REVUE FRANÇAISE D'AUTOMATIQUE, INFORMATIQUE, RECHERCHE OPÉRATIONNELLE. RECHERCHE OPÉRATIONNELLE

JEAN-MARIE PROTH
ANDRÉ RAZAFINDRAKOTO

Amélioration d'un ordonnancement dans le cas d'une fabrication de type linéaire

Revue française d'automatique, informatique, recherche opérationnelle. Recherche opérationnelle, tome 6, nº V3 (1972), p. 17-26
<http://www.numdam.org/item?id=RO_1972__6_3_17_0>

R.A.I.R.O.
(6e année, novembre 1972, V-3, p. 17 à 26)

AMELIORATION D'UN ORDONNANCEMENT DANS LE CAS D'UNE FABRICATION DE TYPE LINEAIRE

par Jean-Marie PROTH [1] et André RAZAFINDRAKOTO [1]

Résumé. — *L'article proposé s'intéresse à la diminution du temps total T de passage d'un ensemble de produits sur une suite de machines.*

Chaque produit subit une transformation sur chacune des machines, dans un ordre fixé. Cet ordre est le même pour tous les produits. D'autre part l'ordre de passage de l'ensemble des produits sur une machine reste le même pour chaque machine, et est fixé au moment du lancement.

Partant d'un ordre de lancement quelconque, nous sommes parvenus à le modifier en modifiant T. Il apparaît que la méthode est d'autant plus avantageuse qu'on se limite à un faible nombre d'essais. Une amélioration de la méthode est également proposée.

I) Position du problème

Le problème que nous nous proposons d'examiner se pose dans l'industrie lorsque les conditions suivantes sont réunies :

1) Chaque produit doit subir, successivement et dans cet ordre, les transformations $O_1, O_2, ..., O_n$. Nous désignerons par M_i l'atelier chargé de la transformation O_i.

2) Lorsqu'un ensemble $\{ C_1, ..., C_m \}$ de produits est lancé en fabrication, l'ordre de passage est fixé à l'entrée de l'atelier M_1 et ne peut être modifié par la suite.

Nous connaissons la matrice $C = [t_{ij}]$, de dimensions $(m \times n)$, où t_{ij} représente le temps nécessaire au i^{me} produit de la suite $C_1, C_2, ..., C_m$ pour subir l'opération O_j.

Il s'agit de trouver un ordre de passage tel que le temps T séparant l'entrée du premier produit dans M_1 de la sortie du dernier produit de M_n soit minimum.

(1) Université de Tananarive, Madagascar.

Remarquons qu'un tel problème sort du cadre du problème central de l'ordonnancement (voir [1] et [2]).

La méthode que nous proposons dans la suite est une méthode heuristique qui a donné, dans la pratique, d'excellents résultats. On trouvera un exemple d'application à la fin de l'exposé. Partant de l'ordre de passage $C_1, C_2, \ldots, C_m$, elle conduit très rapidement, par modification de cet ordre, à une diminution de T. Comme nous le verrons, il est cependant souhaitable d'interrompre le processus suffisamment tôt pour assurer sa rentabilité.

II) Notations

R désignant l'ensemble des réels, considérons $E = RU\{\alpha\}$, où α peut être interprété comme $-\infty$.

Alors (voir [1] et [3]) nous pouvons définir sur E l'addition symbolique, notée $\dot{+}$ de la manière suivante :

$$\forall (x, y) \in R^2 : x \dot{+} y = \text{Max}\,(x, y)$$

et

$$\forall x \in R \quad : x \dot{+} \alpha = \alpha \dot{+} x = x$$

Cette loi est commutative, associative, et admet α comme élément neutre.

III) Recherche du temps T pour un ordre de passage donné

Soit $\{ C_1, C_2, \ldots, C_m \}$ l'ordre de passage fixé à l'entrée de M_1 et $C = [t_{ij}]$ la matrice des temps correspondante.

Dans toute la suite, nous prendrons comme origine des temps l'instant où C_1 entre dans M_1 et nous désignerons par $s_{p,l}$ l'instant où le produit C_p sort de l'atelier M_l.

