

CAHIER DU LAMSADE

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

(Université Paris-Dauphine)

Equipe de Recherche Associée au C.N.R.S. N° 656

PROBLEMES D'ALLOCATION DE MOYENS LIMITES

EN PRESENCE DE CRITERES MULTIPLES

CAHIER N° 36
juin 1981

R. SLOWINSKI (*)

(*) de l'Institut d'Automatique de l'Ecole Polytechnique de Poznan.
Travail réalisé dans le cadre d'un séjour au LAMSADE.

Je tiens à adresser mes vifs remerciements au Professeur Bernard Roy qui a rendu possible mon séjour fructueux au LAMSADE ainsi qu'à Catherine Dupont-Gatelmand dont les remarques m'ont permis d'améliorer ce cahier.

SOMMAIRE

	<u>Pages</u>
<u>ABSTRACT</u>	I
<u>RESUME</u>	II
1. <u>INTRODUCTION</u>	1
2. <u>DEFINITION DU SYSTEME D'ALLOCATION DE MOYENS</u>	3
3. <u>LE CHAMP DE LA RECHERCHE ET LES DOMAINES D'APPLICATIONS</u>	12
4. <u>APPROCHES DES PROBLEMES D'ALLOCATION DE MOYENS DE CATEGORIES DIFFERENTES POUR DES OPERATIONS INTERRUPTIBLES</u>	17
4.1 L'approche en une phase	17
4.2 L'approche en deux phases	21
4.3 La minimisation du nombre d'interruptions des opérations	27
5. <u>METHODES D'ALLOCATION DE MOYENS DE CATEGORIES DIFFERENTES POUR DES OPERATIONS NON INTERRUPTIBLES</u>	31
6. <u>REMARQUES CONCERNANT L'ASPECT MULTICRITERE DES PROBLEMES D'ALLOCATION DE MOYENS</u>	35
<u>REFERENCES</u>	37

ABSTRACT

This paper constitutes a brief synthesis of author's research results in scheduling. The particular problems involved are those of deterministic allocation of constrained resources of different categories among precedence related operations in the presence of multiple criteria. Three categories of resources are considered : renewable (only total usage at every moment is constrained), non-renewable (only total consumption over a time period is constrained) and doubly-constrained (both usage and consumption are constrained). For every feasible combination of resource amounts, the performing time of each operation is known. These combinations constitute a finite set of performing modes given for each operation. Time and cost criteria are considered for project performance evaluation. For these problems, several algorithms are proposed, and are comprehensively described from the viewpoint of their computational complexity and the range of problems being handled. The number of preemptions introduced by the algorithms for the case of splittable activities is considered both in the sense of proving its upper bound for the obtained schedules and as an additional criterion. The problems considered in this paper constitute a large class of practical situations which until now have not been investigated or formalized, although they can be encountered in many technical applications. Some examples of them are given here.

RESUME

Ce cahier constitue une brève synthèse des travaux de l'auteur en matière d'ordonnancement. Il s'agit en particulier des problèmes déterministes d'allocation de moyens limités de catégories différentes à des opérations reliées par des contraintes de précédence, en présence de critères multiples. On prend en compte trois catégories de moyens : renouvelables, non renouvelables et doublement limités, selon que les contraintes de disponibilité concernent la quantité disponible à chaque instant, la consommation sur un intervalle de temps ou les deux conjointement. Le temps d'exécution de toute opération dépend du mode d'affectation des moyens choisi pour cette opération parmi un ensemble fini de modes. Les critères considérés sont de type temps et coût. On propose des algorithmes de résolution de ces problèmes, caractérisés du point de vue de la complexité de calcul et de la capacité d'assimiler différents cas. Dans le cas des opérations interruptibles, on tient compte du nombre d'interruptions, premièrement par l'évaluation de sa borne supérieure dans un ordonnancement construit selon un algorithme proposé et, deuxièmement, par sa minimisation dans cet ordonnancement. Les problèmes abordés dans cet article englobent une large classe des situations réelles qui n'étaient pas considérées jusqu'ici et qu'on retrouve dans plusieurs applications techniques dont les exemples sont cités.

1. INTRODUCTION

La problématique d'allocation de moyens qui servent à exécuter un ensemble d'opérations (tâches) décrites par certains modèles mathématiques et reliées par des dépendances temporelles embrasse aujourd'hui plusieurs directions de recherche. De manière générale, elle englobe des classes de problèmes telles que l'ordonnancement d'atelier, l'allocation de moyens soumis aux contraintes de disponibilité concernant soit la quantité accessible à chaque instant (type machines), soit la consommation sur un intervalle de temps (type argent), l'équilibrage d'une chaîne de montage, la gestion de production et l'ordonnancement des systèmes informatiques, tout cela dans un contexte déterministe ou probabiliste. Chacune de ces classes est caractérisée par une terminologie et une méthodologie bien spécifiques. Il est clair que la dispersion des résultats et les différences de terminologie empêchent à la fois la reconnaissance de l'état de l'art dans la problématique en question et la définition des directions de recherche qui seraient souhaitables et réellement nouvelles. Toutefois, il est nécessaire d'avoir une vue d'ensemble de cette problématique afin de pouvoir modéliser et résoudre des problèmes plus réels, donc plus complexes, d'allocation de moyens, notamment d'un point de vue des caractéristiques des ensembles de moyens et d'opérations.

A cette fin, nous avons proposé un langage commun pour modéliser les problèmes d'allocation de moyens. Ce langage nous a permis de formaliser une large classe de problèmes réels et de proposer une méthodologie pour leur résolution ([21], [29]). Ce langage s'est bien évidemment formé en parallèle avec des travaux effectués sur certains problèmes particuliers d'allocation de moyens.

Dans la section suivante, nous introduisons certains éléments de ce langage, c'est-à-dire ceux qui sont nécessaires à la définition du système d'allocation de moyens que nous abordons dans cette étude. En utilisant cette terminologie, nous précisons, dans la section 3, comment est située cette étude par rapport aux autres travaux effectués en matière d'ordonnancement, quel est son champ d'applications et quel est le contenu des sections suivantes.

2. DEFINITION DU SYSTEME D'ALLOCATION DE MOYENS

Le système d'allocation de moyens est une réunion de trois ensembles $\mathcal{S} = (\mathcal{R}, \mathcal{A}, \mathcal{Q})$ où $\mathcal{R}$ est un ensemble de moyens (ressources), $\mathcal{A}$ est un ensemble d'opérations (activités, tâches) et $\mathcal{Q}$ est un ensemble de critères d'exécution de l'ensemble $\mathcal{A}$. Nous allons définir chacun de ces ensembles ainsi que la commande dans le système $\mathcal{S}$.

L'ensemble $\mathcal{R}$ est composé d'unités de moyens qui sont classés en types selon leurs fonctions. Au même type on associe toutes les unités de moyens dont les fonctions sont identiques. L'identité de fonctions ne signifie pourtant pas l'identité de toutes les caractéristiques de ces unités comme la vitesse ou l'efficacité de travail. Nous parlons donc d'unités identiques ou non-identiques de moyens du même type.

Indépendamment du classement en types, nous distinguons les catégories de moyens. Le classement en catégories peut se réaliser de différents points de vue ([18], [29]); ainsi, en considérant le type des contraintes portant sur la disponibilité des moyens, nous distinguons trois catégories suivantes :

- les moyens renouvelables $R_1^0, \dots, R_p^0$ pour lesquels seulement le nombre d'unités disponibles à chaque instant est limité et égal à N_k unités pour le type $k, k = 1, \dots, p$;

- les moyens non renouvelables $R_1^n, \dots, R_v^n$ pour lesquels seulement la consommation dans une période donnée est limitée et égale à B_k unités pour le type $k, k = 1, \dots, v$;

- les moyens doublement limités $R_1^p, \dots, R_u^p$ pour lesquels le nombre d'unités à chaque instant ainsi que la consommation dans une période donnée sont limités et égaux à N_k^p et B_k^p respectivement pour le type $k, k = 1, \dots, u$.

Ajoutons que les contraintes de disponibilité ne sont pas nécessairement actives ; on connaît des cas où le niveau de charge ou la consommation sont les critères d'allocation de moyens.

La plupart des moyens-matériaux est doublement limitée : ainsi la puissance électrique dont la disponibilité à chaque instant et la consommation (l'énergie) sont limitées, le débit du combustible dont l'intensité et le volume sont limités, enfin l'argent dans le cas où le taux d'investissement ainsi que la somme totale sont limités. Notons également que des moyens tels que la main-d'oeuvre, des machines-outils, la puissance et le débit sont considérés en général comme renouvelables et des moyens tels que l'argent, l'énergie, le combustible, les matières premières, comme non renouvelables. Cette simplification est souvent justifiée dans la pratique mais on en abuse aussi souvent.

Pour achever la description de l'ensemble $\mathcal{R}$, nous pouvons préciser que chaque unité de moyen peut être affectée à une opération au plus dans chaque instant.

L'ensemble $\mathcal{A}$ est composé d'opérations $A_1, \dots, A_n$ avec $n > 1$. Une opération A_i , $i = 1, \dots, n$, est définie par les caractéristiques suivantes :

- les besoins en moyens ;
- le modèle mathématique ;
- l'information sur la possibilité d'interruption ;
- la date de début au plus tôt (la date d'arrivée) a_i ;
- la date de fin au plus tard d_i .

Expliquons ces caractéristiques pour une opération $A_i \in \mathcal{A}$.

