Journal of Theoretical and Applied Information Technology
28th February 2013. Vol. 48 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

COMMON DUE WINDOW SCHEDULING WITH BOTH TIME AND POSITION EFFECTS ON A SINGLE MACHINE

[1]JINTAO MENG, [2]SHISHENG LI
[1]Zhengzhou Institute of Aeronautical Industry Management, Zhengzhou 450015, China
[2] College of Science, Zhongyuan University of Technology, Zhengzhou 450007, China
E-mail: [1] mjint@zzia.edu.cn , [2] lss668@tom.com

ABSTRACT

We consider the problem of common due window location scheduling on a single machine, where the processing times of jobs depend both on their starting times and positions in a sequence. The problem is to determine the optimal earliest due date, the due window size, and the job schedule simultaneously to minimize costs for earliness, tardiness, earliest due date assignment and due window size penalties. An $O(n\log n)$ time optimal algorithm is presented to solve the problem.

Keywords: *Scheduling, Single-Machine, Due Window, Time/Position Effect*

1. INTRODUCTION

In the traditional scheduling theory, it is usually assumed that the processing time of a job is fixed and has a constant value [14]. In many real-life situations, however, the processing conditions may vary and affect the actual durations of jobs. The phenomenon that the actual processing time of a job is variable, is traditionally attributed to one of the following causes: (i) deterioration, (ii) learning and (iii) resource allocation. Usually, in scheduling with deterioration we assume that the later a job starts, the longer it takes to process. On the other hand, in scheduling with learning the actual processing time of a job gets shorter, provided that the job is scheduled later. Scheduling problems with these two effects have received considerable attention in the recent fifteen years; we refer to Alidaee and Womer [1], Cheng et al. [4], Gawiejnowicz [5], Biskup [2], Janiak et al. [8] and Rustogi and Strusevich [15] for recent state of the art reviews in these areas, as well as for references to practical applications of these models.

In contemporary competitive market, firms are required to improve customer service as a means to gain competitive advantage. In operational terms, good customer service means finishing jobs (or orders) as close as possible to their due dates. In practice, the supply contract between the supplier and the customer often specifies a time interval such that a job completed within the time interval is considered to be on time and will not incur any penalty. Jobs completed prior to or after the time window are penalized according to their earliness/tardiness values. It is clear that a late and wide due window increases the production flexibility and the delivery options of the supplier. On the other hand, in this case his competitiveness is reduced. The main question is, therefore, when to schedule the due-window. Scheduling problems with due-window have been studied by several researchers; e.g., Liman et al. [10], Chen and Lee [3] Mosheiov and Sarig [11, 12], Yang et al. [16] and Meng et al. [13]. See also the survey papers by Gordon et al. [6] and Kaminsky and Hochbaum [9] for more relevant motivation and models.

The remaining part of the paper is organized as follows. In Section 2, we formulate the scheduling problem under consideration and present some basic lemmas. In Section 3, we present some optimal properties of optimal solutions. In Section 4, we provide an $O(n\log n)$ time optimization algorithm. Some concluding remarks are given in the last section.

2. PROBLEM FORMULATION AND PRELIMINARIES

We are given a single machine and a set $J=\{J_1,J_2,\ldots,J_n\}$ of n independent jobs, which are non-preemptive and available at time zero for processing. The job processing times depend on both their starting times and positions in a schedule. Specifically, if job $J_j(j=1,2,\ldots,n)$ is scheduled in

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

position r and starts at time t in a sequence, its actual processing time is

$$p_{jr} = \left(a_j + bt\right) g(r) \qquad (1)$$

where a_j is the normal processing time of J_j, b is a common job-independent deterioration rate, and $g(r)$ is a general positional factor defined as a known function.

