

CAHIER DU LAMSADE

Laboratoire d'Analyse et Modélisation de Systèmes pour l'Aide à la Décision

(Université de Paris-Dauphine)

Unité Associée au C.N.R.S. n° 825

PRODUCTION SCHEDULING ON PARALLEL MACHINES
SUBJECT TO STAIRCASE DEMANDS

CAHIER N° 74
janvier 1987

R. SLOWINSKI

CONTENTS

	<u>Pages</u>
RESUME	I
ABSTRACT	II
1. PROBLEM STATEMENT	2
2. EXTENDED TWO-PHASE METHOD	3
3. MINIMIZATION OF THE TOTAL MACHINE SET-UP COST	11
4. CONCLUSIONS	13
REFERENCES	14

ORDONNANCEMENT DE LA PRODUCTION SUR DES MACHINES EN PARALLELE
AVEC DEMANDES EN ESCALIER

RESUME

Un ensemble de n produits indépendants doit être fabriqué par un ensemble de m machines indépendantes travaillant en parallèle en utilisant p ressources supplémentaires de type : moyens de transport, équipement supplémentaire, opérateurs, etc. ; α_i unités de la ressource i sont disponibles à chaque instant. Une machine ne peut pas travailler sur plus d'un seul produit à la fois mais un produit peut être traité sur plusieurs machines simultanément. Chaque paire (machine, produit) demande au plus une unité de la ressource i à chaque fois. Les plans de production sont donnés sous forme de demandes par périodes pour chaque produit. Pour chaque triplet (machine, produit, ressource), on connaît la productivité en nombre d'unités du produit par une unité de temps. L'objectif est de minimiser la somme de pénalités affectées aux écarts (algébriques) entre prévisions et réalisations pour chaque période. Le coût total de changements de fabrication des machines est considéré comme un critère secondaire. Le problème ainsi formulé est résolu par une méthode en deux phases généralisée [5] ; la première phase revient à résoudre un problème de goal programming linéaire ; la deuxième phase consiste dans la construction d'un ordonnancement obtenu à la suite de la résolution d'une série de recherches d'un flot réalisable dans un réseau.

PRODUCTION SCHEDULING ON PARALLEL MACHINES
SUBJECT TO STAIRCASE DEMANDS

ABSTRACT

We are given a set of m unrelated machines working in parallel and a set of n independent products which have to be produced on these machines using p additional resources (e.g. transportation facilities, additional equipment, operators, ...); α_i units of resource i are available at any time. A machine cannot work on more than one product at a time but a product can simultaneously be produced on different machines. Each pair (machine, product) requires at most one unit of resource i at a time. Production plans are imposed in form of periodic demands for each product. For each triple (machine, product, resource) we are given a productivity per time unit. The objective is to minimize the total penalty following from deviations above and below production plans. The sum of machine set-up costs is considered as a secondary performance measure. The problem is solved by an extension of the two-phase method [5]; phase 1 consists in solving an LP problem and phase 2 is the construction of the schedule which reduces to a sequence of compatible flow problems.

1. PROBLEM STATEMENT

The production scheduling problem considered in this paper arised in a workshop manufacturing plastic tanks for different types of car batteries. Its formal statement seems to be of more general interest, so we will drop all technical terms connected with this specific application.

We are given set $\{M_i\}_{i=1}^m$ of unrelated machines working in parallel and set $\{P_j\}_{j=1}^n$ of independent products which have to be produced on these machines using additional renewable resources $\{R_l\}_{l=1}^p$ (e.g. transportation facilities, additional equipment, operators, ...); α_l units of R_l ($l = 1, \dots, p$) are available at any time.

Each pair (M_i, P_j) requires at most 1 unit of specified resource R_l at a time. A machine and a resource unit cannot produce more than one product at a time but a product can be produced simultaneously by various machines with additional resources.

The production of each product on one machine can be arbitrarily interrupted and resumed on another machine or on the same machine but possibly with another resource (preemption). The total machine set-up cost is considered as a secondary schedule performance measure.

The planning horizon is divided into periods of equal length I_t ($t = 1, \dots, q$). Production plans are imposed in form of periodic demands for each product P_j : D_{jt} ($j = 1, \dots, n$; $t = 1, \dots, q$), i.e. the demands are staircase functions of time.

For each triple (M_i, P_j, R_l) , we are given productivity k_{ijl} ($i = 1, \dots, m$; $j = 1, \dots, n$; $l = 1, \dots, p$) in units of P_j produced by (M_i, R_l) per one time unit.

Deviations above and below production plans are penalized by cost coefficients c_{jt}^+ and c_{jt}^- ($j = 1, \dots, n$; $t = 1, \dots, q$), respectively. The objective is to minimize the total cost following from deviations from production plans.