Les besoins en moyens concernent les nombres ou les indices des unités de moyens de types particuliers qui sont nécessaires pour l'exécution de A_i . Nous distinguons des besoins discrets, si les nombres (ou indices) d'unités de moyens que l'on peut affecter à l'opération A_i sont éléments

d'un ensemble fini, et des besoins continus, si les nombres d'unités de moyens sont éléments d'un ensemble non dénombrable (un intervalle).

A partir des besoins en moyens connus pour A_i , on peut définir tous les modes possibles d'exécution de A_i , c'est-à-dire les affectations possibles des moyens de types et catégories particuliers à cette opération. Evidemment, si les besoins en moyens sont de type continu, l'ensemble des modes d'exécution est non dénombrable et, dans le cas de besoins discrets, fini. Dans ce dernier cas, nous représentons cet ensemble sous la forme d'une matrice $\underline{R}_i$ de dimensions $h_i \times (p + v + u)$ où h_i désigne le nombre de modes d'exécution possibles de l'opération A_i :

$$\underline{R}_i = \begin{bmatrix} r_{i1}^0 & \dots & r_{ip}^0 & r_{i1}^n & \dots & r_{iv}^n & r_{i1}^p & \dots & r_{iu}^p \end{bmatrix}$$

où r_{ik}^0 ($k = 1, \dots, p$), r_{ik}^n ($k = 1, \dots, v$), r_{ik}^p ($k = 1, \dots, u$) sont vecteurs-colonnes composés de h_i éléments chacun. La ligne j ($j = 1, \dots, h_i$) de la matrice $\underline{R}_i$,

$$\begin{bmatrix} r_{ij1}^0 & \dots & r_{ijp}^0 & r_{ij1}^n & \dots & r_{ijv}^n & r_{ij1}^p & \dots & r_{iju}^p \end{bmatrix}$$

est composée d'éléments indiquant le nombre d'unités de moyens renouvelables, non renouvelables et doublement limités de tous les types qui peuvent prendre part simultanément à l'exécution de l'opération A_i . La ligne j de la matrice $\underline{R}_i$ définit le mode j d'exécution de A_i , $j = 1, \dots, h_i$. A chaque instant, chaque opération peut être exécutée selon un seul (arbitraire) de ses modes.

Faisons quelques remarques sur les éléments d'une ligne de $\underline{R}_i$. La consommation d'un moyen doublement limité R_k^p , $k = 1, \dots, u$, par l'opération A_i exécutée selon le mode j dans une période de temps est égale au produit de la longueur de cette période et r_{ijk}^p . Si les unités d'un moyen renouvelable R_k^0 (resp. doublement limité R_k^p) ne sont pas identiques, r_{ijk}^0 (resp. r_{ijk}^p) représente l'indice d'unité de R_k^0 (resp. R_k^p).

Nous admettons que des unités de chaque type de moyens non renouvelables sont identiques. Dans ce dernier cas, la consommation d'un moyen doublement limité est alors équivalente au temps d'utilisation de celui-ci.

Pour l'opération A_i , les besoins continus en moyens peuvent être représentés de la façon suivante :

$$\begin{aligned} r_{ik}^0 &\in \langle a_{ik}^0, b_{ik}^0 \rangle, 0 \leq a_{ik}^0 \leq b_{ik}^0, k = 1, \dots, p, \\ r_{ik}^n &\in \langle a_{ik}^n, b_{ik}^n \rangle, 0 \leq a_{ik}^n \leq b_{ik}^n, k = 1, \dots, v, \\ r_{ik}^p &\in \langle a_{ik}^p, b_{ik}^p \rangle, 0 \leq a_{ik}^p \leq b_{ik}^p, k = 1, \dots, u. \end{aligned}$$

Finalement, nous dirons que l'opération A_i est exécutée à l'instant t si et seulement si ses besoins en moyens sont satisfaits.

Le modèle mathématique de l'opération $A_i \in \mathcal{A}$ décrit son exécution en fonction de la quantité et de la qualité des moyens affectés. Dans le cas où des besoins en moyens de A_i sont discrets, son modèle mathématique est représenté par le vecteur des temps d'exécution selon les modes particuliers :

$$\underline{p}_i = [\underline{p}_{ij}], j = 1, \dots, h_i, i = 1, \dots, n.$$

Dans le cas où des besoins en moyens de A_i sont continus, son modèle mathématique est une fonction continue, par exemple "temps d'exécution vs. quantité de moyens", "temps d'exécution vs. coût" ou "vitesse d'exécution vs. quantité de moyens". Dans le dernier cas ([23], [26], [28]), le modèle mathématique est représenté par l'équation différentielle de type :

$$\frac{d x_i(t)}{dt} = g_i [\underline{r}_i^0(t), \underline{r}_i^p(t)], t \in \langle t_i, T_i \rangle,$$

où $x_i(t)$ est l'état de l'opération A_i à l'instant t ; $\underline{r}_i^0(t) = [\underline{r}_{ik}^0(t)]$,

$k = 1, \dots, p$, $\underline{r}_i^p(t) = [\underline{r}_{ik}^p(t)]$, $k = 1, \dots, u$ sont des fonctions vectorielles déterminant les quantités de moyens R_k^0 et R_k^p affectées à A_i à l'instant t ; $g_i(0)$ est une fonction continue non décroissante, $g_i(0, 0) = 0$ où t_i et T_i correspondent respectivement à la date de début et de fin de A_i ; $x_i(t_i) = 0$, $x_i(T_i) = w_i > 0$ où w_i est l'état final, que l'on appelle la dimension de l'opération A_i . La dernière condition est équivalente à l'expression suivante :

$$\int_{t_i}^{T_i} g_i[\underline{r}_i^0(t), \underline{r}_i^p(t)] dt = w_i, i = 1, \dots, n.$$

Considérons maintenant les possibilités d'interruption des opérations.

Nous disons qu'une opération est interruptible si son exécution peut être interrompue à n'importe quel moment puis reprise sans encombre éventuellement selon un autre mode.

Si ces conditions ne sont pas remplies, l'opération est dite non interrompible.

Nous admettons ensuite que l'intensité avec laquelle une opération consomme des moyens non renouvelables et doublement limités pendant son exécution est constante.

La date de début au plus tôt (a_i) pour l'opération A_i définit la date à laquelle A_i est prête à être exécutée. La date de fin au plus tard (d_i) pour l'opération A_i définit la date à laquelle A_i devrait être achevée. Si d_i est une date infranchissable, elle est dite critique.

Indépendamment de l'information concernant les dates a_i et d_i pour chaque opération A_i particulière, on définit sur l'ensemble $\mathcal{A}$ des opérations une relation binaire (irréflexive) $<$ déterminant les contraintes de précedence entre les opérations. $A_i < A_j$ signifie que l'exé-

cution de A_i doit être achevée avant l'exécution de A_j . Si dans l'ensemble $\mathcal{A}$ on a défini au moins une contrainte de précédence, nous disons que $\mathcal{A}$ est un ensemble d'opérations dépendantes ; sinon, $\mathcal{A}$ est un ensemble d'opérations indépendantes. Les contraintes de précédence seront représentées par un graphe avec la convention : "une opération correspond à un arc" ou "une opération correspond à un sommet".

Une structure particulière de l'ensemble $\mathcal{A}$ permet de définir le concept d'atelier. Pour que le système $\mathcal{S}$ soit un atelier, il faut que l'ensemble $\mathcal{A}$ soit divisé en sous-ensembles indépendants et disjonctifs qui constituent des travaux (jobs) composés des opérations (tasks) qui doivent être exécutées en série dans le cadre de chaque travail. Un tel système modélise un atelier de production composé de machines utilisées dans un processus technologique. Si les nombres d'opérations sont égaux pour tous les travaux et si l'ordre d'exécution des opérations d'un travail est arbitraire, l'atelier est dit open shop. Si cet ordre est le même pour tous les travaux, l'atelier est dit flow shop et si les nombres d'opérations ainsi que leur ordre d'exécution sont différents pour chaque travail, l'atelier est dit job shop.

On considère maintenant l'ensemble $\mathcal{Q}$ qui est composé de $g \geq 1$ critères, $Q_1, \dots, Q_g$, d'évaluation de la commande dans le système $\mathcal{S}$. Nous distinguons essentiellement deux groupes de critères : critères de temps et de coût. Parmi les critères de temps, nous prenons en compte par exemple :

- le temps d'exécution de l'ensemble $\mathcal{A}$

$$T = \max [T_i]$$

où T_i est la date de fin de A_i , $i = 1, \dots, n$;

- le retard maximum d'achèvement

$$L = \max_i [\bar{T}_i - d_i] ;$$

- le retard moyen pondéré

$$\bar{L} = 1/n \sum_i w_i \max [0, T_i - d_i]$$

où w_i est un coefficient de pondération attribué à A_i ;

- le nombre d'opérations en retard

$$U = \sum_i U_i$$

$$\text{où } U_i = \begin{cases} 1 & \text{si } T_i > d_i ; \\ 0 & \text{sinon} \end{cases}$$

- le temps moyen de passage d'une opération

$$\bar{F} = 1/n \sum_i w_i (T_i - d_i),$$

c'est-à-dire le temps moyen d'attente et d'exécution d'une opération.