All the jobs share a common due window. Let $d(\geq 0)$ and $h(\geq 0)$ denote the due window starting time and finishing time, respectively, and $D = h - d$ denote the due window size. In our problem, both d and D are to be determined. Hence, an optimal solution to the problem will consist of the job sequence, the actual job starting times, the due window starting time and the window size. A job completed within the due window is regarded as on time and will not be penalized. If a job is completed on or before the due window starting time d, an earliness penalty will be incurred. On the other hand, if a job is completed after the due window finishing time h, a tardiness penalty will be incurred. Let $C_j\left(\pi\right)$ denote that completion time of job J_j in a feasible schedule π. If there is no ambiguity, we omit π and use C_j to denote $C_j\left(\pi\right)$. Our objective is to find an optimal schedule π^* to minimize a cost function that includes earliness, tardiness, earliest due date assignment and due window size penalties, given by the following equation:

$$F = \sum_{j=1}^{n}\left(\alpha E_j + \beta T_j + \gamma d + \delta D\right) \qquad (2)$$

where $E_j = \max\left\{0, d - C_j\right\}$ is the earliness of job J_j; $T_j = \max\left\{0, C_j - h\right\}$ is the tardiness of job J_j; α, β, γ and δ are non-negative parameters representing the cost of one unit of earliness, tardiness, due date and due window penalty, respectively. Let "CDW" denote that the studied problem is a common due window scheduling problem. Using the traditional three-field notation, the problem can be denoted as $1\left|p_{jr} = \left(a_j + bt\right)g(r), CDW\right|\sum_{j=1}^{n}\left(\alpha E_j + \beta T_j + \gamma d + \delta D\right)$.

In the remaining part of this section, we present several lemmas which are very useful to our subsequent analysis. For ease of exposition, let $J_{[j]}$ denote the job that occupies the j-th position in a schedule, and let $p_{[j]}$, $a_{[j]}$, $C_{[j]}$, $E_{[j]}$ and $T_{[j]}$ be defined correspondingly.

Lemma 2.1. For the schedule $1\left|p_{jr} = \left(a_j + bt\right)g(r), CDW\right|\sum_{j=1}^{n}\left(\alpha E_j + \beta T_j + \gamma d + \delta D\right)$, if the job sequence is $\pi = \left(J_{[1]}, J_{[2]}, \ldots, J_{[n]}\right)$, then the makespan of π is

$$C_{\max} = C_{[n]} = \sum_{j=1}^{n} a_{[j]} g(j) \prod_{i=j+1}^{n}\left(1 + bg(i)\right)$$

where an empty product equals one.

Proof. Since $\pi = \left(J_{[1]}, J_{[2]}, \ldots, J_{[n]}\right)$, then the actual processing time of a job scheduled in the first position in π is

$$p_{[1]} = \left(a_{[1]} + b \cdot 0\right) g(1) = a_{[1]} g(1),$$

and its completion time is

$$C_{[1]} = t_{[1]} + p_{[1]} = 0 + p_{[1]} = a_{[1]} g(1).$$

Similarly, we have

$$p_{[2]} = \left(a_{[2]} + b \cdot C_{[1]}\right) g(2) = \mathrm{b}a_{[1]} g(1) g(2) + a_{[2]} g(2),$$

$$C_{[2]} = C_{[1]} + p_{[2]} = a_{[1]} g(1) + \mathrm{b}a_{[1]} g(1) g(2) + a_{[2]} g(2)$$

$$= a_{[1]} g(1)\left(1 + \mathrm{b}g(2)\right) + a_{[2]} g(2)$$

Generally, the actual processing time and the completion time of job $J_{[j]}$ are

$$\begin{aligned} p_{[j]} =\ & \mathrm{b}a_{[1]} g(1)\left(1 + bg(2)\right)\cdots\left(1 + bg(j-1)\right) g(j) \\ & + \mathrm{b}a_{[2]} g(2)\left(1 + bg(3)\right)\cdots\left(1 + bg(j-1)\right) g(j) + \cdots \\ & + \mathrm{b}a_{[j-2]}\left(1 + bg(j-1)\right) g(j) + \mathrm{b}a_{[j-1]} g(j) + a_{[j]} g(j) \end{aligned}$$

$$\begin{aligned} C_{[j]} =\ & a_{[1]} g(1)\left(1 + \mathrm{b}g(2)\right)\cdots\left(1 + \mathrm{b}g(j)\right) \\ & + a_{[2]} g(2)\left(1 + bg(3)\right)\cdots\left(1 + \mathrm{b}g(j)\right) + \cdots \\ & + a_{[j-1]} g(j-1)\left(1 + bg(j)\right) + a_{[j]} g(j) \\ =\ & \sum_{l=1}^{j} a_{[l]} g(l) \prod_{i=l+1}^{j}\left(1 + bg(i)\right) \qquad (3) \end{aligned}$$