Alors :

$$s_{1,1} = t_{11} \quad \text{et} \quad s_{p,1} = s_{p-1,1} + t_{p,1}$$

si bien que :

$$\text{(1)} \qquad \boxed{s_{p,1} = \sum_{i=1}^{p} t_{i,1} \quad \text{pour} \quad 1 \leqslant p \leqslant m}$$

De plus : $s_{1,l} = s_{1,l-1} + t_{1,l}$ ce qui entraîne :

$$\text{(2)} \qquad \boxed{s_{1,l} = \sum_{i=1}^{l} t_{1,i} \quad \text{pour} \quad 1 \leqslant l \leqslant n}$$

Enfin, si l'on remarque que, pour que le produit C_p entre dans l'atelier M_l il faut que C_p soit sorti de M_{l-1} et que C_{p-1} soit sorti de M_l, la relation suivante est évidente :

$$\boxed{\begin{aligned} & s_{p,l} = (s_{p,l-1} \dotplus s_{p-1,l}) + t_{p,l} \\ & \text{pour} \quad \begin{cases} 1 < p \leqslant m \\ 1 < l \leqslant n \end{cases} \end{aligned}} \tag{3}$$

Ces trois relations nous permettent de construire la matrice $B = [s_{ij}]$ de dimensions $(m \times n)$ et :

$$\boxed{T = s_{m,n}}$$

Nous sommes donc en mesure de calculer T pour un ordre de passage donné.

IV) Remarques et définitions

Considérons l'ensemble des suites de $m + n - 1$ éléments $(\tau_1, \tau_2, \ldots, \tau_{m+n-1})$ définies comme suit :

1) $\tau_1 = t_{11}$

2) *si* $\tau_j = t_{p,l}(1 < j < m + n - 1,\ p + l = j + 1)$ alors :

   si $l = n$, alors $\tau_{j+1} = t_{p+1,l}$

   sinon

      si $p = m$, alors $\tau_{j+1} = t_{p,l+1}$

      sinon $\tau_{j+1} = t_{p+1,l}$ ou $\tau_{j+1} = t_{p,l+1}$

Remarquons que $\tau_{m+n-1} = t_{m,n}$, quelle que soit la suite envisagée. Une telle suite sera désignée par $\mathcal{C}$ et appelée « chemin de la matrice C ». La longueur de ce chemin sera la somme :

$$l_{\mathcal{C}} = \sum_{i=1}^{m+n-1} \tau_i$$

La matrice C étant de dimensions $m \times n$, il est facile de voir qu'il existe

$$\frac{(m + n - 2)\,!}{(m - 1)\,!\,(n - 1)\,!} \text{ chemins différents de } C.$$

Nous définissons de la même manière les chemins $\mathcal{C}' = (\tau'_1, \ldots, \tau'_{m+n-1})$ sur B.

Nous dirons qu'un chemin $\mathcal{C}$ défini sur C et un chemin $\mathcal{C}'$ défini sur B sont correspondants si :

$$t_{ij} \in \mathcal{C} \Leftrightarrow s_{ij} \in \mathcal{C}'(i = 1, \ldots, m \quad \text{et} \quad j = 1, \ldots, n)$$

Considérons alors le chemin $\mathcal{C}'_1$ de B défini de la manière suivante :

1) $s_{m,n} \in \mathcal{C}'_1$

2) si $s_{p,q} \in \mathcal{C}'_1$

alors :

si $p = 1$, alors $s_{1,q-1}, s_{1,q-2}, \ldots, s_{1,1} \in \mathcal{C}'_1$

sinon

si $q = 1$, alors $s_{p-1,1}, s_{p-2,1}, \ldots, s_{1,1} \in \mathcal{C}'_1$

sinon

si $s_{p-1,q} \dotplus s_{p,q-1} = s_{p,q-1}$, alors $s_{p,q-1} \in \mathcal{C}'_1$

sinon $s_{p-1,q} \in \mathcal{C}'_1$

V) Proposition

Pour le chemin $\mathcal{C}'_1$ ainsi défini, nous pouvons établir la proposition suivante :

Si le chemin $\mathcal{C}_1$ de C est le chemin correspondant à $\mathcal{C}'_1$, alors $\mathcal{C}_1$ est le chemin de longueur maximum de C et $l_{\mathcal{C}_1} = s_{m,n} = T$, où m est le nombre de lignes et n le nombre de colonnes de C et B.

Démonstration

Dans la suite, pour $u \leqslant m$ et $p \leqslant n$, nous noterons $C_{u,p}$ la matrice formée des éléments t_{ij} de C, pour $i = 1, \ldots, u$ et $j = 1, \ldots, p$. De même, la matrice $B_{u,p}$ sera formée des éléments $s_{i,j}$ de B pour $i = 1, \ldots, u$ et $j = 1, \ldots, p$.

Enfin, $\mathcal{C}'_1(u, p)$ sera le chemin de $B_{u,p}$ défini comme $\mathcal{C}'_1$ sur B, et $\mathcal{C}_1(u, p)$ sera le chemin de $C_{u,p}$ correspondant à $\mathcal{C}'_1(u, p)$.

