
Chapitre V

ORDONNANCEMENT EN ATELIERS SPÉCIALISÉS

La version CD-Rom de ce chapitre permet d'accéder directement à des logiciels exploitant les principales techniques ou approches décrites dans ce chapitre. La présence d'une icône en marge du texte (comme celle figurant dans cette page) indique la possibilité d'utilisation de la technique décrite dans le texte et le lien hypertextuel placé sur cette icône permet d'accéder à l'exemple numérique utilisé dans le texte. Certaines figures et tableaux de ce chapitre ont été établis à partir de ce logiciel mais n'ont pas été repris sous leur forme originale pour des raisons de lisibilité. Vous avez la possibilité de créer vos propres exemples pour exploiter les possibilités offertes¹.



Les problèmes d'**ordonnancement**², c'est-à-dire de la **détermination conjointe des dates d'exécution d'un ensemble d'opérations et des ressources mobilisées dans cette exécution**, ont été introduits au chapitre précédent relatif à la gestion des projets. Ces problèmes d'ordonnancement se retrouvent dans toutes les formes d'organisation de la production et la qualité des solutions proposées conditionne largement la performance, voire la survie, de l'organisation.

- Dans le cas d'une production en ligne de fabrication et / ou d'assemblage, comme on le rencontre fréquemment dans la production de masse de produits manufacturés, ces problèmes sont peu importants pour la production sur une ligne dédiée à la production d'une seule famille de produits obtenue par des combinaisons de modules interchangeables (un moteur à choisir parmi 5 possibles, un tableau de commande à choisir parmi 3 possibles, etc.). Le problème pratique posé concerne surtout le système d'information à mettre en place, dans la mesure où il s'agit de faire «converger» au bon moment les composants qui vont permettre de diversifier la production. Les problèmes sont d'une tout autre nature si le choix de certaines options fait varier le temps de travail sur certains postes de la ligne (c'est le cas, par exemple, de l'option «toit ouvrant» dans l'industrie automobile) et, alors, se pose un véritable problème d'ordonnancement, qui sera présenté au **chapitre IX**.

1. Voir avertissement de la note du bas de la **page 8**.

2. La littérature spécialisée distingue le problème de l'ordonnancement de celui du **séquencement** qui n'est autre que celui de la détermination de l'ordre de passage d'un ensemble d'opérations devant être exécuté sur une ressource donnée. Plusieurs ordonnancements traitant un même ensemble de commandes impliquant chacune l'exécution de plusieurs opérations, dans un système productif comportant plusieurs ressources, peuvent conduire à un même séquençement d'un ensemble d'opération à exécuter sur une même ressource mais, si l'ordre de ces opérations est le même, leurs dates d'exécution changent. Pour approfondir la problématique traitée dans ce chapitre, le lecteur est invité à se reporter à Esquirol & Lopez (1999, [140]); il peut également consulter les ouvrages de Pinedo (1995, [338]), Pinedo & Chao (1999, [339]), Morton & Pentico ([309], 1993) et Carlier & Chretienne (1988, [85]). L'ouvrage collectif coordonné par Lopez & Roubellat (2001, [286]) aborde de manière intelligible les nouvelles approches de résolution de cette classe de problèmes.

- Dans le cas d'industries de process, comme la chimie et aussi dans certaines industries agroalimentaires, le problème d'ordonnancement posé est plutôt celui de la définition d'un schéma répétitif de fabrication de différents produits, pour faire face à une demande relativement stable. Ce problème est souvent assez simple pour que l'appel à des formulations classiques de la recherche opérationnelle s'avère pertinent et particulièrement rentable¹.

Dans ce chapitre, on s'attaquera au cas de l'ordonnancement d'un ensemble de commandes (ou d'ordres de fabrication), à exécuter dans un système productif organisé en ateliers spécialisés (défini au [chapitre I, § I-2.2.2, page 51](#)), chaque commande pouvant porter sur une ou plusieurs unités (dans ce dernier cas, on parle de [série](#)) et impliquant l'exécution de plusieurs opérations mobilisant des ressources variées. Ce problème diffère de celui traité au [chapitre IV](#) (qui pouvait concerner un portefeuille de quelques projets) sur l'horizon de planification et sur le nombre de commandes.

- Les problèmes d'ordonnancement en ateliers spécialisés se posent sur le très court terme (moins d'une semaine), peuvent porter sur un nombre élevé de commandes (plusieurs centaines) et concernent, dans le détail et à chaque instant du temps, la planification de l'utilisation des ressources disponibles en hommes et en machines, afin de mieux contrôler les coûts et de maîtriser les délais de fabrication des productions décidées.
- Les problèmes d'ordonnancement de projet s'attaquent au plus à une dizaine de projets², se posent sur un horizon allant de plusieurs semaines à plusieurs mois, et à un niveau généralement agrégé (découpage temporel, détail des ressources).

Les techniques d'ordonnancement en ateliers spécialisés relèvent de la problématique de la régulation à court terme de l'entreprise, et plus particulièrement de celle du contrôle de l'utilisation de la main-d'œuvre et des équipements productifs. Elles complètent donc les techniques de gestion des stocks, traitées au [chapitre XI](#) (gestion calendaire) et au [chapitre XII](#) (gestion de type «quantité économique de commande, point de commande») et les techniques en Juste-À-Temps, abordées au [chapitre VII](#) et qui assurent un autre contrôle simultané des flux et des ressources, très efficace dans certains cas.

L'organisation en ateliers spécialisés est le plus répandu des modes d'organisation des systèmes productifs. C'est sans doute aussi celui qui pose les problèmes les plus compliqués, non pas en termes de formalisation de problèmes réalistes car les techniques de modélisation permettent une description satisfaisante de ces problèmes, mais en termes de résolution en raison du nombre élevé de variables de commande, de la nature discrète de ces variables et de la prise en compte des contraintes de disponibilité de ressources qui conduit à la résolution de problèmes

1. Par exemple, certains problèmes rencontrés dans la fabrication de certaines gammes de produits laitiers (yaourts...) se résolvent en faisant appel à des modèles initialement mis au point pour résoudre des problèmes de transport.

2. On observe, depuis quelques années, une tendance à une gestion par projet. Le nombre de projets à gérer peut alors être de plusieurs centaines. Dans ces conditions, les logiciels d'ordonnancement de projets sont difficilement appropriés pour prendre en compte efficacement les contraintes cumulatives. On observe souvent alors que les contraintes cumulatives sont ignorées et que se mettent en place des procédures d'ajustements mutuels pour résoudre les conflits lorsqu'ils se produisent.

combinatoires de grande taille¹ et implique l'utilisation d'heuristiques. Une **heuristique** est un algorithme de résolution d'un problème complexe, exploitant au mieux la structure du problème considéré et permettant de lui trouver rapidement une réponse numérique; des analyses comparatives permettent d'évaluer la performance d'heuristiques alternatives pour traiter une classe de problèmes; aucune garantie d'optimalité n'est donnée et on peut, en général, simplement espérer obtenir rapidement une «bonne» solution.

Les apports de la recherche opérationnelle sont cependant loin d'être négligeables, comme nous allons le voir en examinant quelques problèmes génériques simples auxquels on peut se ramener, avec deux approches radicalement différentes de cette classe de problèmes: celle des modèles statiques ([section I](#)) et celle des modèles dynamiques ([section II, page 409](#)). Ce sera l'occasion de présenter certaines méthodes efficaces de résolution de ce type de problèmes. On évoquera enfin ([section III, page 416](#)) les fondements retenus par certains logiciels de résolution de problèmes d'ordonnancement en ateliers spécialisés. Mais au préalable, il faut préciser la signification de quelques termes utilisés en ordonnancement.

Résoudre un problème d'ordonnancement, c'est définir où et à quel moment précis un certain nombre de **tâches**² doivent être réalisées. Une tâche peut correspondre à une fabrication d'un objet ou à la fourniture d'une prestation de service. Chaque tâche se décompose en un certain nombre d'**opérations** dans des centres de production. En ordonnancement, le niveau de détail retenu pour définir une opération est tel que deux opérations successives sont exécutées dans deux centres de production différents. Selon le problème examiné, un **centre de production** peut être aussi bien une usine, un département de production, un atelier, un groupe de machines ou une **machine**. Il peut également être tout autre chose si la tâche est une prestation de service: cabinet médical, fauteuil de dentiste, caisse de supermarché, etc. *Dans ce chapitre on utilisera le terme centre de production, de préférence à celui de machine habituellement utilisé dans la littérature spécialisée, pour bien marquer que ce problème d'ordonnancement ne concerne pas uniquement la production de biens mais aussi celle des services.* Conformément à ce qui vient d'être dit, lorsque la réalisation d'une tâche requiert successivement plusieurs centres de production, on dira que chaque centre de production réalise une opération de la tâche considérée. La **préemption** désigne la possibilité d'interrompre une opération pour en passer une autre avant de la reprendre plus tard et l'on parle de **problèmes préemptifs** et de **problèmes non préemptifs** d'ordonnancement. Les problèmes étudiés ici seront tous de type non préemptif.

Lorsque l'ordre de passage d'un ensemble de tâches sur un même ensemble de centres de production est toujours le même, la littérature spécialisée de recherche opérationnelle désigne ce cas particulier d'organisation en ateliers spécialisés par

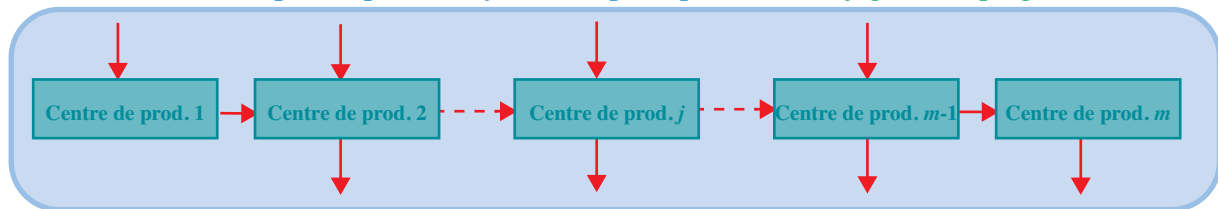
1. Les spécialistes qualifient de **NP-complets** ou **NP-difficiles** les problèmes pour lesquels on n'a pas trouvé d'algorithme de résolution en un temps limité supérieurement par une fonction de la taille N du problème; on qualifie de faciles les problèmes pour lesquels une telle fonction polynomiale existe (par exemple N^2) et de difficiles ceux pour lesquels le temps de résolution n'est limité supérieurement que par une fonction exponentielle de N (par exemple 2^N).
2. Ici, la **tâche** (*job* dans la littérature spécialisée anglo-saxonne, d'où les termes de *flow shop* et de *job shop* – *shop* signifiant atelier) se définit par un ensemble d'opérations associées à la fabrication d'un produit (ou d'un lot de produits) ou à la délivrance d'une prestation de service; dans le premier cas, on parle souvent d'**Ordre de Fabrication** ou OF. Cette acception du mot tâche diffère donc ici de celle retenue en analyse de projets.

le terme de *flow shop*. On parle encore de structure de type « flux unidirectionnel » en ce sens que, sur tous les centres de production, l'ordre d'exécution reste le même; on trouve encore, dans la littérature française le terme d'**atelier à cheminement unique**. Dans un *flow shop*, le temps opératoire de certaines opérations pouvant être nul, on peut alors décrire un *flow shop* quelconque à m centres de production par la **figure 103**, dans laquelle :

- les entrées directes sur un centre de production de rang supérieur à 1 s'expliquent par des temps opératoires nuls sur tous les centres de production précédents;
- les sorties directes antérieurement au dernier centre de production s'expliquent par des temps opératoires nuls sur tous les centres de production suivants;
- un temps opératoire nul entre le centre de production d'entrée et celui de sortie conduit, en pratique, à « sauter » ce centre de production.

FIGURE 103

Schéma de principe d'un flow shop (reprise de la figure 5, page 53)



Ce type de production se distingue de celui de la chaîne de production par trois caractéristiques :

- le fait qu'une tâche peut ne pas faire appel à tous les centres de production (un nombre quelconque de centres pouvant être « sautés »),
- la dispersion importante des temps opératoires des opérations exécutées sur un même poste de travail (en raison d'une absence de spécialisation étroite du poste dans l'exécution d'une même opération),
- l'existence de files d'attente, de longueur variable au cours du temps, en amont des différents postes de travail.

Le **tableau 66** fournit un exemple de données d'un problème de *flow shop* avec dix OF (ordre de fabrication) : l'OF 5 commence directement sur la machine C ; l'OF 7 sort de l'atelier après exécution de l'opération sur la machine C ; la machine B est « sautée » par les OF 9 et 10. Vous pouvez visualiser le Gantt associé au séquençement suivant des OF : 9, 10, 4, 6, 1, 8, 3, 2, 7.



On réserve le terme de *job shop* pour le cas général d'une coexistence de très nombreux cheminements des flux de production dans un même système productif (cas illustré à la **figure 3**, page 52); on trouve encore, dans la littérature française, le terme d'**atelier à cheminements multiples**. Pour en finir avec ce problème de terminologie, on rencontre des problèmes d'**atelier à cheminements libres** (connus aussi sous le nom de problèmes d'*open shop*) dans lesquels les opérations des tâches à exécuter sur un ensemble de centres de production différents peuvent l'être dans un ordre quelconque (cas, par exemple, de certains contrôles de qualité ou d'examens cliniques en milieu hospitalier).

TABLEAU 66

Exemple de problème de flow shop (temps opératoires des opérations à exécuter sur chaque machine, pour chaque ordre de fabrication)

OF	1	2	3	4	5	6	7	8	9	10
Machine A	10	12	10	8	0	11	7	6	8	14
Machine B	9	14	17	10	0	12	14	13	0	0
Machine C	13	11	13	12	13	8	14	15	11	13
Machine D	14	17	14	14	15	12	0	8	17	11
Machine E	22	8	13	15	10	19	0	17	11	14

Les **modèles statiques d'ordonnancement** se caractérisent par la recherche de l'ordonnancement optimal d'un ensemble donné de tâches sur une période donnée, ces tâches n'étant pas forcément toutes disponibles en début de période. Les tâches autres que celles prévues initialement sont stockées en attendant une régénération de l'ordonnancement, sur la base de modèles statiques, pour l'ensemble des tâches en attente et en cours d'exécution. Les **modèles dynamiques d'ordonnancement** se caractérisent au contraire par des arrivées successives de tâches, sans référence précise à une période d'ordonnancement.

Le clivage des approches ne s'effectue pas seulement sur la base d'une opposition «statique – dynamique» : dans les modèles statiques on se situe toujours en univers certain, tandis que l'approche dynamique s'effectue le plus souvent en univers aléatoire. Le lecteur pourra se rendre compte de la complexité du problème posé, même dans un cadre de type «statique-certain», en cherchant à résoudre le cas **MÉCANO S.A.** proposé à la fin de ce chapitre.

Table des matières

Index thématique

SECTION I INTRODUCTION AUX MODÈLES STATIQUES D'ORDONNANCEMENT

On abordera successivement et succinctement deux catégories de modèles statiques d'ordonnancement : ceux où les coûts de lancement sont indépendants de l'ordonnancement retenu (§ I-1), puis ceux où le coût de lancement de chaque tâche dépend de la tâche qui la précède (§ I-2, page 388). Dans les deux cas et malgré le caractère très simpliste de ces modèles statiques, des enseignements précieux peuvent être tirés comme guide d'action dans l'approche dynamique de cette classe de problèmes. Dans un troisième paragraphe (§ I-3, page 404), on tentera de dégager les traits essentiels de cette approche statique et les approches retenues aujourd'hui pour résoudre des problèmes de dimension raisonnable.

I-1 Modèles statiques – Cas des coûts de lancement indépendants de l'ordonnancement retenu

Le problème général est à n tâches et m centres de production (§ I-1.4, page 376), mais nous examinerons tout d'abord des cas plus simples dans lesquels m sera égal à 1 (§ I-1.1) ou 2 (§ I-1.2, page 372), ou dans lequel n sera égal à 2 (§ I-1.3, page 375). On analysera ensuite le cas dans lequel m est quelconque mais avec un ordre de passage entre les centres de production, identique pour toutes les tâches (§ I-1.4, page 376). On abordera ensuite le cas des cheminements libres

(§ I-1.5, page 384) et celui du job shop pour traiter le problème des goulots d'étranglement (§ I-1.6, page 385).

Ces différents modèles ont en commun un certain nombre d'hypothèses implicites: l'ordre technique de passage des tâches sur les centres de production est intangible, les temps de transport et de lancement sont nuls, les temps opératoires sont certains, une opération ne peut commencer que si la précédente est terminée. Pour relâcher ces contraintes, il faut faire appel à une formulation générale de type programmation linéaire en nombres entiers (cf. § I-1.4.2, page 378).

I-1.1 Ordonnancement de n tâches nécessitant l'intervention d'un seul centre de production

Ce cas de figure, le plus simple de tous, est moins évident qu'on ne pourrait le croire de prime abord. En effet, la seule chose que l'on sait, c'est que, quel que soit l'ordonnancement¹ choisi, le temps nécessaire pour réaliser les n tâches est le même. On examinera donc les conséquences de plusieurs règles d'ordonnancement sur certains indicateurs permettant de juger la qualité de l'ordonnancement choisi, dans une optique donnée. Les enseignements obtenus ne manquent pas d'intérêt pratique si le centre de production auquel on s'intéresse est un centre de production constituant le goulot d'étranglement du système productif².

On examinera également, sur les exemples numériques retenus, une technique graphique de visualisation d'un ordonnancement, le graphique de Gantt.

I-1.1.1 L'ordonnancement suivant la règle du temps opératoire minimum (règle TOM)

I-1.1.1.1 Exemple introductif

Supposons que pour la période à venir, 5 tâches, toutes disponibles, soient à exécuter sur le centre de production A. On notera t_i , la durée d'exécution (y compris le temps de préparation) de la tâche i ($i = 1, 2, \dots, 5$), cette durée est encore appelée **temps opératoire**. Ces durées peuvent comporter des temps de préparation mais ceux-ci sont supposés indépendants de l'ordre de passage des tâches sur le centre de production. Le centre de production est réputé disponible le temps nécessaire. Les données du problème sont portées dans le [tableau 67](#).

TABLEAU 67
Données de base

Tâche i	1	2	3	4	5
Temps opératoire t_i (en centième d'heure)	50	150	80	200	30

Supposons que l'ordonnancement retenu soit le suivant: tâche 3, puis tâche 4, puis tâche 1, puis tâche 5, puis tâche 2. On notera j le numéro d'ordre de passage du travail (par exemple, $j = 3$ repère la troisième tâche programmée, c'est-à-dire

1. Puisqu'il n'y a qu'un seul centre de production (à machine unique), on aurait pu tout aussi bien parler de séquençement.

2. Le cas d'un pool de machines identiques ou non sur l'une desquelles chacune des n tâches doit passer est voisin mais plus complexe. Le lecteur intéressé pourra consulter Hax & Candea (1984, [224]), p. 284-288 qui résument les acquis dans ce domaine.

dans notre exemple la tâche 1). On notera T_j (avec un T majuscule et non un t minuscule) la durée d'exécution de la tâche programmée en $j^{\text{ème}}$ position et A_j , la date d'achèvement de cette tâche j . Dans notre exemple, on a donc :

TABLEAU 68
Exemple d'ordonnancement

Ordre de passage j	1	2	3	4	5
Tâche programmée j	3	4	1	5	2
Temps d'exécution T_j	80	200	50	30	150
Date A_j de fin de la tâche j	80	280	330	360	510

I-1.1.1.2 Graphique de Gantt

Cette programmation se visualise facilement sur le diagramme de Gantt de la figure 104.

FIGURE 104

Exemple de visualisation d'un ordonnancement sur un graphique de Gantt

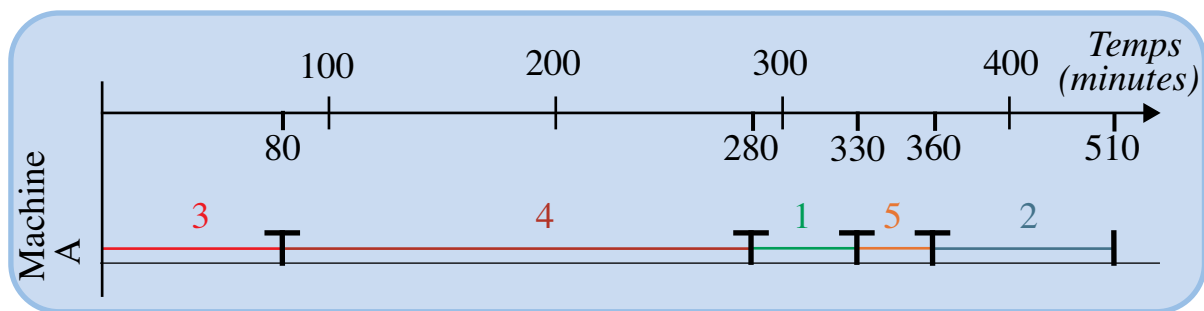


Table des
matières

Index
thématique

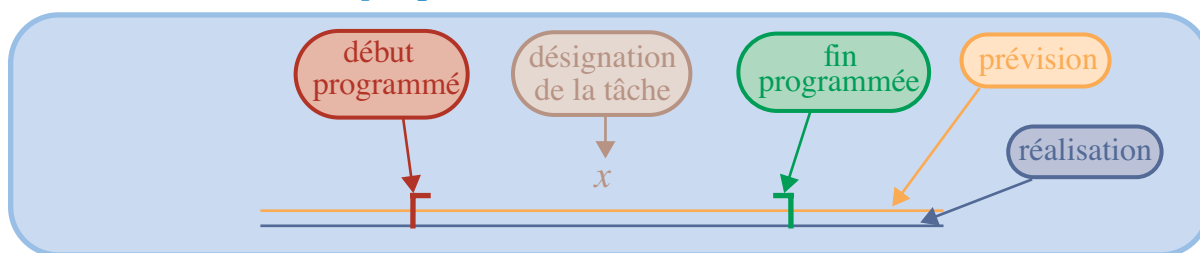
Le **graphique de Gantt** encore appelé **diagramme de Gantt**, est une technique que de visualisation de l'utilisation de moyens productifs et/ou de l'avancement de l'exécution de tâches popularisée par Gantt en 1917 (mais dont on retrouve des utilisations chez les prêtres égyptiens de l'antiquité) et est classiquement utilisée en ordonnancement en atelier. Une tâche y est représentée sur un axe, habituellement horizontal¹, par un segment dont la longueur est, en principe, proportionnelle au temps d'exécution. Lorsque l'on étudie l'évolution de l'utilisation de plusieurs facteurs productifs (par exemple des machines), l'utilisation de chaque facteur productif est portée sur un axe différent; on a alors un faisceau de droites parallèles sur un même document (tableau mural par exemple).

