Lecture Notes in Management Science (2014) Vol. **6**: 128–137
6th International Conference on Applied Operational Research, Proceedings

ISSN 2008-0050 (Print), ISSN 1927-0097 (Online)

From MaxPlus algebra to general lower bounds for the total weighted completion time in flowshop scheduling problems

Nhat-Vinh Vo and Christophe Lenté

Université François-Rabelais de Tours,
CNRS, LI EA 6300, OC ERL CNRS 6305, Tours, France
{nhat.vo, christophe.lente}@univ-tours.fr

Abstract. The flowshop scheduling problem has been largely studied for 60 years. As a criterion, the total weighted completion time reflects the total weighted waiting time of all customers. There have not been many studies about this criterion and they are limited in the number of machines or constraints. MaxPlus algebra is also applied to the scheduling theory but the literature focuses on some concrete constraints. Therefore, this study addresses a general permutation flowshop problem, with several additional constraints such as delays, blocking or setup times, to elaborate on lower bounds for the total weighted completion time. These lower bounds imply solving a Traveling Salesman Problem. The principle, based on a MaxPlus modeling of flowshop problems, is developed and experimental results are presented.

Keywords: scheduling; flowshop; weighted total completion time; lower bound

Introduction

Flowshop scheduling problem has been largely studied for 60 years. The total completion time criterion reflects "the total manufacturing waiting time experienced by all customers" (Emmons and Vairaktarakis (2013)). However, not all customers should be equally evaluated and the total completion time does not fully reflect that total waiting time. Therefore, the total weighted completion time is worth being studied. Even with only two machines, problem $F_2||\sum C_i$ is NP-hard in the strong sense and so are problems with more machines and with $\sum w_i C_i$ criterion. Therefore, results that help to solve these problems are interesting.

There have not been actually many studies on this criterion. Nevertheless, we can cite some studies on the single-machine scheduling to minimize total weighted completion time. A branch-and-bound algorithm to solve problem $1|r_i|\sum w_i C_i$ is proposed in Nessah and

Kacem (2012). In another study, an improved branch-and-bound algorithm to solve problem $1|d_i|\sum w_i C_i$ is proposed in Pan (2003). Two-machine non-classical flowshop problems where processing time of a task is a decreasing or non-decreasing function of its execution start time are investigated (Wang and Wang (2013), Wagneur *et al* (2011)). In fact, these aforementioned studies only deal with a few constraints or limited number of machines. It is unsure to generalize them to m-machine flowshop problems or to other constraints.

In this study, MaxPlus algebra (see §2.1) is the basis of the proposed approach. This approach has been rarely used in the scheduling theory but widely in control systems, especially in relation with Petri Nets. However, there are some studies that can be cited on project scheduling problems (Giffler (1963)), on cyclic parallel machine problems (Hanen *et al* (1995)), on cyclic flowshop/jobshop scheduling problems (Gaubert (1992), Gaubert *et al* (1999)). Moreover, MaxPlus algebra is applied to modeling and to tackling flowshop scheduling problems with minimal delays, setup and removal times in Lenté (2001) and Bouquard and Lenté (2006), and with minimal and maximal delays for two machines in Bouquard *et al* (2006) and for m machines in Augusto *et al* (2006). In these studies, each job is associated to a MaxPlus square matrix and a lower bound, an upper bound and/or dominance conditions are derived by applying transformations to those matrices. This approach is applied effectively to modeling flowshop problems with minimal-maximal delays, setup and removal times and to highlighting a central problem (Vo and Lenté (2013)). Lower bounds for the total completion time in flowshop scheduling problems are also presented (Vo *et al* (2014)).

This study deals with a m-machine permutation flowshop problem submit to various constraints (see §3). Elaborated lower bounds for the total weighted completion time are based on the resolution of two sub-problems: one problem similar to the one machine total weighted completion time minimization problem and the other similar to a traveling salesman problem. A branch-and-bound algorithm using these lower bounds is developed to obtain experimental results. The background of the study including MaxPlus algebra and flowshop scheduling problem is presented in the next section. It is also recalled in section 3 how MaxPlus algebra can be used to model a general flowshop problem. Section 4 then explains the lower bound construction. Finally, after the presentation of a branch-and-bound algorithm, some tests associated to m-machine permutation flowshop problems subject to minimize criterion $\sum w_i C_i$ are presented as experimental results.