1) Alors la proposition ci-dessus est évidemment vraie dans les deux cas suivants :

a) pour les matrices $C(1, p)$ et $B(1, p)$ $(p = 1, \ldots, n)$.

En effet, dans ce cas, le chemin est unique et

$$l_{\mathcal{C}_{1(1,p)}} = \sum_{i=1}^{p} t_{1,i} = s_{1,p}$$

b) pour les matrices $C(u, 1)$ et $B(u, 1)$ $(u = 1, ..., m)$.

Dans ce cas également, le chemin est unique et

$$l_{\mathcal{C}_{1(u,1)}} = \sum_{j=1}^{u} t_{j,1} = s_{u,1}$$

2) Montrons que si la proposition est vraie pour $C(1, p)$, $B(1, p)$ $(p = 1, ..., n)$ et pour $C(u, r)$, $B(u, r)$ $(u = 1, ..., m$ et $1 \leqslant r < n)$, alors elle est encore vraie pour $C(u, r+1)$, $B(u, r+1)$.

a) La proposition est vraie pour $C(1, r+1)$, $B(1, r+1)$, par hypothèse.

b) Si la proposition est vraie pour $C(u, r+1)$, $B(u, r+1)$ $(1, \leqslant u \leqslant m)$, elle est encore vraie pour $C(u+1, r+1)$, $B(u+1, r+1)$.

En effet :

$$\begin{aligned} s_{u+1,r+1} &= (s_{u,r+1} + s_{u+1,r}) + t_{u+1,r+1} \\ &= (l_{\mathcal{C}_{1\,(u,r+1)}} + l_{\mathcal{C}_{1(u+1\,r)}}) + t_{u+1,r+1} \\ &= (l_{\mathcal{C}_{1\,(u,r+1)}} + t_{u+1,r+1}) + (l_{\mathcal{C}_{1\,(u+1,r)}} + t_{u+1,r+1}) \\ &= l_{\mathcal{C}_{1\,(u+1,\,r+1)}} \end{aligned}$$

ce qui achève la démonstration.

L'algorithme qui suit est basé sur cette proposition.

VI) Présentation de l'algorithme

Changer l'ordre de passage des produits correspondant à un temps T, c'est effectuer une permutation des lignes de la matrice C. Soit C_1 la matrice obtenue et T_1 la nouvelle valeur de T.

Remarquons que :

1) si $(t_{i,p-1}, t_{i,p}, ..., t_{i+r,p}, t_{i+r,p+1}) \subset \mathcal{C}_1$, chemin de longueur maximum de C $(i \geqslant 1,\ i+r \leqslant m,\ 1 < p < n)$, alors toute permutation effectuée sur les lignes $i+1,\ i+2,\ ..., i+r-1$ est telle que la longueur du chemin $\mathcal{C}_{11}$ formé des éléments de C_1 de même position que les éléments de $\mathcal{C}_1$ dans C, est égale à la longueur de $\mathcal{C}_1$:

$$l_{\mathcal{C}_{11}} = l_{\mathcal{C}_1}$$

Or, compte tenu de la proposition établie précédemment :

$$T_1 \geqslant l_{\mathcal{C}_{11}} = l_{\mathcal{C}_1} = T$$

T ne peut donc être diminué par une telle permutation, donc aucune permutation de ce type ne sera essayée.

2) Avec les mêmes notations, si nous considérons une permutation telle que $l_{\mathcal{C}_{11}} \geqslant l_{\mathcal{C}_1}$, alors : $T_1 \geqslant l_{\mathcal{C}_{11}} \geqslant l_{\mathcal{C}_1} = T$.

Nous ne calculerons donc pas non plus les valeurs de T pour les permutations telles que $l_{\mathcal{C}} \geqslant l_{\mathcal{C}_1}$.

Seules les permutations donnant $l_{\mathcal{C}_{11}} < l_{\mathcal{C}_1}$ peuvent entraîner $T_1 < T$.

L'algorithme peut donc se résumer comme suit :

1) Donnée de la matrice C correspondant à un ordre de passage $(C_1, ..., C_m)$ quelconque.

2) Construction de la matrice B correspondante.

3) Recherche du chemin $\mathcal{C}_1'$ de B.

4) Recherche du chemin $\mathcal{C}_1$ de C correspondant à $\mathcal{C}_1'$.

5) Recherche d'une permutation telle que $l_{\mathcal{C}_{11}} < l_{\mathcal{C}_1}$ (notations précédentes). Soit C_1 la matrice correspondant au nouvel ordre de passage.