Parmi les critères de coût, nous considérons les critères suivants :

- la consommation totale K_k^n du moyen non renouvelable R_k^n , $k = 1, \dots, v$;
- la consommation totale K_k^p du moyen doublement limité R_k^p , $k = 1, \dots, u$;
- la consommation pondérée, c'est-à-dire le coût d'exécution de l'ensemble $\mathcal{A}$

$$K = \sum_{k=1}^v c_k^n K_k^n + \sum_{k=1}^u c_k^p K_k^p$$

où c_k^n et c_k^p sont des poids représentant le coût unitaire de moyens R_k^n et R_k^p ;

- le nombre d'interruptions des opérations pendant leur exécution.

Notons qu'en général les critères de ces deux groupes sont conflictuels et, dans certains sens, sont attachés aux différentes catégories de moyens. Les critères de temps évaluent le degré d'exploitation de moyens renouvelables (et doublement limités selon l'aspect "vertical" (*) des contraintes de disponibilité) tandis que les critères de coût évaluent le degré de consommation de moyens non renouvelables (et doublement limités selon l'aspect "horizontal" (*) des contraintes de disponibilité). On voit donc que, dans les problèmes d'allocation de moyens de catégories différentes, il faut envisager en général plusieurs critères.

Enfin, on définit la commande dans le système Δ comme une affectation, dans le temps, de l'ensemble des moyens $\mathcal{R}$ à l'ensemble des opérations $\mathcal{A}$ qui garantit l'exécution de toutes les opérations et respecte les contraintes imposées.

La commande qui assure le "meilleur" compromis (au sens défini par un décideur) entre critères de l'ensemble $\mathcal{Q}$ est dite satisfaisante. On appelle commande approximative la commande la plus proche du "meilleur" compromis. Dans la suite, nous assimilons la notion d'ordonnancement à celle de commande.

(*) L'aspect "vertical" désigne des contraintes instantanées tandis que l'aspect "horizontal" traduit des contraintes dans le temps.

Il est évident que les définitions du système d'allocation de moyens et de sa commande peuvent être facilement généralisées, notamment dans un contexte probabiliste ou flou ([29]). Nous ne décrivons cependant pas ces généralisations dans le cadre de cette présentation.

3. LE CHAMP DE LA RECHERCHE ET LES DOMAINES D'APPLICATIONS

Jusqu'à un passé récent, dans la problématique d'ordonnancement, on a considéré séparément l'allocation de moyens renouvelables et non renouvelables ([4], [5], [12]).

En ce qui concerne les moyens renouvelables et les besoins en moyens de type discret, on a jusqu'à présent uniquement considéré le cas d'un seul mode d'exécution pour chaque opération et le critère de temps d'exécution de l'ensemble $\mathcal{A}$. Pour ce cas, les méthodes actuellement les plus performantes sont : la méthode d'énumération implicite de Talbot et Patterson ([24]) pour les opérations non interruptibles et la méthode ARSME ([30]) pour des opérations interruptibles.

Dans le cas de besoins en moyens de type continu, on a considéré des modèles mathématiques d'opérations se présentant sous la forme d'équations différentielles qui montrent la liaison entre la vitesse de changement d'état de l'opération donnée et la quantité de moyens affectée à cette opération pendant son exécution. Le critère d'optimisation est le temps d'exécution de l'ensemble $\mathcal{A}$. Les premiers résultats concernant un seul type de moyens renouvelables ([26]) ont été ensuite généralisés pour prendre en compte plusieurs types de moyens renouvelables ([23]).

L'allocation de moyens non renouvelables, quant à elle, a d'abord été étudiée pour les besoins en moyens de type continu avec des modèles mathématiques d'opérations se présentant sous la forme de fonctions continues "temps vs. coût" ([1], [5]). Puis on l'a abordée pour les besoins en moyens de type discret ([6]). Le coût d'exécution de l'ensemble $\mathcal{A}$ a toujours été le seul critère d'optimisation envisagé dans ces études.

Il est important de souligner que les résultats obtenus dans ces deux classes de problèmes d'ordonnancement concernent des situations où les moyens sont tous de même catégorie et les besoins en moyens des opérations sont discrets ou continus. La réalité est pourtant plus complexe car la plupart des moyens-matériaux est doublement limitée et, dans les problèmes pratiques d'allocation de moyens, on a, en général, des moyens et/ou des besoins en moyens de catégories différentes. La variété de ces situations est très riche. Citons ici quelques exemples pratiques.

1) Dans les problèmes d'ordonnancement d'atelier, on n'envisageait que l'affectation des machines à des opérations. Pourtant, il est fréquent que les machines ne soient pas les seuls moyens d'exécution des opérations ; il y a le personnel, la matière première, l'énergie et les outils supplémentaires qu'il faut prendre en compte et qui représentent des moyens de différentes catégories. De plus, il y a souvent plus qu'un seul mode possible d'exécution pour une opération ([16]).

2) Un autre exemple concerne les systèmes multiprocesseurs de calcul où, en plus des processus, il faut considérer l'allocation d'autres moyens comme les canaux de transmission et la mémoire ([13], [20]). Puisque la mémoire peut être considérée comme un moyen divisible approximativement de manière continue, les besoins en moyens des opérations (programmes) sont à la fois discrets (processeurs, canaux) et continus (mémoire) ([27]).

3) Citons enfin l'exemple du processus d'électroaffinage du cuivre. Ce processus s'effectue dans des cuves d'électrolyse où des anodes de cuivre brut et des cathodes constituées par des feuilles de cuivre pur qui servent d'amorce au processus sont branchées en parallèle dans un circuit électrique. L'électrolyse se déroule en cycles dits anodiques qui se composent de deux à trois phases cathodiques. Une anode s'utilise pendant un cycle anodique et, dans chaque phase cathodique, se forme sur chaque cathode une plaque de cuivre pur. L'ensemble de moyens contient deux catégories de moyens limités : le courant électrique - renouvelable, divisible de manière continue, dont l'intensité est limitée à chaque instant, et deux types de matière première, des anodes et des feuilles d'amorçage cathodiques - moyens doublement limités, divisibles de manière dis-

crête dont la disponibilité est limitée à chaque instant (cette contrainte est due à la capacité du stock entre l'atelier d'électrolyse et l'atelier métallurgique) ainsi que la consommation dans une période de travail (cette contrainte est due à la capacité de production de l'atelier métallurgique). L'ensemble $\mathcal{A}$ est composé d'opérations non interruptibles correspondant aux phases cathodiques réalisées dans des groupes de cuves pendant un certain nombre de cycles anodiques. Pour un même groupe de cuves, les phases cathodiques se succèdent dans le temps. Ce processus est réalisé simultanément dans différents groupes de cuves. Chaque opération requiert pour son exécution un ensemble d'anodes et des feuilles d'amorçage (besoins discrets) ainsi qu'une certaine intensité du courant électrique appartenant à un intervalle (besoins continus). L'état d'exécution d'une opération est défini par le poids de la cathode correspondante. On connaît la dépendance entre l'accroissement du poids d'une cathode et la diminution du poids de son anode. La vitesse de diminution du poids d'une anode dépend de l'intensité du courant électrique. Le modèle mathématique d'une opération montre la liaison entre la vitesse d'accroissement du poids de la cathode et l'intensité du courant électrique :

$$\frac{d x_i(t)}{dt} = \begin{cases} g_i [\bar{r}_i^0(t)] & \text{si, à l'instant } t \in \langle t_i, T_i \rangle, \text{ les besoins} \\ & \text{en moyens discrets sont satisfaits,} \\ 0 & \text{sinon} \end{cases}$$

où $x_i(t)$ - le poids de la cathode i dans l'instant t ; $\bar{r}_i^0(t)$ - l'intensité du courant électrique attribué à A_i à l'instant t compris entre le début (t_i) et la fin (T_i) de son exécution ; $g_i(\cdot)$ - une fonction continue, non décroissante, qui inclut le coefficient de transport électrolytique du cuivre et l'équivalent électrochimique du groupe des cuves correspondant à A_i . La dimension de l'opération A_i , c'est-à-dire son état final, est définie par le poids normalisé w_i d'une plaque de cuivre pur sur la cathode i :

$$\int_{t_i}^{T_i} g_i [\bar{r}_i^0(t)] dt = w_i.$$

L'objectif est la minimisation du temps d'exécution de l'ensemble $\mathcal{A}$, qui est équivalent à la maximisation de la productivité de l'atelier d'électrolyse ([18]).

On voit donc que les problèmes d'allocation de moyens de catégories différentes reflètent mieux la réalité et, pour cela, sont vraiment dignes d'intérêt. Il faut pourtant se rendre compte des conséquences qu'apportent ces problèmes complexes et, en particulier, de la nécessité de considérer des critères multiples d'ordonnancement. Cela provient du fait déjà mentionné (cf. p. 10) que des moyens de catégories différentes engagent des critères à la fois de temps et de coût.

C'est cette direction de recherche qui nous a préoccupé dans nos derniers travaux. Nous nous limitons en principe aux cas de besoins en moyens de type discret et cela pour les raisons suivantes :

- des moyens-matériaux sont divisibles de manière discrète par leur nature ;
- s'ils sont divisibles approximativement de manière continue, alors les besoins en moyens sont en principe de type discret ;
- si les moyens d'une part et les besoins d'autre part sont approximativement continus, l'approche initialisée par Burkov ([3]) est efficace seulement sous des hypothèses assez fortes.

