 \le \Delta\hat{\pi}_2$. Summing up, there is $\Delta\pi_2 \le \Delta\hat{\pi}_2$.

Similarly, as for π_2, a partial schedule π_3 cannot be shifted to the left on M_2 in π' and, in consequence, $(\hat{I}_3 - I_3)$ is equal to the change of the difference in the schedule offsets between machines M_1 and M_2 in π' and π, i.e.:

$$\begin{aligned}(\hat{I}_3 - I_3) &= \{-\Delta\hat{\pi}_2\} - \{(p_{1k}) - (\Delta\pi_2 + p_{2k})\}\\ &= -(p_{1k} - p_{2k}) + \Delta\pi_2 - \Delta\hat{\pi}_2.\end{aligned} \tag{23}$$

Combining (23) with (8), the total idle time change (4) is equal to

$$\begin{aligned}\Delta I &= (\hat{I}_2 - I_2) + (\hat{I}_3 - I_3) + (\hat{I}_k + \hat{I}_s - I_s)\\ &= (p_{1k} - p_{2k}) - (\hat{I}_k + \hat{I}_s - I_s) + (\hat{I}_3 - I_3)\\ &\quad + (\hat{I}_k + \hat{I}_s - I_s)\\ &= (p_{1k} - p_{2k}) + (\hat{I}_3 - I_3)\\ &= (p_{1k} - p_{2k}) - (p_{1k} - p_{2k}) + \Delta\pi_2 - \Delta\hat{\pi}_2\\ &= \Delta\pi_2 - \Delta\hat{\pi}_2 \le 0.\end{aligned} \tag{24}$$

This means that the total idle time cannot increase in consequence of executing job J_k before J_s. The possible idle time which may appear in π' before a subschedule π_3 (if $(\hat{I}_3 - I_3) > 0$, cf. Fig. 2) is a result of shifting some idle time units to the right. The total number of idle time units does not increase; some of them can be shifted from the left to the right part of a schedule. If this shift is large enough, some idle time units can be moved after a common due date, causing an additional decrease of the late work in the system.

Summing up, it is shown that $\Delta I \le 0$ and no unit of idle time is shifted to the left as a result of a schedule modification. Executing job J_k before J_s leads to the new schedule π', which is not worse than π with regard to the total late work value. This analysis can be repeated for the remaining pairs of jobs J_s, $J_k \in J^1$ such that $J_s \in E \wedge J_k \in L \wedge p_{1k} \le p_{1s}$ proving Theorem 1 (note that Theorem 1 holds, if J_s is early as well as it is late in π').

THEOREM 2. In some optimal solution of problem $F2|d_j = d|Y$, there is no pair of jobs $J_k \in J^1$, $J_s \in J^2$ such that $J_s \in E \wedge J_k \in L \wedge p_{1k} \le p_{1s}$.

Proof. Let's assume that there exists an optimal solution π, for which the condition of Theorem 2 does not hold, i.e. there exist two jobs $J_s \in J^2$ (i.e. $p_{1s} > p_{2s}$), $J_k \in J^1$ (i.e. $p_{1k} \le p_{2k}$), such that $J_s \in E$ and $p_{1k} \le p_{1s}$, but $J_k \in L$. Moreover, let assume without the loss of generality, that J_s is the first job in the sequence of early jobs for which the theorem does not hold. It will be shown that processing job J_k early, one obtains a schedule π' satisfying the condition of Theorem 2 with the criterion value not worse than π. Taking into account the assumptions that $J_k \in J^1$ and $J_s \in J^2$, if J_k is early, then it has to be processed before J_s in π', because all early jobs from set J^1 (including J_k) precede all early jobs from set J^2 (including J_s) in Johnson's order. Exemplary solutions π and π' are depicted in Fig. 3 (obviously, some partial schedules $\pi_1, \pi_2, \pi_3, \pi_4$ can be empty as well as some idle times can be equal to zero). To prove the theorem, it is enough to show that the total idle time does not increase as a result of shifting J_k before J_s and that no unit of idle time is shifted to the left (before a common due date d).

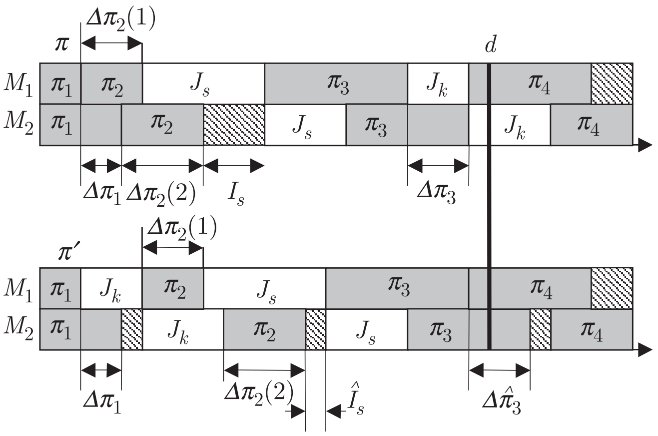

Fig. 3. Exemplary schedules π and π', where $J_k \in J^1$ is shifted before $J_s \in J^2 (p_{1k} \le p_{1s})$

Let's denote with I_1, I_2, I_3, I_4, I_s the amount of idle time within partial schedules $\pi_1, \pi_2, \pi_3, \pi_4$ and before job J_s in schedule π (a possible idle time before J_k is included in I_3). The values $\hat{I}_1, \hat{I}_2, \hat{I}_3, \hat{I}_4, \hat{I}_s, \hat{I}_k$ denote the corresponding idle times in schedule π'. The total idle times for schedules π and π' are equal to $I = I_1 + I_s + I_2 + I_3 + I_4$ and $\hat{I} = \hat{I}_1 + \hat{I}_k + \hat{I}_2 + \hat{I}_s + \hat{I}_3 + \hat{I}_4$ respectively. In consequence, the total idle time difference between π and π' is equal to:

$$\Delta I = (\hat{I}_1 - I_1) + (\hat{I}_2 - I_2) + (\hat{I}_3 - I_3) + (\hat{I}_4 - I_4) + (\hat{I}_k + \hat{I}_s - I_s). \quad (25)$$

Because the partial schedule π_1 does not change in π' in comparison to π, there is $(I_1 = \hat{I}_1) \Rightarrow (\hat{I}_1 - I_1 = 0)$ and Eq. (25) leads to

$$\Delta I = (\hat{I}_2 - I_2) + (\hat{I}_3 - I_3) + (\hat{I}_4 - I_4) + (\hat{I}_k + \hat{I}_s - I_s). \quad (26)$$