The completion time of the last job can be expressed as

$$C_{[n]} = a_{[1]} g(1)\left(1 + \mathrm{b}g(2)\right)\cdots\left(1 + \mathrm{b}g(n)\right)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$+a_{[2]}g(2)\left(1+bg(3)\right)\cdots\left(1+\mathrm{b}g(n)\right)+\cdots$$

$$+a_{[n-1]}g(n-1)\left(1+bg(n)\right)+a_{[n]}g(n)$$

$$=\sum_{j=1}^{n}a_{[j]}g(j)\prod_{i=l+1}^{n}\left(1+bg(i)\right)$$

This completes the proof of Lemma 2.1. □

Lemma 2.2. For the schedule

$1\left|p_{jr}=\left(a_j+bt\right)g(r),CDW\right|\sum_{j=1}^{n}\left(\alpha E_j+\beta T_j+\gamma d+\delta D\right)$, if the job sequence is $\pi=\left(J_{[1]},J_{[2]},\ldots,J_{[n]}\right)$, then the total completion time of π is

$$\sum_{j=1}^{n}C_j=\sum_{j=1}^{n}C_{[j]}=\sum_{j=1}^{n}a_{[j]}g(j)\left[\sum_{i=j}^{n}\prod_{k=j+1}^{i}\left(1+bg(k)\right)\right].$$

Proof. From (3), the total completion time can be calculated as follows:

$$\sum_{j=1}^{n}C_{[j]}=\sum_{j=1}^{n}\sum_{l=1}^{j}a_{[l]}g(l)\prod_{i=l+1}^{j}\left(1+bg(i)\right)$$

$$=\sum_{j=1}^{n}a_{[j]}g(j)\left[\sum_{i=j}^{n}\prod_{k=j+1}^{i}\left(1+bg(k)\right)\right]$$

This completes the proof of Lemma 2.2. □

Moreover, the following lemma says that the minimization of the sum $\sum_i x_i y_i$ is obtained by matching the largest x_j value with the smallest y_j value, the second largest x_j value with the second smallest y_j value, etc.

Lemma 2.3. ([7]) Let there be two sequences of numbers x_i and y_i ($i=1,2,\ldots,n$), the sum $\sum_i x_i y_i$ of products of the corresponding elements is the least if the sequences are monotonic in the opposite sense.

3. PROPERTIES OF OPTIMAL SOLUTIONS

In this section, we present several properties for an optimal schedule of the problem. First, it is clear that an optimal schedule starts at time zero. In addition, an optimal schedule exists with no idle time between consecutive jobs. Therefore, it is sufficient to consider permutation schedules to find optimal solutions. Clearly, the objective function (2) can be alternatively expressed as follows:

$$F=\alpha\sum_{j=1}^{n}E_j+\beta\sum_{j=1}^{n}T_j+\gamma nd+\delta nD \qquad (4)$$

Lemma 3.1. For any given job sequence $\pi=\left(J_{[1]},J_{[2]},\ldots,J_{[n]}\right)$, there exists an optimal common due window such that the due window's starting time d and finishing time w coincide with some jobs' completion times.

proof. Suppose that we are given a job sequence $\pi=\left(J_{[1]},J_{[2]},\ldots,J_{[n]}\right)$ that starts at time zero and contains jobs at the k-th and the l-th positions such that $C_{[k]}=C_{[k+1]}-p_{[k]}<d<C_{k+1}$ and $C_{[l]}=C_{[l+1]}$ $-p_{[l+1]}<h=d+D<C_{[l+1]}$, where $0\le k\le l\le n-1$

First, we show that a small shift of h either to the right or to the left can only decrease (does not increase) the total cost. When we shift Δ units of time to the right for h, the total tardiness cost decreases, whereas the cost of the due-window size increases. The change in the total cost is given by: $\Delta Z_1=-\beta\left(n-l\right)\Delta+\delta n\Delta=\Delta\left(\delta n-\beta\left(n-l\right)\right)$. When we shift Δ units of time to the left for h, the total tardiness cost increases, whereas the cost of the due-window size decreases. The change in the total cost is clearly given by $-\Delta Z_1$. If ΔZ_1 is positive, a shift of h to the left is worthwhile, and otherwise shift h to the right. (If $\Delta Z_1=0$, then a shift to either size does not increase the total cost.) Therefore, there exists an optimal schedule in which h coincides with a job's completion time.