The problem has been solved by an extension of the two-phase method (cf. [2, 3, 5]). Phase 1 consists in solving a linear programming (LP) problem and phase 2 is the construction of the schedule which reduces to a sequence of compatible flow problems.

In the next section, the extended two-phase method is presented. In section 3, minimization of the total machine set-up cost for a schedule is considered. The final section draw conclusions.

2. EXTENDED TWO-PHASE METHOD

The so-called two-phase method was initially proposed for solving a problem of preemptive scheduling of tasks on parallel unrelated machines (cf. [2] for historical background) and then it was extended in different ways to handle additional resources of various categories (cf. [2, 7]).

In this paper, we define the problem in terms of production scheduling and the products correspond to tasks in previous studies. The characteristics of tasks (products) are different, however, since task duration has been replaced by productivity, simultaneous production of the same product on different machines (task multiprocessing) is allowed, and the usual time criterion has been replaced by a cost criterion (weighted sum of deviations).

Phase 1

The decision variables are :

$x_{ijlt} \geq 0$, the time during which (M_i, R_l) works on product P_j in period t ($i = 1, \dots, m$; $j = 1, \dots, n$; $l = 1, \dots, p$; $t = 1, \dots, q$),

$y_{jt}^+, y_{jt}^- \geq 0$, incrementing variables corresponding to the amount above and below production plan for product P_j in period t , respectively ($j = 1, \dots, n$; $t = 1, \dots, q$).

Phase 1 consists in finding the values of decision variables which minimize the weighted sum of deviations from production plans. In order to find them we have to solve the following linear goal programming problem :

$$\left[\begin{array}{l} \text{Minimize : } f = \sum_{t=1}^q \sum_{j=1}^n (c_{jt}^+ y_{jt}^+ + c_{jt}^- y_{jt}^-) \end{array} \right. \quad (1)$$

$$\left[\begin{array}{l} \text{s.t. } \sum_{i=1}^m \sum_{j=1}^n x_{ijlt} \leq \alpha_l I_t \quad (l = 1, \dots, p ; t = 1, \dots, q) \end{array} \right. \quad (2)$$

$$\left[\begin{array}{l} \sum_{l=1}^p \sum_{j=1}^n x_{ijlt} \leq I_t \quad (i = 1, \dots, m ; t = 1, \dots, q) \end{array} \right. \quad (3)$$

$$\left[\begin{array}{l} \sum_{h=1}^t \sum_{i=1}^m \sum_{l=1}^p k_{ijl} x_{ijlt} + y_{jt}^- - y_{jt}^+ = \sum_{h=1}^t D_{jh} \quad (j = 1, \dots, n ; t = 1, \dots, q) \end{array} \right. \quad (4)$$

$$\left[\begin{array}{l} x_{ijlt}, y_{jt}^+, y_{jt}^- \geq 0 \quad (i = 1, \dots, m ; j = 1, \dots, n ; l = 1, \dots, p ; t = 1, \dots, q) \end{array} \right. \quad (5)$$

Constraint (2) says that production times must not exceed the availability of R_l ; (3) ensures that machine M_i does not work more than I_t units of time in period t ; (4) defines the incrementing variables.

It can easily be proved that for an optimal solution of problem (1)-(5), $y_{jt}^+ y_{jt}^- = 0$ ($j = 1, \dots, n$; $t = 1, \dots, q$), i.e. the incrementing variables are mutually exclusive.

Phase 2

Let $z_{il}^t = \sum_{j=1}^n x_{ijl}t$ ($i = 1, \dots, m$; $l = 1, \dots, p$; $t = 1, \dots, q$). Given the $(m \times \alpha_1)$ matrix Z^t with real entries z_{il}^t , we have to find a schedule in I_t time units ($t = 1, \dots, q$) by determining a positive integer r_t , nonnegative numbers $\lambda_1^t, \dots, \lambda_{r_t}^t$ and (0-1)-matrices $S_1^t, \dots, S_{r_t}^t$ satisfying

$$Z = [z_{il}^t] = \lambda_1^t S_1^t + \dots + \lambda_{r_t}^t S_{r_t}^t \quad (6)$$

$$I_t = \lambda_1^t + \dots + \lambda_{r_t}^t \quad (7)$$

$$\sum_{l=1}^p (s_{il}^t)_h \leq 1 \quad (i = 1, \dots, m; h = 1, \dots, r_t) \quad (8)$$

$$\sum_{i=1}^m (s_{il}^t)_h \leq \alpha_l \quad (l = 1, \dots, p; h = 1, \dots, r_t) \quad (9)$$

for $t = 1, \dots, q$.