Context and definitions

MaxPlus algebra

The proposed approach used in this study is based on MaxPlus Algebra. It is shortly described as follows; a more detailed introduction is given in Gunawardena (1998).

In MaxPlus algebra, we denote the maximum by $\oplus$ and the addition by $\otimes$. The former operator, $\oplus$, is idempotent, commutative, associative and has a neutral element $-\infty$ denoted by $\mathbb{O}$. The latter, $\otimes$, is associative, distributive on $\oplus$ and has a neutral element (0) denoted by $\mathbb{I}$. The null element, $\mathbb{O}$, is an absorbing element for $\otimes$. These properties lead to the statement that $\mathbb{R}_{max} = (\mathbb{R} \cup \{-\infty\}, \oplus, \otimes)$ is a dioid. It is important to note that in MaxPlus algebra in particular, and in dioids in general, the first operator $\oplus$ cannot be simplified, i.e. $a \oplus b = a \oplus c \nRightarrow b = c$. Moreover, in $\mathbb{R}_{max}$, the second operator $\otimes$ is commutative, and except $\mathbb{0}$, every element is invertible and the inverse of x ($x \in \mathbb{R}_{max}$) is denoted by x^{-1} or $^{\mathbb{I}}/_{x}$. For simplicity, we denote the ordinary subtractions by $\frac{x}{y}$ instead of $x \otimes y^{-1}$ and by xy the product $x \otimes y$. Moreover, $\forall a \in \mathbb{R}_{max}$ and $\forall (p,q) \in \mathbb{R}^2$, we denote the product of a^p and a^q by $a^p \otimes p^q$ or by a^{p+q} and not by $a^{p \otimes q}$. In the other hand, we set by convention that for any array $u_i \in \mathbb{R}_{max}$, if $p > q$, $\otimes_{i=p}^{q} u_i = \mathbb{I}$ and $a^{\mathbb{I}} = \mathbb{I}\ (= 0)$.

We are also able to extend these two operators to $m \times m$ matrices of elements of $\mathbb{R}_{max}$. Let A and B be two matrices of size $m \times m$, operators $\oplus$ and $\otimes$ are defined by

$$\forall (i,j) \in \{1, \cdots, m\}^2, [A \oplus B]_{ij} = [A]_{ij} \oplus [B]_{ij}$$

$$\forall (i,j) \in \{1, \cdots, m\}^2, [A \otimes B]_{ij} = \bigoplus_{k=1}^{m} [A]_{ik} \otimes [B]_{kj}$$

where $[\cdot]_{ij}$ is the element at the i^{th} row and j^{th} column of the corresponding matrix. The set of $m \times m$ matrices in $\mathbb{R}_{max}$ is a dioid. However, $\otimes$ is not commutative and not every matrix is invertible.

Flowshop Scheduling Problem

A flowshop scheduling problem basically consists of a set of n-jobs $\mathcal{J} = \{J_1, J_2, \cdots, J_n\}$ and another set of m-machines $\{M_1, M_2, \cdots, M_m\}$. Each job of m-operations must be processed sequentially by all machines in the same predefined order, let us say from M_1 to M_m and each machine can be used by only one job at a time (Brucker (2006)). We limit this study to permutation flowshop problems where all jobs are launched in the same order over all machines. An operation is at least described by its processing times: the processing time of job J_i on machine M_k (or equivalently, the processing time of the k^{th} operation of job J_i) p_{ik}. The completion time of job J_i on machine M_k (C_{ik}) and the completion time of job J_i (C_i) are related by $C_i = C_{im}$.

As mentioned above, additional constraints have been studied. The $no - wait$ constraint is studied in problems where there is no delay allowed between two successive operations of a job. On the contrary, constraints of $min - delay, max - delay, min - \max delay$ indicate a flowshop problem with delays between two successive operations of a job. Depending on the case, these delays may have to meet a lower bound, an upper bound or both. It may also exist separate non-sequence dependent setup times (S_{nsd}) and/or removal times (R_{nsd}) before and/or after each operation. Some authors have considered blocking constraints, due to the non-existence of intermediate storage between consecutive machines or to specific interactions between machines. These constraints are referred to as RSb, RCb and RCb^* in Trabelsi *et al* (2012).

As previously stated, in this study we focus on the total weighted completion time ($\sum w_i C_i$) which is the weighted sum of the completion times of the different jobs in a given schedule.