Now we show that a small shift of d either to the right or to the left decreases (does not increase) the total cost. When d is moved Δ units of time to the left, the total earliness cost and the cost of the due-window starting time decrease, whereas the cost of the due-window size increases. The change in the total cost is given by: $\Delta Z_2=-\alpha k\Delta-\gamma n\Delta+\delta n\Delta=\Delta\left(\delta n-\gamma n-\alpha k\right)$. When we shift Δ units of time to the right for d, the change in the total cost is easily shown to be $-\Delta Z_2$. Again, a shift of d either to the right or to the left does not increase the total cost.

Therefore, an optimal schedule exists such that the common due-window's starting time and finishing time are located at the completion times of some jobs, respectively. This completes the proof of Lemma 3.1. □

Lemma 3.2. For any given job sequence π, there exists an optimal schedule in which the index of the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

job completed at the due-window's starting time is $K=\lceil n(\delta-\gamma)/\alpha\rceil$, and the index of the job completed at the due window's finishing time is $L=\lceil n(\beta-\delta)/\beta\rceil$.

Proof. For a given job sequence, from Lemma 2.1, each job's starting time and completion time can be easily determined. Note that the result has been proved by Liman et al. [10] when the job processing times are all constants. In their proof, it is immaterial whether processing times are time/position-dependent, it is only concerned with how many jobs are completed before and after the due window. By Lemma 3.1, the result holds. □

Write $K^*=\lceil n(\delta-\gamma)/\alpha\rceil$ and $L^*=\lceil n(\beta-\delta)/\beta\rceil$. Furthermore, write $\Delta_1=\alpha K^*+\gamma n-\delta n$ and $\Delta_2=\beta L^*+\delta n-\beta n$. In the next lemma, we introduce the positional weights (as in Liman et al. [10]) and present the total cost as a function of these weights.

Lemma 3.3. The total cost can be written as:

$$F=\sum_{j=1}^{n}W_j a_{[j]} \tag{5}$$

Where

$$W_j=\begin{cases} g(j)\Big\{\Delta_1\prod_{i=j+1}^{K^*}(1+bg(i)) \\ +\Delta_2\prod_{i=j+1}^{L^*}(1+bg(i)) \\ -\alpha\sum_{i=j}^{K^*}\prod_{k=j+1}^{i}(1+bg(k)) \\ +\beta\sum_{i=L^*+1}^{n}\prod_{k=j+1}^{i}(1+bg(k))\Big\} \quad j=1,\cdots,K^*; \\ \Delta_2\prod_{i=j+1}^{L^*}(1+bg(i)) \\ +\beta\sum_{i=L^*+1}^{n}\prod_{k=j+1}^{i}(1+bg(k)),\ j=K^*+1,\cdots,L^*; \\ \beta\sum_{i=j}^{n}\prod_{k=j+1}^{i}(1+bg(k)),\quad j=L^*+1,\cdots,n. \end{cases} \tag{6}$$

Proof. Given K^* and L^*, by Lemma 3.2, the total cost F can be expressed as follows:

$$F=\alpha\sum_{j=1}^{n}E_j+\beta\sum_{j=1}^{n}T_j+\gamma nd+\delta nD$$

$$=\alpha\sum_{j=1}^{K^*}\left(d-C_{[j]}\right)+\beta\sum_{j=L^*+1}^{n}\left(C_{[j]}-h\right)$$

$$+\gamma nd+\delta nD$$

$$=\alpha K^*d-\alpha\sum_{j=1}^{K^*}C_{[j]}+\beta\sum_{j=L^*+1}^{n}C_{[j]}$$

$$-\beta\left(n-L^*\right)h+\gamma nd+\delta nD$$

$$=\left(\alpha K^*+\gamma n-\delta n\right)d+\left(\beta L^*+\delta n-\beta n\right)h$$