La même échelle temporelle est utilisée pour tous les axes, ce qui fait que les intersections de ces droites avec une même perpendiculaire repèrent le même instant. Pour faciliter ce repérage, un papier quadrillé est utilisé et l'échelle des temps est explicitée en haut du document. Un curseur vertical permet de visualiser un instant précis du temps, ce qui facilite l'utilisation de ce document pour le suivi d'un atelier. Au-dessus de chaque segment on porte le code d'identification de la tâche (ou ordre de fabrication), ou des quantités s'il s'agit de la production d'un article connu sans ambiguïté.

1. Les programmes informatiques disponibles sur le CD-Rom utilisent, pour des raisons de commodité, la convention d'une représentation sur un axe vertical.

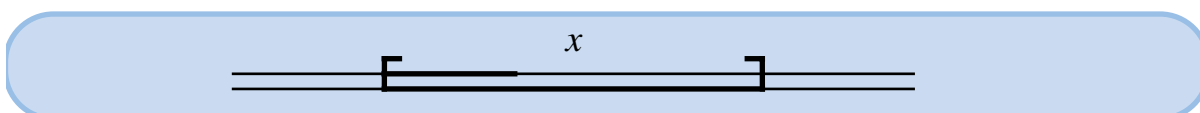
Ce document est utilisé pour le lancement des travaux en ateliers, lorsque la production est organisée en ateliers spécialisés. Le plus souvent en face d'un poste de travail, on porte deux axes parallèles distants de quelques millimètres seulement. Sur l'axe supérieur on trace en trait fin les prévisions, tandis que l'axe inférieur est destiné aux réalisations, en trait gras. Cette « astuce de présentation » permet d'éviter les gommages. Conventionnellement une tâche est programmée comme à la [figure 105](#), où la longueur du segment est proportionnelle à la durée d'exécution et x est le numéro de la tâche, ou la quantité à produire.

FIGURE 105

Graphique de Gantt : structure de base

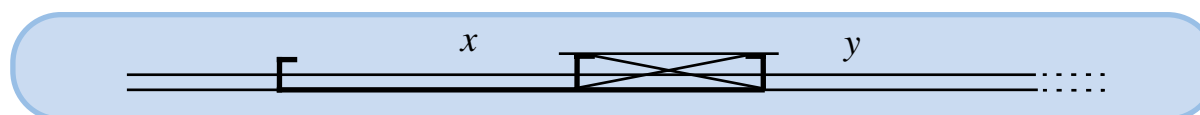
À l'exécution, des écarts peuvent apparaître avec les prévisions, si la production effective est supérieure à la production prévue (parce que le rendement du poste est supérieur à celui défini dans la gamme opératoire, lorsqu'il s'agit d'une production répétitive), l'information correspondante est visualisée comme dans la [figure 106](#) (la longueur du trait supérieur étant proportionnelle au dépassement de quantités produites par rapport aux quantités prévues).

FIGURE 106

Graphique de Gantt : cas d'une production réalisée supérieure à la production prévue

En cas de dépassement prévu, on portera sur le graphique (voir [figure 107](#)) l'information suivante pour indiquer le temps supplémentaire qui a été nécessaire pour réaliser l'objectif prévisionnel (y est le code de l'opération suivante).

FIGURE 107

Graphique de Gantt : cas d'un dépassement de temps

On porte également sur le graphique les causes d'inutilisation du poste de travail, à l'aide des lettres conventionnelles suivantes : Z (aucun travail exécuté), A (exécutant absent), M (manque de matière première), R (réparation). Selon l'horizon de planification utilisé, la précision de l'échelle est plus ou moins grande, le niveau de détail des centres de production repéré par les axes est plus ou moins poussé et les nomenclatures d'articles fabriqués plus ou moins agrégées : l'ordonnancement prévisionnel doit être d'autant plus global que l'horizon de planification retenu est éloigné.

Par ailleurs, lorsque l'on ne s'intéresse qu'à l'avancement de tâches, ce qui est le cas lorsque la production est de type unitaire, on associe habituellement à chaque tâche, un axe différent, cessant alors de privilégier la notion de centre de production ou de poste de travail (voir [chapitre IV](#), [page 307](#)).

I-1.1.1.3 La règle TOM

L'ordonnancement retenu à titre d'exemple dans le [tableau 68](#) n'est que l'un des 5! ordonnancements possibles. Si l'exécution des 5 tâches nécessite 510 centièmes d'heures quel que soit l'ordonnancement retenu, l'ordre de passage de ces tâches a des conséquences sur le moment à partir duquel chaque tâche est terminée. Notons A_j , le moment à partir duquel la tâche programmée en $j^{\text{ème}}$ position est terminée. On a par exemple $A_4 = \sum_{h=1}^4 T_h$ et d'une façon générale :

$$A_j = \sum_{h=1}^j T_h \quad \text{relation 65}$$

Le calcul des différents temps d'achèvement A_j est immédiat (et a déjà été effectué en dernière ligne du [tableau 68](#) et reporté dans la [figure 104 de la page 365](#)) :

TABLEAU 69
Calcul du temps d'achèvement moyen

Ordre de passage j	1	2	3	4	5
T_j	80	200	50	30	150
A_j	80	280	330	360	510

$$\bar{A} = 312$$

Le temps d'achèvement moyen est alors :

$$\bar{A} = \frac{1}{5} \cdot \sum_{j=1}^5 A_j = \frac{80 + 280 + 330 + 360 + 510}{5} = 312$$

et peut s'écrire encore d'une façon générale :

$$\bar{A} = \frac{1}{n} \cdot \sum_{j=1}^n A_j = \frac{1}{n} \cdot \sum_{j=1}^n \left(\sum_{k=1}^j T_k \right) = \frac{1}{n} \cdot \sum_{j=1}^n (n-j+1) T_j \quad \text{relation 66}$$

La règle d'ordonnancement qui *minimise le temps d'achèvement moyen* est celle du Temps Opérateur Minimum, règle que l'on notera **TOM** (elle est connue dans la littérature anglo-saxonne opérationnelle sous le nom de **SPT rule**, pour *Shortest Processing Time rule*, ou encore **SOT rule**, pour *Shortest Operation Time*). Elle consiste, comme son nom l'indique, à exécuter immédiatement la tâche ayant le plus faible temps opératoire. Cette règle récurrente est donc :

$$T_1 \leq T_2 \leq \dots \leq T_j \leq T_{j+1} \leq \dots \leq T_n \quad (\text{règle d'ordonnancement TOM} \Rightarrow \text{minimisation du temps d'attente moyen et du retard algébrique moyen}) \quad \text{relation 67}$$

L'application de cette règle conduit, dans notre exemple, à l'ordonnancement décrit au [tableau 70 de la page 368](#).



TABLEAU 70
Ordonnancement suivant la règle TOM

Ordre de passage j	1	2	3	4	5
Tâche programmée	5	1	3	2	4
T_j	30	50	80	150	200
A_j	30	80	160	310	510

$$\bar{A} = 218$$

Notons dès à présent que la priorité des tâches varie en sens inverse de la valeur du critère utilisé dans la règle d'ordonnancement, c'est-à-dire que la tâche que l'on considère comme la plus « urgente » est celle pour laquelle la valeur du critère retenu est la plus faible. Ce sera le cas pour toutes les règles d'ordonnancement présentées dans ce chapitre.

Cette règle d'ordonnancement ne conduit pas à la plus faible dispersion des résultats. En effet, comme vous pouvez le vérifier, l'écart-type des temps d'achèvement est de 174,06 en suivant la règle TOM, alors qu'il n'est que de 139,05 dans l'ordonnancement initialement choisi.

Par contre, on peut montrer que la règle TOM minimise le temps d'attente moyen ainsi que le retard algébrique moyen¹. Le **retard algébrique** d'une tâche se définit comme la différence positive ou négative $T_j - d_j$ entre le moment où la tâche est terminée T_j et le moment d_j où elle aurait dû l'être et qui est une donnée exogène de l'ordonnancement. Le retard algébrique ne doit pas être confondu avec le **retard vrai**, qui correspond à l'usage courant du mot retard, c'est-à-dire qu'il est nul si la tâche n'est pas achevée postérieurement à la date de livraison attendue et égale au retard algébrique dans le cas contraire: $\text{Max}(0, T_j - d_j)$. Dans la suite de ce chapitre, lorsque l'on parlera du retard sans préciser de quel type de retard il s'agit, il est sous-entendu que c'est le retard vrai qui est visé. Ajoutons enfin que l'**attente** d'une tâche se définit comme l'intervalle de temps séparant l'arrivée d'une tâche dans le système, du début de son exécution.

Si les tâches ne sont pas toutes disponibles à l'instant 0 et qu'il est possible d'interrompre une tâche en cours pour la reprendre ultérieurement au stade d'avancement où elle se trouvait (ce que l'on a défini comme étant la préemption), la règle TOM appliquée à chaque nouvelle arrivée de tâche sur l'ensemble des tâches non encore exécutées ainsi que la tâche en cours (sur la base du temps opératoire résiduel) conserve toutes les propriétés évoquées dans ce paragraphe. L'application de la règle peut donc s'effectuer de manière « myope », c'est-à-dire que lorsqu'une tâche est terminée, on ne se préoccupe que de la détermination de la première tâche à exécuter dans l'ensemble des tâches disponibles, sans tenir compte des dates d'arrivée.

I-1.1.2 La règle TOM pondéré

Dans la pratique, les travaux à effectuer peuvent ne pas tous présenter le même intérêt (articles en rupture de stock, commandes urgentes, etc.). On peut alors introduire un coefficient u_i ($u_i \geq 1$) traduisant la plus ou moins grande priorité accordée à la tâche i , par exemple à partir de l'inverse de la marge (notion intro-

1. Voir Baker (1978, [30]), p. 24-25.

duite au chapitre IV, page 301, que l'on utilisera au § I-1.1.4, page 371) ou d'un indicateur en valeur comme la marge financière associée à la commande à exécuter ou la valeur de ses composants¹. On cherche alors à minimiser le temps d'attente moyen pondéré.

$$\bar{A} = \frac{1}{n} \cdot \sum_{j=1}^n u_j A_j \quad (\text{Temps d'attente moyen pondéré}) \quad \text{relation 68}$$

On démontre² que le temps d'attente moyen pondéré est obtenu en donnant la priorité la plus forte à la tâche qui a le plus faible quotient du temps opératoire au coefficient de pondération. Cette règle récurrente, que l'on désignera sous le nom de **règle TOM pondéré**, (connue aussi sous le nom de **règle de Smith**) est donc :

$$\frac{T_1}{u_1} \leq \frac{T_2}{u_2} \leq \dots \leq \frac{T_j}{u_j} \leq \frac{T_{j+1}}{u_{j+1}} \leq \dots \leq \frac{T_n}{u_n} \quad (\text{règle TOM pondérée}) \quad \text{relation 69}$$

Illustrons cette règle par une modification de notre exemple. Supposons maintenant que les tâches ont les priorités suivantes : 1, 2, 1, 2, 3. L'ordonnancement optimal est alors 5, 1, 2, 3, 4 (voir tableau 71).

TABLEAU 71
Ordonnancement suivant la règle TOM pondérée

Tâche i	1	2	3	4	5
Temps opératoire t_i	50	150	80	200	30
Pondération u_i	1	2	1	2	3
t_i/u_i	50	75	80	100	10
Ordre de passage de la tâche i	2	3	4	5	1
Ordre de passage j	1	2	3	4	5
Tâche programmée	5	1	2	3	4
T_h/u_h	10	50	75	80	100
$T_h \cdot u_h$	90	50	300	80	400
$\sum_{h=1}^j T_h \cdot u_h$	90	140	440	520	920

$$\bar{A} = 422$$

I-1.1.3 Ordonnancement suivant la règle de la date de livraison minimale

Dans notre exemple, nous avons supposé implicitement qu'a priori n'importe quelle tâche pouvait être la dernière, c'est-à-dire que les 5 tâches devaient être

1. Dans ce dernier cas, on retrouve, à une constante multiplicative près, la valeur des encours générée par l'ordonnancement, hors valorisation du passage sur le centre de production. Économiquement, un coût de possession, lié au passage dans cet atelier, peut se déduire de cette valeur.

2. Voir Conway, Maxwell & Miller (1967, [104], p. 43-44). L'interprétation des coefficients de pondération pourrait conduire, dans cette relation, à remplacer n par la somme des coefficients de pondération. Cette transformation n'a pas d'incidence sur la solution ; la date d'achèvement moyenne pondérée passe alors de 422 à 234,4.



terminées 5,1 heures après la livraison simultanée des 5 travaux à effectuer sur le centre de production considéré. Un cas fréquent dans la pratique est celui d'une date maximale souhaitée pour la fin de l'exécution d'une ou de plusieurs tâches. Le plus souvent, aucun ordonnancement ne permet de satisfaire simultanément toutes ces dates de livraison et un arbitrage doit être effectué. Nous allons examiner maintenant les implications de règles de priorité fréquemment utilisées, mais il nous faut préalablement modifier notre exemple et introduire dans le **tableau 72** des dates de livraison d_i (l'origine du temps se situant au moment de la remise des différents travaux à effectuer, ce qui fait que d_i peut fort bien être négatif). On calculera également la **marge** de chaque tâche, que l'on définit comme la différence entre la date de livraison souhaitée et le temps opératoire et qui mesure la marge de manœuvre dont on dispose pour une tâche si l'on ne se préoccupe pas des autres tâches (ce qui correspond au concept de marge totale en ordonnancement de projet).

TABLEAU 72
Introduction des dates de livraison

Tâche i	1	2	3	4	5
Date de livraison d_i souhaitée (en centième d'heures)	100	300	410	400	200
Temps opératoire t_i (en centième d'heures)	50	150	80	200	30
Marge $d_i - t_i$	50	150	330	200	170

Table des
matières

Calculons les conséquences de l'ordonnancement TOM sur les retards de livraison, avant de présenter d'autres règles d'ordonnancement (**tableau 73**).

TABLEAU 73
Conséquences de l'ordonnancement TOM

Ordre de passage j (règle TOM)	1	2	3	4	5
Tâche programmée	5	1	3	2	4
A_j	30	80	160	310	510
Date de livraison d_j souhaitée	200	100	410	300	400
Retard vrai: $\max(0, A_j - d_j)$	0	0	0	10	110

Retard minimal: 0
Retard maximal: 110
Retard moyen: 24

Index
thématique

On montre¹ que le retard vrai maximum est minimisé (et donc le retard algébrique aussi) si l'on adopte l'ordonnancement programmant les tâches selon les dates croissantes de livraison (connu sous le nom de **règle de Jackson**). Ceci implique que l'on ait (j repérant toujours l'ordre de passage) :

$$d_1 \leq d_2 \leq \dots \leq d_j \leq d_{j+1} \leq \dots \leq d_n \quad (\text{règle de la date de livraison minimale} \Rightarrow \text{minimisation du retard maximum}) \quad \text{relation 70}$$

Appliquons cette règle sur notre exemple dans le **tableau 74**.

1. Voir Conway, Maxwell & Miller (1967, [104]) p. 30-71 (*non decreasing due-date rule*), ou Baker (1974, [30]), p. 24-25. On peut ajouter que cette règle minimise le nombre de tâches en retard si l'ordonnancement trouvé comporte moins de 2 tâches en retard (voir Baker, [30], p. 26-27 et 36). Dans le cas contraire, il faut faire appel à l'algorithme de Hodgson pour obtenir ce résultat (voir Baker, [30], p. 27-28).

TABLEAU 74
Ordonnancement selon la règle de la date de livraison minimale

Ordre de passage j	1	2	3	4	5
Date de livraison d_j souhaitée	100	200	300	400	410
Tâche programmée	1	5	2	4	3
Temps opératoire T_j	50	30	150	200	80
A_j	50	80	230	430	510
Retard vrai maximal	0	0	0	30	100

Retard minimal : 0
Retard maximal : 100
Retard moyen : 26
 $\bar{A} = 260$
 $\sigma = 183,74$



Cet ordonnancement *minimise le plus grand retard possible*, mais il ne minimise pas le retard moyen : l'application de la règle TOM avait donné par exemple un retard moyen de 24 centièmes d'heures, contre 26 ici. On n'a pas trouvé de règle simple d'ordonnancement permettant de minimiser le retard moyen, le retard d'une tâche étant maintenant nul si la date de livraison est postérieure à sa date de fin¹. Dans notre exemple, on peut assez facilement le trouver par tâtonnement, il s'agit de l'ordonnancement 1 - 5 - 2 - 3 - 4.

En cas de non disponibilité de toutes les tâches à l'instant 0 et de préemption, la démarche à utiliser pour minimiser le plus grand retard possible est analogue à celle évoquée pour la règle TOM.

I-1.1.4 Ordonnancement suivant la règle de la marge minimale

Une autre technique d'ordonnancement est parfois envisagée. Elle consiste à programmer les tâches par valeurs croissantes de leurs marges ($d_i - t_i$), les tâches ayant la plus faible marge étant supposées offrir le plus grand risque d'être en retard. Ce raisonnement est spécieux, car un tel ordonnancement ne conduit ni à une minimisation du retard moyen ni à une minimisation du retard maximum. On montre en effet que l'ordonnancement qui programme les tâches par valeurs croissantes de leur marge conduit au résultat suivant (intuitivement très peu évident) : cet ordonnancement *maximise le retard le plus faible possible*.

$$d_1 - T_1 \leq d_2 - T_2 \leq \dots \leq d_j - T_j \leq d_{j+1} - T_{j+1} \leq \dots \leq d_n - T_n \text{ (règle de la marge minimale)} \Rightarrow \text{minimisation du retard le plus faible} \quad \text{relation 71}$$

Dans notre exemple, le plus grand retard minimum possible est nul quel que soit l'ordonnancement choisi, comme on peut le voir en appliquant cette règle d'ordonnancement (tableau 75, page 372).

I-1.1.5 Modélisation générale

Pour trouver la solution de ce problème d'optimisation pour lequel aucune règle simple n'a été trouvée, on peut passer par la programmation linéaire en variables mixtes (voir chapitre XVI) qui permet de décrire, par une double contrainte, l'interdiction d'un chevauchement dans la programmation de 2 tâches quelconques². Le lecteur intéressé est invité à se reporter au chapitre IV de l'ouvrage d'Esquirol et Lopez (1999, [140]) qui illustre en détail cette formalisation pour traiter le cas d'une minimisation de la somme des retards pondérés.

1. L'un des algorithmes les plus efficaces semble être proposé par Schrage & Baker (1978, [375]).

TABLEAU 75

Ordonnancement selon la règle de la marge minimale



Ordre de passage j	1	2	3	4	5
$d_j - T_j$	50	150	170	200	330
Tâche programmée	1	2	5	4	3
Temps d'exécution T_j	50	150	30	200	80
A_j	50	200	230	430	510
d_j	100	300	200	400	410
Retard vrai maximal	0	0	30	30	100

Retard minimal: 0
 Retard maximal: 100
 Retard moyen: 32
 $\bar{A} = 284$
 $\sigma = 165,6$

I-1.2 Ordonnancement de n tâches nécessitant l'intervention de 2 centres de production

Chaque tâche comporte maintenant deux opérations distinctes effectuées par 2 centres de production différents. On suppose que la préemption est interdite. Pour simplifier l'exposé, on conviendra que ces 2 centres de production sont les centres de production A et B. On notera t_{iA} et t_{iB} les temps d'exécution de la tâche i dans chacun des centres de production. Il faut distinguer deux cas de figure: celui où l'ordre de passage des tâches est le même sur les 2 centres de production, c'est-à-dire que la tâche doit d'abord s'effectuer sur le centre de production A puis sur le centre de production B pour tous les travaux à effectuer, et celui où l'ordre technologique diffère selon les travaux. Dans un cas comme dans l'autre, *le seul critère que l'on retiendra*¹ pour juger de la performance de l'ordonnancement est celui de la *minimisation du temps total d'exécution de tous les travaux* (et qui inclut les temps d'attente devant le centre de production occupé). Deux cas de figure doivent être distingués, selon que l'ordre de passage sur les centres de production A et B est identique ou non, pour toutes les tâches.

I-1.2.1 Cas du même ordre de passage sur les centres de production A et B

Le problème traité ici est un problème de *flow shop* à 2 centres de production. Supposons que cinq tâches soient à exécuter successivement sur les centres de production A et B (les données sont celles du [tableau 76 de la page 373](#), où les temps sont exprimés en centièmes d'heure).

L'ordonnancement qui minimise le temps d'exécution de tous les travaux se trouve en utilisant l'**algorithme de Johnson** (publié en 1954) et dont on ne démontrera pas l'optimalité², et qui remplit progressivement le tableau d'affectation comme suit:

2. *Note de la page précédente.* Si la tâche j est programmée comme l'une des tâches suivant la tâche i , on doit avoir $A_j \geq A_i + T_j$; si elle est programmée comme l'une des tâches qui la précède, on doit avoir au contraire $A_i \geq A_j + T_i$. Pour exprimer le fait que la tâche j suit la tâche i , on introduit la variable indicatrice δ_{ij} qui vaut 1 dans ce cas et 0 dans le cas contraire et l'on introduit le double jeu de contraintes pour exprimer qu'une seule des contraintes précédentes soit à satisfaire: $A_j \geq A_i + T_j - (1 - \delta_{ij})M$ et $A_i \geq A_j + T_i - \delta_{ij}M$, où M est une valeur positive supérieure aux valeurs que peuvent prendre A_j et A_i (ceci correspond au cas traité dans la [note de la page 1140](#)).