MaxPlus modeling of flowshop scheduling problems

Using notations proposed in Graham *et al* (1979), we denote our problem by $F_m|perm;\beta|\gamma$. It addresses a m-machine permutation flowshop problem with a set of constraints β that is a subset of
$\{min - \max delay, no - wait,\ S_{nsd}, R_{nsd}, RSb, RCb, RCb^*\}$.
As it does not interfere in the modeling process, criterion γ can be whatever we desire. The total weighted completion time criterion is studied in the following. The following scheme basically reflects the modeling process:

- Consider four dates corresponding to each k^{th} operation O_{ik} of a job J_i: date δ_k of availability of machine M_k (before execution of operation O_{ik}), starting time St_{ik} of operation O_{ik}, its completion time C_{ik} and date of liberation D_{ik} of machine M_k (after execution of operation O_{ik}), that is the date when job J_i leaves machine M_k to be placed in a stock or on the following machine. In most flowshop problems, dates C_{ik} and D_{ik} are equal; however, they can be different if there exists removal times. In this case, D_{im} is equal to C_i plus the removal time of operation O_{im}.
- Link these different variables by formulating the system S of inequalities among them.
- Calculate the smallest D_{ik} $(1 \le k \le m, 1 \le i \le n)$ solutions of the system S.

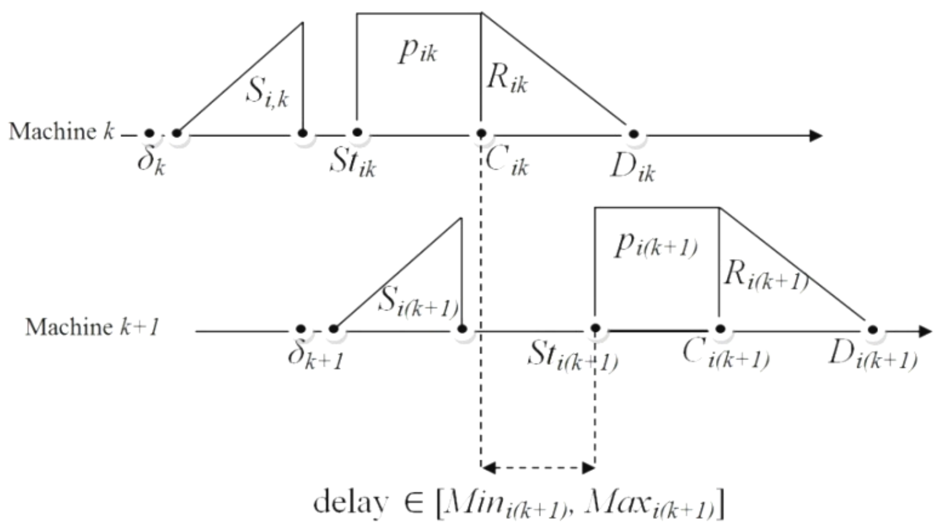

Fig. 1. Example of flowshop problem with time lags, setup and removal times

Whatever the set of constraints β is, these calculations lead to a MaxPlus linear relation between dates of liberation D_{ik} and dates of availability δ_k (Lenté (2011), Vo and Lenté (2013)). More precisely, the following proposition can be stated:

Proposition 1 (Matrix associated to a job)
Let $\vec{\delta}$ (resp. $\vec{D_i}$) be the row vector of the m-dates δ_k (resp. D_{ik}): it exists a $m \times m$ MaxPlus matrix T_i computed from data of job J_i such that

$$\vec{D_i} = \vec{\delta} \otimes T_i \quad (1)$$

Each element of matrix T_i will be denoted t_{lc}^i, in other words, $t_{lc}^i = [T_i]_{lc}$ (see §2.1). This matrix sums up the job data (processing times, setup times, delays and so on) and the flowshop constraints.

$$T_i = \begin{pmatrix} t_{11}^i & \cdots & t_{1m}^i \\ \vdots & \ddots & \vdots \\ t_{m1}^i & \cdots & t_{mm}^i \end{pmatrix} \quad (2)$$

These results can be generalized to a sequence of jobs (Lenté, 2011; Bouquard *et al*, 2006).