Un autre aspect nouveau de nos travaux consiste à prendre en compte le problème d'interruption d'opérations interruptibles et cela de deux façons :

- par l'évaluation d'une borne supérieure du nombre d'interruptions dans un ordonnancement construit selon une approche que l'on explicite dans la section suivante et
- par une minimisation du nombre d'interruptions dans cet ordonnancement.

Dans la section 4, nous présentons d'une part deux approches de problèmes d'ordonnancement incluant différentes catégories de moyens pour des opérations interruptibles et, d'autre part, la méthode de minimisation du nombre d'interruptions. La section 5 décrit une méthode d'allocation de moyens de catégories différentes pour des opérations non interruptibles et la section 6 regroupe des remarques concernant les méthodes de résolution des programmes linéaires (continus et discrets) multicritères formulés dans les sections précédentes.

4. APPROCHES DES PROBLEMES D'ALLOCATION DE MOYENS DE CATEGORIES DIFFERENTES POUR DES OPERATIONS INTERRUPTIBLES

Le problème général d'allocation de moyens limités des trois catégories réunies, dans le cas d'opérations interruptibles et en supposant des besoins discrets, peut être formulé comme un problème de Programmation Linéaire Multicritère (PLM). Cette formulation n'est cependant ni évidente ni univoque. En effet, celle-ci dépend des critères d'ordonnancement, du type de besoins en moyens, des dates de début au plus tôt et de fin au plus tard des opérations, etc. Il existe aussi une classe de problèmes pour lesquels il y a différentes formulations qui se distinguent entre elles par la taille du problème de PLM, la complexité de calcul et la borne supérieure du nombre d'interruptions des opérations.

Nous présentons dans la suite deux approches pour résoudre ce problème d'allocation de trois catégories de moyens : elles se différencient par la manière d'utiliser la PLM ([17]). La première approche, dite "en une phase", donne une allocation satisfaisante à la suite de la résolution d'un problème de PLM. Dans la deuxième approche, dite "en deux phases", la solution d'un problème de PLM sert, dans la deuxième phase, à la construction d'une allocation satisfaisante.

Toutes les notations utilisées dans la suite sont homogènes avec celles qui ont été définies précédemment.

4.1 L'approche en une phase

Considérons un système d'allocation de moyens définis de la façon suivante.

L'ensemble de moyens comprend les trois catégories de moyens $\mathcal{R} = \{R_1^0, \dots, R_p^0, R_1^n, \dots, R_v^n, R_1^p, \dots, R_u^p\}$ dont la disponibilité est respectivement égale à N_k , $k = 1, \dots, p$, B_k , $k = 1, \dots, v$, et N_k^p, B_k^p ,

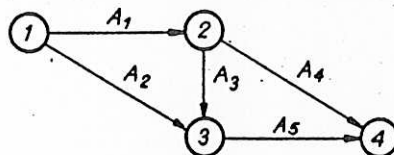
$k = 1, \dots, u$. L'ensemble d'opérations est composé de n opérations interruptibles. Les relations de succession temporelle parmi les opérations sont représentées par un graphe orienté, connexe et sans circuits avec la convention "une opération correspond à un arc". Les sommets du graphe (événements) sont numérotés de 1 à w où w est le nombre de sommets. suivant la règle qui dit que le sommet j précède k si et seulement si $j < k$. L'opération $A_i, i = 1, \dots, n$, est caractérisée par :

- h_i modes d'exécution définis par des lignes de la matrice $\underline{R}_i$ avec $h_i \geq 1$;
- le vecteur $\underline{p}_i = [p_{ij}]$, $j = 1, \dots, h_i$ où p_{ij} est le temps d'exécution de A_i selon le mode j ;
- $a_i = 0, d_i = \infty$

L'ensemble des critères $Q = \{T, K_1^n, \dots, K_v^n, K_1^p, \dots, K_u^p\}$.

Un numérotage des sommets du graphe étant donné, nous pouvons définir l'ensemble F_g de toutes les opérations qui peuvent être exécutées simultanément entre deux événements (sommets du graphe) successifs : g et $g + 1$, $g = 1, \dots, w - 1$ ([20]). Il est clair que les parties d'opérations exécutées dans le cadre d'un ensemble F_g sont indépendantes. Pour formuler ce problème en termes de PLM, il faut créer les différents sous-ensembles non vides de l'ensemble F_g pour $g = 1, \dots, w - 1$ et ensuite engendrer toutes les différentes possibilités d'affectation des modes d'exécution aux opérations dans les sous-ensembles ainsi générés. On appellera variantes d'allocation les possibilités qui vérifient les contraintes de disponibilité de moyens renouvelables et doublement limités. On les numérote de 1 à s . Donnons un exemple de construction des variantes d'allocation avec les données suivantes :

$$p = v = u = 1, n = 5, w = 4$$



$$\mathcal{R} = \{R^o, R^n, R^p\}, N = 4, N^p = 3$$

$$\begin{aligned} \underline{R}_1 &= \begin{bmatrix} 1^o & 1^n & 1^p \\ 3 & 3 & 1 \\ 3 & 6 & 1 \end{bmatrix} \\ \underline{R}_2 &= \begin{bmatrix} 1 & 4 & 1 \end{bmatrix} \\ \underline{R}_3 &= \begin{bmatrix} 2 & 2 & 1 \\ 4 & 5 & 0 \end{bmatrix} \\ \underline{R}_4 &= \begin{bmatrix} 1 & 3 & 1 \\ 3 & 9 & 3 \end{bmatrix} \\ \underline{R}_5 &= \begin{bmatrix} 3 & 7 & 2 \end{bmatrix} \end{aligned}$$

F_g	sous-ensembles des ensembles F_g	variantes d'allocation					
		n° de mode d'exécution					l
		A_1	A_2	A_3	A_4	A_5	
$\{A_1, A_2\}$	$\{A_1, A_2\}$	1	1				1
		2	1				2
	$\{A_1\}$	1					3
		2					4
	$\{A_2\}$		1				5
$\{A_2, A_3, A_4\}$	$\{A_2, A_3, A_4\}$		1	1	1		6
	$\{A_2, A_3\}$		1	1			7
	$\{A_2, A_4\}$		1		1		8
	$\{A_3, A_4\}$			1	1		9
	$\{A_3\}$			1			10
				2			11
	$\{A_4\}$				1		12
					2		13
$\{A_4, A_5\}$	$\{A_4, A_5\}$				1	1	14
	$\{A_5\}$					1	15

Figure 1 : Génération des variantes d'allocation pour un problème illustratif

La variable du problème de PLM est le temps $x_l \geq 0$ d'utilisation de la variante l. L'allocation satisfaisante de moyens est donc définie par la suite $\{x_l\}_{l=1}^S$ qui se déduit de la solution du Programme Linéaire Multicritère suivant :

Minimiser :

$$T = \sum_{l=1}^s x_l \quad (1)$$

$$K_k^n = \sum_{i=1}^n \sum_{l \in V_i} (x_l / p_{ij(l)}) r_{ij(l)k}^n \quad k = 1, \dots, v \quad (2)$$

$$K_k^p = \sum_{i=1}^n \sum_{l \in V_i} x_l r_{ij(l)k}^p \quad k = 1, \dots, u \quad (3)$$

sous les contraintes :

$$\sum_{l \in V_i} x_l / p_{ij(l)} = 1 \quad i = 1, \dots, n \quad (4)$$

$$K_k^n \leq B_k \quad k = 1, \dots, v \quad (5)$$

$$K_k^p \leq B_k^p \quad k = 1, \dots, u \quad (6)$$

où $p_{ij(l)}$ est le temps d'exécution de A_i selon le mode j affecté à cette opération dans la variante d'allocation l , V_i est l'ensemble de numéros des variantes qui contiennent l'opération A_i . Par exemple, les V_i pour le problème illustratif présenté ci-dessus sont les suivants : $V_1 = \{1, 2, 3, 4\}$, $V_2 = \{1, 2, 5, 6, 7, 8\}$, $V_3 = \{6, 7, 9, 10, 11\}$, $V_4 = \{6, 8, 9, 12, 13, 14\}$, $V_5 = \{14, 15\}$.

Une application directe d'une des méthodes de PLM se heurte pourtant aux problèmes sérieux que sont notamment la génération, le codage et la mémorisation de la matrice des coefficients qui a une taille importante. Pour éviter cet encombrement, on a développé une méthode appelée ARSME ([30]) qui fait appel à la méthode du simplexe révisée. En profitant de la structure particulière du programme linéaire, on a programmé une procédure qui engendre automatiquement des colonnes successives de la matrice des coefficients, c'est-à-dire des variantes d'allocation à chaque itération du simplexe. Ainsi, à chaque instant, on ne mémorise qu'une seule variante d'allocation et son ensemble F_g d'origine. La procédure est

construite de manière à minimiser le nombre de variantes qui sont vérifiées du point de vue des contraintes de disponibilité de moyens. Ajoutons que le nombre de types de moyens ne restreint pas la capacité de la méthode ; son accroissement est même favorable car cela peut réduire le nombre de variantes d'allocations, c'est-à-dire de variables du programme linéaire.

Remarquons que l'approche en une phase réclame un numérotage des sommets du graphe qui n'est pas en général unique. Il faudrait alors considérer tous les numérotages possibles, ce qui peut s'avérer pénible s'ils sont nombreux. Dans le dernier cas, on peut choisir un numérotage établi à l'aide de l'heuristique NODORD ([14]) : celle-ci détermine un numérotage qui donne un ordonnancement proche, à quelques pourcents près, de l'optimum au sens d'un critère d'ordonnancement.