1. Il s'agit en réalité du seul critère pour lequel quelques résultats significatifs ont été trouvés.

2. Voir Conway, Maxwell & Miller (1967, [104]), p. 83-89, ou Baker (1974, [30]), p. 144-146.

TABLEAU 76
Exemple de problème d'ordonnancement à 2 centres de production, avec même ordre de passage

Numéro de la tâche i	1	2	3	4	5
t_{iA}	50	150	80	200	30
t_{iB}	60	50	150	70	200



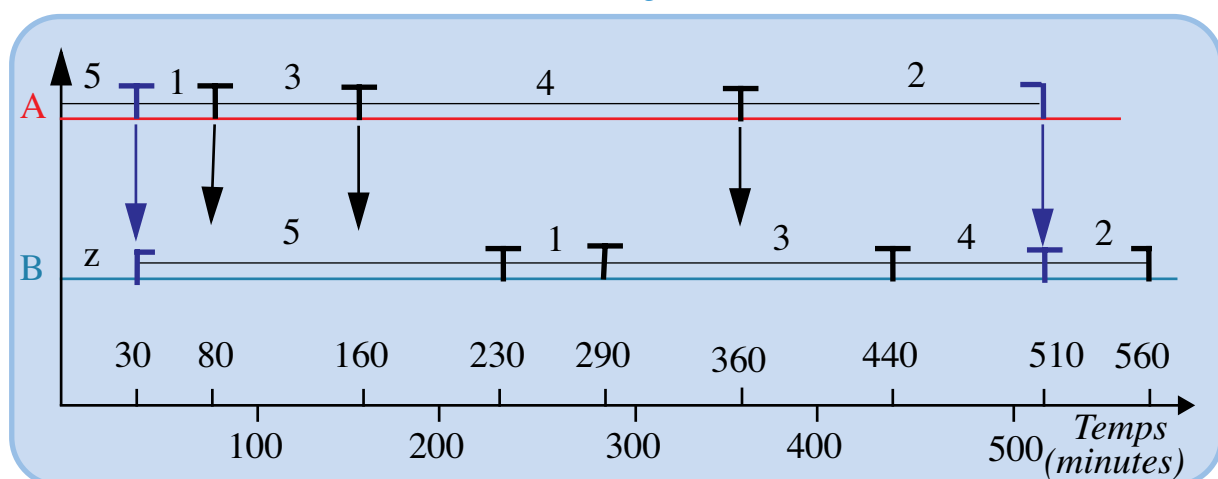
- **Étape 1** : rechercher la tâche i dont le temps d'exécution t_{ij} (avec $j = A$ ou B) est le plus faible possible.
- **Étape 2** : si $j = A$ placer cette tâche à la première place disponible en début de la séquence d'ordonnancement; si $j = B$ placer cette tâche à la dernière place disponible.
- **Étape 3** : supprimer la tâche i des tâches restant à programmer; s'il reste plus d'une tâche à programmer, revenir en étape 1 ; s'il n'en reste qu'une, sa position est imposée puisqu'il ne reste plus dans le tableau d'affectation qu'une seule place à prendre.

L'application de cet algorithme conduit à placer tout d'abord la tâche 5 en première position ($t_{5A} = 30$) puis la tâche 1 en deuxième position ($t_{1A} = 50$), puis la tâche 2 en dernière position ($j = 5$, $t_{2B} = 50$), puis la tâche 4 en avant-dernière position (dernière place disponible en fin de classement, $t_{2B} = 70$). Il ne reste plus pour la tâche 3 qu'une place possible, la troisième. La programmation optimale est alors la suivante: 5 – 1 – 3 – 4 – 2. Le calcul du temps total d'exécution est facilité en visualisant l'ordonnancement sur un graphique de Gantt (voir [figure 108](#)) qui permet en outre de voir rapidement les moments d'occupation et de non-occupation des centres de production.

Table des matières

Index thématique

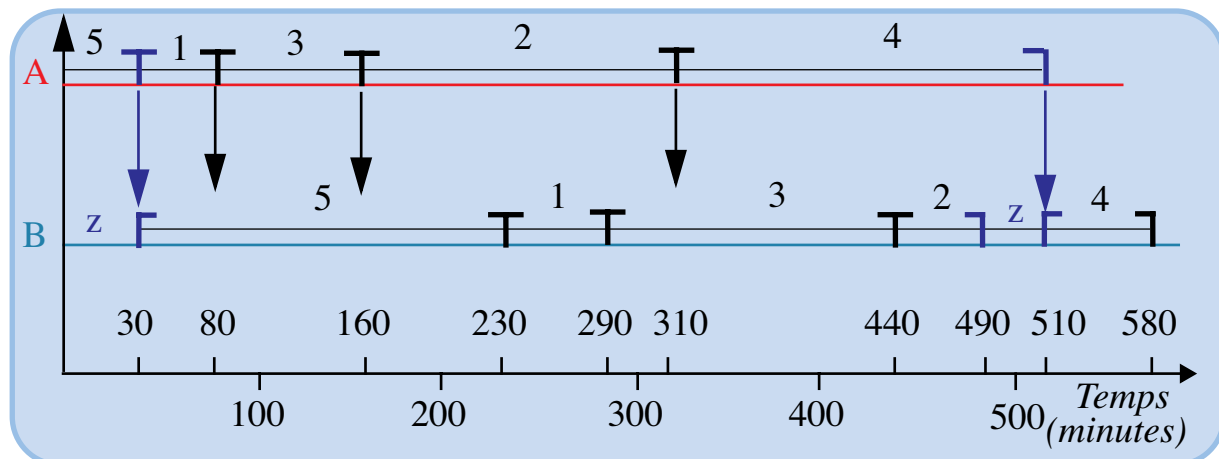
FIGURE 108
Ordonnancement selon l'algorithme de Johnson



Pour information, la [figure 109 de la page 374](#) retrace le Gantt correspondant à un ordonnancement suivant la règle TOM appliquée aux temps opératoires du centre de production A.

L'ordonnancement optimal permet de gagner 20/100 d'heure sur la date de fin d'exécution de tous les travaux, ce qui correspond à l'économie de temps réalisée

FIGURE 109
Ordonnancement suivant la règle TOM sur la machine A



sur l'utilisation du centre de production B (attente, dans la solution, de la fin de la tâche 4 sur le centre de production A, avant de pouvoir occuper le centre de production B, libre à partir de 4,9 heures).

L'utilisation de graphiques de Gantt permet, comme ces exemples l'illustrent, de visualiser le passage d'une tâche d'un centre de production à un autre, à l'aide de flèches verticales (ou obliques). Cette convention est souvent utilisée dans la représentation de **nomenclatures d'ordonnancement** qui repère différents ensembles et sous-ensembles d'un produit complexe et fournit sur un graphique de Gantt les temps opératoires correspondants ainsi que les enclenchements d'assemblage des différents composants.

I-1.2.2 Cas de la non-unicité de l'ordre de passage sur les centres de production A et B

Jackson a montré en 1957 comment adapter l'algorithme de Johnson pour tenir compte de la possibilité d'avoir des tâches ne s'effectuant pas toutes dans le même ordre (ce qui transforme le problème en un problème de *job shop*). Comme précédemment, nous ne démontrerons¹ pas l'optimalité de l'**algorithme de Jackson**, nous contentant de décrire la procédure à suivre.

Il faut tout d'abord effectuer une partition de l'ensemble initial des n tâches en quatre sous-ensembles :

- l'ensemble $\{A\}$ comprend toutes les tâches qui ne nécessitent que l'intervention du centre de production A ;
- l'ensemble $\{B\}$ comprend toutes les tâches qui ne nécessitent que l'intervention du centre de production B ;
- l'ensemble $\{AB\}$ comprend toutes les tâches qui nécessitent de passer d'abord par le centre de production A avant de passer par le centre de production B ;
- l'ensemble $\{BA\}$ comprend toutes les tâches qui nécessitent de passer d'abord par le centre de production B avant de passer par le centre de production A.

1. Voir démonstration dans Conway, Maxwell & Miller (1967, [104], p. 105-106).

On définit tout d'abord l'ordonnancement optimal sur le sous-ensemble $\{AB\}$ à l'aide de l'algorithme de Johnson. Puis on fait de même pour le sous-ensemble $\{BA\}$. Pour les sous-ensembles $\{A\}$ et $\{B\}$, l'ordre de passage des tâches n'ayant aucune importance, on sélectionne une séquence arbitraire de passage des tâches dans chaque sous-ensemble. On combine les résultats obtenus dans chaque sous-ensemble de la façon suivante :

- centre de production A : séquence optimale du sous-ensemble $\{AB\}$, puis tâches du sous-ensemble $\{A\}$, et enfin séquence optimale du sous-ensemble $\{BA\}$;
- centre de production B : séquence optimale du sous-ensemble $\{BA\}$, puis tâches du sous-ensemble $\{B\}$, puis séquence optimale du sous-ensemble $\{AB\}$.

I-1.3 Ordonnancement de 2 tâches nécessitant l'intervention de m centres de production

Une solution graphique à ce problème d'ordonnancement est connue de façon correcte depuis 1963. Elle concerne le cas de 2 tâches ne suivant pas la même séquence d'opérations (problème de *job shop*), mais utilisant le même nombre de centres de production (dans le cas contraire, on peut toujours se ramener au cas général en attribuant un temps d'exécution nul aux centres de production n'intervenant pas dans le processus productif d'une tâche). Cette procédure¹ est facile à comprendre sur un exemple. Supposons que 5 centres de production, repérés par les lettres A à E, soient nécessaires pour réaliser la tâche 1 (dans l'ordre suivant : D, B, E, A, C) et la tâche 2 (dans l'ordre technique suivant A, B, C, D, E). Les temps d'exécution, en heures, sont donnés dans le [tableau 77 de la page 376](#).

Table des
matières

Index
thématique

On établit un graphique sur les axes duquel le montant cumulé du travail accompli de chacune des tâches (l'axe vertical pour la tâche 1 et l'axe horizontal pour la tâche 2, ou l'inverse) est repéré. On commence par placer sur ces axes les temps passés sur chaque centre de production, dans l'ordre imposé techniquement. Dans cet espace à 2 dimensions, on représente par des rectangles rouges les incompatibilités liées à la demande simultanée d'un même centre de production par les 2 tâches à effectuer. Tout point du graphique correspond à l'état d'avancement simultané des 2 tâches : le point p du graphique indique après lecture de ses coordonnées que les opérations D et B de la tâche 1 sont terminées et que l'on passe à la tâche E, tandis que les opérations A et B de la tâche 2 sont terminées, et que l'opération C est aux 2/3 terminée ; l'heure correspondant à ce point est 7 heures.

Une programmation réalisable se visualise par une ligne brisée partant de l'origine 0 des axes, au point 0' (repérant la fin simultanée des 2 travaux), et ne comportant que :

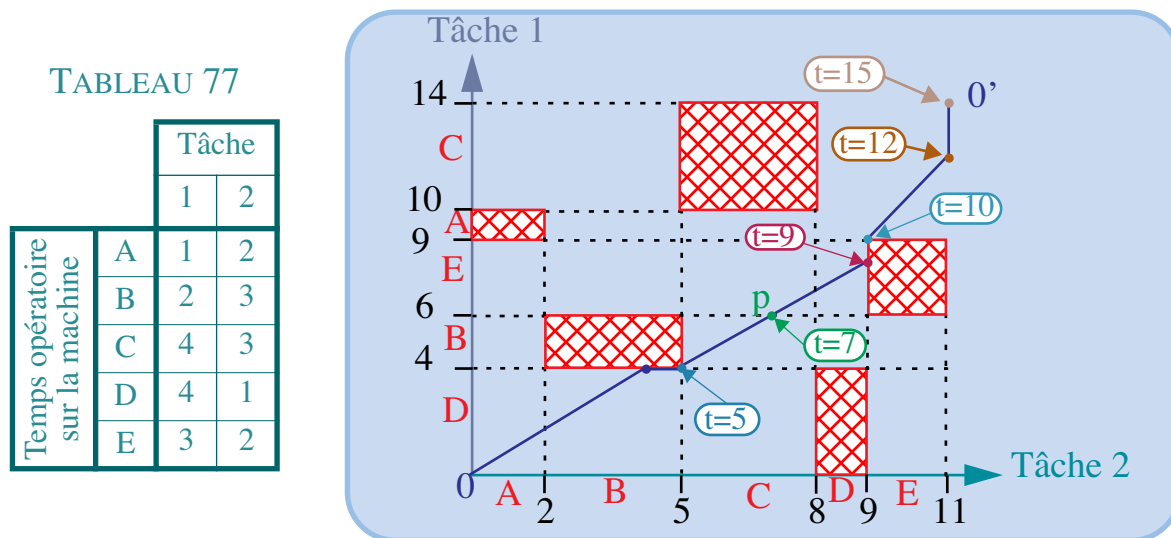
- des segments verticaux (seule la tâche 1 est en cours d'exécution),
- des segments horizontaux (seule la tâche 2 est en cours d'exécution)
- et des segments à 45° avec les axes (exécution simultanée des 2 tâches²).

1. Voir Conway, Maxwell & Miller (1967, [104]), p. 98-100 et 106-107.

2. En effet, s'il pouvait exister des segments faisant une autre pente, cela signifierait que la durée d'exécution d'une des tâches sur la machine correspondant à ce segment serait supérieure à celle annoncée.

FIGURE 110

Résolution graphique du problème d'ordonnancement des deux tâches



Le minimum du temps total d'exécution de 2 tâches (y compris donc les temps d'attente devant un centre de production occupé) est obtenu par l'ordonnancement dont la représentation sur ce graphe minimise la somme des segments horizontaux et verticaux. Cette solution optimale s'obtient généralement très vite après quelques tâtonnements.

I-1.4 Ordonnancement de n tâches nécessitant l'intervention de m centres de production

On examinera tout d'abord le cas de 3 centres de production avec un même ordre de passage (§ I-1.4.1) avant d'aborder le cas à m centres de production, avec une prise en compte d'hypothèses de fonctionnement plus générales (§ I-1.4.2, page 378).

I-1.4.1 Ordonnancement de n tâches nécessitant l'intervention de 3 centres de production (ordre identique de passage)

Johnson, dans l'article présentant sa méthode, fait remarquer que dans certains cas particuliers un problème d'ordonnancement sur 3 centres de production se ramène à un problème d'ordonnancement sur 2 centres de production. C'est le cas où le centre de production B, qui, techniquement, doit intervenir avant le centre de production C et après le centre de production A, est complètement dominé par l'un ou l'autre de ces 2 centres de production, c'est-à-dire que le plus grand temps d'exécution t_{iB} est plus faible (ou égal) que le plus petit temps d'exécution observé sur le centre de production qui le domine (étant entendu qu'il est possible que le centre de production B soit dominé par les deux centres de production).

Lorsque ce cas se produit, on reformule le problème en un problème à 2 centres fictifs de production, le premier groupant les centres de production A et B en un centre de production virtuel noté $\{AB\}$ (avec un temps opératoire $t_{iAB} = t_{iA} + t_{iB}$) et le second groupant les centres de production B et C en un centre de production virtuel noté $\{BC\}$ (avec un temps opératoire $t_{iBC} = t_{iB} + t_{iC}$). Par exemple, le

problème d'ordonnancement posé dans le [tableau 78](#) se ramène à celui du [tableau 79](#).

TABLEAU 78
Exemple de problème de flow shop à 3 centres de production

Tâche i	t_{iA}	t_{iB}	t_{iC}
1	7	1	6
2	4	3	2
3	3	2	4
4	8	2	1
5	5	1	3
	$\min t_{iA} = 3$	$\min t_{iB} = 3$	$\min t_{iC} = 1$



TABLEAU 79
Définition du problème fictif à 2 centres de production obtenu par regroupement des 2 premiers et des 2 derniers centres de production du [tableau 78](#)

Tâche i	t_{iAB}	t_{iBC}
1	8	7
2	7	5
3	5	6
4	10	3
5	6	4

Table des
matières

Index
thématique

L'application de l'algorithme de Johnson sur les données du [tableau 79](#) permet de déterminer l'ordonnancement optimal sur les 3 centres de production: tâches 3, 1, 2, 5 et 4 (voir [figure 111](#), [page 378](#)).

Certains auteurs¹ ont remarqué que lorsque les conditions imposées pour se ramener au cas de 2 centres de production n'étaient pas exactement satisfaites, la solution optimale ou une solution à performance voisine était néanmoins trouvée en utilisant cette procédure. Il a été montré² que l'algorithme de Johnson reste optimal si quelle que soit la tâche i , on a $t_{iB} \leq t_{iA}$ et $t_{iB} \leq t_{iC}$.

Le cas général du problème à 3 centres de production a toutefois une solution optimale en appliquant l'un des algorithmes utilisés en programmation linéaire en nombres entiers, et qui est connu sous le nom de *branch and bound* ; cependant la présentation de la solution «correcte» du problème à 3 centres de production dépasse le propos introductif que l'on s'est fixé ici³.

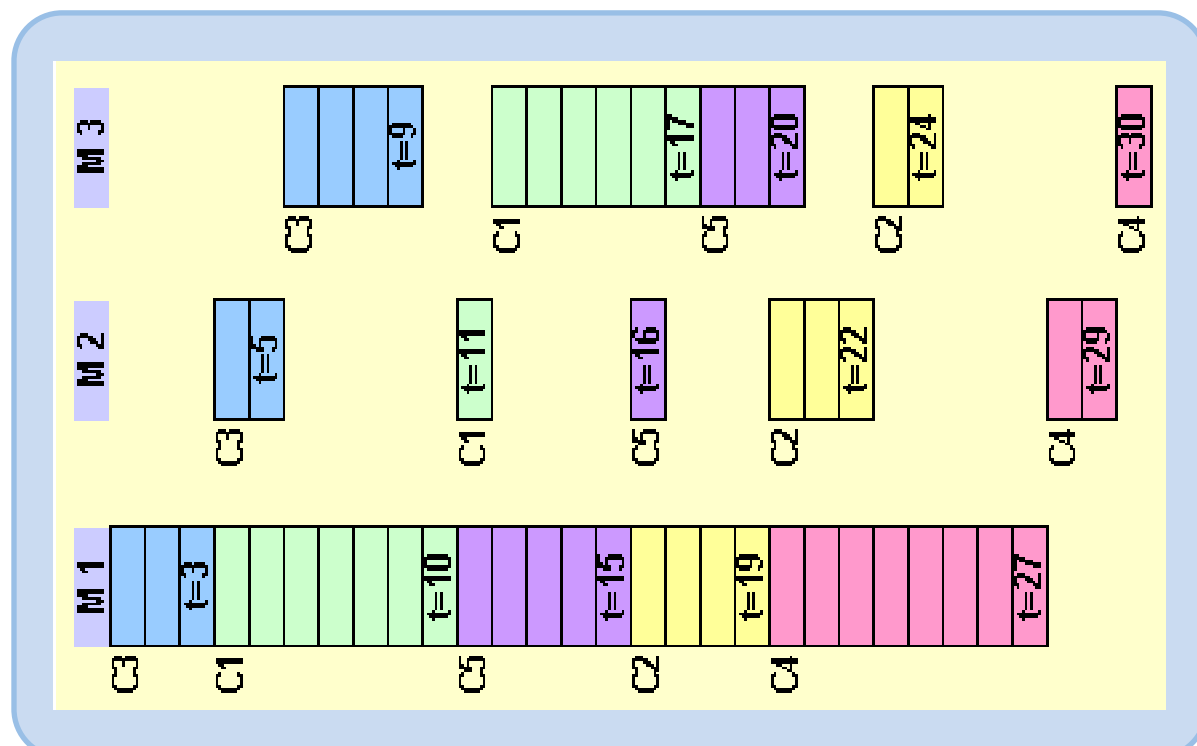
1. Giglio & Wagner (1964, [203]).

2. Burns et Rooker (1978, [79]).

3. Sur ce point, voir Johnson & Montgomery (1974, [245], p. 329-336) qui présentent la méthode et l'illustrent d'un exemple numérique traité de bout en bout; voir également Baker (1974, [30], p. 149-156 et p. 178-210).

FIGURE 111

Résolution graphique du problème d'ordonnancement à 3 centres de production (ordre identique de passage)



I-1.4.2 Ordonnancement de n tâches nécessitant l'intervention de m centres de production (ordre identique de passage)

Après avoir examiné le modèle de base dans lequel chaque centre ne possède qu'une machine (§ I-1.4.2.1), on examinera successivement des variantes de ce problème pour tenir compte des temps de montage/démontage d'outillage (§ I-1.4.2.2, page 380), de contraintes de continuité dans l'exécution des opérations d'une même tâche (§ I-1.4.2.3, page 381) et de la possibilité qu'un centre de production dispose de plusieurs machines (§ I-1.4.2.4, page 383).

I-1.4.2.1 Le modèle de base

Le problème combinatoire posé est formidable, il y a en effet $(n!)^m$ ordonnancements possibles, dans le cas (simplifié) où une même tâche ne passe qu'une fois sur chaque poste de travail. Le problème général a été formalisé en termes de programmation dynamique et en termes de programme linéaire en nombres entiers¹. La formulation générale permet d'intégrer des contraintes supplémentaires (date de livraison, capacité variable de production) et de choisir plusieurs formulations différentes de la fonction-objectif (minimisation du temps d'achèvement total, minimisation du cumul des retards par rapport aux dates de livraison souhaitées, minimisation de l'appel à des ressources supplémentaires...). Par contre, elle devient difficilement exploitable au-delà d'une trentaine de tâches ou de postes de travail.

1. Voir Conway, Maxwell & Miller (1967, [104], p. 107-108), Holloway & Nelson (1979, [234]) et Baker (1974, [30], p. 206-208 et 277-278).

Lorsque l'ordre de passage des tâches est identique et que le nombre de centres de production ne dépasse pas quelques dizaines, une solution souvent proche de la solution optimale peut être trouvée en utilisant l'algorithme de Johnson sur des groupements de centres de production successifs, connu sous le nom de l'**algorithme CDS**¹. La procédure à suivre est plus facile à comprendre à partir d'un exemple qu'à partir de sa description formelle. Prenons le cas de 5 centres de production repérés dans leur ordre d'intervention par les lettres A à E, il faut résoudre les 4 problèmes suivants (en reprenant les notations de regroupement de centres de production introduites au § I-1.4.1):

$$\{A\} - \{E\}; \{AB\} - \{DE\}; \{ABC\} - \{CDE\}; \{ABCD\} - \{BCDE\}$$

et d'une façon générale, pour m centres, on aura à résoudre $m - 1$ problèmes.