Similarly as in the proof for Theorem 1, in order to avoid a possible violation of precedence constraints within jobs, the partial schedules π_2, π_3, π_4 cannot be shifted to the left on M_2 in π' with regard to π (cf. e.g. subschedule π_4 in Fig. 3). In consequence, the possible change in idle time within π_2, $(\hat{I}_2 - I_2)$, is equal to the change of the offsets between schedules on machines M_1 and M_2 in π' and π, respectively, i.e.:

$$\begin{aligned}(\hat{I}_2 - I_2) &= \{p_1k - (\Delta\pi_1 + \hat{I}_k + p_2k)\} - \{-\Delta\pi_1\} \\ &= p_{1k} - p_{2k} - \hat{I}_k.\end{aligned} \quad (27)$$

Taking into account the fact that $J_k \in J^1 \Rightarrow p_{1k} \le p_{2k} \Rightarrow p_{1k} - p_{2k} \le 0$, as well as the fact that an idle time value cannot be negative, $\hat{I}_k \ge 0 \Rightarrow -\hat{I}_k \le 0$, Eq. (27) and (26) lead to

$$(\hat{I}_2 - I_2) \le 0, \quad (28)$$

$$\Delta I \le (\hat{I}_3 - I_3) + (\hat{I}_4 - I_4) + (\hat{I}_k + \hat{I}_s - I_s). \quad (29)$$

Equation (28) implies that the offset between partial schedules on M_1 and M_2 before π_2 cannot decrease in π' in comparison to π. In consequence, no additional idle time appears before the first job of π_2 in π' and the internal idle time within π_2 (if any) can be only reduced.

Moreover, the change of the schedule offset before a subschedule π_3, i.e. the possible idle time change for π_3, can be determined as

$$\begin{aligned}(\hat{I}_3 - I_3) &= \{(p_{1k} + \pi_2(1) + p_{1s}) \\ &\quad - (\Delta\pi_1 + \hat{I}_k + p_{2k} + \pi_2(2) + \hat{I}_s + p_{2s})\} \\ &\quad - \{(\pi_2(1) + p_{1s}) - (\Delta\pi_1 + \pi_2(2) + I_s + p_{2s})\} \\ &= (p_{1k} - p_{2k}) - (\hat{I}_k + \hat{I}_s - I_s),\end{aligned} \quad (30)$$

where $\pi_2(i)$ denotes the length of a partial schedule π_2 on machine M_i and

$$\hat{I}_k = \max\{0, p_{1k} - \Delta\pi_1\}, \quad (31)$$

$$\begin{aligned}\hat{I}_s &= \max\{0, p_{1k} + \pi_2(1) + p_{1s} - \Delta\pi_1 - \hat{I}_k \\ &\qquad - p_{2k} - \pi_2(2)\} \\ &= \max\{0, (p_{1k} - p_{2k}) - \hat{I}_k + (\pi_2(1) + p_{1s} \\ &\qquad - \Delta\pi_1 - \pi_2(2))\},\end{aligned} \quad (32)$$

$$I_s = \max\{0, \pi_2(1) + p_{1s} - \Delta\pi_1 - \pi_2(2)\}. \quad (33)$$

To prove that the change of the schedule offset before π_3, $(\hat{I}_3 - I_3)$, cannot be positive and to show that no unit of idle time is shifted to the left, the following two cases have to be considered.

Case 1. Let's assume that

$$\hat{I}_k = 0 \Rightarrow p_{1k} - \Delta\pi_1 \le 0 \Rightarrow p_{1k} \le \Delta\pi_1. \quad (34)$$

Eq. (34) and (32) imply

$$\begin{aligned}\hat{I}_s &= \max\{0, (p_{1k} - p_{2k}) - \hat{I}_k + (\pi_2(1) \\ &\quad + p_{1s} - \Delta\pi_1 - \pi_2(2))\} \\ &= \max\{0, (p_{1k} - p_{2k}) + (\pi_2(1) + p_{1s} \\ &\quad - \Delta\pi_1 - \pi_2(2))\}.\end{aligned} \quad (35)$$

Subcase 1. Let's assume that

$$I_s = 0 \Rightarrow \pi_2(1) + p_{1s} - \Delta\pi_1 - \pi_2(2) \le 0. \quad (36)$$

From Eq. (36) and $p_{1k} - p_{2k} \le 0$, Eq. (35) reduces to $\hat{I}_s = \max\{0, (p_{1k}-p_{2k})+(\pi_2(1)+p_{1s}-\Delta\pi_1-\pi_2(2))\} = 0$. Combining this observation with (34) and (36), one obtains $(\hat{I}_k+\hat{I}_s-I_s) = 0$, and Eq. (30) reduces to $(\hat{I}_3-I_3) = (p_{1k}-p_{2k}) - (\hat{I}_k+\hat{I}_s-I_s) = (p_{1k}-p_{2k}) \le 0$.

Subcase 2. Let's assume that

$$I_s > 0 \Rightarrow \pi_2(1) + p_{1s} - \Delta\pi_1 - \pi_2(2) > 0. \quad (37)$$

Equations (33) and (37) denote that

$$I_s = \pi_2(1) + p_{1s} - \Delta\pi_1 - \pi_2(2), \quad (38)$$

while Eq. (35) and (38) lead to

$$\begin{aligned}\hat{I}_s &= \max\{0, (p_{1k}-p_{2k}) + (\pi_2(1)+p_{1s}-\Delta\pi_1-\pi_2(2))\}\\ &= \max\{0, (p_{1k}-p_{2k}) + I_s\}.\end{aligned} \quad (39)$$

First, let's assume that

$$\hat{I}_s = 0 \Rightarrow (p_{1k}-p_{2k}) + I_s \le 0. \quad (40)$$

Eq. (34), (37) and (40) lead to $(\hat{I}_k+\hat{I}_s-I_s) = 0+0-I_s < 0$, and reduce Eq. (30) to $(\hat{I}_3-I_3) = (p_{1k}-p_{2k}) - (\hat{I}_k+\hat{I}_s-I_s) = (p_{1k}-p_{2k}) + I_s \le 0$.