The interpretation of phase 2 is as follows : $S_h^t = [s_{il}^t]_h$ defines a feasible assignment of resources to machines which is used for λ_h^t time units in period t ($h = 1, \dots, r_t$; $t = 1, \dots, q$). A schedule can always be found in phase 2, as asserted by the following theorem :

Theorem : For any $x_{ijl}t$ ($i = 1, \dots, m$; $j = 1, \dots, n$; $l = 1, \dots, p$; $t = 1, \dots, q$) satisfying (2)-(5) and $Z^t = [z_{il}^t]$ such that $z_{il}^t = \sum_{j=1}^m x_{ijl}t$ ($i = 1, \dots, m$; $l = 1, \dots, p$; $t = 1, \dots, q$), there exists a positive finite integer r_t , nonnegative numbers $\lambda_1^t, \dots, \lambda_{r_t}^t$ and (0-1)-matrices $S_1^t, \dots, S_{r_t}^t$ satisfying (6)-(9) for $t = 1, \dots, q$.

Proof : Let us define a square $((m + p) \times (m + p))$ matrix W^t composed of four matrices :

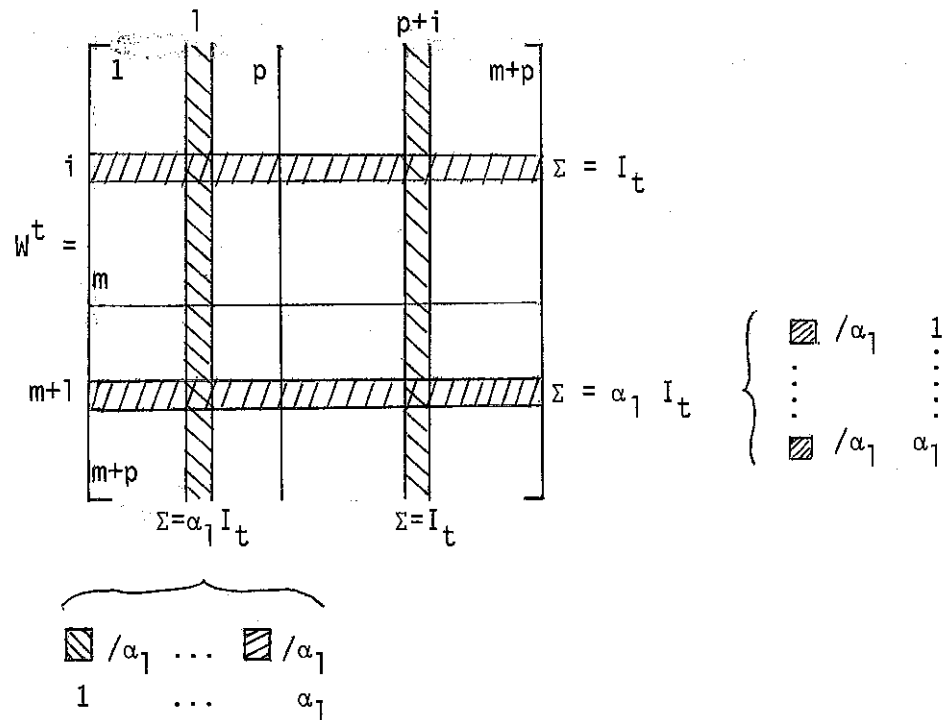
$$W^t = \left[\begin{array}{c|c} Z^t & U^t \\ \hline V^t & (Z^t)^T \end{array} \right] \quad (t = 1, \dots, q) \quad (10)$$

where $(Z^t)^T$ is Z^t transposed, $U^t = [y_{ij}^t]$ is an $(m \times m)$ diagonal matrix of machine idle times, $u_{ij} = I_t - \sum_{l=1}^p z_{il}^t$ ($i = 1, \dots, m$), $V^t = [v_{ij}^t]$ is a $(p \times p)$ diagonal matrix of resource idle times, $v_{ll}^t = \alpha_l I_t - \sum_{i=1}^m z_{il}^t$ ($l = 1, \dots, p$).

Machine M_i is said to be critical in period t iff $y_{ii}^t = 0$ ($i = 1, \dots, m$). Similarly, resource R_l is said to be critical in period t iff $v_{ll}^t = 0$ ($l = 1, \dots, p$).

Let us remark that in matrix W^t , the sum of row elements is equal to I_t in rows $1, \dots, m$ and it is equal to $\alpha_l I_t$ in rows $m+1, \dots, m+p$; the sum of column elements is equal to $\alpha_l I_t$ in columns $1, \dots, p$ and it is equal to I_t in columns $p+1, \dots, p+m$.

Let us substitute each row (and column) in matrix W^t whose sum is equal to $\alpha_l I_t$ by α_l rows (columns) with elements divided by α_l , as it is shown below :



The modified matrix W^t is denoted by $\hat{W}^t$ and its dimension is $(m + p \alpha_l) \times (m + p \alpha_l)$. In $\hat{W}^t$, the sum of elements in each row as well as the sum of elements in each column is equal to I_t , so matrix $\frac{1}{I_t} \hat{W}^t$ is a bistochastic matrix ($t = 1, \dots, q$).