Illustrons cette procédure par un exemple numérique ne comportant que 4 centres de production (**tableau 80**). La solution obtenue (3 – 4 – 6 – 2 – 5 – 1) est

TABLEAU 80
Exemple de problème de flow shop à 4 centres de production

Tâche i	Temps d'exécution en 1/10ème d'heure			
	t_{iA}	t_{iB}	t_{iC}	t_{iD}
1	50	43	15	4
2	89	99	95	77
3	7	47	20	98
4	8	64	12	94
5	61	19	65	14
6	1	80	66	78

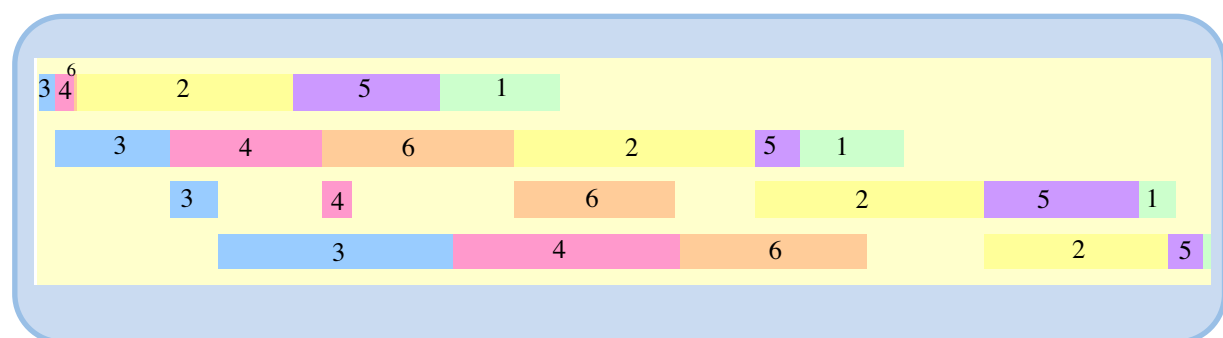


Table des
matières

Index
thématique

est proche (48,7 heures) de la solution optimale (trouvée par dénombrement) qui donne un temps minimal de 48,5 heures avec l'ordonnancement suivant 6 – 3 – 4 – 2 – 5 – 1. Le premier problème fictif (machines A et D) conduit à un temps de 51,2 heures, pour l'ordre 6 – 3 – 4 – 2 – 5 – 1. Les deux autres problèmes fictifs sont analysés dans les tableaux 81 et 82.

FIGURE 112
Ordonnancement proposé par l'algorithme CDS au problème de Flow Shop proposé au tableau 80



1. Cette méthode heuristique, due à Campbell, Dudek et Smith (1970, [84]), semble être l'une des plus performantes. Voir sur ce point Baker (1974, [30]), p. 163-165, ainsi que D.G. Dannenbring (1977, [118]).

TABLEAU 81
Deuxième problème fictif de l'algorithme CDS

Tâche i	$t_{i, A+B}$	$t_{i, C+D}$
1	93	19
2	188	172
3	54	118
4	72	106
5	80	79
6	81	144

Ce deuxième problème ($\{AB\} - \{CD\}$) donne la solution suivante: 3 – 4 – 6 – 2 – 5 – 1
Avec cet ordonnancement, l'ensemble des travaux sera terminé au bout de 48,7 heures.

TABLEAU 82
Troisième problème fictif de l'algorithme CDS

Tâche i	$t_{i, A+B+C}$	$t_{i, B+C+D}$
1	108	62
2	283	271
3	74	165
4	84	170
5	145	98
6	147	224

Ce dernier problème ($\{ABC\} - \{BCD\}$) donne la même solution que le deuxième problème

Cette solution approchée peut constituer une bonne réponse à un problème périodiquement posé: par exemple celui de l'ordonnancement du jour ouvrable suivant, avec indépendance des ordonnancements successifs. Les efforts faits au niveau de l'ordonnancement sont généralement très payants au niveau de l'utilisation des moyens productifs (on a déjà dit qu'il n'est pas rare de voir ces moyens utilisés à moins de 70% de leur capacité pour des raisons d'organisation et, au premier chef, d'ordonnancement). Mais une autre démarche peut être suivie et consiste à utiliser des règles de priorité dans l'affectation des tâches aux centres de production (*dispatching rules*) qui permettent de décider simplement pour chaque centre de production, celle des tâches en attente qu'il convient d'exécuter lorsque le centre de production sera disponible. Cette approche diffère fondamentalement de celle étudiée jusqu'ici en ce sens que les décisions sont prises «au fil de l'eau» et non en une fois pour tout un ensemble de tâches. Nous examinerons ces techniques au § III-1, page 418.

Table des
matières

Index
thématique

I-1.4.2.2 Prise en compte des temps de montage / démontage dépendants de l'ordre de passage des tâches

Dans la pratique, le problème d'ordonnancement du *flow shop* à m centres de production ne peut pas ne pas tenir compte de l'existence de **temps de montage** s_{ij} d'outillages pour permettre l'exécution de la tâche i sur la machine du centre de production j et de son temps de démontage r_{ij} , sachant que la tâche passe au centre de production suivant en même temps que commence l'opération de démontage. Pour résoudre ce problème, Proust, Drogou, Foucher & Foucheyrand (1988, [353]) proposent¹ un algorithme particulièrement efficace en se ramenant à

1. Voir également Proust, Gupta & Deschamps (1991, [353]).

l'heuristique CDS que nous venons d'examiner en définissant les temps des groupements fictifs de k centres de production pour se ramener à l'algorithme de Johnson comme suit :

- temps du premier centre de production fictive = somme des temps opératoires de i sur les k premiers centres de production augmenté du temps de montage s_{i1} et diminué du temps de montage du premier centre de production du second groupement fictif $s_{i, m-k+1}$;
- temps du second centre de production fictive = somme des temps opératoires de i sur les k derniers centres de production augmenté du temps de démontage sur le dernier centre de production r_{im} et diminué du temps de démontage du dernier centre de production du premier groupement fictif r_{ik} .

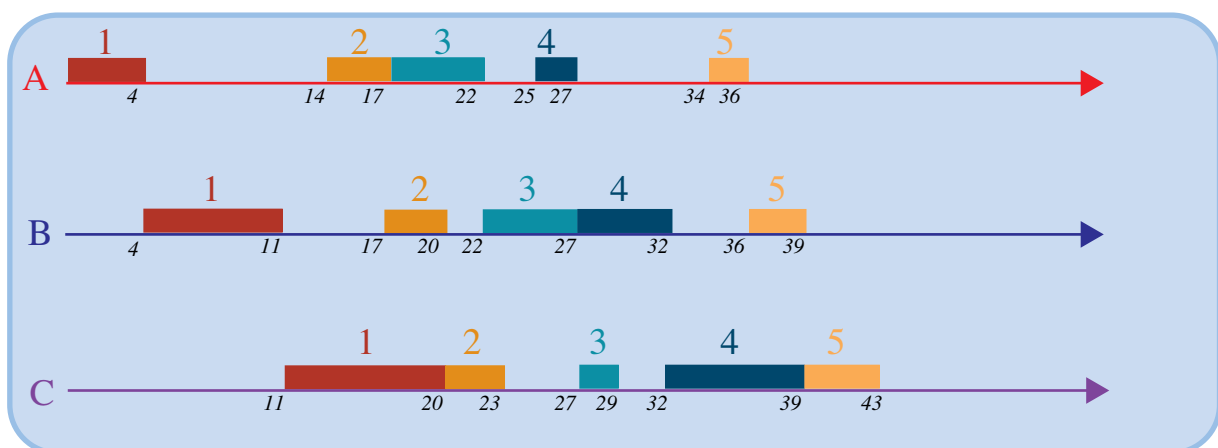
On peut ajouter que l'on peut modifier légèrement la formulation du problème pour prendre en compte les dates de livraison. Il suffit d'ajouter un centre de production fictive en fin du processus et de définir le temps opératoire de chaque tâche sur ce centre de production fictive comme étant la différence entre la plus tardive des dates de livraison et la date de livraison de cette tâche.

I-1.4.2.3 Ordonnancement de n tâches nécessitant l'intervention de m centres de production (ordre identique de passage – *sans attente*)

Dans certaines industries (industries alimentaires notamment), les opérations de chaque tâche doivent s'enchaîner sans attente et tous les centres de production sont utilisés¹ (autrement dit, il n'y a pas de temps opératoire nul). Illustrons ce problème par un exemple faisant intervenir 5 tâches comportant chacune 3 opérations, chacune d'entre elles étant réalisée successivement sur les centres de production A, B et C; les temps opératoires sont donnés dans le [tableau 83 de la page 382](#) et une programmation possible (suivant l'ordre des numéros de tâches) est donnée dans la [figure 113](#).

FIGURE 113

Exemple d'ordonnancement acceptable pour l'exemple de problème de Flow shop sans attente



Si l'on considère deux tâches i et j programmées l'une après l'autre au plus tôt, il est évident qu'une fois programmé i de telle sorte que ses opérations s'enchaînent sans attente, il convient, pour que toutes les opérations de la tâche j s'enchaî-

1. Voir Esquirol & Lopez (1999, [140]), p. 74. Une généralisation de la démarche est proposée ici.

TABLEAU 83
Temps opératoire du problème de flowshop sans attente

	Tâche 1	Tâche 2	Tâche 3	Tâche 4	Tâche 5
Machine A	4	3	5	2	2
Machine B	7	3	5	5	3
Machine C	9	3	2	7	4

nent également sans attente et que cette tâche s'achève au plus tôt, de retarder le début de j d'un certain temps, noté δ_{ij} (δ_{ij} n'ayant aucune raison d'être égal à δ_{ji}). On notera c_{ij} , somme du temps opératoire de la première opération de la tâche i et de δ_{ij} , le **décalage minimal** qui sépare le début de la première opération de la tâche i , du début de la première opération de la tâche j supposée programmée immédiatement après la tâche i (dans notre exemple, $c_{12} = 4 + 10 = 14$).

Il est facile de comprendre, à partir de notre exemple numérique, que la durée d'exécution totale des tâches dans une programmation au plus tôt sans attente (= 43, dans l'exemple ci-dessus) est la somme des opérations ($2 + 3 + 4 = 9$) de la dernière tâche (tâche 5) et des décalages successifs de la séquence de tâches antérieurement programmées ($\{4 + 10\} + \{3 + 0\} + \{5 + 3\} + \{2 + 7\} = 34$).

Si l'on est en présence d'un problème de définition d'une séquence répétitive, conduisant, dans notre exemple, à recommencer la programmation d'une tâche 1, sitôt terminée l'exécution de la séquence $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5$, la durée du cycle répétitif est $c_{12} + c_{23} + c_{34} + c_{45} + c_{51}$ et c 'est la durée la plus faible qui assure le débit le plus élevé. Ce type de problème est alors formellement identique à celui que l'on va examiner au § I-2 et qui trouve une solution avec l'algorithme de Little.

S'il s'agit simplement d'un problème d'ordonnancement unique (par opposition à périodique) d'un ensemble de tâches, la recherche d'une solution est un peu plus complexe. Dans le cas précédent, le problème d'ordonnancement posé pouvait être considéré comme équivalent à un problème de création d'une tournée de voyageur de commerce qui doit visiter chaque ville d'un ensemble, une fois et une seule, et revenir dans la ville de départ, qui peut être quelconque; dans l'analogie, la visite d'une ville correspond à l'exécution d'une tâche et le transport d'une ville i à une ville j , au décalage minimum c_{ij} . Dans le cas du problème d'ordonnancement unique, l'analogie à utiliser est celle de la tournée filaire (introduite au § II-1.3, page 918, du chapitre XIII, qui est une tournée partant d'une ville qui non prédéterminée et, après la visite unique de toutes les autres villes, ne se préoccupe pas d'un éventuel retour à la ville de départ). Il faut alors, en gardant cette analogie:

- créer une ville fictive de départ Alpha avec une distance nulle avec chacune des villes j et une interdiction de se rendre dans la ville Alpha; ceci correspond à la création d'une tâche fictive α avec les décalages minimaux suivants avec les autres tâches: $c_{\alpha j} = 0, \forall j \neq \omega$ et $c_{i\alpha} = \infty, \forall j \neq \omega$;
- créer une ville fictive Oméga de destination finale, distante de chaque ville i mais avec interdiction de partir d'Oméga vers une ville autre qu'Alpha; ceci correspond à la création d'une tâche fictive ω avec un décalage minimal avec une tâche i , autre que la tâche α , égal au cumul des temps opératoires de i

dans les m centres de production : $c_{i\omega} = \sum_{h=1}^m t_{ih}$, $\forall i \neq \alpha$ avec $c_{\omega j} = \infty$,

$\forall j \neq \alpha$, $c_{\omega\alpha} = 0$ et $c_{\alpha\omega} = \infty$;

- la distance entre deux villes quelconques autres qu'Alpha et Oméga est égale au décalage minimal c_{ij} .

Dans ces conditions, on se ramène au problème classique de la tournée du voyageur de commerce: en partant d'Alpha, on passe successivement dans toutes les villes, sans savoir par laquelle on commence, puis on arrive à Oméga avant de repartir nécessairement sur Alpha. Dans l'analogie, on traite toutes les tâches, sauf la dernière, en cumulant les décalages minimaux et on ajoute le cumul des temps opératoires de la dernière tâche. Reprenons notre exemple, la matrice des coûts est, dans le cas d'ordonnancement non répétitif, celle du [tableau 84](#), la solution optimale étant $\alpha \rightarrow 4 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 5 \rightarrow \omega$, pour une durée totale de 33.



TABLEAU 84
Tableau des décalages minimaux, avec tâches fictives

		Successeur						
		α	1	2	3	4	5	ω
Prédécesseur	α	∞	0	0	0	0	0	∞
	1	∞	∞	14	10	13	15	52
	2	∞	3	∞	3	4	4	14
	3	∞	6	7	∞	8	8	29
	4	∞	3	8	4	∞	9	24
	5	∞	2	2	2	3	∞	9
	ω	0	∞	∞	∞	∞	∞	∞

Table des
matières

Index
thématique

I-1.4.2.4 Le flow shop hybride

Cette classe de problèmes se caractérise par le fait que chaque centre de production peut disposer de plusieurs machines substituables, ce nombre pouvant varier d'un centre de production à l'autre. Une solution empirique satisfaisante consiste à travailler en deux temps¹ :

- On commence par déterminer un ordre de passage des tâches dans les centres de production (problème de séquençement), par exemple en utilisant l'algorithme CDS.
- On affecte ensuite les opérations aux machines. On commence par les opérations à réaliser sur l'une des machines du premier centre, selon des règles définies ci-après ; il en découle des dates d'arrivée des tâches devant les machines du deuxième centre. On affecte ensuite les tâches aux machines du deuxième centre, en fonction de leurs dates d'arrivée, des disponibilités des machines et de règles. On recommence ensuite ce processus jusqu'au dernier centre de production. L'affectation s'effectue à l'aide de l'une des règles de priorité suivante :

1. Ce problème est traité en détail dans Lopez & Roubellat (2001, [286], chap. X) ; voir également Vignier, Billaut & Proust (1999, [432]).

- La **règle FAM** (*First Available Machine*), argement utilisée, affecte la tâche à la machine qui se libère le plus tôt.
- La **règle LSM** (*Latest Start Machine*) choisit la machine libre le plus tard parmi les machines libres quand l'opération est disponible ou la machine libre le plus tôt si aucune machine n'est libre lorsque l'opération est disponible. Cette règle tend à minimiser le temps d'inactivité des machines utilisées et donc à n'utiliser que celles qui sont nécessaires.
- La **règle ECT** (*Earliest Completion Time*), utilisée si la durée de l'opération varie en fonction de la machine retenue, affecte la tâche à la machine qui permet l'exécution la plus rapide. Cette règle tend à maximiser l'utilisation des machines les plus performantes.

I-1.5 Ordonnancement de n tâches nécessitant l'intervention de m centres de production (cheminement libre – *open shop*)

Très peu de résultats analytiques existent¹. Si deux centres de production seulement sont mobilisés, on démontre que l'ordonnancement minimisant la durée d'achèvement de l'ensemble des tâches est obtenu en appliquant la **règle LAPT** (*Longest Alternate Processing Time first*) qui consiste à sélectionner, sur le centre de production libre, l'opération de plus grande durée sur l'autre centre. Dans l'application de cette règle, les tâches dont une opération a été réalisée ont la même priorité, la plus faible, et, lorsque ces dernières sont considérées, elles le sont de manière arbitraire. Cette règle reste optimale dans le cas préemptif.

Illustrons par un exemple l'application de cette règle, en partant des données du **tableau 85**.

TABLEAU 85
Exemple de problème d'ordonnancement à cheminement libre dans 2 centres de production

Numéro de la tâche i	1	2	3	4
t_{iA}	50	150	80	200
t_{iB}	60	50	100	70

L'application de la règle LAPT conduit aux étapes suivantes:

- en $t = 0$, l'opération la plus longue est l'opération A de la tâche 4, on charge donc l'opération de la tâche 4 à exécuter sur B (date d'achèvement en $t = 70$) (choisie arbitrairement parmi les 2 centre libres);
- en $t = 0$, le centre A étant libre, les opérations des tâches 1, 2 et 3 sont donc candidates; l'opération la plus longue sur le centre B parmi ces candidats est celle de la tâche 3 (100), ce qui conduit à charger l'opération A de la tâche 3 (date d'achèvement en $t = 80$);
- en $t = 70$, le centre B se libère, les opérations des tâches 1 et 2 sont donc candidates (3 étant en cours); l'opération la plus longue sur A de ces candi-

1. Voir Esquirol & Lopez (1999, [140], chap. IV) et le chapitre de Prins dans Lopez & Roubellat (2001, [286], chap. XI).

- ats est celle de la tâche 2 (150); on charge donc l'opération B de la tâche 2 (date d'achèvement en $t = 70 + 50 = 120$);
- en $t = 80$, le centre A se libère, les opérations des tâches 1 et 4 sont donc candidates; l'opération de la tâche 1 (qui n'a encore aucune réalisation de tâche) est plus prioritaire que celle de la tâche 4 (qui a déjà une opération réalisée); on charge donc l'opération A de la tâche 1 (date d'achèvement en $t = 80 + 50 = 130$);
 - en $t = 120$, le centre B se libère, l'opération de la tâche 3 est candidate unique (1 étant en cours); on charge donc l'opération B de la tâche 3 (date d'achèvement en $t = 120 + 100 = 220$);
 - en $t = 130$, le centre A se libère, sont donc candidates les opérations des tâches 2 et 4 qui ont déjà toutes deux une opération exécutée sur B; elles sont équivalentes et on charge arbitrairement l'opération A de la tâche 2 (date d'achèvement en $t = 130 + 150 = 280$), puis l'opération A de la tâche 4 (date d'achèvement en $t = 280 + 200 = 480$) qui est la dernière opération à exécuter sur ce centre;
 - en $t = 220$, le centre B se libère, on charge donc l'opération B de la tâche 1 (date d'achèvement en $t = 220 + 60 = 2800$) qui est la dernière à réaliser.

Le travail est donc achevé en $t = 480$.

Lorsqu'il y a plus de 2 centres de production, on peut utiliser des heuristiques, lesquelles ne garantissent pas l'optimum. Les travaux effectués montrent qu'en général la meilleure règle est la **règle MWR** (*Most Work Remaining*) qui privilégie la tâche candidate dont la durée cumulée des opérations non encore traitées est maximale.

Table des
matières

I-1.6 Ordonnancement de n tâches nécessitant l'intervention de m centres de production (ordre de passage quelconque)

Index
thématique

Ce cas général que l'on a qualifié de problème de *job shop* (voir page 362), se caractérise par le fait que l'ordre de passage entre les centres de production varie d'une tâche à l'autre et que, pour une même tâche, certains centres de production peuvent être utilisés plusieurs fois et d'autres pas. On verra ultérieurement (§ I-3, page 404) comment caractériser plus complètement cette classe de problèmes. Aucun résultat analytique n'est disponible pour résoudre cette classe de problèmes. On peut toujours trouver une formulation d'un problème concret par la programmation mathématique, mais la résolution numérique optimale de problèmes d'une certaine dimension, est le plus souvent hors de portée (voir le § I-2.2.4, page 404). On cherchera à les résoudre plutôt à l'aide d'approches simulatoires s'appuyant sur des méthodes heuristiques (voir § II-2, page 410).

On examinera cependant ici le problème statique de l'ordonnancement en cas d'existence d'un **goulot d'étranglement**, c'est-à-dire de centre de production plus sollicité que les autres au point de conditionner le débit global de production du système productif étudié. La démarche suivie depuis très longtemps – et formalisée à partir de la fin des années soixante-dix – consiste à piloter le système en s'appuyant sur un ordonnancement défini pour ce goulot d'étranglement et qui sert de pivot pour l'ordonnancement global et à faire en sorte qu'il ne puisse être affecté par une quelconque rupture de charge (en particulier grâce à un stock-tampon), parce que l'heure de travail perdu dans un goulot d'étranglement est

perdue pour l'ensemble du système productif. De nombreuses approches ont été proposées¹ ; on présentera ici une démarche heuristique donnant de bons résultats qui connaît de nombreuses sophistications, la démarche utilisée ici n'étant que l'une des démarches possibles. Elle s'appuie sur cinq étapes. Elle sera illustrée par l'exemple du [tableau 87](#).