Definition 1 *(Matrix associated to a sequence)*
Let σ be a sequence of ϑ jobs: its associated matrix is matrix T_σ defined by

$$T_\sigma = \bigotimes_{i=1}^{\vartheta} T_{\sigma(i)} \quad (3)$$

Proposition 2 *If $\vec{\delta}$ is the vector of dates of availability of machines and $\vec{D}_\sigma$ the vector of dates of liberation of machines, after the execution of sequence σ, we have the relation*

$$\vec{D}_\sigma = \vec{\delta} \otimes T_\sigma \quad (4)$$

Proposed lower bounds

This section presents lower bounds for problem $F_m|perm;\beta|\sum w_i C_i$, with $\beta \in \{min - \max delay, no - wait, S_{nsd}, R_{nsd}, RSb, RSb, RCb^*\}$. To develop the calculations, we assume that $C_i = D_{im}$ $(1 \leq i \leq n)$. It is true unless there exists removal times: this particular case will be discussed in §4.3.

Lower bound of completion time of a job

A lower bound of the completion time of the k^{th} job in a sequence is firstly presented and then a lower bound for the total weighted completion time criterion is elaborated.

Proposition 3 *Let σ a sequence of jobs and $\vec{\delta}$ the line vector of dates of availability of the machines ($\vec{\delta} = (\delta_1, \delta_2, \cdots, \delta_m)$). The completion time of the job in the k^{th} position in the sequence verifies relation:*

If $k = 1$:
$$C_{\sigma(1)} \geq \delta_1 [T_{\sigma(1)}]_{1m}$$
If $k \geq 2$:
$$C_{\sigma(k)} \geq \delta_1 \bigotimes_{j=1}^{k-2} [T_{\sigma(j)}]_{11} [T_{\sigma(k-1)} T_{\sigma(k)}]_{1m}$$

Proof. The proof can be found in Vo *et al* (2014).

Lower bound of the total weighted completion time

Definition 2 ($A_1(\sigma)$ and $B_1(\sigma)$)
Given a sequence σ of n-jobs, we define:

$$A_1(\sigma) = \bigotimes_{i=1}^{n-1} \left(\left[T_{\sigma(i)}\right]_{11}\right)^{\sum_{j=i+1}^{n} w_{\sigma(i)}} \quad \text{and} \quad B_1(\sigma) = \left[T_{\sigma(1)}\right]_{1m}^{w_{\sigma(1)}} \left(\bigotimes_{i=2}^{n} \frac{\left[T_{\sigma(i-1)}T_{\sigma(i)}\right]_{1m}^{w_{\sigma(i)}}}{\left[T_{\sigma(i-1)}\right]_{11}^{w_{\sigma(i)}}}\right)$$

Proposition 4
$\forall\sigma$ sequence:

$$\bigotimes_{i=1}^{n} C_{\sigma(i)}^{w_{\sigma(i)}} \geq \delta_1^{\left(\sum_{i=1}^{n} w_{\sigma(i)}\right)} \otimes A_1(\sigma) \otimes B_1(\sigma)$$

At this point, we can obtain a lower bound of the total weighted completion time by computing the optimal values of factors $A_1(\sigma)$ and $B_1(\sigma)$. The two following propositions explain how to do.

Proposition 5 (Minimisation of $A_1(\sigma)$)
Let σ_{SWPT}^1 the sequence obtained by sorting jobs in non-decreasing order of ratio $\frac{[T_i]_{11}}{w_i}$ (in usual notations). This sequence minimizes criterion $A_1(\sigma)$.

Proof. It is a straightforward application of Smith's rule (Smith (1956)).

Proposition 6 (Minimisation of $B_1(\sigma)$)
Let us consider an Asymmetric Traveling Salesman Problem (ATSP) defined by the following distances between $n+1$ towns, numbered from 0 to n:

$$\begin{cases} \forall i \in \{1,2,\cdots,n\}: d(0,i) = [T_i]_{1m}^{w_i} \\ \forall i \in \{1,2,\cdots,n\}: d(i,0) = \mathbb{I}\ (=0) \\ \forall (i,j) \in \{1,2,\cdots,n\}^2: d(i,j) = \dfrac{\left[T_iT_j\right]_{1m}^{w_j}}{[T_i]_{11}^{w_j}} \end{cases} \tag{5}$$

Let sequence σ_{ATSP}^1 be an optimal cycle of this ATSP: $B_1(\sigma_{ATSP}^1)$ is the optimal value of criterion $B_1(\sigma)$.