Then, let's assume that

$$\hat{I}_s > 0 \Rightarrow (p_{1k}-p_{2k}) + I_s > 0, \quad (41)$$

which is equivalent to $\hat{I}_s = (p_{1k}-p_{2k}) + I_s$. This observation combined with Eq. (34) results in $(\hat{I}_k+\hat{I}_s-I_s) = 0 + (p_{1k}-p_{2k}+I_s) - I_s = p_{1k}-p_{2k} \le 0$, and with Eq. (30) in $(\hat{I}_3-I_3) = (p_{1k}-p_{2k}) - (\hat{I}_k+\hat{I}_s-I_s) = (p_{1k}-p_{2k}) - (p_{1k}-p_{2k}) = 0$.

All subcases investigated within Case 1, showed that $(\hat{I}_3-I_3) \le 0$ and $(\hat{I}_k+\hat{I}_s-I_s) \le 0$. Since $(\hat{I}_3-I_3) \le 0$, the schedule offset before π_3 can only increase (or remain the same) in π' with regard to π, possibly reducing the internal idle time within π_3. Moreover, no new idle time appears between time zero and the end of subschedule π_3 in π', because in Eq. (25) there is $(\hat{I}_1-I_1) = 0$, $(\hat{I}_2-I_2) \le 0$, $(\hat{I}_3-I_3) \le 0$ and $(\hat{I}_k+\hat{I}_s-I_s) \le 0$.

Case 2. Let's assume that

$$\hat{I}_k > 0 \Rightarrow p_{1k} - \Delta\pi_1 > 0 \Rightarrow p_{1k} > \Delta\pi_1. \quad (42)$$

Equations (31) and (42) imply

$$\hat{I}_k = p_{1k} - \Delta\pi_1. \quad (43)$$

Subcase 1. Let's assume that

$$I_s = 0 \Rightarrow \pi_2(1) + p_{1s} - \Delta\pi_1 - \pi_2(2) \le 0. \quad (44)$$

Taking into account (42), (44) and $p_{1k} - p_{2k} \le 0$, Eq. (32) reduces to $\hat{I}_s = 0$ and, in consequence, Eq. (30) is reduced to $(\hat{I}_3-I_3) = (p_{1k}-p_{2k}) - (\hat{I}_k+\hat{I}_s-I_s) = (p_{1k}-p_{2k}) - (\hat{I}_k+0-0) \le 0$. This means that the schedule offset before π_3 cannot decrease in π' with regard to π, there is rather possible a reduction of an internal idle time within π_3. Moreover, idle time between time zero and the end of subschedule π_3 is not larger in π' than in π, because $(\hat{I}_1-I_1)+(\hat{I}_2-I_2)+(\hat{I}_k+\hat{I}_s-I_s)+(\hat{I}_3-I_3) = 0+(p_{1k}-p_{2k}-\hat{I}_k)+(\hat{I}_k+0-0)+(\hat{I}_3-I_3) = (p_{1k}-p_{2k})+(\hat{I}_3-I_3) \le 0$.

Subcase 2. Let's assume that

$$I_s > 0 \Rightarrow \pi_2(1) + p_{1s} - \Delta\pi_1 - \pi_2(2) > 0. \quad (45)$$

Equations (33) and (45) imply

$$I_s = \pi_2(1) + p_{1s} - \Delta\pi_1 - \pi_2(2), \quad (46)$$

while Eq. (32) and (46) result in

$$\begin{aligned}\hat{I}_s &= \max\{0, (p_{1k}-p_{2k}) - \hat{I}_k + (\pi_2(1)+p_{1s}\\ &\quad -\Delta\pi_1-\pi_2(2))\}\\ &= \max\{0, (p_{1k}-p_{2k}) - \hat{I}_k + I_s\}.\end{aligned} \quad (47)$$

First, let's assume that

$$\hat{I}_s = 0 \Rightarrow (p_{1k}-p_{2k}) + I_s - \hat{I}_k \le 0. \quad (48)$$

Equations (30) and (43), (46) and (48) result in

$$\begin{aligned}(\hat{I}_3-I_3) &= (p_{1k}-p_{2k}) - (\hat{I}_k+\hat{I}_s-I_s)\\ &= (p_{1k}-p_{2k}) - (\hat{I}_k+0-I_s)\\ &= (p_{1k}-p_{2k}) - (p_{1k}-\Delta\pi_1)\\ &\quad + (\pi_2(1)+p_{1s}-\Delta\pi_1-\pi_2(2))\\ &= \pi_2(1)+p_{1s}-p_{2k}-\pi_2(2).\end{aligned} \quad (49)$$

Because according to Eq. (42) there is idle time before J_k in $\pi'(\hat{I}_k > 0)$, Eq. (49) determines idle time before J_s in π', i.e. $\hat{I}_s$ (note, that there is assumed that a partial schedule π_2 cannot be shifted to the left on M_2 in π' in comparison to π). Eq. (48) states that $\hat{I}_s = 0$, and in consequence $(\hat{I}_3-I_3) = \hat{I}_s = 0$. This means that the schedule offset before π_3 is the same in π' as in π. Moreover, idle time between time zero and the end of subschedule π_3 is not larger in π' than in π, because $(\hat{I}_1-I_1)+(\hat{I}_2-I_2)+(\hat{I}_k+\hat{I}_s-I_s)+(\hat{I}_3-I_3) = 0+(p_{1k}-p_{2k}-\hat{I}_k)+(\hat{I}_k+0-I_s)+0 = p_{1k}-p_{2k}-I_s \le 0$.

Now, let's assume that

$$\hat{I}_s > 0 \Rightarrow p_{1k}-p_{2k}+I_s-\hat{I}_k > 0. \quad (50)$$

Equations (47) and (50) imply that

$$\hat{I}_s = p_{1k}-p_{2k}+I_s-\hat{I}_k. \quad (51)$$

Equations (51) results in $(\hat{I}_k+\hat{I}_s-I_s) = \hat{I}_k + p_{1k} - p_{2k} + I_s - \hat{I}_k - I_s = p_{1k}-p_{2k} \le 0$, and Eq. (30) reduces to $(\hat{I}_3-I_3) = (p_{1k}-p_{2k}) - (\hat{I}_k+\hat{I}_s-I_s) = (p_{1k}-p_{2k}) - (p_{1k}-p_{2k}) = 0$. In consequence, the idle time change between time zero and the end of subschedule π_3 can be estimated as $(\hat{I}_1-I_1)+(\hat{I}_2-I_2)+(\hat{I}_k+\hat{I}_s-I_s)+(\hat{I}_3-I_3) \le 0$.