TABLEAU 86
Problème de job shop

1		2		3		4		5	
Machine	durée	Machine	durée	Machine	durée	Machine	durée	Machine	durée
A	5	A	3	C	8	B	5	D	7
C	7	B	5	A	4	D	4	C	15
D	9	C	10	B	3	C	6	A	4
-	-	D	4	-	-	B	7	-	-

- On commence par détecter le centre de production qui constitue un goulot d'étranglement. On peut, par exemple, retenir celui qui a la plus forte charge de travail (dans notre exemple, c'est la machine C), mais les choses ne sont pas toujours aussi simples et des approches plus sophistiquées peuvent être utilisées (voir [page 433](#)). En cas de doute sur l'identité du centre qui rythme la production du système productif, on peut toujours recommencer la démarche décrite ci-après pour chacun des goulots possibles et retenir la meilleure des solutions trouvées.
- On considère qu'en amont et en aval de ce goulot, on est à capacité infinie, ce qui revient à dire que les opérations de chacune des tâches peuvent s'exécuter au plus tôt, aucun conflit dans l'utilisation d'une même machine n'étant censé arriver. Ceci revient à considérer que :
 - avant le goulot, le travail s'effectue sur une machine fictive sur laquelle sont réalisées toutes les opérations antérieures, pour une durée égale à la somme des durées (par exemple, pour la tâche 2, on cumule les durées des opérations effectuées sur les machines A et B), ces durées sont considérées ensuite comme les dates d'arrivées (au plus tôt) dans le centre considéré comme goulot d'étranglement ; les tâches qui n'utilisent pas ce centre critique peuvent être fusionnées dans cet ensemble ou traitées à part, une fois réalisé l'ordonnancement des tâches utilisant ce goulot d'étranglement ;

Table des
matières

Index
thématique

1. Voir en particulier Morton & Pentico ([309], 1993) qui consacrent de longs développements à ces approches heuristiques (en particulier aux chapitres IX et X) ; voir également Narasimhan, McLeavey & Billington (1995, [314], chap. XVI). Cette approche du goulot d'étranglement a été popularisée par la méthode OPT, proposée par Goldratt et Fox (1984, [206] et 1986, [207]) et qui fait appel à la [théorie des contraintes](#). Ce logiciel OPT, qui a connu son heure de gloire, a connu des difficultés aux États-Unis en raison de la confidentialité des algorithmes utilisés. Il semblerait que l'usage raisonné d'heuristiques donne des résultats de qualité similaires (Morton & Pentico ([309], 1993, chap. X). L'idée que Goldratt et Fox ont popularisée est celle d'une synchronisation des flux, rythmés par la production du goulot d'étranglement, puisque toute accélération de débit en amont du goulot ne peut que conduire à la création inutile d'encours et que l'accélération en aval conduit seulement à créer des périodes sans sortie de production, le débit moyen du système productif restant inchangé ; par ailleurs, l'introduction d'un stock-tampon (*time buffer*) en amont du goulot permet d'éviter toute rupture de charge à ce niveau, sachant que celle-ci conduit à une sous-activité de l'ensemble du système productif.

- après le goulot, le travail s'effectue sur une machine fictive sur laquelle sont réalisées toutes les opérations postérieures, pour une durée égale à la somme des durées (par exemple, pour la tâche 3, on cumule les durées des opérations effectuées sur les machines A et B); si des dates de livraison sont assignées aux tâches, on retranche à ces dates ces durées pour déterminer des dates de livraison (au plus tard) à retenir pour le goulot d'étranglement; si la tâche 3 avait comme date de livraison 20, la date de livraison pour la machine C serait alors $20 - 7 = 13$.

Le résultat de ce travail est consigné dans le [tableau 88](#).

TABLEAU 87
Transformation du problème de job shop autour de la machine critique C

Tâche 1		Tâche 2		Tâche 3		Tâche 4		Tâche 5	
Machine	durée	Machine	durée	Machine	durée	Machine	durée	Machine	durée
Avant C	5	Avant C	8	Avant C	0	Avant C	9	Avant C	7
C	7	C	10	C	8	C	6	C	15
Après C	9	Après C	4	Après C	7	Après C	7	Après C	4

- On cherche alors à résoudre le problème du goulot d'étranglement, en partant des dates d'arrivées qui viennent d'être calculées et, le cas échéant, en tenant compte des dates de livraison rectifiées comme on vient de l'indiquer. Ce problème à résoudre porte sur une machine unique et est connu dans la littérature spécialisée comme le *simple embeded one-resource problem*. L'une des règles introduites au § I-1.1, page 364, peut être utilisée; dans notre exemple, c'est la règle TOM qui a été utilisée dynamiquement, sur la base des seules opérations exécutables et sans préemption:
 - en $T = 0$: chargement de la tâche 3 immédiatement disponible (durée 8);
 - en $T = 5$: arrivée de la tâche 1 (durée 7);
 - en $T = 7$: arrivée de la tâche 5 (durée 15);
 - en $T = 8$: fin de la tâche 3, libération de la machine C; arrivée de 2 (durée 10); chargement de 1 (en application de la règle TOM, les tâches 1 et 5 étant candidates);
 - en $T = 9$: arrivée de 4 (durée 6);
 - en $T = 15$: fin de la tâche 1, libération de la machine C; chargement de la tâche 4 (en application de la règle TOM, les tâches 4 et 5 étant candidates);
 - en $T = 21$: fin de 4, libération de la machine C; chargement de la tâche 2 (en application de la règle TOM, les tâches 2 et 5 étant candidates);
 - en $T = 31$: fin de la tâche 2, libération de la machine C; chargement de la tâche 5 (candidat unique);
 - en $T = 46$: fin de la tâche 5.
- On cherche ensuite à résoudre le problème des centres situés en amont du goulot, en prenant comme dates de livraison celles retenues pour définir les arrivées dans le goulot (autrement dit, celles reposant sur l'hypothèse de la capacité infinie du «système productif amont»). Là encore, plusieurs règles sont possibles (notamment la règle S/OPN qui sera introduite à la [page 413](#)); on a utilisé ici, en cas de conflit dans la mobilisation d'une ressource, la tâche devant débiter le plus tôt sur le goulot (critère de la date de livraison minimale introduit à la [page 370](#) et utilisé sur le groupe restreint des opérations immédiatement exécutables). Il en résulte généralement une modification des dates d'arrivées au goulot d'étranglement que l'on considère comme sans grande importance si la date de fin d'exécution du travail sur le goulot n'est

- pas retardée mais qui peut conduire à recommencer l'étape précédente (réordonnancement sur le goulot, en partant des nouvelles dates d'arrivées).
- Les dates de fin des tâches sur le goulot sont ensuite considérées comme les dates d'arrivées dans le système-aval; un ordonnancement est alors calculé en utilisant des règles de priorité. Si certaines machines se retrouvent simultanément en amont et en aval du goulot (dans notre exemple, c'est le cas de la machine B, à cause de la tâche 4), il convient de considérer comme non révisable la programmation décidée en amont du goulot (sauf acceptation d'une nouvelle itération).

L'ordonnancement obtenu à l'issue de cette démarche dans notre exemple est consigné à la **figure 114**.

FIGURE 114
Ordonnancement du problème de job shop

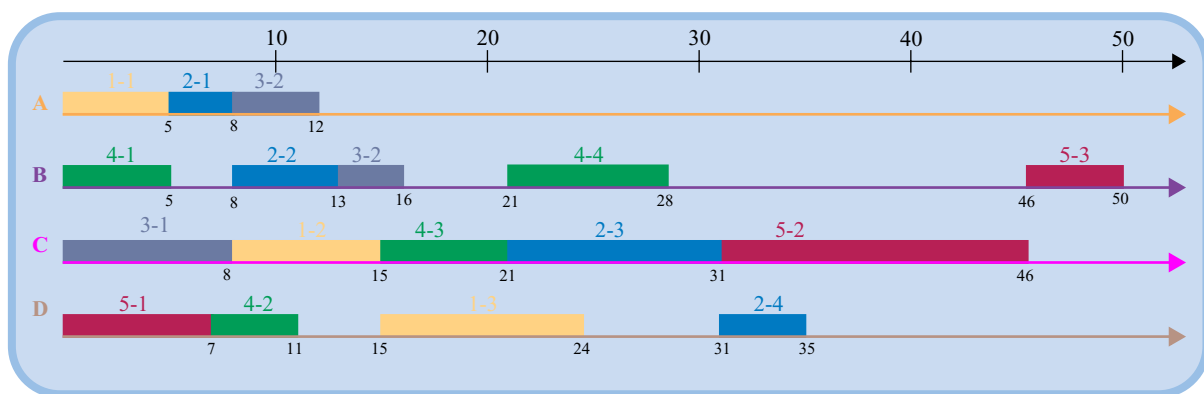


Table des
matières

I-2 Modèles statiques: cas du coût de lancement total variable avec l'ordonnancement retenu

Nous n'examinerons ici que le cas de n tâches nécessitant l'intervention d'un seul centre de production ($m = 1$) et pour lequel l'ordre de passage influe sur les coûts de lancement. Prenons l'exemple d'une usine de peinture: différentes couleurs de peinture sont produites en séquence sur le même équipement, lequel doit être soigneusement nettoyé lorsque l'on change de couleur. Le temps total de nettoyage (en heures) passé par l'équipe chargée du nettoyage est le suivant (en limitant ici le problème à 4 couleurs) :

TABLEAU 88
Temps de nettoyage pour passer d'une couleur à l'autre

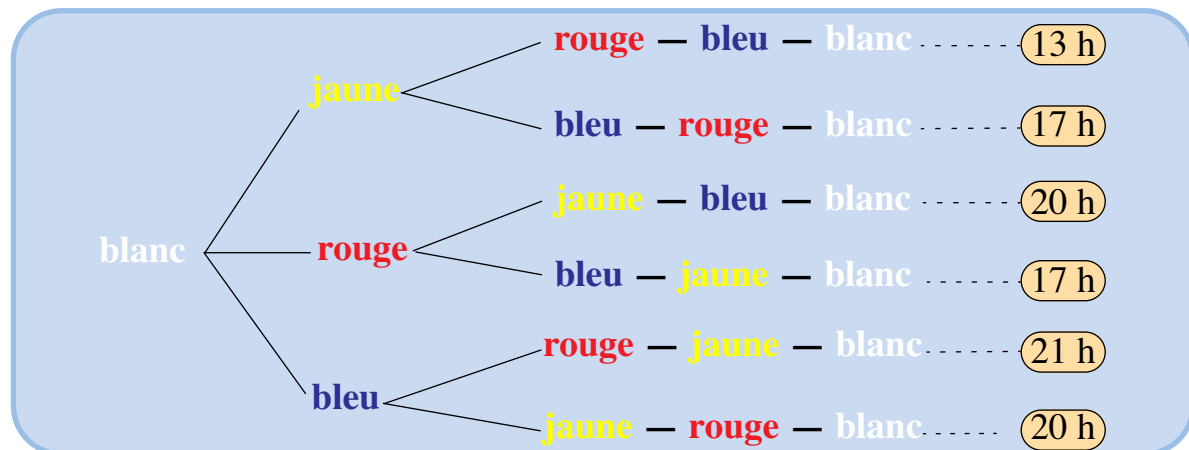
		Nouvelle couleur			
		blanc	jaune	rouge	bleu
Couleur initiale	blanc	0	1	2	3
	jaune	6	0	1	2
	rouge	8	6	0	1
	bleu	10	8	6	0

Index
thématique

Le problème posé revient à trouver l'ordre de passage qui minimise le temps total de nettoyage. On peut, bien sûr, trouver une solution à ce problème en explicitant toutes les combinaisons possibles au nombre de $(n - 1)! = 3! = 6$, ce qu'illustre la [figure 115](#).

FIGURE 115

Explicitation de tous les ordres de changement de couleurs du [tableau 88](#)



Mais cette technique n'est envisageable que si le nombre m de tâches est faible : pour 6 couleurs seulement, le dénombrement conduit à 120 comparaisons du résultat de 6 additions, c'est-à-dire que cette solution d'énumération exhaustive est rapidement inapplicable. Aussi a-t-on recours, pour résoudre ce problème, à l'analogie que ce problème présente avec celui du voyageur de commerce pour lequel un algorithme de résolution efficace a été trouvé. Dans ce problème, le voyageur de commerce quitte une ville de départ et doit visiter au cours d'une tournée, $m - 1$ villes différentes, en n'y passant qu'une fois, avant de retourner à la ville de départ, et cherche à minimiser son coût de transport. Ce coût de transport c_{ij} entre une ville i et une ville j est de même nature que le coût de lancement de la tâche j lorsque celle-ci succède à la tâche i .

Pour résoudre le problème du voyageur de commerce, on peut faire appel à la programmation dynamique (que l'on étudiera en [section II, page 562](#), du [chapitre VIII](#)), mais si les performances de celle-ci sont légèrement meilleures que celles de l'explicitation de toutes les solutions possibles¹, elles restent cependant mauvaises par rapport à l'utilisation de l'algorithme que l'on va étudier. On reviendra en détail sur ces problèmes de transport au [chapitre XIII](#).

Sur le plan formel, le problème du voyageur de commerce peut se formuler sous la forme du programme linéaire suivant, où x_{ij} est une variable susceptible de prendre seulement les valeurs 0 ou 1, la valeur 0 signifiant que le voyageur de commerce ne part pas de la ville i pour se rendre à la ville j , la valeur 1 ayant la signification contraire.

1. Voir Conway, Maxwell & Miller (1967, [\[104\]](#), p. 61-65) pour la formulation du problème du voyageur de commerce sous la forme d'un programme dynamique et les performances comparées de cette méthode avec celles de l'énumération et de l'algorithme de *branch and bound*. Voir également l'article de Crowder et Padberg (1980, [\[113\]](#)).

- Minimiser $\sum_{i=1}^m \sum_{j=1}^m c_{ij} \cdot x_{ij}$ (minimisation du coût total de transport)
- sous contrainte $\sum_{i=1}^m x_{ij} = 1$ pour $j = 1, \dots, m$ et $\sum_{j=1}^m x_{ij} = 1$ pour $i = 1, \dots, m$.

Les premières contraintes s'interprètent comme le fait qu'il ne saurait y avoir qu'une seule ville i dans laquelle se trouvait le voyageur de commerce avant de se rendre dans la ville j et les secondes contraintes s'interprètent comme le fait qu'en quittant la ville i le voyageur de commerce ne peut se rendre que dans une seule ville j .

L'algorithme de résolution d'un programme linéaire en nombres entiers le plus utilisé pour cette classe de problèmes est sans aucun doute celui connu du *branch and bound*, qui est une méthode de recherche par tâtonnements de la solution optimale entière, en s'aidant des solutions trouvées par un algorithme classique de programme linéaire du type simplex. Nous ne présenterons pas cette technique¹ car en réalité lorsque les valeurs que les variables sont susceptibles de prendre sont 0 ou 1, cet algorithme général se simplifie. Nous présenterons au § I-2.1 l'adaptation de cette méthode², en nous appuyant sur un exemple numérique. Indiquons tout de suite que la description détaillée de l'algorithme sera longue car chaque calcul sera au départ amplement expliqué, mais qu'en pratique l'utilisation de cet algorithme est très rapide pour des problèmes comportant quelques dizaines de villes. Le logiciel disponible sur le CD-Rom détaille les calculs. Nous ferons ensuite quelques remarques complémentaires (§ I-2.2, page 401).

I-2.1 Présentation de l'algorithme de Little, Marty, Sweeney & Karel

Supposons que notre voyageur de commerce ait à visiter successivement chacune des 6 villes suivantes: A, B, C, D, E, F, en partant de l'une quelconque d'entre elles, puis retourne à son point de départ. On conviendra ici d'appeler trajet, noté $i \rightarrow j$, l'acheminement direct entre les villes i et j . Pour des raisons diverses, le coût de transport de la ville i à la ville j n'est pas égal au coût de transport de la ville j à la ville i (mais s'il en était ainsi, l'algorithme que l'on va présenter resterait toujours applicable). L'unité dans laquelle ces coûts, portés dans le [tableau 89](#) de gauche, sont exprimés importe peu ici, car notre propos est d'illustrer une démarche. Les coûts portés dans la diagonale du [tableau 89](#) sont infinis. D'une manière générale, la convention $c_{ij} = \infty$ a pour objet d'empêcher (coût prohibitif) le voyage direct de i vers j ; l'application de cette convention au trajet $i \rightarrow j$ revient à dire que lorsque l'on quitte la ville i , c'est pour se rendre dans une ville différente.

1. Voir Wagner (1975, [435], p. 484-490) qui a une présentation très claire de la démarche suivie avec cette méthode et s'appuie sur un exemple numérique traité de bout en bout.

2. Trouvée par Little, Murty, Sweeney & Karel en 1963; voir Conway, Maxwell & Miller (1967, [104], p. 56-61), l'exemple numérique donné ici est tiré de cet ouvrage (il n'est pas évident en effet de bâtir un exemple numérique simple permettant d'illustrer toutes les implications de l'algorithme); voir également Johnson & Montgomery (1974, [245], p. 340-345).

La recherche de la tournée optimale retenue dans cet algorithme est de type **Procédure par Séparation et Évaluation** (PSE), procédure connue également sous le nom de **branch and bound**. Elle passe par la résolution successive d'un certain nombre de problèmes partiels à l'issue desquels on prend la décision d'inclure ou d'exclure de la solution finale (ici la tournée du voyageur de commerce) une solution partielle (ici la décision de partir de la ville i pour se rendre dans la ville j). Cette procédure que l'on va décrire en détail à partir de cet exemple s'effectue en un certain nombre d'itérations au cours desquelles on restreint le domaine de recherche en adoptant une solution partielle (étape de séparation) et on évalue la performance de la concaténation des solutions partielles adoptées (étape d'évaluation) pour vérifier s'il faut ou non poursuivre la recherche dans le sous-ensemble de solution auquel on est arrivé. Examinons l'application de ces principes au problème du voyageur de commerce proposé¹ par Little, Marty, Sweeney & Karel (1963, [282]).

TABLEAU 89



Données du problème du voyageur de commerce

Tableau intermédiaire du processus de réduction

		Ville d'arrivée						minimum
		A	B	C	D	E	F	
ville de départ	A	∞	1	7	3	14	2	1
	B	3	∞	6	9	1	24	1
	C	6	14	∞	3	7	3	3
	D	2	3	5	∞	9	11	2
	E	15	7	11	2	∞	4	2
	F	20	5	13	4	18	∞	4
		minimum						
		0	0	3	0	0	0	

Avant toute chose, il faut procéder à la *réduction de la matrice*. Cette opération consiste tout d'abord à soustraire de chaque ligne du tableau de gauche le plus petit des éléments de la ligne (porté en marge de la dernière colonne du **tableau 89** de gauche, **page 391**) des autres éléments de cette ligne. Dans ce nouveau tableau (**tableau 89** de droite), on soustrait ensuite le plus petit des éléments d'une colonne (ce minimum, porté en marge de la dernière ligne du **tableau 89** de droite, sera le plus souvent nul) des autres éléments de la colonne. On obtient alors la **matrice réduite** (**tableau 90**), qui se caractérise par la présence d'un zéro au moins dans chaque ligne et dans chaque colonne.

L'interprétation de la première étape de la réduction de la matrice est simple : on a remplacé chaque coût de transport entre une ville i et une ville j par la différence entre ce coût de transport et le coût de transport le plus faible pour quitter cette ville i . La somme de ces coûts minimaux ($1 + 1 + 3 + 2 + 2 + 4 = 13$) corres-

1. Pour une étude complète de cette classe de problèmes, voir Lawler, Lenstra, Rinnooy Kan & Shmoys (1992, [274]).

TABLEAU 90
Résultat du processus de réduction de la matrice des coûts

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0	3	2	13	1
	B	2	∞	2	8	0	23
	C	3	11	∞	0	4	0
	D	0	1	0	∞	7	9
	E	13	5	6	0	∞	2
	F	16	1	6	0	14	∞

pond au coût minimal qui sera à la charge du voyageur de commerce lorsqu'il aura quitté chacune des 6 villes.

La matrice obtenue à la fin de la première étape de réduction (tableau 89 de droite) s'analyse donc comme une matrice de coûts de transports qui s'ajoutent au coût de 13. L'examen de ce tableau montre que, pour se rendre dans la ville C, il faudra encore dépenser au moins 3 : la seconde étape de la réduction de la matrice vise à obtenir des coûts d'arrivée calculés par différence avec le coût d'arrivée minimal. Le tableau 90 s'interprète donc comme le tableau de coûts de transport à ajouter à une dépense minimale de base $x = 13 + 3 = 16$ (c'est-à-dire la somme des valeurs minimales des lignes du tableau initial et de celle des valeurs minimales des colonnes du tableau obtenu après la première étape de réduction) en dessous de laquelle on ne saurait descendre, puisque le voyageur de commerce dans sa tournée quittera toutes les villes, et se rendra dans toutes les villes.

C'est sur cette matrice réduite que s'effectue la recherche de la tournée idéale. Le branchement (le *branch*, du *branch and bound*) consiste à décider quel trajet $i \rightarrow j$ doit être ajouté à un ensemble de trajets déjà sélectionnés¹, les décisions successives pouvant se visualiser par un arbre de décision, comme on le verra rapidement. Chaque décision de sélection ou de rejet d'un trajet correspond à la résolution d'un problème partiel. Dans cet exemple, on devra résoudre successivement six problèmes partiels pour trouver une solution optimale par cet algorithme, ce que nous allons examiner maintenant.

I-2.1.1 Résolution du problème de la première itération

Au départ, aucun trajet n'est sélectionné, et il semble raisonnable de restreindre le choix aux seuls trajets qui n'accroissent pas le coût minimal de 16. Seuls les trajets qui ont des zéros dans la matrice réduite répondent à ce critère. Le raisonnement est simple, *on retiendra le trajet qui, s'il n'était pas retenu, accroîtrait le plus la dépense minimale x.*

Examinons les implications de ce que l'on vient de dire, sur l'un quelconque de ces trajets, le trajet $A \rightarrow B$. Si l'on n'inclut pas $A \rightarrow B$, il faudra ajouter à x :

- au moins 1 pour se rendre en B, qui est le plus faible coût pour se rendre en B sans partir de A,

1. Cette étape est celle de la séparation dans la terminologie, moins imagée, retenue en France (approche PSE).

- et au moins 1, qui est le plus faible coût pour partir de A et se rendre dans une ville autre que B.

La pénalité a_{AB} associée à la décision de ne pas retenir $A \rightarrow B$, décision que l'on notera $A \nrightarrow B$, est donc de $a_{AB} = 1 + 1 = 2$, ce qui revient à dire que le coût d'une tournée qui n'inclurait pas (A, B) ne saurait être inférieur à $16 + 2 = 18$. On peut systématiser le raisonnement qui nous a permis de calculer la pénalité a_{ij} associée à la décision $i \nrightarrow j$ comme suit :

$$a_{ij} = \underset{\substack{k \\ k \neq j}}{\text{Min}(c_{ik})} + \underset{\substack{h \\ h \neq i}}{\text{Min}(c_{hj})} \quad \text{relation 72}$$

Le calcul de ces pénalités est rapide. Le plus simple est de le porter, entre parenthèses, à côté des zéros du tableau de la matrice réduite (tableau 90 de droite), ce qui donne le tableau 91 de gauche. On retiendra le trajet auquel est associé la plus

TABLEAU 91
Résolution du problème de la première itération dans l'utilisation de l'algorithme de Little et al.