Dans [17], [18], on a démontré que la borne supérieure du nombre d'interruptions dans un ordonnancement construit selon l'approche en une phase est égale à $M(n + v + u - G)$ où M est le nombre maximum d'opérations qui peuvent être exécutées simultanément et $G = \min_i (w - 1 - g_i)$ où g_i est le nombre d'ensembles F_g qui contiennent A_i .

On a également calculé la complexité de calcul de cette approche en admettant qu'on minimise un seul critère (les autres étant bornés) et que, pour la résolution du programme linéaire, on utilise l'algorithme polynomial de Khachiyan ([8]). La complexité de calcul de l'approche en une phase est alors en $O(n^{4M} \mathcal{L})$ ([18]) où $\mathcal{L}$ est le nombre de bits nécessaires pour coder l'ensemble des données du programme linéaire.

4.2 L'approche en deux phases

Cette approche couvre une classe de problèmes qui diffère de celle couverte par l'approche en une phase ; néanmoins, l'intersection de ces deux classes n'est pas vide. L'approche en deux phases permet de prendre en compte des dates différentes de début au plus tôt et de fin au plus tard des opérations ainsi que le critère de retard maximum d'achèvement

de l'ensemble $\mathcal{A}$; elle n'accepte pas cependant tous les types de besoins en moyens qui sont acceptés par l'approche en une phase. L'utilisation de cette approche est recommandée dans le cas où dans l'ensemble $\mathcal{Q}$ se trouvent des unités non identiques de moyens renouvelables ou doublement limités de même type, par exemple m machines de même type dont les efficacités de travail sont différentes. Les différentes caractéristiques de ces deux approches ainsi que leur comparaison sont données dans [17], [18].

Pour présenter cette approche, nous abordons un système d'allocation de moyens relativement simple et ensuite nous en montrons les possibilités de généralisation.

L'ensemble de moyens comprend les trois catégories de moyens $\mathcal{R} = \{M_1, \dots, M_m, R_1^0, \dots, R_p^0, R_1^n, \dots, R_v^n, R_1^p, \dots, R_u^p\}$ où $M_j, j = 1, \dots, m$, sont des machines non identiques de même type et la disponibilité des autres moyens (dont les unités sont identiques dans le cadre de chaque type) est respectivement égale à $N_k, k = 1, \dots, p, B_k, k = 1, \dots, v$, et $N_k^p, B_k^p, k = 1, \dots, u$. L'ensemble des opérations est composé de n opérations interruptibles et indépendantes. L'opération A_i est caractérisée par :

- $h_i \geq 1$ modes d'exécution définis par des lignes de la matrice $R_i = [r_{i0}^0 \ r_{i1}^0 \ \dots \ r_{ip}^0 \ r_{i1}^n, \dots \ r_{iv}^n \ r_{i1}^p, \dots \ r_{iu}^p]$ où r_{ijo}^0 indique le numéro de la machine, $r_{ijk}^0 \in \{0, 1\}, j = 1, \dots, h_i, k = 1, \dots, p$, $r_{ijk}^p \in \{0, 1\}, j = 1, \dots, h_i, k = 1, \dots, u$, et $r_{ijk}^n, j = 1, \dots, h_i, k = 1, \dots, v$, indique le nombre d'unités de R_k^n qui peuvent prendre part à l'exécution de A_i ;
- le vecteur de temps d'exécution $p_i = [p_{ij}], j = 1, \dots, h_i$;
- $a_i = 0, d_i = \infty$.

L'ensemble des critères $\mathcal{Q} = \{T, K_1^n, \dots, K_v^n, K_1^p, \dots, K_u^p\}$. Remarquons que les besoins en moyens $R_k^0, k = 1, \dots, p$ et $R_k^p, k = 1, \dots, u$, concernent au plus une unité de tous ces moyens qui est la même pour tous les modes d'exécution de A_i .

L'approche en question est composée de deux phases. Dans la première phase, on cherche les temps totaux $x_{ij} \in \langle 0, p_{ij} \rangle$ d'exécution de A_i selon le mode j , $i = 1, \dots, n$; $j = 1, \dots, h_i$, dans un ordonnancement satisfaisant (au sens des critères appartenant à l'ensemble $\mathcal{Q}$). A partir des x_{ij}^* , $i = 1, \dots, n$; $j = 1, \dots, h_i$, obtenus dans la première phase, on construit dans la deuxième phase un ordonnancement satisfaisant, réalisable pour les mêmes valeurs des critères, en forme d'une suite d'ordonnements partiels. L'ordonnement partiel est une affectation des modes d'exécution à m opérations différentes pour une période de temps $\delta > 0$, affectation qui garantit que, à chaque instant, chacune de ces m opérations est exécutée selon un seul mode.

Dans la figure 2, nous présentons un exemple des ordonnancements résultant de la première et de la deuxième phase de l'approche en question.

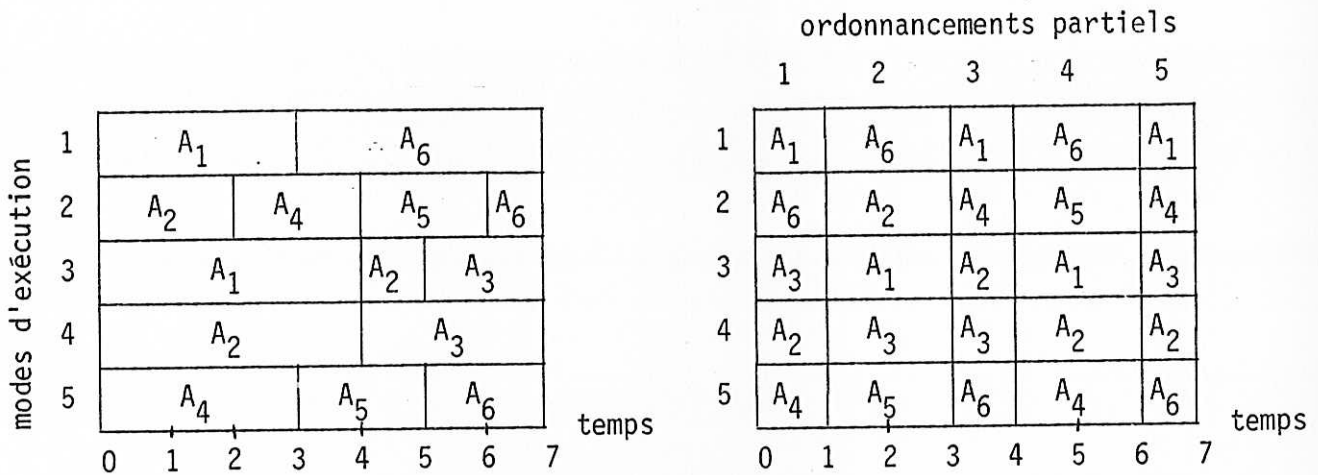


Figure 2 : Ordonnements résultants de la première et de la deuxième phase de l'approche pour un problème illustratif.

Une allocation satisfaisante de moyens est donc définie par la suite $\{x_{ij}(l)\}_{l=1, i=1, j=1}^{s, n, h_i}$ où $\sum_{l=1}^s x_{ij}(l) = x_{ij}^*$ et $x_{ij}(l)$ est le temps d'exécution de A_i selon le mode j dans l'ordonnement partiel l , $l = 1, \dots, s$.

Pour trouver les valeurs de x_{ij}^* dans la première phase, on doit résoudre le Programme Linéaire Multicritère suivant :

$$\text{Minimiser } T \quad (7)$$

$$K_k^n = \sum_{i=1}^n \sum_{j=1}^{h_i} (x_{ij}/p_{ij}) r_{ijk}^n \quad k = 1, \dots, v \quad (8)$$

$$K_k^p = \sum_{i=1}^n \sum_{j=1}^{h_i} x_{ij} r_{ijk}^p \quad k = 1, \dots, u \quad (9)$$

sous les contraintes :

$$T - \sum_{(i,j) \in C_f} x_{ij} \geq 0 \quad f = 1, \dots, m \quad (10)$$

$$T - \sum_{j=1}^{h_i} x_{ij} \geq 0 \quad i = 1, \dots, n \quad (11)$$

$$T N_k - \sum_{i \in D_k} \sum_{j=1}^{h_i} x_{ij} \geq 0 \quad k = 1, \dots, p \quad (12)$$

$$T N_k^p - \sum_{i \in D_k^p} \sum_{j=1}^{h_i} x_{ij} \geq 0 \quad k = 1, \dots, u \quad (13)$$

$$K_k^p \leq B_k^p \quad k = 1, \dots, u \quad (14)$$

$$K_k^n \leq B_k \quad k = 1, \dots, v \quad (15)$$

$$\sum_{j=1}^{h_i} x_{ij}/p_{ij} = 1 \quad i = 1, \dots, n \quad (16)$$

$$\text{où } C_f = \{(i, j) : f = r_{ij0}^0\} \quad f = 1, \dots, m$$

$$D_k = \{i : \sum_j r_{ijk}^0 > 0\} \quad k = 1, \dots, p$$

$$D_k^p = \{i : \sum_j r_{ijk}^p > 0\} \quad k = 1, \dots, u.$$

La condition (10) garantit que le temps actif de chaque machine ne dépassera pas la durée d'ordonnancement T , (11) que chaque opération sera achevée avant T , (16) que chaque opération sera exécutée. Les conditions (12)-(15) représentent les contraintes de disponibilité de moyens renouvelables, doublement limités et non renouvelables respectivement.