According to the Birkhoff-von Neumann theorem (see [1]), $\frac{1}{I_t} \hat{W}^t$ can always be decomposed into a finite number r_t of permutation matrices $\hat{S}_h^t$.

($h = 1, \dots, r_t$) :

$$\frac{1}{I_t} \hat{W}^t = \delta_1^t \hat{S}_1^t + \dots + \delta_{r_t}^t \hat{S}_{r_t}^t \quad (11)$$

where $\delta_h^t \geq 0$ ($h = 1, \dots, r_t$; $t = 1, \dots, q$) and $\sum_{h=1}^{r_t} \delta_h^t = 1$ ($t = 1, \dots, q$).

Thus, $\delta_h^t I_t = \lambda_h^t$, and by summing up the rows and the columns corresponding to the same machine or additional resource, respectively, we can transform $\hat{S}_h^t$ to S_h^t ($h = 1, \dots, r_t$; $t = 1, \dots, q$) so that conditions (6)-(9) are satisfied.

Let us pass to the construction of S_h^t and calculation of λ_h^t ($h = 1, \dots, r_t$; $t = 1, \dots, q$) for any solution satisfying (2)-(5).

Matrix S_h^t appearing in the decomposition (6) may be obtained by considering the following network $\mathcal{N}^1$ (Fig. 1).

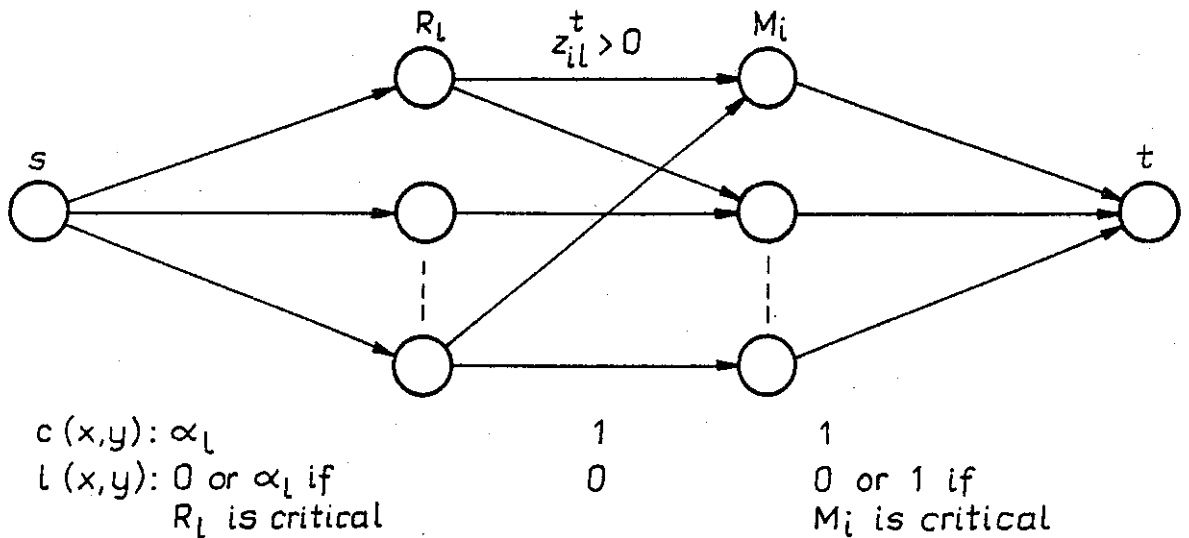
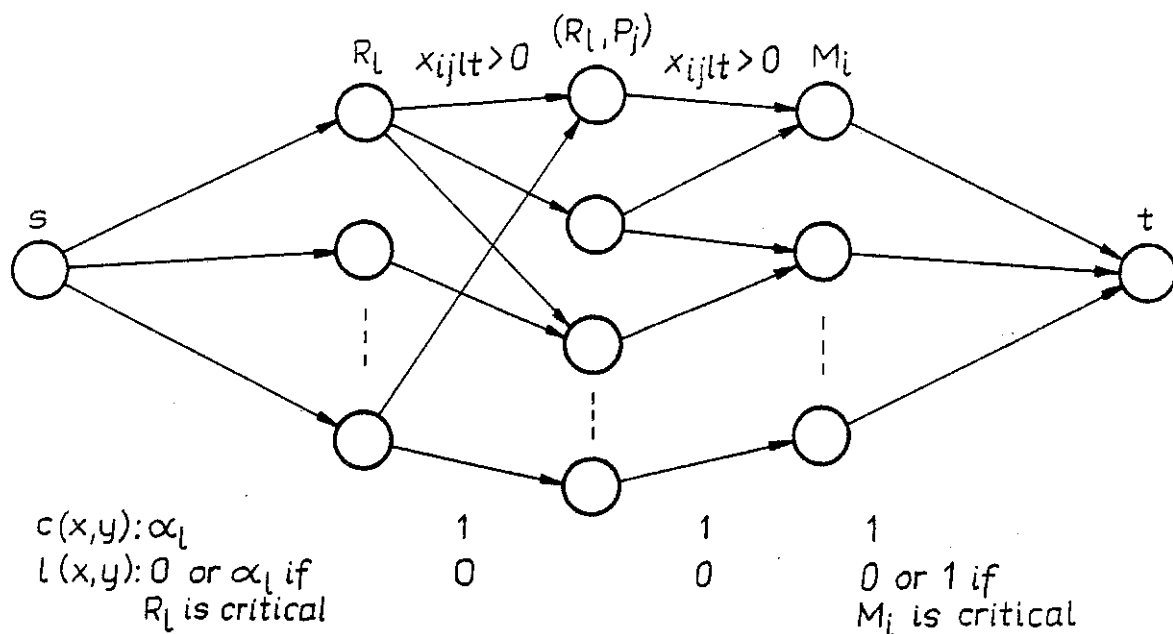