$$-\alpha\sum_{j=1}^{K^*}C_{[j]}+\beta\sum_{j=L^*+1}^{n}C_{[j]}. \tag{7}$$

From Lemma 3.1 and 3.2, we have $C_{K^*}=d$, $C_{L^*}=h=d+D$. Then in virtue of Lemmas 2.1 and 2.2, (7) can be expressed as

$$F=\Delta_1 C_{K^*}+\Delta_2 C_{L^*}-\alpha\sum_{j=1}^{K^*}C_{[j]}+\beta\sum_{j=1}^{n}C_{[j]}-\beta\sum_{j=1}^{L^*}C_{[j]}$$

$$=\Delta_1\left(\sum_{j=1}^{K^*}a_{[j]}g(j)\prod_{i=j+1}^{K^*}(1+bg(i))\right)$$

$$+\Delta_2\left(\sum_{j=1}^{L^*}a_{[j]}g(j)\prod_{i=j+1}^{L^*}(1+bg(i))\right)$$

$$-\alpha\sum_{j=1}^{K^*}a_{[j]}g(j)\left[\sum_{i=j}^{K^*}\prod_{k=j+1}^{i}(1+bg(k))\right]$$

$$+\beta\sum_{j=1}^{n}a_{[j]}g(j)\left[\sum_{i=j}^{n}\prod_{k=j+1}^{i}(1+bg(k))\right]$$

$$-\beta\sum_{j=1}^{L^*}a_{[j]}g(j)\left[\sum_{i=j}^{L^*}\prod_{k=j+1}^{i}(1+bg(k))\right]$$

$$=\sum_{j=1}^{K^*}a_{[j]}g(j)\Big\{\Delta_1\prod_{i=j+1}^{K^*}(1+bg(i))$$

$$+\Delta_2\prod_{i=j+1}^{L^*}(1+bg(i))-\alpha\sum_{i=j}^{K^*}\prod_{k=j+1}^{i}(1+bg(k))$$

$$+\beta\sum_{i=L^*+1}^{n}\prod_{k=j+1}^{i}(1+bg(k))\Big\}$$

$$+\sum_{j=K^*+1}^{L^*}a_{[j]}g(j)\Big\{\Delta_2\prod_{i=j+1}^{L^*}(1+bg(i))$$

$$+\beta\sum_{i=L^*+1}^{n}\prod_{k=j+1}^{i}(1+bg(k))\Big\}$$

$$+\sum_{j=L^*+1}^{n}a_{[j]}g(j)\Big\{\beta\sum_{i=j}^{n}\prod_{k=j+1}^{i}(1+bg(k))\Big\}$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$= \sum_{j=1}^{n} W_j a_{[j]},$$

where w_j is defined in (6). □

Lemma 3.4. There exists an optimal schedule such that the processing sequence of the jobs can be obtained by matching the elements of w_j with a_j in opposite orders.

Proof. Note that the term w_j defined in (6) can be viewed as a positional, job-independent penalty for any job scheduled in the j-th position. Therefore, by Lemma 2.3, the total cost (5) can be minimized by the matching procedure. □

4. AN OPTIMAL ALGORITHM

Summarizing the above discussion, we present the following optimization algorithm.

Algorithm CDW

Step 1: Set $K^* = \lceil n(\delta-\gamma)/\alpha \rceil$ and

$$L^* = \lceil n(\beta-\delta)/\beta \rceil.$$

Step 2: Calculate w_j according to Equation (6) for j =1, 2, $\cdots$, n.

Step 3: Determine the processing order of the jobs according to the matching procedure. Let π be the obtained job sequence. If necessary, renumber the jobs such that $\pi = (J_1, J_2, \ldots, J_n)$.

Step 4: Calculate the earliest due date as $d = \sum_{j=1}^{K^*} a_j g(j) \prod_{i=j+1}^{K^*} (1+bg(i))$, the due window size as $D = \sum_{j=1}^{L^*} a_j g(j) \prod_{i=j+1}^{L^*} (1+bg(i)) - \sum_{j=1}^{K^*} a_j g(j) \prod_{i=j+1}^{K^*} (1+bg(i))$, and the total cost by $F = \sum_{j=1}^{n} W_j a_j$.