Proof. With these notations, $B_1(\sigma)$ can be rewritten as the length of a cycle:

$$B_1(\sigma) = d\big(0,\sigma(1)\big)\left(\bigotimes_{i=1}^{n-1} d\big(\sigma(i),\sigma(i+1)\big)\right) d(\sigma(n),0) \tag{6}$$

All these results lead to the next proposition.

Proposition 7 (Lower Bound LB_{VFL}^1)
Let $LB_{VFL}^1 = \delta_1^{\left(\sum_{i=1}^{n} w_{\sigma(i)}\right)} \otimes A_1(\sigma_{SWPT}^1) \otimes B_1(\sigma_{ATSP}^1)$: LB_{VFL}^1 is a lower bound of the total weighted completion time. In usual notations, this lower bound is defined by:

$$LB_{VFL}^{1} = \sum_{i=1}^{n} w_i\,\delta_1 + A_1(\sigma_{SWPT}^{1}) + B_1(\sigma_{ATSP}^{1}) \tag{7}$$

It is needed to solve a traveling salesman problem to compute this lower bound; however, the procedures for solving that problem are rather effective on medium size instances. This lower bound is similar to the one presented in Della Croce *et al* (1996) for two machines.

Existence of removal times

If there are removal times, the date of liberation of machine M_m by job J_i (D_{im}) is equal to the weighted sum of completion time C_i of job J_i and removal time of the last operation of O_{im} of J_i. Thus, the total weighted sum of D_{im} $(1 \le i \le n)$ is equal to the total weighted completion time plus a constant term which is equal to the weighted sum of removal times of all last operations. Therefore, to obtain a lower bound of the total weighted completion time we only have to subtract this constant from LB_{VFL}^{1}.

Additional similar lower bounds

A similar approach can be used to elaborate on lower bound LB_{VFL}^{l} ($\forall l \in \{1,2,\cdots,m\}$).

Let us define $A_l(\sigma)$ and $B_l(\sigma)$:

$$A_l(\sigma) = \bigotimes_{i=1}^{n-1} \left(\left[T_{\sigma(i)}\right]_{ll}\right)^{\sum_{j=i+1}^{n} w_{\sigma(i)}} \quad \text{and} \quad B_l(\sigma) = \left[T_{\sigma(1)}\right]_{lm}^{w_{\sigma(1)}} \left(\bigotimes_{i=2}^{n} \frac{\left[T_{\sigma(i-1)}T_{\sigma(i)}\right]_{lm}^{w_{\sigma(i)}}}{\left[T_{\sigma(i-1)}\right]_{ll}^{w_{\sigma(i)}}}\right)$$

The l^{th} lower bound is given by:

$$LB_{VFL}^{l} = \delta_l^{(\sum_{i=1}^{n} w_i)} \otimes A_l(\sigma_{SWPT}^{l}) \otimes B_l(\sigma_{ATSP}^{l}) \tag{8}$$

Branch-and-bound algorithm

To evaluate the lower bounds we proposed, we have incorporated them in a branch-and-bound procedure. A branch-and-bound is an enumeration method that builds dynamically a search tree. Lower bounds or dominance relations are used to cut some useless branches. We have used the separation scheme introduced in Ignall and Schrage (1965): a partial sequence is progressively built as we go deeper in the search tree. A node corresponds to a partial sequence and a set of free jobs. The separation of a node consists in adding a free job at the end of the sequence. A node has as many children as its free jobs. The branching strategy is Depth-First-Search. An upper bound is computed at the root node and updated at each node. For this purpose, we have used heuristic HA and its modified version HA^{o} (only at the rood node) presented in Rajendran and Ziegler (1997).

Experimental results

To the best of our knowledge, there has not been any study on the exact resolution of m-machine permutation flowshop scheduling problems with total weighted completion time criterion. Therefore, we implemented our branch-and-bound procedure for problem $F_m|perm; min - \max delay\,|\sum w_i C_i$ to test effectiveness of LB_{VFL}s. We generated benchmarks corresponding to three different problems: the first one is associated to problems with minimal and maximal delays (denoted by P_1), the second one relates to classical flowshop problems (P_2) and the last one corresponds to $no - wait$ flowshop problems (P_3). We considered both criterion $\sum w_i C_i$ and criterion $\sum C_i$. The number of machine m belongs to $\{5,10,15\}$ and the number of jobs n to $\{5,10,13\}$. For each class of (m, n), twenty-five instances were generated. The processing times, the maximal and minimal delays were randomly generated from the uniform distribution on $[10,100]$, $[0,300]$ and $[0,200]$, respectively. In case of the total weighted completion time, the job weights were randomly generated between 1 and 5. All machines were available from the time zero. To compute lower bounds LB_{VFL}s, we used the ATSP solving procedure developed in Carpaneto *et al* (1995). The used machine is based on an Intel Duocore 2.6GHz 4GB RAM.