All subcases investigated within Case 2, showed that the schedule offset before π_3 cannot decrease in π' with regard to π, because $(\hat{I}_3-I_3) \le 0$. Actually, the internal idle time within π_3 may become smaller, if $(\hat{I}_3-I_3) < 0$. Moreover, idle time between time zero and the end of subschedule π_3 in π' can be only reduced (or it remains the same) in π' in comparison to π, because $(\hat{I}_1-I_1)+(\hat{I}_2-I_2)+(\hat{I}_k+\hat{I}_s-I_s)+(\hat{I}_3-I_3) \le 0$.

Finally, it is necessary to show that the proposed modification of a schedule does not increase the criterion value within a partial schedule π_4.

Equations (29) and (30) result in

$$\Delta I \leq (p_{1k} - p_{2k}) - (\hat{I}_k + \hat{I}_s - I_s) + (\hat{I}_4 - I_4) + (\hat{I}_k + \hat{I}_s - I_s) = (p_{1k} - p_{2k}) + (\hat{I}_4 - I_4). \quad (52)$$

Similarly as for π_2, it is proven that the offset between partial schedules on M_1 and M_2 before π_3 cannot decrease (because $(\hat{I}_3 - I_3) \leq 0$) and, in consequence, no additional idle time before the first job of π_3 in π' appears. The offset increase before π_3 can only reduce an internal idle time within this subschedule (if any) or make the difference between the schedule lengths on M_1 and M_2 after π_3 larger in $\pi'(\Delta\hat{\pi}_3)$ than in π $(\Delta\pi_3)$, i.e. $\Delta\hat{\pi}_3 \geq \Delta\pi_3$.

In consequence, the change of the schedule offset before π_4, equivalent to the change of idle time, can be calculated as (note, that if there is an idle time before J_k in π, then it is included in I_3):

$$(\hat{I}_4 - I_4) = (-\Delta\hat{\pi}_3) - (p_{1k} - \Delta\pi_3 - p_{2k}) = (\Delta\pi_3 - \Delta\hat{\pi}_3) - (p_{1k} - p_{2k}). \quad (53)$$

Applying (53) to Eq. (52), one obtains

$$\Delta I \leq (p_{1k} - p_{2k}) + (\hat{I}_4 - I_4) = (p_{1k} - p_{2k}) + (\Delta\pi_3 - \Delta\hat{\pi}_3) - (p_{1k} - p_{2k}) = \Delta\pi_3 - \Delta\hat{\pi}_3 \leq 0. \quad (54)$$

Summing up, the total idle time in the system cannot increase, $\Delta I \leq 0$, and no unit of idle time is shifted to the left as a result of a schedule modification (actually, some idle time can be shifted to the right, before π_4, if $\hat{I}_4 - I_4 > 0$, cf. Fig. 3). Shifting job J_k before J_s leads to the schedule π' not worse than π with regard to the total late work value. The repeated analysis of the remaining pairs of jobs $J_s \in J^2$, $J_k \in J^1$ such that $J_s \in E \wedge J_k \in L \wedge p_{1k} \leq p_{1s}$, proves Theorem 2 (note that Theorem 2 holds if J_s is early as well as it is late in π').

Theorems 1 and 2 state that, if one decides to execute a certain job $J_s \in J$ early, then executing all jobs $J_k \in J^1$ with $p_{1k} \leq p_{1s}$ early leads to a schedule not worse than an original one.

Unfortunately, this rule cannot be applied for jobs $J_k \in J^2$ straightforwardly. If there is a solution with two jobs $J_s \in E$ and $J_k \in J^2 \cap L$, and J_k precedes J_s in Johnson's order, $J_k \to J_s$, (where $J_k \in J^2$ and $J_k \to J_s$ imply that $J_s \in J^2$ and $p_{2s} \leq p_{2k}$), then introducing job J_k into a sequence of early jobs may lead to a schedule worse than an original one. The quality of a schedule may deteriorate, if there is idle time before J_s in an original sequence (cf. Fig. 4), as well as if there is no idle time before this job (cf. Fig. 5).

Nevertheless, for such a pair of jobs J_k, $J_s \in J^2$ and $p_{2s} \leq p_{2k}$ a dominance relation analogous to Theorems 1 and 2 can be still formulated. In the case of activities from set J^2 it is profitable to interchange job J_s with J_k (Theorem 3) instead of shifting J_k before J_s (cf. Theorems 1 and 2).

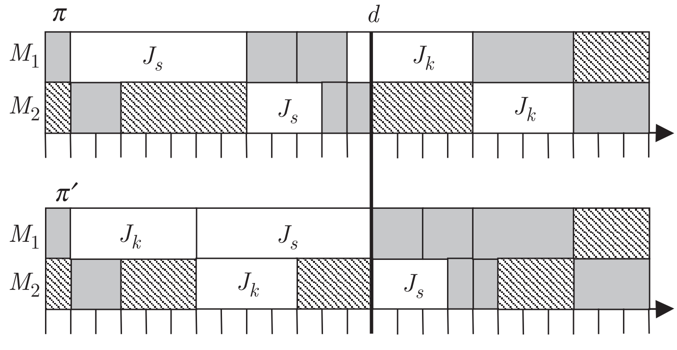

Fig. 4. Exemplary schedules π and π', where $J_k \in J^2$ is shifted before $J_s \in J^2(p_{1k} \leq p_{1s})$ and there is idle time before J_s in π

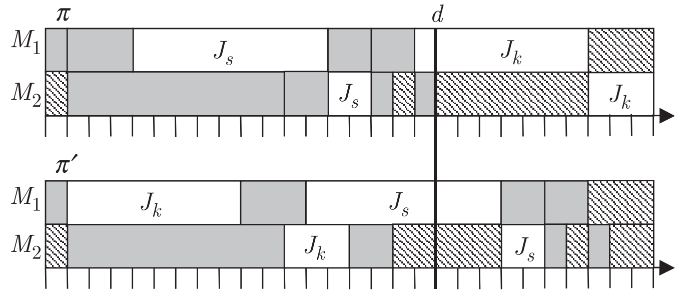

Fig. 5. Exemplary schedules π and π', where $J_k \in J^2$ is shifted before $J_s \in J^2(p_{1k} \leq p_{1s})$ and there is no idle time before J_s in π

THEOREM 3. In some optimal solution of problem $F2|d_j = d|Y$, there is no pair of jobs J_k, $J_s \in J^2$ such that $J_s \in E \wedge J_k \in L \wedge p_{1k} \leq p_{1s} \wedge p_{2s} \leq p_{2k}$.