Fig. 1. Network $\mathcal{N}^1$

$\{s, t\} \cup \{R_l\}_{l=1}^p \cup \{M_i\}_{i=1}^m$ is the set of vertices of $\mathcal{N}^1$ and $c(x, y)$ and $l(x, y)$ denote lower and upper bounds for a flow on arc (x, y) , respectively. There is an arc between R_l and M_i iff $z_{il}^t > 0$. Taking the values $(s_{il}^t)_h$ of any integral feasible flow on arcs (R_l, M_i) we get a (0-1)-matrix S_h^t .

Let us remark that $\mathcal{N}^1$ is equivalent to $\mathcal{N}^2$ shown in Fig. 2.

Fig. 2. Network $\mathcal{N}^2$

$\{s, t\} \cup \{R_l\}_{l=1}^p \cup \{(R_l, P_j)\}_{l=1, j=1}^{p, n} \cup \{M_i\}_{i=1}^m$ is the set of vertices of $\mathcal{N}^2$. There is an arc between R_l and (R_l, P_j) on the one hand and between (R_l, P_j) and M_i on the other hand iff $x_{ijlt} > 0$.

Since $z_{i1}^t > 0$ iff there exists j such that $x_{ijlt} > 0$, and since $c(R_l, (R_l, P_j)) = c((R_l, P_j), M_i) = c(R_l, M_i)$, we claim that if there exists an integral flow in $\mathcal{N}^1$, then it exists also in $\mathcal{N}^2$, and conversely.

Thus, any integral flow on the arcs $((R_l, P_j), M_i)$ defines the assignment of product P_j to machine M_i and resource R_l during the time λ_h^t in period t - this assignment is called partial schedule h .

Now, we shall explain the calculation of λ_h^t for a given t and $h = 1, \dots, r_t$.

Let $h = 1$ and suppose that we know a feasible integral flow in $\mathcal{N}^2$. Let us also define set $E_1 = \{(i, j, 1) \mid \text{flow on } ((R_i, P_j), M_1) \text{ is equal to } 1\}$; of course, $\forall (i, j, 1) \in E_1, x_{ij1t} > 0$.

$$\lambda_1^t = \min\{x_{\min}, x_{\max}\}, \text{ where}$$

$$x_{\min} = \min_{(i,j,1) \in E_1} \{x_{ij1t}\},$$

$$x_{\max} = \min\left\{\min_i \left\{ \frac{\alpha_i I_t - \sum_{(i,j) \in G_1} x_{ij1t}}{\alpha_i - |G_1|} \right\}, \min_i \left\{ I_t - \sum_{(1,j) \in H_1} x_{ij1t} \right\} \right\}.$$

$$G_1 = \{(i, j) \mid (i, j, 1) \in E_1\}$$

$$H_1 = \{(1, j) \mid (1, j, 1) \in E_1\}$$

In order to find λ_2^t , one should decrease I_t and x_{ijl}^t for $(i, j, l) \in E_1$ by λ_1^t and restart calculation for a new feasible integral flow in $\mathcal{M}^2$, i.e. with E_2 . This procedure ends when $I_t = 0$.

Let us observe that λ_h^t is chosen so that either one of the positive elements x_{ijl}^t is reduced to zero or one more resource R_l or machine M_j becomes critical in period t . Either of these cases may happen a finite number of times thus it is another proof of the fact that the procedure is finite.

We shall illustrate the above procedure by the following numerical example.

Example

Two products $\{P_1, P_2\}$ have to be produced on two machines $\{M_1, M_2\}$ using two additional resources $\{R_1, R_2\}$; $\alpha_1 = \alpha_2 = 1$ units of each resource is available at any time. For the sake of clarity we consider only one period, $I = 10$ units long. Production plans concerning the two products are: $D_1 = 60$, $D_2 = 100$. The productivity of each triple (M_i, P_j, R_l) is the following:

$$\begin{aligned} k_{111} &= 8, k_{121} = 9, k_{112} = 10, k_{122} = 8, \\ k_{211} &= 6, k_{221} = 7, k_{212} = 8, k_{222} = 8. \end{aligned}$$

The cost coefficients penalizing deviations from production plans are all equal to 1.