La construction de l'ordonnancement réalisable dans la deuxième phase peut se faire à l'aide d'un algorithme polynomial qui a été proposé et décrit dans [13], [18], [20], [22]. Le point crucial de cet algorithme consiste dans la recherche d'un sous-ensemble de l'ensemble des opérations qui forme un ordonnancement partiel. On a proposé deux méthodes originales pour la construction de ces sous-ensembles : la première est une méthode de type "affectation" qui opère directement dans la matrice composée d'éléments x_{ij}^* ([13], [18]) et la deuxième est une méthode qui conduit à la recherche d'un flot dans un réseau ([18], [20]).

En ce qui concerne l'approche en deux phases dont l'idée de base vient d'être présentée, on a déjà développé plusieurs généralisations. Citons les plus importantes :

a) Contraintes de précédence ([15], [17], [20])

On fait appel à la notion de l'ensemble F_g définie pour l'approche en une phase. Les fractions des opérations exécutées dans le cadre d'un ensemble F_g , $g = 1, \dots, w - 1$, étant indépendantes, on a $w - 1$ ensembles distincts pour lesquels on peut appliquer l'approche développée pour l'ensemble des opérations indépendantes. La durée d'ordonnancement pour l'ensemble F_g est égale à T_g . La variable du programme linéaire multicritère est le temps total $x_{ijg} \in \langle 0, p_{ij} \rangle$ d'exécution de A_i selon le mode j dans le cadre de l'ensemble F_g , $i = 1, \dots, n$; $j = 1, \dots, h_i$; $g = 1, \dots, w - 1$. Dans la deuxième phase, l'algorithme proposé pour le cas des opérations indépendantes doit être appliqué à chaque ensemble F_g , $g = 1, \dots, w - 1$.

b) $a_i \geq 0, i = 1, \dots, n$ ([17])

On suppose que les opérations sont classées par ordre de leurs dates d'arrivée croissantes $a_1 \leq a_2 \leq \dots \leq a_n$. $T = a_{n+1}$ et la variable du programme linéaire multicritère est le temps total $x_{ij}^{(h)} \in \langle 0, p_{ij} \rangle$ d'exécution de A_i selon le mode j dans l'intervalle $\langle a_h, a_{h+1} \rangle$, $h = 1, \dots, n$; $j = 1, \dots, h_i$; $i = 1, \dots, n$. Dans la deuxième phase, l'algorithme proposé dans le cas des opérations indépendantes doit être appliqué à chaque intervalle h .

c) $d_i \leq \infty, i = 1, \dots, n, L \in \mathbb{Q} ([15], [17], [20])$

On suppose que les opérations sont classées par ordre de leurs dates de fin au plus tard croissantes $d_1 \leq d_2 \leq \dots \leq d_n$. Le critère de temps est le retard maximum L et la variable du programme linéaire multicritère est le temps total $x_{ij}^{(h)} \in \langle 0, p_{ij} \rangle$ d'exécution de A_i selon le mode j dans l'intervalle $\langle d_{h-1} + L, d_h + L \rangle$ où $d_0 + L = 0, h = 1, \dots, i; j = 1, \dots, h_i; i = 1, \dots, n$. La deuxième phase est analogue au cas b).

d) $a_i \geq 0, d_i \leq \infty, i = 1, \dots, n$, et d_i sont critiques ([17])

On définit une suite croissante des valeurs de a_i et $d_i, b_1 \leq b_2 \leq \dots \leq b_{2n}$ et une variable $x_{ij}^{(h)}$ comme dans le cas b). Evidemment, $x_{ij}^{(h)} \equiv 0$ si $a_i > b_h$ ou $d_i < d_{h+1}, h = 1, \dots, 2n - 1; j = 1, \dots, h_i; i = 1, \dots, n$. La deuxième phase ne change pas par rapport au cas b).

e) Ordonnancement de coût minimal pour l'atelier de type "open shop" soumis à des contraintes de moyens, de dates, de disponibilité et de fin au plus tard ([16]).

Dans [17], [18] on a démontré que la borne supérieure du nombre d'interruptions dans un ordonnancement construit selon l'approche en deux phases était égale à $PR[3m^2 + z_N(m - 1) + (2m - 1)(2m + v + u - 1)]$ où $z_N = \max \{0, \min[n - 3m - v - u, m(p + u + 1)]\}$ et PR est le nombre d'intervalles qui, selon les cas, prend les valeurs suivantes :

$$PR = \begin{cases} 1 & \text{pour le problème de base (opérations indépendantes),} \\ w - 1 & \text{pour (a),} \\ n & \text{pour (b) ou (c),} \\ 2n - 1 & \text{pour (d)} \end{cases}$$

La complexité de calcul de la première phase de cette approche calculée avec les mêmes hypothèses que celles que l'on a admises pour l'approche

en une phase est en $O(n^4 m^4 PR^4 \mathcal{X})$ ([18]). La complexité de la deuxième phase est en $O(z^2 m^2 PR)$ où $z = \max(z_N, m)$ ([18]). On voit donc que l'approche en deux phases est polynomiale, ce qui prouve son efficacité dans le sens de la complexité des calculs.

4.3 La minimisation du nombre d'interruptions des opérations

Introduire la possibilité d'interrompre des opérations entraîne plusieurs conséquences, notamment certaines propriétés et méthodes de construction d'ordonnements satisfaisants. Premièrement, en admettant l'interruption, on peut augmenter le degré d'utilisation des moyens renouvelables, c'est-à-dire diminuer le temps mort des machines et de la main-d'oeuvre, ce qui raccourcit le temps d'exécution de l'ensemble $\mathcal{X}$. Deuxièmement, l'hypothèse d'interruption augmente la performance des méthodes d'ordonnement car une large classe de problèmes peut être réduit de façon polynomiale à des problèmes de programmation linéaire. Ils appartiennent ainsi à la classe P alors que les problèmes correspondants avec des opérations non interruptibles sont NP-complets. Il convient de noter cependant que chaque interruption est plus ou moins coûteuse et qu'en dépit de certaines applications, notamment informatiques où ce coût est négligeable par rapport au gain qui en résulte, ce coût existe réellement ; il est donc raisonnable d'en tenir compte. Une méthode cherchant à minimiser ce coût a été proposée dans [19] pour des ordonnements construits par une des approches présentées ci-dessus. C'est ainsi la première fois que le nombre d'interruptions a pu intervenir comme un des critères multiples d'ordonnement.

Pour décrire l'essentiel de cette méthode, il est utile de rappeler qu'un ordonnancement obtenu selon l'approche en une phase est une suite de variantes d'allocation et que celui qui est obtenu selon l'approche en deux phases est une suite d'ordonnements partiels. Une interruption ne peut se faire qu'entre deux variantes d'allocation ou entre deux ordonnements partiels. Remarquons que l'ordre des éléments de ces suites est

n'est pas unique dans le cadre de chaque ensemble F_g , $g = 1, \dots, w - 1$. Il faut donc chercher un ordre qui minimise le nombre d'interruptions des opérations dans un ordonnancement donné. Ce problème a été résolu en se ramenant au problème du voyageur de commerce ([18], [19]).

Pour chaque variante d'allocation appartenant à un ordonnancement obtenu selon l'approche en une phase (*), on définit un vecteur $\underline{z}_l = [\underline{z}_{lq}]$, $q = 1, \dots, ve$; $l = 1, \dots, wp$,

$$z_{lq} = \begin{cases} 1 & \text{si la position } q \text{ correspond à une opération et son mode} \\ & \text{d'exécution dans la variante } l, \\ 0 & \text{sinon.} \end{cases}$$

wp est le nombre de variantes et ve est le nombre de modes d'exécution distincts utilisés dans l'ordonnancement en question. On introduit également un vecteur $\underline{z}_0 = [\underline{z}_{0q}]$ où $z_{0q} = 0$ pour $q = 1, \dots, ve$. Pour chaque couple $\underline{z}_i, \underline{z}_j$ on définit une matrice carrée $\underline{s} = [\underline{s}_{ij}]$, $i, j = 0, 1, \dots, wp$ où

$$s_{ij} = \begin{cases} \infty & \text{si la variante } j \text{ ne peut pas suivre immédiatement la va-} \\ & \text{riante } i \text{ (par exemple à cause des contraintes de précé-} \\ & \text{dence),} \\ \sum_{q=1}^{ve} (z_{iq} + z_{jq} - 2 z_{iq} z_{jq}) & \text{sinon.} \end{cases}$$

$s_{ij} \neq \infty$ définit le nombre de différences entre deux variantes. Chaque différence correspond à un des quatre événements :

- | | | |
|---|---|--|
| <ul style="list-style-type: none"> - début - suspension - reprise - fin | } | d'exécution d'une opération dans une des deux va-
riantes voisines. |
|---|---|--|

(*) Dans le cas de l'approche en deux phases, il faut substituer la variante d'allocation par l'ordonnancement partiel.