Proof. Let's assume that there exists an optimal solution π, for which the condition of Theorem 3 does not hold, i.e. there exist two jobs J_k, $J_s \in J^2$, such that $J_s \in E$, $p_{1k} \leq p_{1s}$, $p_{2s} \leq p_{2k}$, but $J_k \in L$. Moreover, let assume that J_s is the first such a job in a sequence of early jobs. It will be shown that interchanging job J_s with J_k leads to a schedule π' satisfying the condition of Theorem 3 with the criterion value not worse than π.

With regard to the assumption that J_s is the first job contradicting Theorem 3, as well as to the fact that all early jobs are sequenced in Johnson's order, if J_k is early, then it starts in π' not later than J_s in π (all early jobs from J^2, are scheduled in non-increasing order of processing times of second tasks and $p_{2s} \leq p_{2k}$). There are two cases possible, that after interchanging job J_s with J_k, job J_k starts at the same time in π' as J_s in π (i.e. there is no early job from J^2 that is processed before J_s in π and after J_k in π', cf. Fig. 6) or J_k starts earlier in π' than J_s in π (i.e. there are some early jobs from J^2 that are processed before J_s in π and after J_k in π', cf. Fig. 7).

Case 1. Let's assume that J_s in π and J_k in π' starts at the same time. Exemplary solutions π and π' are depicted in Fig. 6. Some partial schedules π_1, π_3, π_4 can be empty as well as some idle times $I_s, I_k, \hat{I}_s$ and $\hat{I}_k$ can be equal to zero.

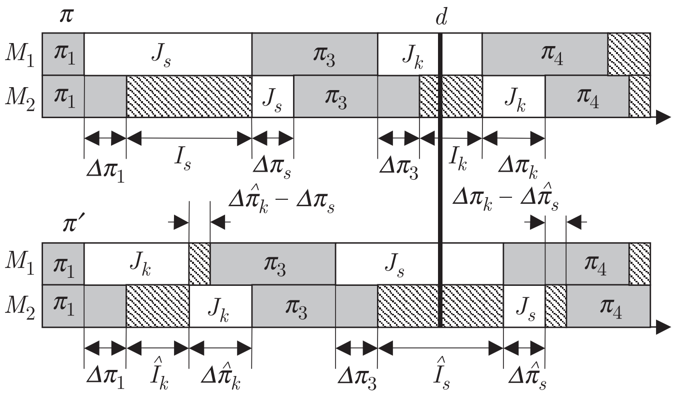

Fig. 6. Exemplary schedules π and π', where $J_k \in J^2$ is interchanged with $J_s \in J^2 (p_{1k} \le p_{1s} \wedge p_{2s} \le p_{2k})$ and there is no job executed before J_s in π and after J_k in π'

Let's assume at the beginning that the structures of subsequences π_3 and π_4 do not change after interchanging jobs J_s and J_k. This means that the offset between M_1 and M_2 is kept in both cases, as it is depicted in Fig. 6 (cf. π_3 on M_1 and π_4 on M_2). Hence, the criterion value is influenced only by the change of idle times before jobs J_s and J_k and by the change of the schedule offsets after these jobs before π_3 and π_4.

First, the schedule from time zero to the end of π_3 will be considered. The difference between idle time before J_s in π and J_k in π' is equal to

$$(I_s - \hat{I}_k) = \max\{0, p_{1s} - \Delta\pi_1\} - \max\{0, p_{1k} - \Delta\pi_1\}. \quad (55)$$

The assumption that $p_{1s} \ge p_{1k}$ implies that $I_s \ge \hat{I}_k$ and $(I_s - \hat{I}_k) \ge 0$. This means that idle time before J_k in π' is not larger than before J_s in π. Moreover, the schedule offset before π_3 changes by the value

$$\begin{aligned} \Delta\pi_s - \Delta\hat{\pi}_k &= (\max\{0, \Delta\pi_1 - p_{1s}\} + p_{2s}) \\ &\quad - (\max\{0, \Delta\pi_1 - p_{1k}\} + p_{2k}) \\ &= \max\{0, \Delta\pi_1 - p_1 s\} \\ &\quad - \max\{0, \Delta\pi_1 - p_{1k}\} - (p_{2k} - p_{2s}). \end{aligned} \quad (56)$$

Since $p_{1s} \ge p_{1k}$, there is, in Eq. (56), $\max\{0, \Delta\pi_1 - p_{1s}\} \le \max\{0, \Delta\pi_1 - p_{1k}\} \Rightarrow \max\{0, \Delta\pi_1 - p_{1s}\} - \max\{0, \Delta\pi_1 - p_{1k}\} \le 0$. Because J_s succeeds J_k in Johnson's order, there is $p_{2s} \le p_{2k} \Rightarrow p_{2s} - p_{2k} \le 0 \Rightarrow -(p_{2k} - p_{2s}) \le 0$ and Eq. (56) reduces to $\Delta\pi_s - \Delta\hat{\pi}_k \le 0$. This means that the schedule offset before π_3 in π' is not smaller than in π. The possible increase of the schedule offset can reduce idle time at the beginning of π_3 or within this subschedule (actually, the value $-(\Delta\pi_s - \Delta\hat{\pi}_k)$ corresponds to the length of the artificial idle time before π_3 on M_1 in Fig. 6).

Now, let's consider the remaining part of the schedule from the end of π_3 to the beginning of π_4. Since it is assumed that π_3 is the same in π' as in π, the schedule offset before J_k in π and J_s in π' is identical and equal to $\Delta\pi_3$. The difference between idle times before J_k in π and J_s in π' is equal to

$$(I_k - \hat{I}_s) = \max\{0, p_{1k} - \Delta\pi_3\} - \max\{0, p_{1s} - \Delta\pi_3\}. \quad (57)$$

Because $p_{1s} \ge p_{1k}$, idle time before J_s in π' is not smaller than before J_k in π and $(I_k - \hat{I}_s) \le 0$. Additionally, the schedule offset before π_4 changes by the value

$$\begin{aligned} \Delta\pi_k - \Delta\hat{\pi}_s &= (\max\{0, \Delta\pi_3 - p_{1k}\} + p_{2k}) \\ &\quad - (\max\{0, \Delta\pi_3 - p_{1s}\} + p_{2s}) \\ &= \max\{0, \Delta\pi_3 - p_{1k}\} \\ &\quad - \max\{0, \Delta\pi_3 - p_{1s}\} - (p_{2s} - p_{2k}). \end{aligned} \quad (58)$$

The assumptions that $p_{1s} \ge p_{1k}$ and $p_{2s} \le p_{2k}$ implies that $\Delta\pi_k - \Delta\hat{\pi}_s \ge 0$. This means that the offset before J_s in π' is not larger than before J_k in π. It may result in an additional idle time before π_4 (cf. Fig. 6).