6) Construction de la matrice B_1 correspondante.

7) *Si* $T_1 \geqslant T$, retour en 5).

Sinon, faire $C = C_1$ et $B = B_1$, puis retour en 3).

Comme il a été dit en début de cet exposé, il est souhaitable d'interrompre ce processus suffisamment tôt pour assurer sa rentabilité. On constate d'ailleurs, dans tous les cas, que le nombre d'essais ne diminuant pas T, et situés entre deux essais successifs diminuant T, augmente au fur et à mesure que le processus se développe.

L'exemple porté en fin d'exposé le montrera clairement.

Nous avons donc décidé d'interrompre le processus lorsque le nombre total d'essais atteint M *ou* lorsque le nombre d'essais successifs ne diminuant pas T devient égal à N. M et N sont donnés en entrée du programme.

VII) Amélioration de l'algorithme

L'amélioration que nous proposons consiste à prendre une situation de départ déjà favorable.

Considérons la matrice C des temps de fabrication et supposons qu'elle puisse s'écrire formellement de la manière suivante :

$$C = \begin{array}{|c|c|} \hline P_{\mathrm{I}} & G_{\mathrm{II}} \\ \hline G_{\mathrm{III}} & P_{\mathrm{IV}} \\ \hline \end{array}$$

Dans ce schéma, P représente un ensemble de temps « faibles » et G un ensemble de temps « élevés », ces notions de temps « faibles et de temps « élevés » étant évidemment relatives.

Nous avons vu que T est la longueur maximum des chemins de C. Or la disposition précédente interdit à tout chemin passant par la zone II de passer par la zone III et réciproquement. Si bien que la disposition précédente correspond nécessairement à un ordonnancement déjà favorable.

Partant de cette idée, nous avons procédé de la manière suivante :

1) Classement des m lignes de C par ordre croissant de la somme des $\left[\frac{n}{2}\right]$ premiers éléments de chaque ligne.

2) Classement des m lignes de C par ordre décroissant de la somme des éléments de rang $\left[\frac{n}{2}\right] + 1, \left[\frac{n}{2}\right] + 2, ..., n$ de chaque ligne.

3) Interclassement, en considérant successivement le dernier élément du premier classement, le premier élément du second classement, l'avant-dernier élément du premier classement, etc...

Si par exemple les classements ont donné les résultats suivants (cas de 6 produits) :

1er classement : 1, 4, 3, 2, 6, 5

2e classement : 6, 2, 4, 3, 1, 5

alors l'interclassement conduira à :

6, 4, 1, 3, 2, 5

L'expérience a montré que, dans la plupart des cas, cette amélioration conduit à une situation de départ bien meilleure que dans le cas général. Nous allons présenter un exemple de problème traité avec chacun de ces deux programmes.

VIII) Résultats obtenus

Nous présentons ici des résultats concernant une matrice $C(20, 10)$ (c'est-à-dire 20 produits et 10 machines). Le processus était interrompu lorsque 100 essais étaient effectués au total *ou* lorsque 11 essais successifs n'entraînaient aucune amélioration de T.

1) *Algorithme simple*

Ordre de passage																				Valeur de T
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	9 9 8 9.0
4	2	3	1	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	9 9 8 4.0
11	2	3	1	5	6	7	8	9	10	4	12	13	14	15	16	17	18	19	20	9 3 6 2.5
2	11	3	1	5	6	7	8	9	10	4	12	13	14	15	16	17	18	19	20	9 2 7 7.0
3	11	2	1	5	6	7	8	9	10	4	12	13	14	15	16	17	18	19	20	9 1 7 9.5
5 essais successifs sans amélioration de T																				
15	11	2	1	5	6	7	8	9	10	4	12	13	14	3	16	17	18	19	20	8 8 3 7.0
2	11	15	1	5	6	7	8	9	10	4	12	13	14	3	16	17	18	19	20	8 7 3 4.6
1	11	15	2	5	6	7	8	9	10	4	12	13	14	3	16	17	18	19	20	8 5 9 4.6
4	11	15	2	5	6	7	8	9	10	1	12	13	14	3	16	17	18	19	20	8 4 7 3.6
1 essai sans amélioration de T																				
13	11	15	2	5	6	7	8	9	10	1	12	4	14	3	16	17	18	19	20	8 3 0 7.0
15	11	13	2	5	6	7	8	9	10	1	12	4	14	3	16	17	18	19	20	8 2 2 2.5
2	11	13	15	5	6	7	8	9	10	1	12	4	14	3	16	17	18	19	20	8 1 9 6.0
1 essai sans amélioration de T																				
5	11	13	15	2	6	7	8	9	10	1	12	4	14	3	16	17	18	19	20	8 1 7 8.0
7	11	13	15	2	6	7	8	9	10	1	12	4	14	3	16	17	18	19	20	8 0 9 3.1
7 essais successifs sans amélioration de T																				
4	11	13	15	2	6	5	8	9	10	1	12	7	14	3	16	17	18	19	20	8 0 5 4.6
5 essais successifs sans amélioration de T																				
17	11	13	15	2	6	5	8	9	10	1	12	7	14	3	16	4	18	19	20	7 9 6 7.0
9 essais successifs sans amélioration de T																				
18	11	13	15	2	6	5	8	9	10	1	12	7	14	3	16	4	17	19	20	7 8 9 8.1
4 essais successifs sans amélioration de T																				
3	11	13	15	2	6	5	8	9	10	1	12	7	14	18	16	4	17	19	20	7 6 7 7.5
9 essais successifs sans amélioration de T																				