Comme pour chaque opération, il y a autant de suspensions que de reprises et un seul début et une seule fin ; le nombre d'interruptions est égal à $[(\sum_i \sum_j s_{ij}^*) - 2n]/2$ où s_{ij}^* est l'un des $(wp + 1)$ éléments de $\underline{S}$ tels que :

(i) dans chaque ligne et dans chaque colonne de $\underline{S}$ il y a exactement un élément s_{ij}^* ;

(ii) les indices des éléments s_{ij}^* forment un cycle :

$0i, ij, jk, kf, \dots, pq, q0.$

Le problème consiste à trouver une suite de $(wp + 1)$ éléments s_{ij}^* qui respecte les conditions (i) et (ii) et minimise

$$(iii) \sum_i \sum_j s_{ij}^*.$$

On voit bien que ce problème est équivalent au problème du voyageur de commerce. Comme c'est un problème NP-complet, il est raisonnable d'utiliser pour le résoudre une des heuristiques connues.

Pour illustrer une application de cette méthode, reprenons l'exemple présenté dans la figure 1. Supposons que les variantes d'allocation suivantes soient entrées dans un ordonnancement construit selon l'approche en une phase : 4, 6, 7, 8, 10, 13, 14. Les vecteurs $\underline{Z}_i$, $i = 1, \dots, 7$, et la matrice $\underline{S}$ prennent alors les formes suivantes :

$$\begin{array}{l} \underline{Z}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\ \underline{Z}_2 = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \\ \underline{Z}_3 = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \\ \underline{Z}_4 = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix} \\ \underline{Z}_5 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \\ \underline{Z}_6 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \\ \underline{Z}_7 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \end{array}$$

n° de la variante d'allocation	s_{ij}	0	1	2	3	4	5	6	7
	0	∞	(<u>1</u>)	∞	∞	∞	∞	∞	∞
4	1	∞	∞	4	3	3	(<u>2</u>)	2	∞
6	2	∞	∞	∞	1	(<u>1</u>)	2	4	3
7	3	∞	∞	(<u>1</u>)	∞	2	1	3	4
8	4	∞	∞	1	2	∞	3	3	(<u>2</u>)
10	5	∞	∞	2	(<u>1</u>)	3	∞	2	3
13	6	(<u>1</u>)	∞	4	3	3	2	∞	3
14	7	2	∞	∞	∞	∞	∞	(<u>3</u>)	∞

Dans la suite des variantes : 0-4-10-7-6-8-14-13-0, le nombre d'interruptions est minimal et égal à 1. Remarquons que dans la suite initiale : 0-4-6-7-8-10-13-14-0, le nombre d'interruptions est égal à 4.

5. METHODES D'ALLOCATION DE MOYENS DE CATEGORIES DIFFERENTES
POUR DES OPERATIONS NON INTERRUPTIBLES

Le problème général d'allocation des trois catégories de moyens, dans le cas des opérations non interruptibles et des besoins discrets, est évidemment NP-complet et peut être formulé en termes de la programmation linéaire multicritère en variables 0-1 ([18]). Cependant, même dans le cas d'un seul critère, une application directe d'une méthode de PL en variables 0-1 à la résolution de ce problème n'est pas satisfaisante car ce sont des méthodes peu performantes et de plus, dans le déroulement de l'algorithme, on ne connaît en général pas de solution réalisable. Ainsi, si l'on est obligé d'arrêter les calculs, on n'est pas sûr d'en obtenir.

C'est pourquoi on a proposé dans [18] une autre technique de résolution qui est une technique d'énumération implicite et qui élimine le dernier inconvénient. Cet algorithme généralise celui de Talbot et Patterson ([24]) développé dans le cas où des moyens sont renouvelables et, pour chaque opération, il y a un seul mode d'exécution.

On considère un système d'allocation de moyens proche de celui qui a été défini dans le paragraphe 4.1 : on admet en effet que les opérations sont non interruptibles et que les relations de succession temporelle parmi les opérations sont représentées par un graphe orienté sans circuits avec la convention "une opération correspond à un sommet" et que $a_i \geq 0$ et $d_i \leq \infty$, $i = 1, \dots, n$.

Soit T_h une borne supérieure du temps d'exécution de l'ensemble $\mathcal{A}$, appelée horizon de temps, calculée selon une règle heuristique. Supposons que T_h et les temps d'exécution des opérations, p_{ij} , $i = 1, \dots, n$; $j = 1, \dots, h_i$, ont des valeurs entières. L'horizon de temps se compose donc de T_h périodes unitaires. Pour chaque opération, on connaît une

date d'achèvement au plus tôt, e_i , et une date d'achèvement au plus tard, l_i , $i = 1, \dots, n$, qui se déduisent de l'analyse du chemin critique à condition que toutes les opérations soient exécutées selon leurs modes les plus rapides et que $T = T_h$. Evidemment, $e_i \geq a_i + \min_j (p_{ij})$ et $l_i \leq d_i$, $i = 1, \dots, n$. On admet aussi, sans perte de généralité, que l'opération A_n est la seule opération sans successeurs ; donc $l_n = T_h$.

L'algorithme d'énumération implicite consiste en l'amélioration itérative d'un ordonnancement réalisable connu a priori. L'ordonnancement de départ peut être construit selon une simple règle heuristique ([18]). Nous présentons dans la suite le principe de cet algorithme dans le cas où l'on minimise un seul critère, les valeurs des autres critères étant bornées. Afin d'accélérer la convergence de l'algorithme en question, il est souhaitable de faire auparavant les rangements suivants :

1) Rangement des opérations dans l'ordre respectant les contraintes de précédence ; pour les ex-aequo, on utilise des règles prioritaires qui dépendent de la nature du critère temps. Notons $P(i)$ la priorité de A_i :

- pour T , $P(i) = 1/\min_j (p_{ij})$;
- pour $\bar{L}$ ou U , $P(i) = w_i/d_i$;
- pour $\bar{F}$, $P(i) = w_i/\min_j (p_{ij})$.

2) Rangement des contraintes de disponibilité de moyens selon un rapport "besoins vs. disponibilité" décroissant.

3) Rangement des modes d'exécution pour chaque opération :

- pour un critère de temps dans l'ordre des p_{ij} croissants ;
- pour un critère de coût dans l'ordre de consommation croissante.

Pour chaque opération A_i , on définit deux variables Y_i et U_i qui prennent respectivement les valeurs t et j si, à la suite de l'affectation du mode j à l'opération A_i , son exécution s'achève pendant la

période t . On définit également des variables N_{kt} , $k = 1, \dots, p$, B_{kt} , $k = 1, \dots, v$, et N_{kt}^p , B_{kt}^p , $k = 1, \dots, u$ qui montrent les quantités respectives de moyens R_k^o , R_k^n , R_k^p disponibles pendant la période t , $t = 1, \dots, T_h$, au cours de l'application de l'algorithme.

L'algorithme commence par la définition de la date d'achèvement au plus tôt de l'opération A_1 exécutée selon son premier mode. Ainsi, on connaît Y_1 et $U_1 = 1$ et il faut diminuer respectivement N_{kt} , $k = 1, \dots, p$, B_{kt} , $k = 1, \dots, v$, N_{kt}^p , $k = 1, \dots, u$, $t = 1, \dots, Y_1$ de r_{11k}^o , r_{11k}^n , r_{11k}^p . Ensuite, commençant avec A_2 et finissant avec A_n , on essaye d'affecter le premier mode d'exécution à ces opérations de sorte qu'elles s'achèvent le plus tôt possible compte tenu des contraintes de moyens et de précédence. Si l'augmentation d'une solution partielle par l'inclusion de l'opération A_i exécutée selon le mode j est non admissible à cause de contraintes de moyens, il faut essayer l'affectation de l'un des autres modes $j + 1, \dots, h_i$ à cette opération. Si c'est impossible, l'algorithme revient à l'opération A_{i-1} . On enlève alors l'opération A_{i-1} de la solution partielle et on essaye de l'y réintroduire de telle sorte que la date de son achèvement soit le plus près possible mais postérieure au Y_{i-1} précédent. Evidemment, on tient compte ici de tous les modes $j \geq U_{i-1}$ et des disponibilités de moyens dans l'intervalle compris entre la date de début au plus tôt e_i (qui, au cours de l'application de l'algorithme, doit vérifier la condition $e_i \geq Y_l$, $l \in P_{i-1}^*$ où P_{i-1}^* est l'ensemble des prédécesseurs immédiats de A_{i-1}) et la date d'achèvement au plus tard, l_{i-1} . Si on réussit, la solution partielle est augmentée et l'algorithme passe à l'opération A_i . Le processus d'augmentation et de retrait se poursuit jusqu'à l'obtention d'une solution complète ou jusqu'à la tentative de retrait au-dessous de la première opération. La première situation signifie qu'on a obtenu une nouvelle solution améliorée et la deuxième que la dernière solution complète est optimale. Il est également possible que Y_n , pour une solution améliorée, soit égale à la borne inférieure de T (par exemple e_n) connue a priori ; cela signifie que cette solution est déjà optimale. Si ce n'est pas le cas, la

solution améliorée remplace la dernière solution complète et, avant l'itération suivante, on actualise les dates d'achèvement au plus tard $l_i' = l_i - (l_n - Y_n)$, $i = 1, \dots, n$.

Dans le cas d'autres critères de temps ou de coût, notamment $\bar{L}$, $\bar{F}$, U , K_k^n ($k = 1, \dots, v$), K_k^p ($k = 1, \dots, u$) ou K , après avoir trouvé une solution améliorée au lieu de changer la valeur de l_i , $i = 1, \dots, n$, on actualise la borne supérieure de la valeur du critère en question. Par rapport à la version présentée ci-dessus, l'augmentation d'une solution partielle peut se faire à la condition supplémentaire que l'inclusion de l'opération A_i exécutée selon le mode j n'implique pas d'augmentation de la valeur du critère au-delà de sa borne supérieure (en admettant que les opérations suivantes soient exécutées selon leurs modes les plus rapides ou les moins chers sans considérer de contraintes de moyens). Sinon, l'algorithme revient tout de suite à l'opération A_{i-1} . En-dehors de cette modification, le principe de l'algorithme reste le même.