Phase 1 consists in solving the following linear goal programming problem:

$$\left[\begin{array}{l} \text{Minimize : } f = y_1^+ + y_1^- + y_2^+ + y_2^- \\ \text{s.t. } x_{111} + x_{211} + x_{121} + x_{221} \leq 10 \\ x_{112} + x_{212} + x_{122} + x_{222} \leq 10 \\ x_{111} + x_{121} + x_{112} + x_{122} \leq 10 \\ x_{211} + x_{221} + x_{212} + x_{222} \leq 10 \\ 8x_{111} + 6x_{211} + 10x_{112} + 8x_{212} - y_1^+ + y_1^- = 60 \\ 9x_{121} + 7x_{221} + 8x_{122} + 8x_{222} - y_2^+ + y_2^- = 100 \\ x_{ijl}, y_j^+, y_j^- \geq 0 \quad (i = 1, 2; j = 1, 2; l = 1, 2) \end{array} \right.$$

The optimal solution is: $f = 0$, $x_{121} = 3$, $x_{221} = 7$, $x_{112} = 6$, $x_{222} = 3$; all other variables are equal to 0. This solution is presented graphically on the Gantt chart in Fig. 3.

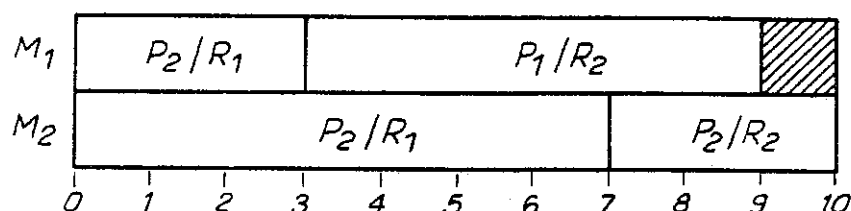


Fig. 3. Result of phase 1

In phase 2, first we have to find an integral feasible flow in the network shown in Fig. 4.

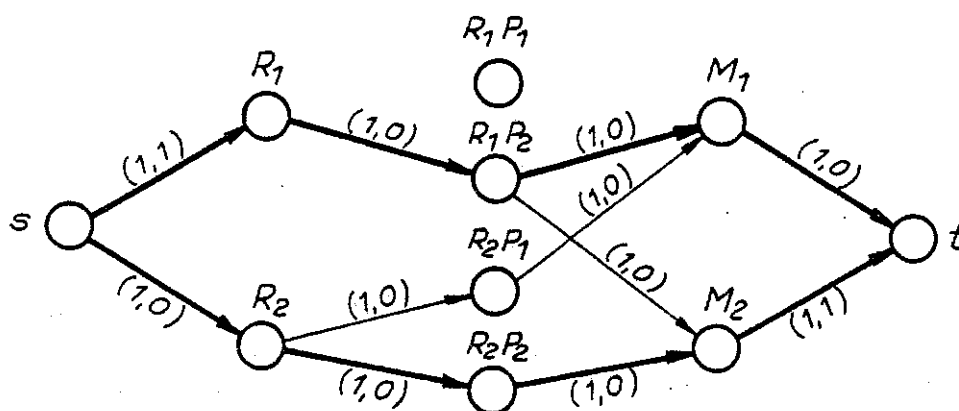


Fig. 4. Network N^2 for $I = 10$

The numbers in brackets indicate lower and upper bounds on arc flows. The feasible integral flow goes through the arcs drawn by thick lines.

The first partial schedule is defined by set $E_1 = \{(1, 2, 1), (2, 2, 2)\}$ and its duration λ_1 is calculated below.

$$x_{\min} = \min \{3, 3\} = 3$$

$$x_{\max} = \min \left\{ \min \left\{ \frac{10 - 10}{0}, \frac{10 - 9}{0} \right\}, \min \{10 - 6, 10 - 7\} \right\} = 3 \quad (1)$$

$$\lambda_1 = \min \{3, 3\} = 3$$

$$I = 10 - 3 = 7, x_{121} = 3 - 3 = 0, x_{222} = 3 - 3 = 0.$$

The second partial schedule is defined by the flow in the network shown in Fig. 5.

- (1) All terms $0/0$ are excluded from the expression.