Ordre de passage																				Valeur de T
20	11	13	15	2	6	5	8	9	10	1	12	7	14	18	16	4	17	19	3	6 9 7 2.0
5	11	13	15	2	6	20	8	9	10	1	12	7	14	18	16	4	17	19	3	6 9 5 5.0
7 essais successifs sans amélioration de T																				
5	9	13	15	2	6	20	8	11	10	1	12	7	14	18	16	4	17	19	3	6 9 1 8.0
5 essais successifs sans amélioration de T																				
5	20	13	15	2	6	9	8	11	10	1	12	7	14	18	16	4	17	19	3	6 9 1 2.0
11 essais successifs sans amélioration de T																				

Soit au total 86 essais qui ont conduit à une amélioration de 30,8 pour cent du temps initial.

2) *Algorithme amélioré*

L'expérience a été reprise avec la même matrice, c'est-à-dire avec l'ordre de passage initial :

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

et un temps de passage :

$$T = 9\ 9\ 8\ 9\ 0$$

Après les classements, puis l'interclassement initial, très rapides, la situation était la première du tableau qui suit, avec un temps T nettement meilleur que dans le cas précédent.

Ordre de passage																				Valeur de T
15	2	7	6	9	4	16	19	18	5	20	1	3	17	14	8	10	13	12	11	7 5 1 3.2
19	2	7	6	9	4	16	15	18	5	20	1	3	17	14	8	10	13	12	11	7 2 9 5.2
2	19	7	6	9	4	16	15	18	5	20	1	3	17	14	8	10	13	12	11	7 2 9 3.8
18	19	7	6	9	4	16	15	2	5	20	1	3	17	14	8	10	13	12	11	7 2 0 8.2
4	19	7	6	9	18	16	15	2	5	20	1	3	17	14	8	10	13	12	11	7 2 0 5.2
2 essais successifs sans amélioration de T																				
13	19	7	6	9	18	16	15	2	5	20	1	3	17	14	8	10	4	12	11	6 9 0 0.2
2 essais successifs sans amélioration de T																				
18	19	7	6	9	13	16	15	2	5	20	1	3	17	14	8	10	4	12	11	6 8 3 5.7
11 essais successifs sans amélioration de T																				

Soit au total 22 essais seulement qui ont conduit à une valeur de T inférieure à la valeur obtenue par l'algorithme simple.

Les expériences effectuées jusqu'à présent ont toutes donné des résultats analogues.

Il semblerait par ailleurs que l'on puisse encore améliorer les algorithmes précédents en choisissant, à partir d'un certain seuil, les permutations qui diminuent peu la longueur du chemin $\mathcal{C}_1$. On évite ainsi de nombreux essais infructueux, mais on complique considérablement le programme.

On pourrait aussi penser à choisir plusieurs chemins de C, dont $\mathcal{C}_1$, et comparer les variations de longueur pour ces différents chemins. Là encore, l'algorithme se trouverait alourdi.

BIBLIOGRAPHIE

[1] R. CRUON et Ph. HERVÉ, *Quelques résultats relatifs à une structure algébrique et à son application au problème central de l'ordonnancement*, Revue Française de Recherche Opérationnelle, n° 34, 1er trim. 1965.

[2] Bernard ROY, *Physionomie et traitement des problèmes d'ordonnancement*, Gestion, numéro spécial, avril 1963.

[3] C. BERGE, *Théorie des graphes et ses applications*, Dunod, Paris, 1963.