Summing up, before the schedule π_3 the solution improves as a result of interchanging jobs J_s and J_k, and it deteriorates after π_3. The possible reduction of idle time at the beginning of the schedule $(I_s - \hat{I}_k \ge 0)$ has to be decreased by the additional idle time which appears at the end of it $(I_k - \hat{I}_s \le 0)$. Moreover, shifting π_3 to the left on M_1 (cf. Fig. 6), no precedence constraint is violated. In consequence, the schedule offset before π_3 increases by the value $-(\Delta\pi_s - \Delta\hat{\pi}_k) \ge 0$. A larger schedule offset can result only in a smaller idle time in the succeeding part of schedule π', i.e. between J_k and π_3. However, this criterion improvement is reduced before π_4 by the value $-(\Delta\pi_k - \Delta\hat{\pi}_s) \le 0$. In consequence, the total change in idle time is equal to:

$$\begin{aligned} \Delta I = (I_s - \hat{I}_k) + (I_k - \hat{I}_s) - (\Delta\pi_s - \Delta\hat{\pi}_k) \\ - (\Delta\pi_k - \Delta\hat{\pi}_s). \end{aligned} \quad (59)$$

Combining Eq. (59) with (55–58), one obtains

$$\begin{aligned} \Delta I &= (\max\{0, p_{1s} - \Delta\pi_1\} - \max\{0, p_{1k} - \Delta\pi_1\}) \\ &\quad + (\max\{0, p_{1k} - \Delta\pi_3\} - \max\{0, p_{1s} - \Delta\pi_3\}) \\ &\quad - (\max\{0, \Delta\pi_1 - p_{1s}\} - \max\{0, \Delta\pi_1 - p_{1k}\} \\ &\quad - (p_{2k} - p_{2s})) \\ &\quad - (\max\{0, \Delta\pi_3 - p_{1k}\} - \max\{0, \Delta\pi_3 - p_{1s}\} \\ &\quad - (p_{2s} - p_{2k})) \\ &= \max\{0, p_{1s} - \Delta\pi_1\} - \max\{0, \Delta\pi_1 - p_{1s}\} \\ &\quad + \max\{0, \Delta\pi_1 - p_{1k}\} - \max\{0, p_{1k} - \Delta\pi_1\} \\ &\quad + \max\{0, p_{1k} - \Delta\pi_3\} - \max\{0, \Delta\pi_3 - p_{1k}\} \\ &\quad + \max\{0, \Delta\pi_3 - p_{1s}\} - \max\{0, p_{1s} - \Delta\pi_3\} \\ &\quad + (p_{2k} - p_{2s}) + (p_{2s} - p_{2k}). \end{aligned} \quad (60)$$

Taking into account that for any integer x

$$\max\{0, x\} - \max\{0, -x\} = x, \quad (61)$$

Equation (60) reduces to $\Delta I = p_{1s} - \Delta\pi_1 + \Delta\pi_1 - p_{1k} + p_{1k} - (\pi_3 + (\pi_3 - p_{1s} + (p_{2k} - p_{2s}) - (p_{2k} - p_{2s}) = 0$. The total idle time does not change after interchanging jobs J_s and J_k. Moreover, some idle time units are shifted to the right, possibly after a common due date, reducing in this way the total late work in the system. In consequence the schedule π' is not worse than π.

Case 2. Let's assume that J_k in π' starts earlier than J_s in π. This means that there exists a sequence π_2 which is scheduled before J_s in π and after J_k in π'. Exemplary solutions π and π' are depicted in Fig. 7. Some partial schedules π_1, π_3, π_4 can be empty. Actually, the proof for Case 2 is a straightforward consequence of Case 1. In the analysis for Case 1, no assumption on the structure of a subschedule preceding J_s in π was formulated. Particularly, the assumption that all early jobs are sequenced in Johnson's order was not crucial for the correctness of the proof for Case 1.

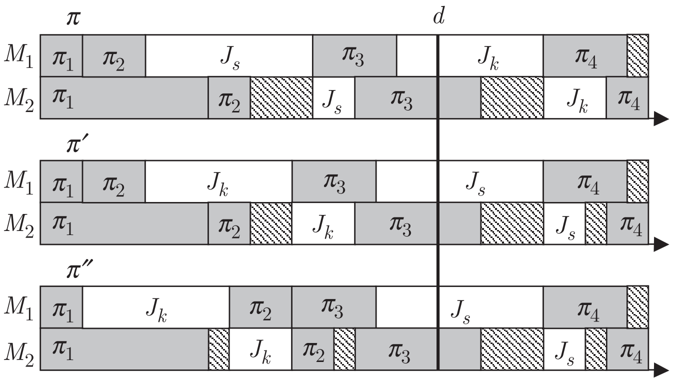

Fig. 7. Exemplary schedules π, π' and π'', where $J_k \in J^2$ is interchanged with $J_s \in J^2(p_{1k} \leq p_{1s} \wedge p_{2s} \leq p_{2k})$ and there exists at least one job executed before J_s in π and after J_k in π'

In consequence, interchanging jobs J_s and J_k in π, one obtain a new solution π' as depicted in Fig. 7, corresponding to Case 1 (where J_k and π_2 are not in Johnson's order). Case 1 shows that the total idle time in π' is not larger than in π. Actually, in π' some idle time can be shifted to the right in comparison to π causing the possible decrease of the criterion value. In consequence π' is not worse than π.

Imposing Johnson's order on all early jobs causes shifting subschedule π_2 after J_k in a final schedule π''. The starting times of π_3 and the succeeding subschedules on M_1 do not change after such a modification. Since Johnson's order is optimal from the schedule length point of view, the completion time of π_2 can be only smaller in π'' than the completion time of J_k in π'. This means that some idle time can be shifted to the right, possibly after a common due date, reducing the criterion value in this way.