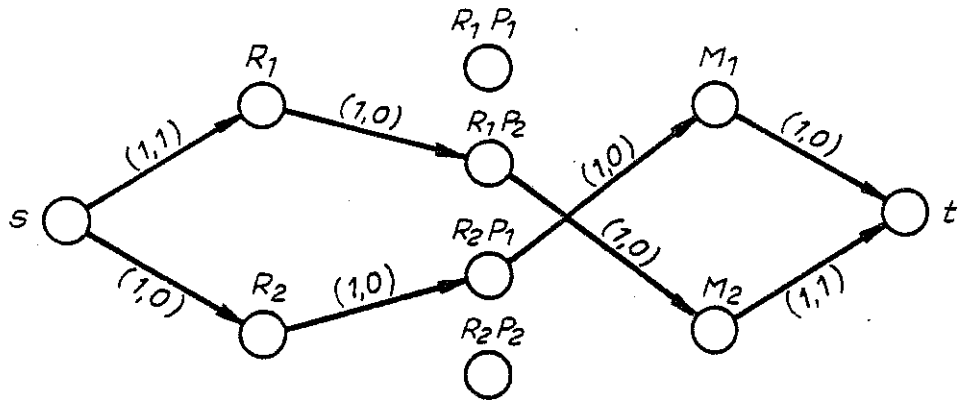


Fig. 5. Network N^2 for $I = 7$

$$E_2 = \{(1, 1, 2), (2, 2, 1)\}$$

$$x_{\min} = \min \{6, 7\} = 6$$

$$x_{\max} = \min \left\{ \min \left\{ \frac{7-6}{0}, \frac{7-7}{0} \right\}, \min \{7-0, 7-0\} \right\} = 7$$

$$\lambda_2 = \min \{6, 7\} = 6$$

$$I = 7 - 6 = 1, x_{112} = 6 - 6 = 0, x_{221} = 7 - 6 = 1.$$

The flow in the network shown in Fig. 6 defines the third partial schedule.

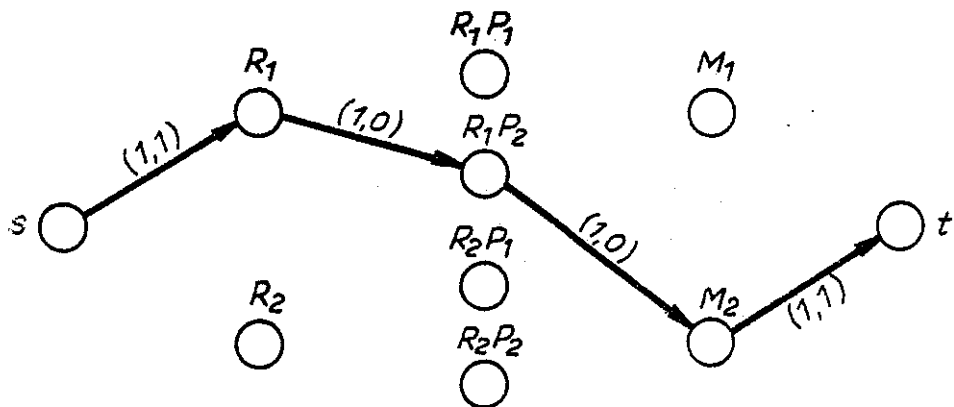


Fig. 6. Network N^2 for $I = 1$

$$\begin{aligned}
E_3 &= \{(2, 2, 1)\} \\
x_{\min} &= 1 \\
x_{\max} &= \min \left\{ \frac{1-1}{0}, 1-0 \right\} = 1 \\
\lambda_3 &= \min \{1, 1\} = 1 \\
I &= 1 - 1 = 0, \quad x_{221} = 1 - 1 = 0.
\end{aligned}$$

Since I has been reduced to 0, the procedure terminates ; the feasible schedule is shown in Fig. 7.

	$\lambda_1=3$	$\lambda_2=6$	$\lambda_3=1$
M_1	P_2/R_1	P_1/R_2	
M_2	P_2/R_2	P_2/R_1	P_2/R_1
	1	2	3 ← <i>partial schedule</i>

Fig. 7. Result of phase 2

3. MINIMIZATION OF THE TOTAL MACHINE SET-UP COST

Let us observe that set-up may take place only between two successive partial schedules. To be precise, set-up cost of machine M_i is greater than zero if there is a change in the assignment of P_j or R_j to M_i between two successive partial schedules (the idle time is also considered as a change because during this time R_j can be moved to another machine). By convention, the set-up cost is greater than zero for the first partial schedule in the sequence and it is equal to zero for the last one.

Of course, the sequence of partial schedules is not unique and it is interesting to find the sequence with the minimum total set-up cost. Similarly to [4], we will show how the problem of finding such a sequence can be formulated as the travelling salesman problem.