Theorem 4.1. Algorithm CDW solves the problem $1\left|p_{jr} = (a_j + bt) g(r), CDW\right| \sum_{j=1}^{n} (\alpha E_j + \beta T_j + \gamma d + \delta D)$ in $O(n\log n)$ time.

Proof. The correctness of algorithm CDW follows from Lemmas 3.1-3.4. To determine the running rime of the algorithm, note that Step 1 requires constant time; Steps 2 and 4 require $O(n)$ time; while Step 3 requires $O(n\log n)$ time. Hence, the time complexity of algorithm CDW is $O(n\log n)$. □

5. CONCLUSIONS

We study the single-machine common due window assignment and scheduling problem with simultaneous consideration of time and position effects. Our goal is to find jointly the optimal location and size of the common due window, as well as the optimal job sequence to minimize the total earliness, tardiness, and due window starting time and size costs. We propose a polynomial time solution for the problem. For further research, it would be interesting to extend the results to the case with multiple machines.

ACKNOWLEDGMENTS

This research was supported in part by NSFC (11171313).

REFRENCES:

[1] B. Alidaee and N.K. Womer, Scheduling with time dependent processing times: Review and extensions, Journal of the Operational Research Society, 50(1999) 711-720.

[2] D. Biskup, A state-of-the-art review on scheduling with learning effects. European Journal of Operational Research, 188(2008), 315-329.

[3] Z.L. Chen and C.Y. Lee, Parallel machine scheduling with a common due window. European Journal of Operational Research, 136(2002), 512-527.

[4] T.C.E. Cheng, Q. Ding and B.M.T. Lin, A concise survey of scheduling with time-dependent processing times, European Journal of Operations Research, 152(2004), 1-13.

[5] S. Gawiejnowicz, Time-Dependent Scheduling, Springer, Berlin/Heidelberg, 2008.

[6] V.S. Gordon, J.M. Proth and V.A. Strusevich, Scheduling with due date assignment, in: J.Y.T. Leung (ed.), Handbook of Scheduling: Algorithms, Models and Performance Analysis. CRC Press, Boca Raton, 2004.

[7] G.H. Hardy, J.E. Littlewood and G. Polya, Inequalities, Cambridge University Press, London, 1967.

[8] A. Janiak, T. Krysiak and R. Trela, Scheduling problems with learning and ageing effects: a

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

survey. Decision Making in Manufacturing Services 5(2011), 19-36.

[9] P. Kaminsky and D. Hochbaum, Due date quotation models and algorithms, in Leung, J.Y.T. (ed.), Handbook of Scheduling: Algorithms, Models and Performance Analysis. CRC Press, Boca Raton, 2004.

[10] S.D.Liman, S.S. Panwalkar and S. Thongmee, Common due window size and location determination in a single machine scheduling problem, Journal of the Operational Research Society, 49(1998), 1007-1010.

[11] G. Mosheiov and A. Sarig, A due-window assignment problem with position-dependent processing times, Journal of the Operational Research Society, 59(2008), 997-1003.

[12] G. Mosheiov and A. Sarig, Scheduling identical jobs and due-window on uniform machines, European Journal of Operational Research, 201(2010), 712-719.

[13] J.T. Meng, J. Yu and X.X. Lu, Scheduling deteriorating jobs with a common due window on a singl machine, Information Technology Journal, 11(2012), 392-395.

[14] M. Pinedo, Scheduling: Theory, Algorithms, and Systems, Prentice-Hall, Upper Saddle River, New York, 2008.

[15] K. Rustogi and V.A. Strusevich, Simple matching vs linear assignment in scheduling models with positional effects: A critical review, European Journal of Operational Research, 222(2012), 393-407.

[16] S.J. Yang, D.L. Yang and T.C.E. Cheng, Single-machine due-window assignment and scheduling with job-dependent aging effects and deteriorating maintenance, Computers and Operations Research, 37(2010) 1510-1514.