Table 1. The performance of branch-and-bound procedure

m	n	P_1				P_2				P_3			
		t_w	N_w	t	N	t_w	N_w	t	N	t_w	N_w	t	N
5	5	0.000	15	0.000	16	0.000	13	0.000	19	0.000	14	0.000	22
	10	0.024	768	0.038	1403	0.028	1167	0.047	3031	0.046	1185	0.065	2633
	13	0.441	12854	1.004	36445	0.254	8793	3.752	194986	0.692	13892	5.421	181477
10	5	0.000	23	0.000	26	0.000	23	0.000	27	0.000	23	0.000	27
	10	0.215	2934	0.332	4976	0.177	2921	0.487	10087	0.229	2494	0.501	7677
	13	7.840	96066	20.986	269617	3.227	39486	26.890	407069	4.343	33864	29.236	364559
15	5	0.004	32	0.004	37	0.002	27	0.002	32	0.003	25	0.003	31
	10	1.260	6452	1.796	9847	0.799	3778	1.286	6995	1.064	3817	2.136	10528
	13	54.468	198799	161.896	703428	35.167	140287	158.467	610115	35.897	100705	155.718	542101

We have reported in table 1 the mean computation time (in seconds) of each class for problems with job weights (column t_w) and without job weights (column t). We have indicated also in this table the average number of visited nodes over all instances of each class (column N_w and column N). For each resolution, we used lower bounds: LB^1_{VFL}, LB^2_{VFL}, LB^{m-1}_{VFL} and LB^m_{VFL}. Over the three groups of benchmarks, computation time as well as number of visited nodes depends not only on the number of jobs but also on the number of machines. Our proposed lower bounds can be adapted for both problems with job weights and problems without job weights. However, it seems that the performance of problems with job weights is better than that of problems without job weights. For example, with instances of 15-machines and 13-jobs of the first group, it takes 54.468 seconds (161.896 seconds, respectively) to complete a branch-and-bound procedure in problems with job weights (without job weights, respectively). The results also show that with a small number of jobs (5 or 10 jobs), problems with delays are easier to be solved than $no-wait$ problems. However, as the number of jobs is greater (13 jobs), $no-wait$ problems have better performance because solving asymmetric traveling salesman problem is easier in $no-wait$ problems. Table 1 also shows that with a small number of jobs, it takes a very short time to achieve the optimum. However, for a larger number of jobs, we need to develop a strategy in order to shorten the computation time. This strategy is under investigation.

Conclusion

This study proposed a MaxPlus approach to tackle a m-machine flowshop problem with several additional constraints. We are able, thanks to the MaxPlus approach, to transform a general flowshop problem into a matrix problem. Then some computations over these matrices allow us to elaborate on new lower bounds for the total weighted completion time criterion, based on the resolution of a one-machine problem and an asymmetric traveling salesman problem. Solving an NP-hard problem is necessary but experimental results have shown the effectiveness of these lower bounds. This branch-and-bound also outperforms the one presented in Allahverdi and Al-Anzi (2006), in a non-weighted jobs case for problem $F_3|perm; S_{nsd}|\sum C_i$ (Vo *et al* (2014)).

In our further research, the objective is to improve these lower bounds LB_{VFL} as well as the branch-and-bound algorithm. In some cases as the number of jobs is large, it is also necessary to carry out a strategy to improve the quality of lower bounds and to shorten the computation time of the whole branch-and-bound algorithm. Moreover, additional constraints for a flowshop problem can be studied such as $no-wait$, S_{nsd}, R_{nsd}, limited stocks between machines or blocking constraints by modifying only matrix T_i associated to job J_i.

References

Allahverdi, A. and Al-Anzi, F. S. (2006). A branch-and-bound algorithm for three-machine flowshop scheduling problem to minimize total completion time with separate setup times. EJOR, 169(3):767–780.