Summing up, π'' is not worse than π', and, consequently, it is not worse than the original schedule π. The repeated analysis of the remaining pairs of jobs J_s, $J_k \in J^2$ such that $J_s \in E \wedge J_k \in L \wedge p_{1k} \leq p_{1s} \wedge p_{2k} \geq p_{2s}$, proves Theorem 3.

The presented theorems suggest a strategy of constructing an optimal solution for problem $F2|d_j = d|Y$. If there is a pair of jobs J_k, J_s such that $p_{1k} \leq p_{1s}$ and J_k precedes J_s in Johnson's order ($J_k \to J_s$), and in a certain solution J_s is processed early, while J_k is late, then one can improve a current schedule by executing J_k before a common due date by:

– shifting J_k before J_s, if $J_k \in J^1$ (for $J_s \in J^1$ based on Theorem 1, for $J_s \in J^2$ based on Theorem 2),
– interchanging J_k and J_s, if $J_k \in J^2$ (it has to be $J_s \in J^2$, based on Theorem 3).

These two cases can be intuitively justified. The shorter processing time on the first machine ($p_{1k} \leq p_{1s}$), denotes that the possible idle time before job J_k can be smaller than before job J_s. Moreover, processing a job from J^1 early is usually profitable for the criterion value, because it does not decrease the offset between machines in a succeeding subschedule: the larger offset the smaller idle time before a succeeding job. On the contrary, jobs from J^2 processed early may cause the offset decrease, since they have shorter second task than the first one. However, if $J_k \in J^2$, then the offset decrease for J_k is smaller than for J_s, because $p_{1k} \leq p_{1s}$ and $p_{2k} \geq p_{2s}$, it makes interchanging these jobs profitable for the criterion value.

All the theorems presented in the paper concern pairs of jobs J_k and J_s such that $p_{1k} \leq p_{1s}$ and J_k precedes J_s in Johnson's order ($J_k \to J_s$). If J_k succeeds J_s in Johnson's order ($J_s \to J_k$), then executing J_k early does not influence the schedule before J_s and one cannot estimate, whether moving J_k to the set of early jobs is profitable for the schedule quality or not with regard to J_s. On the other hand, interchanging J_s with J_k does not always result in the improvement of the criterion value, as it is shown in Fig. 8 (for $J_k \in J^2$ and $J_s \in J^1$) and in Fig. 9 (for J_k, $J_s \in J^2$ and $p_{2s} > p_{2k}$).

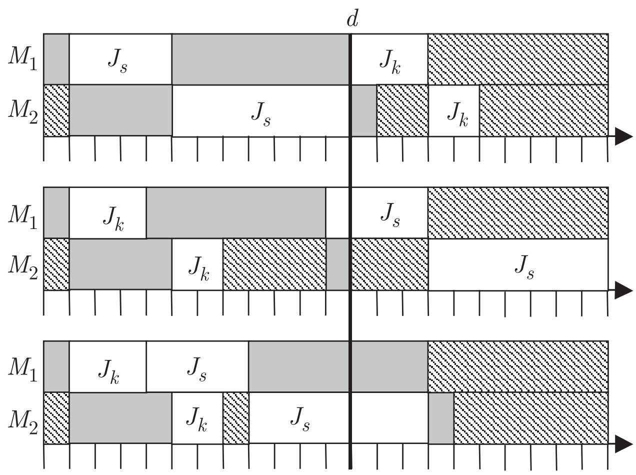

Fig. 8. Exemplary schedules with $J_k \in J^2 \wedge J_s \in J^1 \wedge p_{1k} \leq p_{1s}$, where executing J_k early instead of J_s is not profitable for the total late work value

These observations complete the analysis of all possible cases for jobs J_k and J_s with $p_{1k} \leq p_{1s}$, which are as follows:

– $J_k \to J_s \wedge J_k \in J^1 \wedge J_s \in J^1$: shifting J_k before J_s is profitable, cf. Theorem 1,
– $J_k \to J_s \wedge J_k \in J^1 \wedge J_s \in J^2$: shifting J_k before J_s is profitable, cf. Theorem 2,

- $J_k \rightarrow J_s \wedge J_k \in J^2 \wedge J_s \in J^1$: the case is impossible, J^1 precedes J^2 in Johnson's order,
- $J_k \rightarrow J_s \wedge J_k \in J^2 \wedge J_s \in J^2$: interchanging J_k and J_s is profitable, cf. Theorem 3,
- $J_s \rightarrow J_k \wedge J_k \in J^1 \wedge J_s \in J^1$: the case is impossible, $J_s \rightarrow J_k$ in Johnson's order means that $p_{1s} \leq p_{1k}$ (if $p_{1s} = p_{1k}$, then Theorem 1 can be applied),
- $J_s \rightarrow J_k \wedge J_k \in J^1 \wedge J_s \in J^2$: the case is impossible, J^1 precedes J^2 in Johnson's order,
- $J_s \rightarrow J_k \wedge J_k \in J^2 \wedge J_s \in J^1$: processing J_k early might not be profitable, cf. Fig. 8,
- $J_s \rightarrow J_k \wedge J_k \in J^2 \wedge J_s \in J^2$: processing J_k early might not be profitable, cf. Fig. 9.

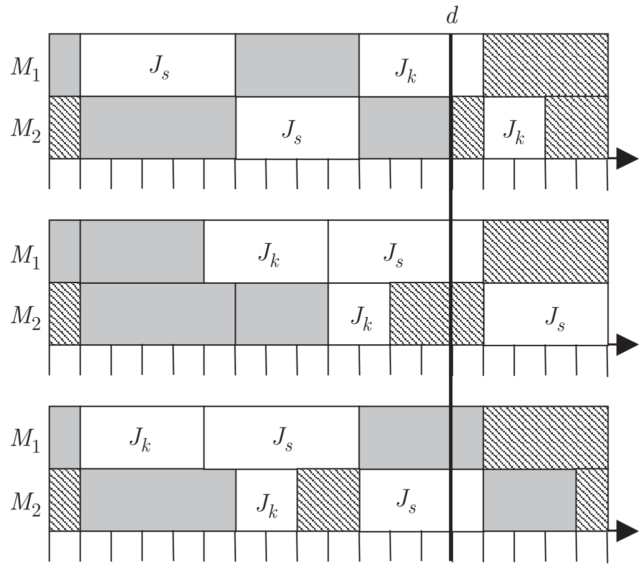

Fig. 9. Exemplary schedules with $J_k, J_s \in J^2 \wedge p_{1k} \leq p_{1s} \wedge p_{2s} > p_{2k}$, where executing J_k early instead of J_s is not profitable for the total late work value

4. Conclusions

The paper presents three dominance relations for the two-machine flow shop problem with a common due date and the late work criterion, $F2|d_j = d|Y$, which describe the special features of an optimal solution of this scheduling case. They state that constructing an optimal sequence of jobs one should select early jobs based on their processing times on the first machine, preferring activities with a shorter first task.