6. REMARQUES CONCERNANT L'ASPECT MULTICRITERE DES PROBLEMES D'ALLOCATION DE MOYENS

Dans le cas d'opérations interruptibles, dans les deux approches proposées, l'aspect multicritère des problèmes d'ordonnancement est pris en compte à l'étape de la résolution du programme linéaire multicritère alors que, dans le cas d'opérations non interruptibles, cet aspect est pris en compte dans une technique arborescente ([19]). En principe, dans les deux cas, l'ordonnancement satisfaisant est une des solutions efficaces du problème d'ordonnancement multicritère à condition que le décideur ne prenne pas en compte d'autres critères, en particulier des critères non analytiques. On connaît trois approches principales au choix d'une solution efficace satisfaisante ([7], [10], [11]) :

A) L'approche par la programmation paramétrique (multiparamétrique) qui met en évidence l'ensemble des solutions efficaces. Le décideur choisit une solution satisfaisante après avoir comparé différents éléments de cet ensemble. Cette approche est particulièrement recommandée lorsqu'il n'y a que deux critères mais devient très inefficace s'il y en a plus.

B) L'approche par agrégation de plusieurs critères en un critère unique. Cette approche fournit au décideur une seule solution efficace qui maximise une fonction d'utilité reflétant ses préférences. Selon les techniques d'agrégation, on distingue les méthodes suivantes :

- minimisation de la somme pondérée des critères ;
- méthode lexicographique ;
- méthode des seuils ;
- goal programming ;
- programmation linéaire floue.

Toutefois, en dépit des techniques proposées, il est fréquent de trouver des problèmes qui soit demandent un travail trop important pour obtenir une image satisfaisante des préférences réelles du décideur, soit conduisent à une représentation arbitraire de ces préférences par une fonction d'utilité trop grossière. De plus, la quantification a priori des importances relatives des critères peut être complètement faussée par une ignorance de la configuration de l'ensemble des solutions réalisables. Ces remarques incitent à définir les préférences en même temps que l'on explore l'ensemble des solutions réalisables ; ceci constitue l'approche C du choix d'une solution satisfaisante.

C) Méthodes interactives qui se composent itérativement d'une phase de calcul et d'une phase de décision. Dans la phase de décision, on propose au décideur une solution efficace et une solution de référence pour qu'il définisse la direction de recherche d'une meilleure solution efficace dans la phase de calcul. Le décideur est aidé par l'analyse de sensibilité qui décrit les tendances des différents critères dans le voisinage de la dernière solution efficace.

Nous présentons dans [19] une analyse des conditions d'application de ces approches dans le cas des problèmes d'ordonnancement avec des opérations interruptibles. Nous insistons sur l'application de l'approche C car seule cette approche permet de prendre en compte le critère du nombre d'interruptions avec une importance équivalente aux autres critères. En effet, la méthode de minimisation du nombre d'interruptions s'applique à un ordonnancement connu ; dans l'approche B, ce critère n'aurait pu intervenir que comme un critère secondaire.

Le cas des opérations non interruptibles est pourtant plus difficile, notamment dans le cadre de l'approche C. Comme nous l'avons mentionné dans la section précédente, le problème d'ordonnancement avec des opérations non interruptibles peut être résolu par Programmation Linéaire Multicritère en variables 0-1. Cette possibilité demeure pourtant théorique car les méthodes connues ([2], [9], [25]) ne sont pas assez performantes dans ce cas là. C'est pourquoi nous travaillons actuellement sur une version multicritère interactive de l'algorithme arborescent présenté dans la section 5 qui répondra spécifiquement à ce type de problème.

REFERENCES

- [1] Berman E.B. : Resource allocation in a PERT network under continuous activity time-cost functions. Management Sci. 10 (1964), n° 4.
- [2] Bitran G.R. : Theory and algorithms for linear multiple objective programs with zero-one variables. Mathematical Programming 17 (1979), n° 3.
- [3] Burkov V.N. : Problems of optimum distribution of resources. Control and Cybernetics 1 (1972), n° 1/2.
- [4] Davis E.W. : Project scheduling under resource constraints - historical review and categorization of procedures. AIIE Transactions 5 (1973), n° 4.
- [5] Elmaghraby S.E. : Activity Networks - Project Planning and Control by Network Models. J. Wiley, New York, 1977.
- [6] Harvey R.T., Patterson J.H. : An implicit enumeration algorithm for the time/cost tradeoff problem in project network analysis. Found. of Control Engrg. 4 (1979), n° 3.
- [7] Hwang C.L., Masud A.S.M. : Multiple Objective Decision Making - Methods and Applications. Springer-Verlag, Berlin, 1979.
- [8] Khachiyan L.G. : Algorithmes polynomiaux en programmation linéaire (en russe). Zh. Vychislit. Mat. mat. Fiz. 20 (1980), n° 1.
- [9] Klein D., Hannan E. : A branch and bound algorithm for the multiple objective zero-one linear programming problem. Working Paper 79-2, Florida International University, Oct. 1979.

- [10] Roy B. : Décisions avec critères multiples : problèmes et méthodes. Revue METRA 11 (1972), n° 1.
- [11] Roy B., Vincke Ph. : L'analyse multicritère. Document du LAMSADE n° 6, Université de Paris-Dauphine, mai 1980.
- [12] Slowinski R. : Optimal and heuristic procedures for project scheduling with multiple constrained resources - A survey. Found. of Control Engrg. 2 (1977), n° 1.
- [13] Slowinski R. : Scheduling preemptable tasks on unrelated processors with additional resources to minimize schedule lenght. Dans Bracchi G., Lockemann P.C. (ed.), Lecture Notes in Computer Science, Vol. 65, Springer-Verlag, Berlin (1978), 536-547.
- [14] Slowinski R. : A node ordering heuristic for network scheduling under multiple resource constraints. Found. of Control Engrg. 3 (1978), n° 1.
- [15] Slowinski R. : Allocation de ressources limitées parmi des tâches exécutées par un ensemble de machines indépendantes. Dans Pelegrin M., Delmas J. (ed.), Comparison of Automatic Control and Operational Research Techniques Applied to Large Systems Analysis and Control, Pergamon Press, Oxford (1979), 189-195.
- [16] Slowinski R. : Cost-minimal preemptive scheduling of independant jobs with release and due dates on open shop under resource constraints. Information Processing Letters 9 (1979), n° 5.
- [17] Slowinski R. : Two approaches to problems of resource allocation among project activities - A comparative study. Journal of the Operational Research Society 31 (1980), n° 8.
- [18] Slowinski R. : Algorithmes pour la commande d'allocation de moyens de catégories différentes dans le complexe d'opérations (en polonais). Editions de l'Ecole Polytechnique de Poznan, ser. Rozprawy n° 114, Poznan, 1980.

- [19] Slowinski R. : Multiobjective network scheduling with efficient use of renewable and non-renewable resources. European Journal of Operational Research 7 (1981), n° 3.
- [20] Slowinski R. : L'ordonnancement des tâches préemptives sur les processeurs indépendants en présence de ressources supplémentaires. RAIRO Informatique 15 (1981), n° 2.
- [21] Slowinski R., Weglarz J. : Systèmes non-linéaires de commande en temps minimal de distribution des tâches et des moyens. Actes du 8e Congrès International de Cybernétique, Namur (1976), 733-743.
- [22] Slowinski R., Weglarz J. : Modèle d'ordonnancement optimal avec plusieurs modes d'exécution des opérations (en polonais). Przegląd Statystyczny 24 (1977), n° 4.
- [23] Slowinski R., Weglarz J. : Solving the general project scheduling problem with multiple constrained resources by mathematical programming. Dans Stoer J. (ed.), Lecture Notes in Control and Information Sciences, Vol. 2, Part 2, Springer-Verlag, Berlin (1978), 278-288.
- [24] Talbot F.B., Patterson J.H. : An efficient integer programming algorithm with network cuts for solving resource-constrained scheduling problems. Management Sci. 24 (1978), n° 11.
- [25] Villarreal B., Karwan M.H., Zionts S. : An interactive branch and bound procedure for multicriterion integer linear programming. Dans G. Fandel, T. Gal (eds.), Lecture Notes in Economic and Mathematical Systems, Vol. 177, Springer-Verlag, Berlin (1980), 448-467.
- [26] Weglarz J. : Time-optimal control of resource allocation in a complex of operations framework. IEEE Trans. Systems, Man, and Cybernet., SMC-6 (1976), n° 11.

- [27] Weglarz J. : Multiprocessor scheduling with memory allocation - A deterministic approach. IEEE Trans. on Computers, C-29 (1980), n° 8.
- [28] Weglarz J. : Project scheduling with continuously divisible, doubly-constrained resources. Management Sci., 27 (1981), n° 3.
- [29] Weglarz J. : Commande des systèmes de type complexe d'opérations (en polonais). PWN, Poznan, Warszawa, 1981.
- [30] Weglarz J., Blazewicz J., Cellary W., Slowinski R. : Algorithme 520. An automatic revised simplex method for constrained resource network scheduling. ACM Trans. on Mathematical Software, 3 (1977), n° 3.