Augusto, V., Lenté, C., and Bouquard, J.-L. (2006). Résolution d'un flowshop avec delais minimaux et maximaux. In MOSIM.

Bouquard, J.-L. and Lenté, C. (2006). Two-machine flow shop scheduling problems with minimal and maximal delays. 4or, 4(1):15-28.

Bouquard, J.-L., Lenté, C. and Billaut, J.-C. (2006). Application of an optimization problem in Max-Plus algebra to scheduling problems. Discrete Applied Mathematics, 154(15):2064-2079.

Brucker, P. (2006). Scheduling Algorithms. Springer, 5 edition.

Carpaneto, G., Dell'amico, M., and Toth, P. (1995). Exact solution of large asymmetric travelling salesman problems. ACM Transactions on Mathematical Software, 21(4):394-409.

Della Croce, F., Narayan, V., and Tadei, R. (1996). The two-machine total completion time flow shop problem. European Journal Of Operational Research, 90:227-237.

Emmons, H. and Vairaktarakis, G. (2013). Flow Shop Scheduling. Springer, New York, 182 edn.

Gaubert, S. (1992). Théorie des systèmes linéaires dans les dioïdes. PhD thesis.

Gaubert, S. and Maisresse, J. (1999). Modeling and analysis of timed Petri nets using heaps of pieces. IEEE trans. Automatic Control, 44(4):683–698.

Giffler, B. (1963). Schedule algebras and their use in formulating general systems simulations. In Industrial scheduling. Prentice Hall, New Jersey.

Graham, R. L., Lawler, E. L., Lenstra, J. K., and Rinnooy Kan, A. H. (1979). Optimization and approximation in deterministic sequencing and scheduling: A survey. Annals of Discrete Mathematics, 5(2):287–326.

Gunawardena, J. (1998). Idempotency. Publications of the Newton Institute.

Hanen C and Munier A (1995). Cyclic scheduling on parallel processors: an overview. John Wiley.

Ignall, E. and Schrage, L. (1965). Application of branch-and-bound technique to some flow shop problems. Operations Research, 13(3):400–412.

Lenté, C. (2001). Analyse Max-Plus de problèmes d'ordonnancement de type Flowshop. PhD thesis, Université François Rabelais de Tours.

Lenté, C. (2011). Mathématiques, Ordonnancement et Santé. HDR, Université François Rabelais de Tours.

Nessah, R. and Kacem, I. (2012). Branch-and-bound method for minimizing the weighted completion time scheduling problem on a single machine with release dates. Computers & OR, 39:471-478.

Pan Y (2003). An improved branch and bound algorithm for single machine scheduling with deadlines to minimize total weighted completion time. Operations Research Letters, 31(6): 492–496.

Rajendran, C. and Ziegler, H. (1997). An efficient heuristic for scheduling in a flowshop to minimize total weighted flowtime of jobs. European Journal of Operational Research, 1(103):129-138.

Smith WE (1956). Various optimizers for single-stage production. Naval Res. Log. Quar 3(1-2):59-66.

Trabelsi, W., Sauvey, C., and Sauer, N. (2012). Heuristics and metaheuristics for mixed blocking constraints flowshop scheduling problems. Computers & Operations Research, 39:2520-2527.

Vo, N. V. and Lenté, C. (2013). Equivalence between Two Flowshop Problems - MaxPlus Approach. In Proceedings of the 2nd International Conference on Operations Research and Enterprise Systems, pages 174–177, Barcelona. SciTePress - Science and Technology Publications.

Vo, N. V., Fouillet, P., and Lenté, C. (2014). General lower bounds for the total completion time in a flowshop scheduling problem - MaxPlus approach. In Proceedings of the 3rd Int'l Conf on OR and Enterprise Systems, Angers, France. SciTePress - Science and Technology Publications.

Wagneur, E., Cheng, E., and Ng, D. (2011). A closedform solution for the optimal release times for the F2/deteriorating jobs/∑w_j C_j problem. Discrete Applied Mathematics, 159(13): 1367–1376.

Wang, J.-B. and Wang, M.-Z. (2013). Solution algorithms for the total weighted completion time minimization flow shop scheduling with decreasing linear deterioration. The International Journal of Advanced Manufacturing Technology, 67(1-4):243–253.