For all pairs of partial schedules of a given schedule we define a square matrix $A = [a_{ij}]$ ($i, j = 0, 1, \dots, r$), where r is the number of partial schedules and $i = 0, j = 0$ represent a dummy partial schedule which starts and finishes each sequence, respectively. $a_{ij} = \infty$ if $i = j$ or partial schedule j cannot precede partial schedule i because j belongs to a later period than i ; $a_{ij} \neq \infty$ is the machine set-up cost when i immediately precedes j . The total set-up cost is equal to $TSC = \sum_{i,j} a_{ij}^*$ where a_{ij}^* is one of $r+1$ elements of A which fulfill the following conditions :

- (a) there is exactly one element a_{ij}^* in each row and each column,
 (b) the indices of a_{ij}^* form a cycle : $0i, ij, jk, \dots, ps, s0$.

It is evident now that the problem of minimizing TSC is equivalent to the travelling salesman problem. Since TSC is a secondary schedule performance measure, a fast heuristic is advised to solve this problem.

Let us illustrate the minimization of TSC in a hypothetical schedule shown in Fig. 8.

	λ_1^1	λ_1^2	λ_3^1	λ_1^2	λ_3^2
M_4	P_2/R_1	P_6/R_3	P_2/R_1	P_2/R_1	P_3/R_1
M_3	P_3/R_2	P_4/R_2	P_4/R_2	P_3/R_2	P_1/R_3
M_2	P_3/R_3	P_6/R_2	P_3/R_3	P_4/R_3	P_5/R_2
M_1	P_1/R_2	P_1/R_1	P_5/R_2	P_6/R_2	
	I_1			I_2	
	1	2	3	4	5 ← partial schedules

Fig. 8. Example of an optimal schedule with regard to deviations from production plans

Matrix A defined from Fig. 8 for equal machine set-up costs has the following form :

A	0	1	2	3	4	5
0	∞	4	④	4	4	4
1	0	∞	4	2	②	4
2	0	4	∞	③	4	4
3	0	②	3	∞	3	4
4	0	∞	∞	∞	∞	④
5	①	∞	∞	∞	4	∞

The encircled entries of A are a_{ij}^* , defining the sequence of partial schedules with the minimum TSC equal to 15 ; the sequence is : 0-2-3-1-4-5-0. For comparison, the original sequence : 0-1-2-3-4-5-0, gives TSC = 18. The final schedule corresponding to the minimum TSC is shown in Fig. 9.

M_4	P_6/R_3	P_2/R_1		P_3/R_1
M_3	P_4/R_2	P_3/R_2		P_1/R_3
M_2	P_6/R_2	P_3/R_3	P_4/R_3	P_5/R_2
M_1	P_1/R_1	P_5/R_2	P_1/R_2	P_6/R_2
	I_1		I_2	

Fig. 9. Optimal schedule from Fig. 8 with the minimum TSC

4. CONCLUSIONS

We have shown the way of modelling and solving a production scheduling problem on parallel machines with additional (renewable) resources. The primary objective is the total cost following from deviations from production plans given as staircase functions of time. The secondary schedule performance measure is the total machine set-up cost. As to the computational complexity of the extended two-phase method, let us observe that the transformation of the problem into the LP problem in phase 1 is polynomial and that the sequence of compatible flow problems in phase 2 can be solved in polynomial time.

Let us end with a comment about possible generalizations of the presented approach. Firstly, as shown in [4], the two-stage approach can easily handle multiple category resources, so apart from renewable resources we could also consider nonrenewable and doubly-constrained resources. Secondly, it may be worthwhile to check whether leaving an optimum with respect to the primary criterion, it is possible to decrease the total set-up cost so that the global utility of the schedule gets increased. The Reverse Simplex Method can then be used in phase 1 to generate feasible solutions differing from the optimum by a given relaxation (cf. the minimization of the number of preemptions in [2]).

REFERENCES

- [1] C. BERGE : Graphes, Gauthier-Villars, Paris, 1983.
- [2] J. BLAZEWICZ, W. CELLARY, R. SLOWINSKI, J. WEGLARZ : Scheduling under Resource Constraints : Deterministic Models, J.C. Baltzer AG, Basel, 1986.
- [3] M. COCHAND, D. de WERRA, R. SLOWINSKI : Preemptive scheduling with staircase and piecewise linear resource availability, O.R. Working Paper 85/1, Ecole Polytechnique Fédérale de Lausanne, Dép. de Math., Lausanne, January 1985.
- [4] R. SLOWINSKI : Multiobjective network scheduling with efficient use of renewable and nonrenewable resources, European J. Operational Res. 7 (1981) 265-273.
- [5] R. SLOWINSKI : L'ordonnancement des tâches préemptives sur les processeurs indépendants en présence de ressources supplémentaires, RAIRO Informatique/Computer Sci. 15 (1981) 155-166.
- [6] R. SLOWINSKI : Preemptive scheduling of independent jobs on parallel machines subject to financial constraints, European J. Operational Res. 15 (1984) 366-373.
- [7] D. de WERRA : Preemptive scheduling, linear programming and network flows, SIAM J. on Algebraic and Discrete Methods 5 (1984) 11-20.