These results make it possible to continue the research on problem $F2|d_j = d|Y$ in two interesting directions. First, the efficiency of dominance relations will be checked in practice, by implementing a branch and bound approach. The presented theorems enable to truncate some branches in the search tree representing the solution process of the B&B method. Partial solutions, partial permutations, which are dominated in terms of the rules formulated in the paper can be discarded, usually reducing the run time of the exact approach.

On the other hand, the features of an optimal solution pointed out in the considered theorems will be important components of heuristic or metaheuristic approaches, whose efficiency will be validated mainly in the computational experiments. However, in the case of heuristic algorithms, the results given in the work might be also the basis for the theoretical analysis of their behavior in the worst case.

Thus, the results on $F2|d_j = d|Y$ presented in the paper are the starting point for the further theoretical and computational studies on this scheduling problem.

Acknowledgements. The author has been supported by a KBN grant.

References

[1] J. Błażewicz, K. Ecker, E. Pesch, G. Schmidt, and J. Węglarz, *Scheduling Computer and Manufacturing Processes*, 2nd ed., Springer, Berlin, 2001.
[2] P. Brucker, *Scheduling Algorithms*, Springer, Berlin, 1998.
[3] M. Pinedo and X. Chao, *Operation Scheduling with Applications in Manufacturing and Services*, McGraw-Hill, Boston, 1999.
[4] J.Y-T. Leung, "Minimizing total weighted error for imprecise computation tasks and related problems", in: *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*, pp. 1–16, edited by J.Y-T Leung, CRC Press, Boca Raton, 2004.
[5] J. Błażewicz, "Scheduling preemptible tasks on parallel processors with information loss", *Recherche Technique et Science Informatiques* 3(6), 415–420 (1984).
[6] J. Błażewicz and G. Finke, "Minimizing mean weighted execution time loss on identical and uniform processors", *Information Processing Letters* 24, 259-263 (1987).
[7] C.N. Potts and L.N. Van Wassenhove, "Single machine scheduling to minimize total late work", *Operations Research* 40(3), 586–595 (1991).
[8] C.N. Potts and L.N. Van Wassenhove, "Approximation algorithms for scheduling a single machine to minimize total late work", *Operations Research Letters* 11, 261–266 (1991).
[9] A.M.A. Hariri, C.N. Potts, and L.N. Van Wassenhove, "Single machine scheduling to minimize total late work", *INFORMS Journal on Computing* 7, 232–242 (1995).
[10] M.Y. Kovalyov, C.N. Potts, and L.N. Van Wassenhove, "A fully polynomial approximation scheme for scheduling a single machine to minimize total weighted late work", *Mathematics of Operations Research* 19(1), 86–93 (1994).
[11] D.S. Hochbaum and R. Shamir, "Minimizing the number of tardy job unit under release time constraints", *Discrete Applied Mathematics* 28, 45–57 (1990).
[12] J.Y-T. Leung, V.K.M. Yu, and W-D. Wei, "Minimizing the weighted number of tardy task units", *Discrete Applied Mathematics* 51, 307–316 (1994).
[13] R.B. Kethley and B. Alidaee, "Single machine scheduling to minimize total late work: a comparison of scheduling rules and search algorithms", *Computers & Industrial Engineering* 43, 509–528 (2002).
[14] S.G. Kolliopoulos and G. Steiner, "Approximation algorithms for minimizing total weighted tardiness on a sin-

gle machine", *Lectures Notes in Computer Science* 2996, 176–186 (2004).

[15] G.J. Woeginger, "When does a dynamic programming formulation guarantee the existence of an FPTAS?" in: *Electronic Colloquium on Computational Complexity Report* TR01-084, University of Trier, 2001.

[16] J. Błażewicz, E. Pesch, M. Sterna, and F. Werner, "Total late work criteria for shop scheduling problems", in: *Operations Research Proceedings* edited by K. Inderfurth, G. Schwödiauer, W. Domschke, F. Juhnke, P. Kleinschmidt, G. Wäscher, pp. 354–359, Springer, Berlin, 2000.

[17] J. Błażewicz, E. Pesch, M. Sterna, and F. Werner, "Open shop scheduling problems with late work criteria", *Discrete Applied Mathematics* 134, 1–24 (2004).

[18] J. Błażewicz, E. Pesch, M. Sterna, and F. Werner, "The two-machine flow-shop problem with weighted late work criterion and common due date", *European Journal of Operational Research* 165(2), 408–415 (2005).

[19] J. Błażewicz, E. Pesch, M. Sterna, and F. Werner, "A note on two-machine job shop with late work criterion", *Journal of Scheduling* 10 (2), 87–97 (2007).

[20] J. Błażewicz, E. Pesch, M. Sterna, and F. Werner, "Flow shop scheduling with late work criterion - choosing the best solution strategy", *Lecture Notes in Computer Science* 3285, 68–75 (2004).

[21] J. Błażewicz, E. Pesch, M. Sterna, and F. Werner, "A comparison of solution procedures for two-machine flow shop scheduling with late work criterion", *Computers & Industrial Engineering* 49, 611–624 (2005).

[22] J. Błażewicz, E. Pesch, M. Sterna, and F. Werner, "Metaheuristics for late work minimization in two-machine flow shop with common due date", *Lecture Notes in Artificial Intelligence* 3698, 222–234 (2005).

[23] M. Sterna, *Problems and Algorithms in Non-Classical Shop Scheduling*, Scientific Publishers of the Polish Academy of Sciences, Poznań, 2000.

[24] M.R. Garey and D.S. Johnson, *Computers and Intractability. A Guide to the Theory of NP-completeness*, W.H. Freeman and Co., San Francisco, 1979.

[25] B.M.T. Lin, F.C. Lin, and R.C.T. Lee, "Two-machine flow shop scheduling to minimize total late work", *Engineering Optimization* 38(4), 501–509 (2006).

[26] S.M. Johnson, "Optimal two- and three-stage production schedules with setup times included", *Naval Research Logistics Quarterly* 1, 61–68 (1954).