Étape 1 de la résolution du problème de l'itération 1

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0 (2)	3	2	13	1
	B	2	∞	2	8	0 (6)	23
	C	3	11	∞	0 (0)	4	0 (1)
	D	0 (2)	1	0 (2)	∞	7	9
	E	13	5	6	0 (2)	∞	2
	F	16	1	6	0 (1)	14	∞

Étape 2 de la résolution du problème de l'itération 1

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0	3	2	∞	1
	B	∞	∞	∞	∞	∞	∞
	C	3	11	∞	0	∞	0
	D	0	1	0	∞	∞	9
	E	13	∞	6	0	∞	2
	F	16	1	6	0	∞	∞

forte pénalité, décision intuitivement évidente, mais que l'on pourra toujours remettre en cause si, au cours des étapes ultérieures, on trouve un coût minimal supérieur à celui de la solution qu'on écarte. La plus grande pénalité trouvée est 6, on retiendra donc une tournée qui fera partir le voyageur de commerce vers la ville E, après qu'il est arrivé dans la ville B. Le coût minimal des tournées qui n'incluent pas $B \rightarrow E$ est donc $x = 16 + 6 = 22$. Nous verrons ultérieurement (dans la résolution du quatrième problème partiel au § I-2.1.4, page 398), comment modifier la matrice de coûts conformément à une décision $i \nrightarrow j$, mais pour l'instant ce travail est inutile, et seule importe l'information du coût minimum de 22 pour toutes les tournées qui n'incluent pas $B \rightarrow E$.

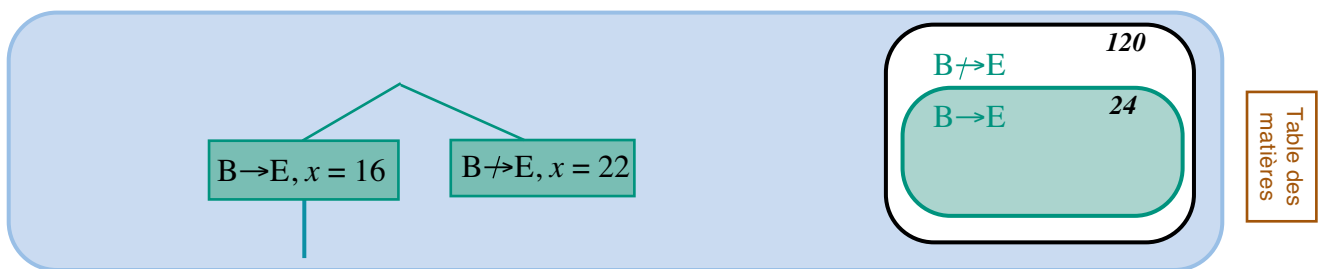
Se pose maintenant la question de savoir si le coût des tournées qui incluent $B \rightarrow E$ est toujours 16 (c'est l'étape du «bound» dans le *branch and bound* ou de l'évaluation dans le SPE). La réponse n'est pas immédiate. Pour continuer l'exploration des solutions qui incluent $B \rightarrow E$, on modifie le tableau en supprimant la ligne correspondant à B (informations de coûts relatifs au départ de B) et la colonne correspondant à E (informations de coûts relatifs à l'arrivée en E), afin de ne pas remettre en cause cette décision. L'autre solution, retenue ici dans les illustrations numériques, consiste à rendre infinis ces coûts, pour éviter de repartir de

B, ou d'arriver de nouveau en E. Ces modifications correspondent à une procédure dite d'**élimination**.

Une dernière modification s'impose : rendre infini le coût c_{EB} de transport de E vers B pour éviter au voyageur de commerce un trajet maintenant interdit. On peut dès à présent indiquer que si, ultérieurement, on sélectionne les trajets $i \rightarrow j$ et $j \rightarrow k$, c'est le coût c_{ki} qu'il faudra rendre infini pour rendre impossible tout «bouclage». Dans ces conditions, la matrice des coûts de transport supplémentaires est le **tableau 91** (étape 2). Ce tableau de coûts comporte au moins un zéro dans chaque ligne et dans chaque colonne, c'est une matrice réduite. On est donc assuré que le coût minimal des tournées qui incluent $B \rightarrow E$ est $x = 16$. On peut ajouter que, sur les $5! = 120$ tournées initialement envisageables, l'exploration ne se poursuit que sur $4! = 24$ d'entre elles (celles qui incluent la décision de se rendre directement en E après avoir quitté B) ; les $5! - 4! = 96$ tournées délaissées sont toutes celles qui font partir de B pour se rendre ailleurs qu'en E. La réduction ainsi opérée est particulièrement sensible. À ce stade de l'analyse, l'arbre des choix et la description de la décomposition de l'ensemble des solutions sont donnés à la **figure 116**.

FIGURE 116

Analyse des décisions prises à la première itération



I-2.1.2 Résolution du problème de la deuxième itération

Il faut maintenant choisir un nouveau trajet parmi ceux possibles, ce qui exclut ceux qui arrivent en E et ceux qui partent de B. Pour savoir quelle décision antagoniste $i \rightarrow j$ et $i \nrightarrow j$ on a intérêt à «brancher» sur le «nœud» $B \rightarrow E$, que l'on privilégie parce que le coût le plus faible lui est associé, on cherche de nouveau quel est le trajet $i \rightarrow j$ pour lequel le coût de transport est nul, et auquel la plus forte pénalité a_{ij} est associée. Cette procédure est dite de **séparation** (ou encore de partition), parce qu'elle remplace un problème considéré comme trop difficile à résoudre directement, par deux problèmes complémentaires. Ce second problème partiel consiste donc à explorer un sous-ensemble de 24 solutions possibles, celui qui contient toutes les tournées qui partent de B pour aller ensuite en E.

Le calcul des pénalités est porté entre parenthèses dans le **tableau 92** (itération 1). Deux trajets ont la même pénalité s'ils ne sont pas retenus : les trajets $D \rightarrow A$ et $D \rightarrow C$. Retenons arbitrairement le premier d'entre eux. Le coût minimal associé à la séquence de décisions $B \rightarrow E$ et $D \nrightarrow A$ est $16 + 3 = 19$, valeur inférieure à l'ensemble des tournées qui n'incluent pas $B \rightarrow E$. Calculons maintenant le coût minimal associé à la séquence de décision $B \rightarrow E$ et $D \rightarrow A$. Comme précédemment, il faut empêcher tout départ de la nouvelle ville de départ retenue D, ainsi que toute arrivée dans la nouvelle ville d'arrivée retenue A. Il faut en outre éviter de

repartir de A pour se rendre en D. La modification du tableau de coûts donne le **tableau 92** (itération 2), d'où l'on tire la matrice réduite du **tableau 93**.

TABLEAU 92
Résolution du problème de la deuxième itération dans l'utilisation de l'algorithme de Little et al.

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0 (2)	3	2		1
	B						
	C	3	11	∞	0 (0)		0
	D	0 (3)	1	0 (3)	∞		9
	E	13	∞	6	0 (2)		2
	F	16	1	6	0 (1)		∞

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0	3	∞	∞	1
	B	∞	∞	∞	∞	∞	∞
	C	∞	11	∞	0	∞	0
	D	∞	∞	∞	∞	∞	∞
	E	∞	∞	6	0	∞	2
	F	∞	1	6	0	∞	∞
Minimum			0	3	0		0

TABLEAU 93
Résolution du problème de la deuxième itération dans l'utilisation de l'algorithme de Little et al. (suite)

Étape 3 de la résolution du problème de l'itération 2

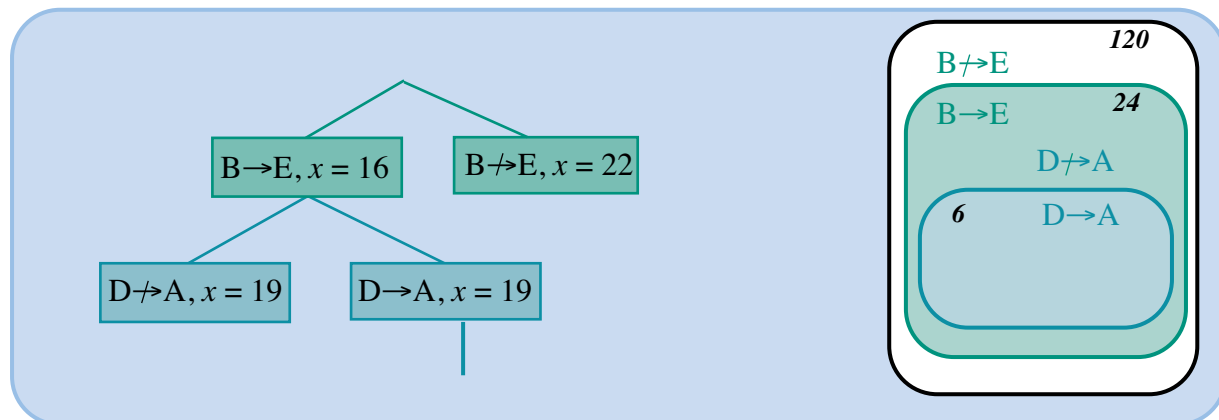
		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0	0	∞	∞	1
	B	∞	∞	∞	∞	∞	∞
	C	∞	11	∞	0	∞	0
	D	∞	∞	∞	∞	∞	∞
	E	∞	∞	3	0	∞	2
	F	∞	1	3	0	∞	∞

Le coût minimal associé à la séquence de décisions $B \rightarrow E$ et $D \rightarrow A$ est donc $16 + 3 = 19$, c'est-à-dire qu'elle a le même coût que la séquence $B \rightarrow E$ et $D \rightarrow A$. On peut donc poursuivre l'exploration à l'un ou l'autre nœud, retenons la première de ces deux séquences, qui est celle qui a sélectionné le plus de trajets. En poursuivant après le nœud $D \rightarrow A$ de notre arbre, il ne reste plus que $3! = 6$ tournées possibles sur les 24 initialement envisagées (après le nœud $B \rightarrow E$) ; $4! - 3! = 24 - 6 = 18$ tournées délaissées se retrouvent toutes après le nœud $D \rightarrow A$. Pour l'instant, on a donc effectué une partition de 120 tournées en 3 sous-ensembles : le premier comporte les 96 tournées qui évitent de partir de B pour se rendre en E, le deuxième comporte les 18 tournées qui imposent de partir de B pour se rendre en E et qui interdisent de partir de D pour se rendre en A et le dernier sous-ensemble qui incluent les tournées qui font partir le voyageur de commerce de B pour se

rendre en E et de D pour se rendre en A. L'arbre des décisions prises est celui de la figure 117.

FIGURE 117

Analyse des décisions prises à la deuxième itération



I-2.1.3 Résolution du problème de la troisième itération

L'étape 1 de la résolution de ce troisième problème consiste à calculer les pénalités, ce qui conduit au tableau 94 de gauche. La pénalité maximale de ce troisième

TABLEAU 94

Résolution du problème de la troisième itération dans l'utilisation de l'algorithme de Little et al.

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0 (1)	0 (3)	∞	∞	1
	B	∞	∞	∞	∞	∞	∞
	C	∞	11	∞	0 (0)	∞	0 (1)
	D	∞	∞	∞	∞	∞	
	E	∞	∞	3	0 (2)	∞	2
	F	∞	1	3	0 (1)	∞	∞

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	∞	∞	∞	∞	∞
	B	∞	∞	∞	∞	∞	∞
	C	∞	11	∞	∞	∞	0
	D	∞		∞	∞	∞	∞
	E	∞	∞	∞	0	∞	2
	F	∞	1	∞	0	∞	∞
Minimum			1		0		0

Table des matières

Index thématique

problème partiel est associée au trajet $A \rightarrow C$ qui accroît de 3 le coût de transport s'il n'est pas retenu, ce qui conduit dans ce cas à un coût de $19 + 3 = 22$. Pour connaître le coût minimal de la séquence $B \rightarrow E, D \rightarrow A$ et $A \rightarrow C$, il faut utiliser la procédure d'élimination de A comme ville de départ, et de C comme ville d'arrivée. Il faut en outre interdire de pouvoir aller de C à D, puisque l'on a déjà retenu la séquence $D \rightarrow A$ et que l'adjonction de $A \rightarrow C$ conduit à la tournée partielle $D \rightarrow A \rightarrow C$, on posera donc $c_{DC} = \infty$. On aboutit alors au tableau 94 de droite qui ne correspond pas à une matrice réduite, mais cette dernière s'obtient immédiatement au prix d'un accroissement du coût minimal de 1 (d'où le tableau 95).

TABLEAU 95

Résolution du problème de la *troisième itération* dans l'utilisation de l'algorithme de Little et al (suite)

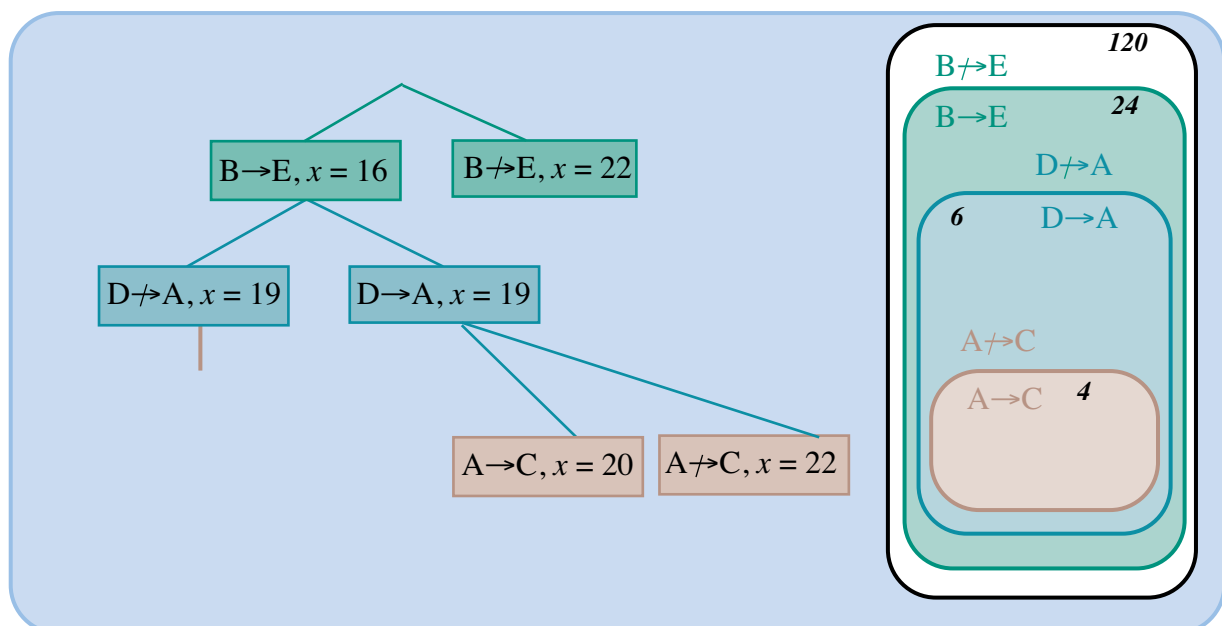
Étape 3 de la résolution du problème de l'itération 3

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	∞	∞	∞	∞	∞
	B	∞	∞	∞	∞	∞	∞
	C	∞	10	∞	∞	∞	0
	D	∞		∞	∞	∞	∞
	E	∞	∞	∞	0	∞	2
	F	∞	0	∞	0	∞	∞

Le coût minimal de la séquence décisionnelle ($B \rightarrow E, D \rightarrow A \rightarrow C$) est donc de 20. Ce coût est supérieur à celui de la séquence ($B \rightarrow E, D \nrightarrow A$ qui n'est que de 19. Il faut donc poursuivre l'exploration au-delà du nœud $D \nrightarrow A$. Remarquons que nous mettons ici en cause une décision antérieure (celle de ne pas continuer au-delà d'un nœud) parce que la borne inférieure de coût, associée à l'un des nœuds que l'on a décidé de ne pas explorer, devient plus intéressante que le coût des deux branches de la dernière alternative explorée; le terme de *bound* (= limite) de l'expression *branch and bound* désigne la plus faible des valeurs limites du critère des nœuds dont l'exploration est jugée « sans intérêt » jusqu'à preuve du contraire. On notera qu'en aval du nœud $D \nrightarrow A$, le nombre de tournées possibles est de $24 - 6 = 18$, qu'en aval du nœud (A, C) il est de $2! = 2$ tournées et qu'en aval du nœud $A \nrightarrow C$, il est de $3! - 2! = 6 - 2 = 4$ possibilités. À cette étape du raisonnement, l'arbre des décisions prises est celui donné à la figure 118.

FIGURE 118

Analyse des décisions prises à la troisième itération



I-2.1.4 Résolution du problème de la quatrième itération

Le quatrième problème partiel que l'on doit examiner maintenant doit partir d'une matrice de coûts tenant compte des décisions déjà prises, à savoir retenir (B, E) et ne pas retenir (D, A). Il faut donc modifier la matrice obtenue à la fin de la résolution du premier problème partiel (tableau 91 de la page 393 - étape 2) en rendant infini le coût pour se rendre de D à A. On obtient alors le tableau 96 de gauche, page 398, qui ne correspond pas à une matrice réduite. On procède donc à sa réduction¹ et on calcule les pénalités (tableau 96 de droite).

TABLEAU 96

Résolution du problème de la quatrième itération dans l'utilisation de l'algorithme de Little et al.

Création du tableau initial du problème de l'itération 4 et réduction de ce tableau

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0	3	2	∞	1
	B	∞	∞	∞	∞	∞	∞
	C	3	11	∞	0	∞	0
	D	∞	1	0	∞	∞	9
	E	13	∞	6	0	∞	2
	F	16	1	6	0	∞	∞
Minimum		3	0	0	0	0	0

Étape 1 de la résolution du problème de l'itération 4

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0 (2)	3	2	∞	1
	B	∞	∞	∞	∞	∞	∞
	C	0 (10)	11	∞	0 (0)	∞	0 (1)
	D	∞	1	0 (4)	∞	∞	9
	E	10	∞	6	0 (2)	∞	2
	F	13	1	6	0 (1)	∞	∞

Table des matières

La pénalité maximale est obtenue en ne retenant pas le trajet $C \rightarrow A$ ($a_{CA} = 10$), ce qui conduit à un coût minimal $x = 29$, associé aux décisions « $B \rightarrow E$, $D \nrightarrow A$, $C \nrightarrow A$ ». Pour calculer le coût associé à la sélection complémentaire de ce trajet, il faut «éliminer» les lignes C et A de cette dernière matrice et interdire le retour de A vers C. On obtient alors le tableau 97. En cherchant à le réduire, on voit immédiatement que le coût minimal associé à l'ensemble des solutions compatibles avec les décisions «(B, E), $D \nrightarrow A$, (C, A)» s'élève à $x = 19 + 1 = 20$. Ce coût est identique à celui obtenu à la fin de l'itération 3. Pour déterminer sur quel sous-ensemble de solutions il convient de poursuivre, il faut calculer le nombre de trajets retenus. En poursuivant avec le sous-ensemble le plus riche en trajets, obtenu à la fin du quatrième problème, on part d'une solution partielle comportant 2 trajets «(B, E), (C, A)». En poursuivant avec le sous-ensemble le plus riche en trajets, obtenu à la fin du troisième problème, on part d'une solution partielle comportant 3 trajets ($B \rightarrow E$, $D \rightarrow A$, $C \rightarrow A$), ce qui milite en faveur d'une exploration du sous-ensemble de 4 solutions restant à examiner. L'arbre des décisions prises est alors celui de la figure 119.

Index thématique

1. On retrouve alors l'accroissement de coût de 3, que l'on avait associé à la décision supplémentaire ($\overline{D, A}$).

FIGURE 119

Analyse des décisions prises à la quatrième itération

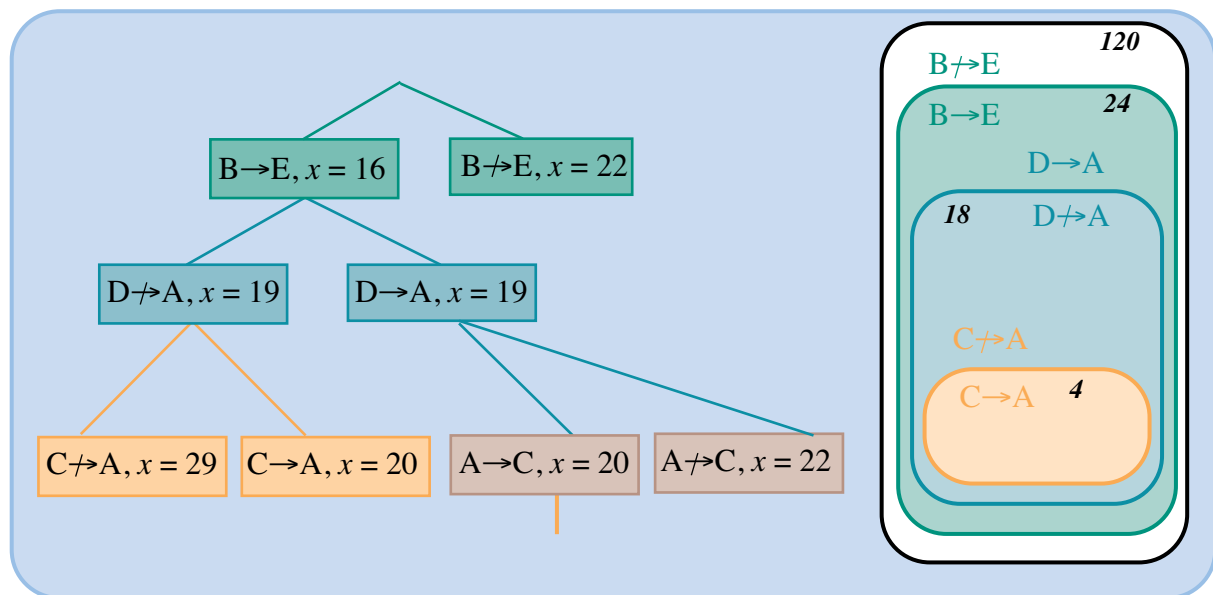


TABLEAU 97

Résolution du problème de la quatrième itération dans l'utilisation de l'algorithme de Little et al.

Étape 2 de la résolution du problème de l'itération 4

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0	∞	2	∞	1
	B	∞	∞	∞	∞	∞	∞
	C	∞	∞	∞	∞	∞	∞
	D	∞	1	0	∞	∞	9
	E	∞	∞	6	0	∞	2
	F	∞	1	6	0	∞	∞
Minimum			0	0	0		1

I-2.1.5 Résolution du problème de la cinquième itération

Le cinquième problème partiel s'étudie à partir de la matrice réduite obtenue à la fin de la résolution du problème partiel de la troisième itération (tableau 95 de la page 397). Le calcul des pénalités est effectué dans le tableau 98 de la page 400 (étape 1). La pénalité la plus forte (12) est associée au trajet $C \rightarrow F$. La procédure d'élimination conduit non seulement à éliminer la ligne C et la colonne F du tableau, mais également à rendre infini le coût c_{FD} pour rendre impossible le retour de F à D (la solution retenue comportant la tournée partielle $D \rightarrow A \rightarrow C \rightarrow F$). On obtient alors la matrice réduite du tableau 98 (étape 2).

TABLEAU 98

Résolution du problème de la cinquième itération dans l'utilisation de l'algorithme de Little et al.

Étape 1 de la résolution du problème de l'itération 5

		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	∞	∞	∞	∞	∞
	B	∞	∞	∞	∞	∞	∞
	C	∞	10	∞	∞	∞	0 (12)
	D	∞		∞	∞	∞	∞
	E	∞	∞	∞	0 (2)	∞	2
	F	∞	0 (10)	∞	0 (0)	∞	∞

Étape 2 de la résolution du problème de l'itération 5

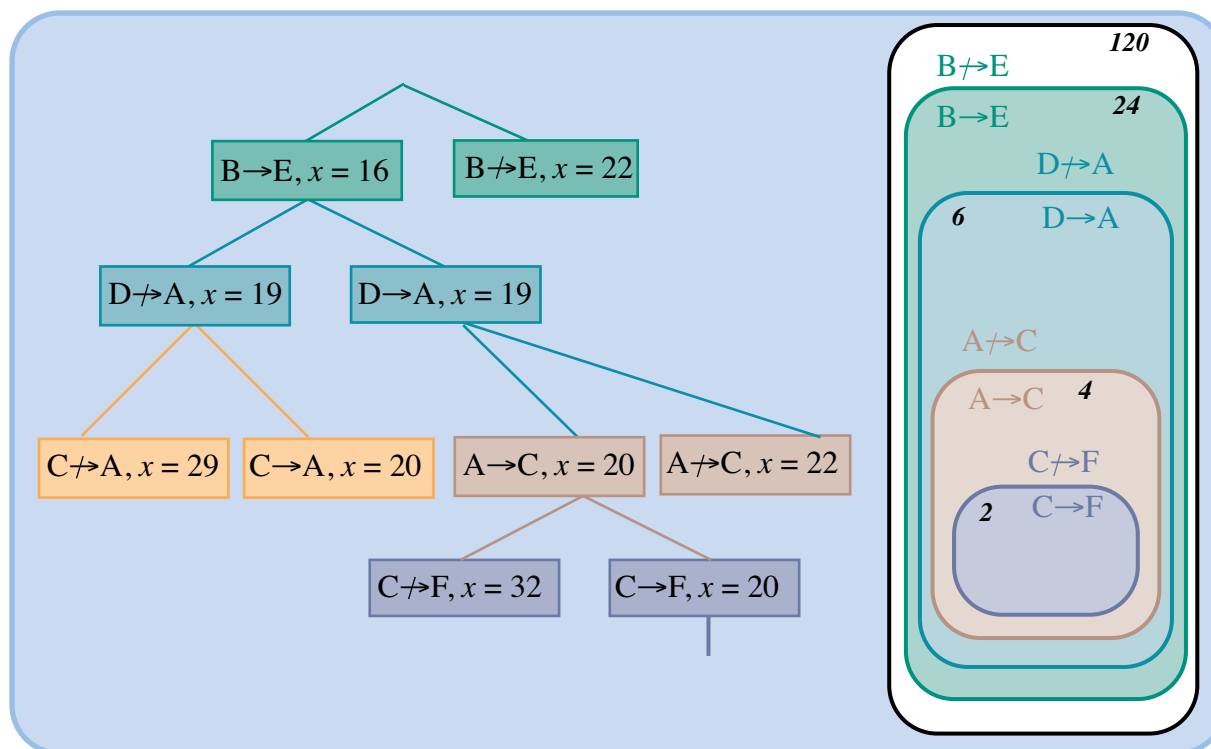
		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	∞	∞	∞	∞	∞
	B	∞	∞	∞	∞	∞	∞
	C	∞	∞	∞	∞	∞	∞
	D	∞		∞	∞	∞	∞
	E	∞	∞	∞	0	∞	∞
	F	∞	0	∞	∞	∞	∞
Minimum			0		0		

I-2.1.6 Résolution du problème de la sixième itération

La matrice de coûts étant réduite, le coût de 20 reste toujours valable et on peut alors sélectionner les deux derniers trajets possibles E→D et F→B. L'arbre des décisions prises est alors celui de la figure 120.

FIGURE 120

Analyse des décisions prises à la cinquième itération



La tournée optimale pour notre voyageur de commerce est donc : F→B→E→D→A→C→F, pour un coût minimum de 20, comme on peut le vérifier

directement sur le tableau de coûts initial $5 + 1 + 2 + 2 + 7 + 3$. L'ensemble des décisions étudiées est représenté à la figure 121.

FIGURE 121

Analyse des décisions prises à la sixième itération

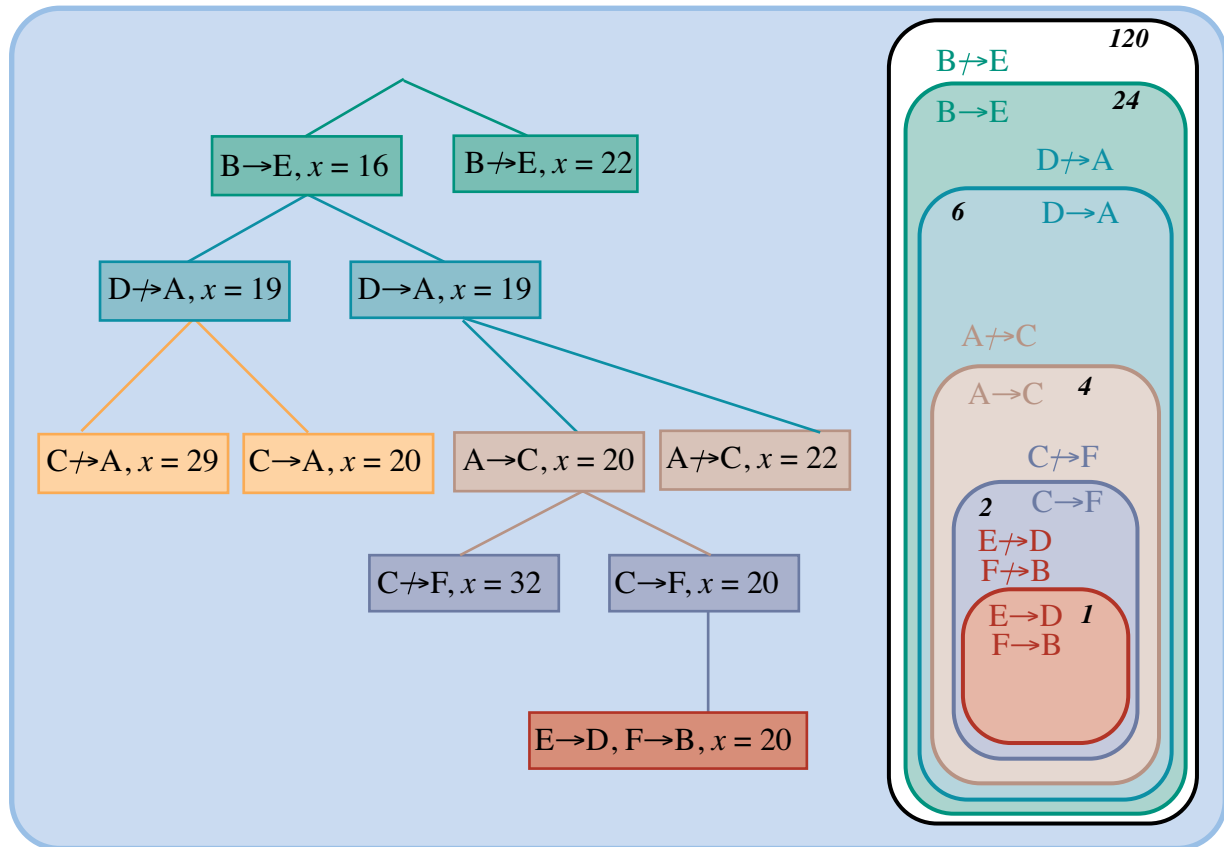


Table des matières

Index thématique

I-2.2 Remarques complémentaires

Quatre remarques complémentaires méritent d'être faites. Elles reposent toutes sur l'analyse d'un problème de tournée, mais les trois premières restent valables pour le problème d'ordonnancement de tâches avec prise en compte du temps de lancement dépendant de l'ordre de passage des tâches.

I-2.2.1 Détermination empirique de la tournée

On a testé¹ la performance du comportement empirique observé chez le voyageur de commerce et qui consiste à choisir comme prochaine étape la ville la plus proche. On obtient alors des solutions souvent acceptables, c'est-à-dire suffisamment voisines de la solution optimale, à condition de systématiser le raisonnement, c'est-à-dire en essayant comme ville de départ toutes les villes de la tournée et en retenant la tournée ayant le coût le plus faible. L'application de ce principe donne ici une solution de 10 % plus coûteuse que la solution optimale. Cette procédure donne de meilleurs résultats (25 % de plus en moyenne, par rapport au coût minimum) si la distribution des coûts de transport est suffisamment voisine d'une

1. Travaux de Gavett, cités par Conway, Maxwell & Miller (1967, [104], p. 66-68). Cet algorithme est connu sous le nom de *Closest Unvisited-city Algorithm*.

distribution normale, que dans le cas d'une distribution proche d'une distribution uniforme (40 % de plus en moyenne).

I-2.2.2 Détermination optimale de tournées multiples

On peut également traiter à l'aide de l'algorithme de Little le problème de la détermination des tournées optimales de m voyageurs de commerce partant tous de la même ville de départ, chaque ville n'étant visitée que par un seul voyageur de commerce et toutes les villes étant visitées. Il suffit¹ de créer $m - 1$ villes supplémentaires, toutes représentatives de la ville de départ et de dupliquer $m - 1$ fois la ligne et la colonne associée à la ville de départ; les distances entre les m villes représentatives de la ville de départ sont considérées comme infinies pour empêcher de «passer d'une tournée à l'autre». On verra au § II-1.2, page 911, du chapitre XIII, une formulation plus générale par la programmation mathématique qui présente un avantage sur la solution proposée ici si l'on souhaite tenir compte de contraintes (capacité, fenêtre de temps) dans l'organisation de ces tournées.

I-2.2.3 Problème stochastique du voyageur de commerce

Dans la réalité, le problème du voyageur de commerce se pose souvent plus en termes de temps que de coûts. La formulation retenue ici repose sur des valeurs moyennes pour pouvoir traiter ce problème en univers certain; or il est évident que le temps de transport est aléatoire. On examinera succinctement ce problème maintenant bien que ce développement se situe dans une section consacrée à l'analyse de problèmes de type «statique-certain».

Si l'on peut définir les distributions de probabilité de ces temps de transport, il est possible de résoudre le problème stochastique du voyageur de commerce en s'appuyant sur l'utilisation conjointe de la méthode de Monte-Carlo et des algorithmes génétiques, selon la démarche décrite en détail au § II-3.4, page 196, du chapitre III. De surcroît:

- la fonction de coût à optimiser peut être plus complexe qu'une simple fonction linéaire du temps mais on peut aussi rester sur un critère minimisation de l'espérance mathématique de la tournée ou sur un critère de minimisation de la probabilité qu'une tournée s'effectue en plus de x heures ou...
- on peut imposer des contraintes supplémentaires du type «le temps de transport de la tournée ne doit pas dépasser x heures» ou «l'écart-type du temps de transport de la tournée ne doit pas dépasser 30 minutes» ou...

Illustrons cette approche², à partir des données du tableau 99 qui reprennent le mode de celles du tableau 89 de la page 391 pour faciliter la comparaison des résultats. La distribution triangulaire a été retenue.

La tournée optimale trouvée est: $F \rightarrow C \rightarrow A \rightarrow D \rightarrow E \rightarrow B \rightarrow F$, pour un coût minimum de 21,43³ (espérance mathématique⁴, écart-type du coût = 1,28); la

1. Voir Christofides (1976, [94]) et Watson et Foulds (1981, [437]).

2. L'exemple proposé ici s'appuie sur l'add-in *Risk Optimizer*® de Palisade, utilisable sous Excel®, en complément de @Risk (autre add-in de Palisade); plusieurs algorithmes génétiques sont disponibles, dont un spécialisé dans les problèmes de séquençement.

3. Cette espérance mathématique est supérieure à la valeur obtenue en univers certain sur la base des modes (qui, d'une manière générale, dans le jeu de données utilisées, sont inférieures aux espérances mathématiques des coûts de transports).

TABLEAU 99
Paramètres (Minimum – Mode – Maximum) des distributions de probabilités triangulaires des coûts de la tournée du voyageur de commerce

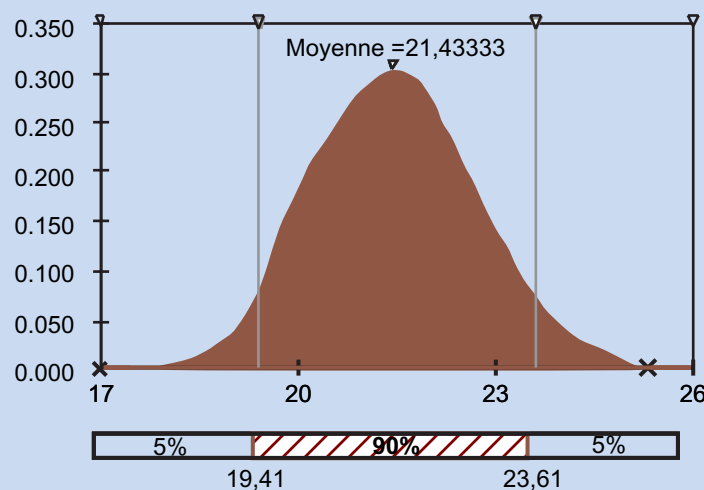
		Ville d'arrivée					
		A	B	C	D	E	F
ville de départ	A	∞	0,9 – 1 – 1,5	6 – 7 – 9	2,5 – 3 – 4,4	12 – 14 – 18	1,3 – 2 – 4
	B	1 – 3 – 4,5	∞	5 – 6 – 8	5 – 9 – 20	0,6 – 1 – 1,5	20 – 24 – 30
	C	5,5 – 6 – 8	12 – 14 – 19	∞	2,2 – 3 – 3,9	5,5 – 7 – 10,6	2 – 3 – 5
	D	1,5 – 2 – 3	2 – 3 – 4,7	3,5 – 5 – 7,4	∞	7 – 9 – 12	10 – 11 – 15
	E	12 – 15 – 20	5 – 7 – 10	10 – 11 – 15	1 – 2 – 4	∞	3 – 4 – 6
	F	13 – 20 – 30	4 – 5 – 6,7	9 – 13 – 18	2,8 – 4 – 5,3	12 – 18 – 20	∞

distribution de probabilité de cette tournée est donnée à la [figure 122](#), avec les éléments d'analyse de sensibilité (voir les explications sur la méthodologie utilisée, au [chapitre III](#), [page 193](#)). Pour illustrer la puissance de cette approche,

FIGURE 122

Distribution de probabilités du coût de la tournée optimale et analyse de sensibilité

Distribution de probabilités du coût de la tournée optimale



on peut ajouter que :

- la tournée qui minimise l'écart-type du coût de la tournée est toujours $F \rightarrow C \rightarrow A \rightarrow D \rightarrow E \rightarrow B \rightarrow F$; c'est aussi celle qui minimise le coût qui n'a que 5 % de chances d'être dépassé ;
- si la solution trouvée n'avait pas été dominante sur les différents critères d'optimisation utilisés, on aurait pu chercher à résoudre un problème du type «trouver la tournée qui minimise le coût et possède un fractile à 95 % inférieur à x ».

I-2.2.4 Complexité des problèmes concrets

La **complexité des problèmes concrets de transport** est habituellement plus grande que ne le laisse supposer cet exemple. Dans la pratique on dispose d'une flotte hétérogène de camions ment, le problème est alors non seulement celui de tournées optimales mais également celui (non indépendant du précédent) d'une affectation optimale de camions aux ponts de desserte, avec prise en compte du passage chez des clients sur des fenêtres de temps précises et, dans le cas de tournées multiples s'accompagnant de livraison, avec prise en compte de contraintes de capacité de transport. La complexité du problème est telle qu'il est souvent préférable d'utiliser, en conversationnel, des systèmes d'aide à la décision s'appuyant sur des algorithmes optimaux pour résoudre des problèmes simplifiés, ce que l'on verra à la [section II, page 904](#), du [chapitre XIII](#).

I-3 Tentative de caractérisation de l'approche statique

Plusieurs milliers d'articles ont été écrits depuis cinquante ans dans ce domaine, ce qui est sans doute beaucoup par rapport aux retombées opérationnelles qu'ils ont suscitées. Pour comprendre pourquoi il en est ainsi et être en mesure d'extraire de cette masse de travaux les éléments susceptibles d'aider à la résolution de problèmes concrets, il est nécessaire de présenter une grille d'analyse de cette approche statique¹. Celle-ci se fonde sur l'utilisation de modèles d'optimisation plus ou moins représentatifs de tout ou partie de situations concrètes. Cette simplification de la réalité s'impose dans le cadre d'une démarche déductive susceptible de résoudre des problèmes d'une complexité croissante. Une typologie de cette classe de travaux peut être faite sur la triple base du critère d'optimisation (§ I-3.1), de la liste des hypothèses du modèle (§ I-3.2, [page 405](#)) et de la méthode de résolution (§ I-3.3, [page 406](#)).

I-3.1 Critère d'optimisation

Celui-ci est presque toujours de nature organisationnelle, c'est-à-dire que la performance de l'ordonnancement n'est pas jugée sur la base de ses conséquences économiques (et peut du reste difficilement l'être si l'on s'attache au très court terme), mais sur celle d'un indicateur physique. Son unité est habituellement le temps (date moyenne d'achèvement des tâches, retard moyen, retard maximum...) ou la tâche (nombre de tâches terminées en retard...) et il est censé être en relation avec la performance économique de l'ordonnancement dans la mesure où il varie dans le même sens (ou en sens inverse, selon l'indicateur retenu) que l'un des coûts partiels du système productif, qu'il soit lié à la saturation du potentiel productif ou à l'insatisfaction de la demande.

Le choix de critère organisationnel se justifie par le fait que l'utilisation d'indicateurs économiques dans le cadre d'un ordonnancement sur une période unique ne pose pas de problèmes de formulation mathématique, mais des problèmes de définition de coefficients de valorisation, nécessairement arbitraire sur un tel horizon. Cela étant, l'utilisation de critère organisationnel connaît d'autres

1. Voir l'analyse faite par Riou-Long (1986, [364]). Une typologie généralement utilisée est celle proposée par Rinnooy Kan (1976, [361]) qui caractérise les problèmes d'ordonnancement par quatre critères: le nombre de tâches, le nombre de centres de production, le critère d'évaluation de l'ordonnancement et une liste d'hypothèses caractérisant le problème posé.

limites, car il est évident que ces coûts partiels auxquels ils sont liés peuvent varier en sens inverse et qu'en conséquence, les conclusions auxquelles on parvient doivent être évaluées avec prudence.

I-3.2 Liste des hypothèses décrivant le système productif

Ces hypothèses, souvent implicites dans la présentation d'un modèle statique, décrivent le système productif et les tâches à exécuter.

Le système productif se caractérise par un certain nombre de centres de production.

- Chaque centre de production est composé d'un ou plusieurs postes de travail. Dans ce dernier cas, ces postes sont identiques ou non identiques (en particulier au niveau de la rapidité d'exécution d'une opération).
- L'exécution d'une opération sur un poste de travail peut nécessiter des réglages préalables de la machine et, dans ce cas, le temps de réglage peut ou non dépendre du réglage antérieur.
- L'opération (ou une séquence d'opérations) peut ou non être exécutée sur des centres de production de nature différente (centre d'usinage contre séquences de machines classiques spécialisées), ce qui revient à prendre en compte des gammes alternatives.
- Un poste de travail peut exécuter une seule opération à la fois, mais dans certains cas, une certaine simultanéité peut être autorisée (par exemple, traitement thermique ou chimique), pour des opérations débutant en même temps ou non.
- Les temps de transport d'un centre de production à un autre centre de production sont considérés comme nuls, ou sont inclus dans le temps de lancement, ou sont isolés (en particulier si le modèle tient compte de l'existence d'un centre de production «manutention»).
- Il est possible ou non (contrainte de stockage ou contrainte technique) d'avoir des tâches en attente devant un poste de travail occupé.
- Un ouvrier (ou un groupe d'ouvriers) n'est affecté ou non qu'à un seul centre de production. Dans ce dernier cas l'ordonnancement est nettement plus complexe parce qu'il porte à la fois sur des machines et sur des hommes (lesquels constituent souvent la ressource rare en matière d'ordonnancement).

En ce qui concerne les tâches :

- Les tâches à exécuter peuvent être ou non simultanément disponibles en début de période.
- Une tâche peut correspondre à la fabrication d'un article unique ou à celle d'un lot d'articles identiques.
- Les temps opératoires sont certains.
- L'ordre d'exécution des opérations d'une tâche est intangible et une opération ne peut débuter avant le début des opérations précédentes (ce qui exclut le cas de gammes arborescentes), sauf spécification contraire¹.
- Une opération peut ou non débuter avant la fin de l'opération précédente (recouvrement portant sur les temps de réglage et rendu possible du fait d'une libération antérieure de la machine à régler et/ou recouvrement lié au

fractionnement de lots définissant une tâche). On retrouve là des implications de la mise en œuvre de démarche SMED¹.

- Des dates-limites d'achèvement des tâches peuvent ou non être introduites.
- Les tâches sont indépendantes (et donc exécutables dans un ordre quelconque) ou non.
- L'exécution d'une opération peut ou non être suspendue sur un poste de travail au profit d'une opération d'une autre tâche (préemption).

Il faut souligner qu'en règle générale les modèles retenus habituellement se définissent par une combinaison des hypothèses les plus restrictives, quelques-unes d'entre elles seulement étant levées. Rappelons enfin que l'on raisonne sur une période unique et donc en supposant implicitement que la mise en œuvre de la solution trouvée reste possible (possibilité d'interrompre en fin de journée une opération en cours et de la reprendre le lendemain au même point).

I-3.3 Méthodes de résolution

Tout d'abord, pour quelques problèmes simples (cheminement unique et deux centres de production, par exemple) et certains critères (minimisation de la date d'achèvement de toutes les tâches, par exemple), il est possible de démontrer analytiquement qu'un ordonnancement suivant la valeur croissante d'un indicateur, ce qui a été abondamment illustré dans les pages qui précèdent.

Pour le problème général, même s'il est restreint au cas du cheminement unique, on n'a pas trouvé de règles simples permettant d'obtenir un ordonnancement optimal. On peut cependant réduire la dimension combinatoire du problème posé et démontrer que si le critère que l'on cherche à minimiser est un **critère régulier**, c'est-à-dire si sa valeur n'augmente que si la date d'achèvement de l'une des tâches s'accroît, l'optimum ne peut se trouver que dans un sous-ensemble de l'ensemble des ordonnancements possibles *a priori*². Cette limitation de l'énumération peut être utilisée pour résoudre le problème par une technique de type *branch and bound*³, présentée au § I-2.1, page 390, mais pour des structures simplifiées du problème et certains critères, on peut encore réduire le champ des possibles en mettant en évidence que le respect de certaines propriétés est une condition nécessaire à l'obtention d'un optimum, ce qui accroît encore l'efficacité potentielle de l'approche de type *branch and bound*.

Néanmoins, le plus souvent, le problème combinatoire posé reste redoutable, même dans le cas simplifié de flux unidirectionnels. C'est pourquoi, dans le cadre de problèmes simplifiés (et plus particulièrement dans le cas du cheminement unique⁴) des heuristiques ont été proposées et testées : elles permettent une

1. Note de la page précédente. On parle dans ce cas de stocks multi-échelons ou encore de stock à étages, mais les travaux de recherche s'intéressant à ce cas de figure sont largement déconnectés de ceux ayant trait à l'ordonnancement en ateliers spécialisés (en particulier au niveau des critères). Voir Williams (1981, [443]).

1. Voir le chapitre VII, page 511.

2. Voir Baker (1974, [30], p. 13 et p. 181) et Esquirol et Lopez (1999, [140], chapitre II) sur la typologie des objectifs et critères d'évaluation. Voir également Lopez et Roubellat (2001, [286], chapitre V) pour une présentation complète sur les problèmes de satisfaction de contraintes et de propagation de contraintes.

3. Voir Baker (1974, [30], p. 55-65, p. 148-156).

4. Voir les synthèses de Gupta (1979, [211]), Dannenbring (1977, [118]) et celle de King et Spachis (1980; [265]).

économie considérable de calculs et ne garantissent pas l'obtention d'une solution optimale mais d'une solution dont les performances sont bonnes en général.

Nous avons indiqué au § I-1.4.2, page 378, que le problème général a été formalisé en termes de programmation dynamique et en termes de programme linéaire en nombres entiers et que s'il est montré que cette dernière approche est la plus performante, il n'en reste pas moins qu'aujourd'hui, elle reste difficilement exploitable au-delà de quelques dizaines de tâches ou de postes de travail par des méthodes exactes (comme celle du *branch and bound*). Pour trouver des solutions satisfaisantes à des problèmes combinatoires d'une certaine dimension, on utilise de nos jours des **méta-heuristiques**¹ qui sont des démarches générales de résolution de ces classes de problèmes permettant d'obtenir assez rapidement des solutions sans pouvoir toutefois en garantir l'optimalité. On présentera les trois principales d'entre elles : la méthode Tabou, le recuit simulé et les algorithmes génétiques (§ I-3.3.2), cette dernière approche étant introduite de manière plus complète (§ I-3.3.2, page 408). Ces approches sont assez performantes pour la résolution de problèmes d'ordonnancement mais elles nécessitent encore une expertise faiblement répandue.

I-3.3.1 La méthode Tabou et le Recuit Simulé

Ces deux méthodes s'appuient sur des algorithmes itératifs qui explorent l'espace des solutions en se déplaçant pas à pas d'une solution à une autre. Le risque que l'on court en explorant les solutions proches est celui de se trouver au voisinage d'un optimum local. Pour contrer ce risque, on a donc développé en particulier deux méthodes efficaces qui acceptent des solutions voisines moins bonnes.

- La **méthode du Recuit Simulé** s'appuie sur des travaux amorcés dans les années cinquante pour simuler l'évolution d'un système physique instable vers un état d'équilibre thermique à une température θ fixée. Au cours d'une itération donnée, on passe à l'une des solutions admissibles voisines de la solution courante, qui est acceptée si elle est meilleure et, dans le cas contraire, avec une certaine probabilité qui est fonction de l'importance de la détérioration constatée dans le critère à optimiser et d'un paramètre θ . Le changement de ce paramètre (analogie avec la température) s'effectue en général chaque fois qu'un nombre donné d'itérations a été effectué et l'algorithme s'arrête lorsqu'aucune solution meilleure n'a été trouvée au cours d'un cycle d'itération. La performance de cet algorithme est étroitement liée au schéma de refroidissement retenu (mécanisme de changement de θ). Un exemple d'utilisation très performante de cette approche pour résoudre le problème de l'ordonnancement d'un projet sous contrainte de ressources non stockables et existence de gammes alternatives d'exécution d'une tâche peut être trouvé dans Boctor (1996, [57]).
- La **méthode Tabou** remonte aux années quatre-vingt. Pour éviter de risquer de s'enfermer dans un minimum local, lorsque celui-ci est atteint (pas de solution admissible voisine meilleure), cette méthode garde en mémoire les dernières solutions visitées et interdit le retour arrière vers celles-ci avant un

1. Pour en savoir plus, le lecteur est invité à lire le texte écrit par Widmer, Hertz et Costa dans l'ouvrage coordonné par Lopez et Roubellat (2001, [286], chap. III).

nombre fixé d'itérations, ce qui permet alors de repartir en arrière pour explorer une solution moins bonne mais permettant peut-être de s'acheminer vers l'optimum. Dans la mesure où le passage de la solution courante à la solution nouvelle peut être entaché d'arbitraire, plusieurs candidats ayant la même performance, l'interdit de retour arrière peut être levé plus tôt sous certaines conditions (définies par une «fonction d'aspiration»).

I-3.3.2 Les algorithmes génétiques

Les **algorithmes génétiques**¹, imaginés par des biologistes dans les années cinquante pour simuler l'évolution des espèces, cherchent une solution optimale en simulant l'évolution naturelle d'une population d'individus, génération après génération, en reproduisant des mécanismes d'hérédité et de sélection, en partant du principe que ce sont les individus les plus forts qui tendent à survivre et à donner une descendance, les enfants héritant stochastiquement des caractéristiques génétiques (bonnes ou mauvaises) de leurs parents.

Dans cette approche, le génome d'un individu correspond à une solution du problème d'optimisation posé² et la **force** (*fitness*) de cet individu est mesurée par la valeur du critère à optimiser prise par cette solution (ou une valeur dérivée). L'algorithme génétique deux mécanismes de base :

- Le **mécanisme de croisement** (*cross-over*) apparie deux individus de la population choisis aléatoirement avec une probabilité proportionnelle à leurs forces respectives (ce qui laisse à aux individus faibles des chances d'être choisis), pour donner naissance à un nouvel individu dont le génome est constitué à partir du génome de chaque parent. Ce mécanisme de croisement est conçu de telle sorte qu'il ait des chances d'augmenter la force de l'enfant³, sachant qu'en régime de croisière, on remplace un couple de parents par couple d'enfants de caractéristiques nouvelles (ou, avec une probabilité fixée, par un couple d'enfants identiques aux parents). Ce mécanisme de croisement est généralement mis en œuvre sur la moitié de la population (tirage aléatoire sans remise) avant de passer au mécanisme suivant.
- Le **mécanisme de mutation** est destiné à éviter le risque de dégénérescence, en introduisant de la variété. Chaque individu est passé en revue et subit ou non avec une probabilité fixée à l'avance, une mutation altérant son génome⁴.

1. Pour en savoir plus, le lecteur est invité à lire le texte écrit par Portmann et Vignier dans l'ouvrage coordonné par Lopez et Roubellat (2001, [286], chap. IV), qui illustre la démarche sur un exemple détaillé d'ordonnement.

2. Par exemple, le séquençement, pour un problème de voyageur de commerce devant explorer un ensemble de n villes, sans revenir à la ville de départ, pour reprendre une adaptation d'un problème déjà traité (avec l'algorithme de Little) et qui sera étudié plus en détail au [chapitre XIII](#), § II-1.3, page 918.

3. Dans l'exemple retenu, on peut imaginer de retenir les k premières villes du séquençement (génome) du père et les $n - k$ dernières villes du séquençement (génome) de la mère, le nombre k étant choisi aléatoirement entre 2 et $n - 2$. Ce mécanisme de création d'une nouvelle solution est connu sous le nom d'opérateur à un point mais d'autres mécanismes existent (le choix du mécanisme étant un point central de la démarche puisqu'il conditionne la possibilité d'amélioration de la solution). La concaténation de ces parties de séquençements a de fortes chances de ne pas comporter toutes les villes et donc d'inclure des doublons (autant que de villes manquantes); on élimine alors une ville de chacun des doublons et on complète la séquence avec les villes manquantes à la césure ou en fin de chaîne. Comme ce mécanisme peut donner naissance à plusieurs enfants différents (en raison de la substitution possible du rôle du père et de la mère mais aussi des mécanismes de recombinaison de la séquence en cas de doublon), on peut choisir de ne garder que les deux enfants ayant les deux meilleurs génomes.

Ces deux mécanismes sont utilisés successivement sur un certain nombre d'itérations fixé à l'avance ou par un test d'arrêt (temps de calcul, stabilité de la meilleure force trouvée depuis un nombre prédéterminé d'opération...). L'un des problèmes à résoudre initialement est celui de la constitution d'une population initiale et donc celui de la recherche de solutions possibles.

Cette approche qui est utilisée depuis plusieurs années avec succès repose sur la pertinence des opérateurs de croisement et de mutation utilisés. Les logiciels commerciaux disponibles (comme celui utilisé dans ce chapitre et le [chapitre III](#)) proposent normalement de rattacher le problème à traiter à une classe de problèmes pour lesquels des opérateurs s'avèrent habituellement efficaces. Il n'en reste pas moins que cette efficacité peut être assez limitée si le problème posé s'éloigne des problèmes type de la classe de problèmes auxquels il est rattaché.

SECTION II L'APPROCHE ALÉATOIRE DYNAMIQUE

Cette approche se caractérise par le double refus de se placer dans un cadre déterministe et de travailler sur un ensemble fini de tâches à exécuter au cours d'une période déterminée. Si le problème posé est décrit par un ensemble de variables aléatoires de caractéristiques stables, le comportement du système étudié est induit par un ensemble de règles de décisions définissant une politique d'ordonnancement administrée systématiquement. Son évaluation repose sur une (ou plusieurs) variable(s) d'état caractérisant le « régime de croisière », ayant le statut d'une espérance mathématique. Deux démarches de résolution de ces problèmes sont alors envisageables : l'utilisation de la théorie des files d'attente (§ II-1) et la simulation (§ II-2, page 410).

Table des
matières

II-1 L'approche par la théorie des files d'attente

Index
thématique

D'une manière générale, dans cette approche, les arrivées des tâches dans le système productif ne sont pas simultanées, mais espacées dans le temps. On définit une distribution de probabilités de l'intervalle de temps séparant deux arrivées successives, cette distribution étant immuable au cours du temps. Le système productif est généralement décrit par un ou plusieurs postes de travail, fonctionnant en parallèle ou en série ; les durées d'exécution d'une opération sont spécifiées de façon certaine ou en probabilité (on parle de **loi de service**) et ce pour chaque poste de travail. Le bon fonctionnement du système est jugé à travers un ou plusieurs critères, comme dans le cas statique. Plusieurs règles d'ordonnancement (dans ce type de « littérature » on parle plutôt de **discipline de file d'attente**) sont possibles. L'objet de cette approche est de fournir des résultats analytiques des valeurs prises par les variables d'état choisies pour évaluer le fonctionnement d'un système géré par une certaine discipline de file d'attente. Ces résultats analytiques sont des espérances mathématiques calculées sur une durée de fonctionnement infinie ; ils caractérisent ce que l'on appelle le **régime stationnaire**.

Cette approche nécessiterait de larges développements pour être abordée correctement¹, ce qui mènerait nettement au-delà de l'objectif de sensibilisation à

4. Dans notre exemple, un opérateur de mutation possible consiste à choisir aléatoirement deux villes consécutives du séquençement associé au génome et à les permuter.

cette problématique poursuivi ici. Une seconde raison, plus péremptoire, conforte ce choix : les résultats obtenus dans cette approche ne sont pas d'une portée très générale. En effet, si l'analyse est «relativement» facile dans le cas d'un seul centre de production, ou de plusieurs centres de production mis en parallèle et fournissant la même prestation (caisses d'un supermarché par exemple), il n'en est plus de même lorsque les centres de production sont en série, et à plus forte raison lorsqu'ils forment un réseau quelconque et que des limitations sont introduites dans la longueur des files d'attente ou les durées maximales d'attente admissibles. Dans ces conditions un problème classique de centres de production spécialisés, se caractérisant par des cheminements entre centres de production, variables selon les produits à fabriquer, n'a aucune chance de trouver de solution analytique, du moins dans l'état actuel des connaissances.

II-2 L'approche simulateur

Lorsqu'il est difficile de trouver la solution analytique d'un problème probabiliste, bien défini sur le plan formel, l'utilisation des méthodes de Monte-Carlo rend les mêmes services que cette solution analytique, c'est-à-dire permet d'évaluer les conséquences des règles de fonctionnement du système étudié. Cette approche a déjà été décrite en détail au [chapitre III, § I-2, page 147](#), et utilisée pour simuler le fonctionnement de systèmes réels. Ici, les distributions de probabilités des différentes variables d'état, retenues pour caractériser la qualité de fonctionnement du système, sont empiriquement obtenues après de nombreuses simulations au lieu d'être calculées analytiquement. En outre, les paramètres de tendance centrale et de dispersion sont calculés numériquement à partir des données de la simulation au lieu de résulter d'un calcul en application d'une formule analytique. Il ne faut donc pas s'étonner que l'extension des facilités informatiques ait entraîné depuis les années soixante, des utilisations croissantes de ce type d'approche. Les travaux se sont orientés dans deux directions : la simulation de systèmes réels (§ II-2.1) et celle de systèmes fictifs (§ II-2.2, page 411).

Table des
matièresIndex
thématique

II-2.1 La simulation de systèmes réels

La première approche vise à trouver la solution d'un problème complexe réel, c'est-à-dire existant physiquement. La recherche de règles de décision s'effectue, après modélisation du système réel :

- soit une fois pour toutes, pour définir des tables de décisions dont la pertinence sera remise en cause lorsque les données structurelles du problème (caractéristiques des flux et/ou des ressources) auront significativement changé ; ces tables de décisions peuvent, le cas échéant, se ramener à une règle simple, comme une formule de calcul de la priorité d'une tâche arrivant dans le système, compte tenu de l'état du système à cet instant (état caractérisé par différents paramètres) ;
- soit périodiquement pour proposer un ordonnancement destiné à résoudre un problème réel, la simulation étant reprise à chaque modification jugée significative du problème (arrivée de nouvelles commandes, modification de la

1. Voir Conway, Maxwell & Miller (1967, [104]), p. 141 à 218, pour un exposé des principaux résultats connus ; voir également l'exposé très didactique de cette problématique des files d'attente dans Wagner (1975, [435]), p. 851 à 901 et 977 à 994. Une présentation des résultats récents peut être trouvée dans Kelly, [259].

disponibilité de certaines ressources, problème d'approvisionnement, déviation significative des réalisations par rapport aux prévisions).

La simulation des systèmes réels a longtemps été d'un coût prohibitif (dû à la puissance de calcul et au volume de données important à manipuler). L'évolution informatique permet maintenant une généralisation de ce type d'approche, à des coûts acceptables¹, pour une mise en œuvre aisée aboutissant à des résultats facilement exploitables. La simulation, se substituant à l'approche de type «essai – erreur», permet d'anticiper des problèmes au lieu de les subir. D'un point de vue opérationnel, l'amélioration des procédures permet une meilleure utilisation du système productif qui diminue les coûts et recule les limites de capacité productive. D'un point de vue stratégique, l'usage de simulations met en évidence l'existence de «goulots d'étranglement» et facilite l'étude de l'amélioration du système productif par adjonctions ou transformations de ressources productives mais aussi par modification des procédures. Cela étant, les conclusions que l'on peut tirer de ce type d'expérimentation n'ont pas forcément de portée générale, la prédominance de certaines règles pouvant être liée aux caractéristiques particulières du système productif étudié. Tel n'est pas le cas de la seconde utilisation de la simulation que nous allons examiner (§ III-2.2), parce qu'elle fournit des enseignements utiles pour aborder la simulation de systèmes réels, sur laquelle, nous reviendrons au § III, page 416.

II-2.2 La simulation de systèmes fictifs

Table des
matières

Index
thématique

Cette approche poursuit d'une certaine façon la démarche utilisée dans les modèles statiques étudiés au début de ce chapitre, à ceci près que la résolution numérique (par des méthodes simulatoires) se substitue à la résolution analytique. Les conditions de fonctionnement du système sont strictement définies : nombre de centres de production utilisés par une tâche, contraintes techniques de passage entre ces centres (ordre de passage pouvant être généré aléatoirement), lois des arrivées des tâches (loi de Poisson le plus souvent), lois des durées d'exécution d'opérations dans un centre de production (durée constante ou loi exponentielle négative, par exemple), temps de transferts entre centres de production (souvent considérés comme nuls), spécifications techniques ayant une incidence sur l'ordonnancement (par exemple, absence de chevauchement entre tâches), fonctionnement perpétuel du système (pas de panne, pas de fractionnement du temps en journées...) ou autre modalité de fonctionnement, etc.

Les investigations ont été conduites dans deux directions : l'ordonnancement dans un ensemble de centres de production indépendants (§ II-2.2.1) ou non (§ II-2.2.2, page 416).

II-2.2.1 Le cas des ateliers spécialisés indépendants

Les travaux particulièrement importants de Conway, Maxwell et Miller², s'appuient sur un même jeu de 8 700 tâches devant être exécutées dans un système fictif de 9 centres de production qui a été utilisé pour tester 25 types différents de

1. On trouve plusieurs logiciels de moins de 1 000 € tournant sur micro-ordinateur, dotés de descripteurs du système productifs, de gammes et de règles de fonctionnement, ainsi que de générateurs aléatoires d'arrivées dans le système, de temps opératoires et de choix de gammes. Une description sophistiquée d'un système complexe requiert des moyens plus conséquents.

règles de priorité. Dans ce jeu de données, on avait généré aléatoirement pour chacune de ces commandes, mais une fois pour toutes :

- l'ordre de passage des opérations d'une commande entre les différents centres de production,
- les dates d'arrivées des commandes dans le système (utilisation d'un processus de Poisson¹),
- la durée constante des opérations d'une commande dans les différents centres (cette durée était générée aléatoirement à partir d'une loi exponentielle négative, de moyenne égale à 1).

L'utilisation des règles de priorité testées ne reposait bien entendu que sur l'information de l'état du système au moment où une commande se présente et non sur les prévisions d'arrivées de commandes.

La définition du Temps d'Achèvement Moyen \bar{A} que nous avons présentée dans le cadre du modèle statique (§ I-1.1.1.2, page 365) doit être adaptée, puisque le calcul ne s'effectue plus sur un ensemble fini de travaux à exécuter pendant une période et tous disponibles en début de période, mais sur un échantillon aléatoire de 8700 commandes arrivant à des dates différentes dans le système. Le calcul de la moyenne des temps de séjour dans le système s'analyse comme une espérance mathématique, puisque les arrivées dans le système et les temps opératoires sont des variables aléatoires. On peut ajouter que le critère de temps de présence dans le système est conforme aux préoccupations de bon nombre de responsables se réclamant ou non de la philosophie du juste à temps (voir chapitre VII).

On évoquera ci-après les principales règles de priorité testées²:

- La règle d'ordonnancement aléatoire (**RANDOM**) consiste à choisir au hasard parmi les commandes qui sont en attente devant un centre de production. Son intérêt pratique est de permettre un étalonnage³ non ambigu dans la comparaison des performances relatives des règles de décision (éventuellement entre travaux de simulation assez différents).
- La règle du **PAPS** (pour **P**remier-**A**rrivé, **P**remier-**S**ervi ou son équivalent anglo-saxon FCFS, pour *First Come First Served*) est un décalque du célèbre FIFO; cette règle d'ordonnancement, largement répandue en pratique, est

2. Note de la page précédente. Conway, Maxwell & Miller (1967, [104]), p. 220 à 247 et 261 à 290, ces travaux ont été conduits en partie à la Rand Corporation. On peut citer également la thèse (non publiée) de Nanot, *An Experimental Investigation and Comparative Evaluation of Priority Discipline in Job-Shop Like Quening Network*, à laquelle il est très fréquemment fait mention et qui est présentée avec un certain détail dans Buffa & Miller (1979, [74]), p. 494 à 499.

1. Ce processus implique que le processus d'arrivée est sans mémoire et que la probabilité d'arrivée d'une nouvelle commande au cours d'un bref intervalle de temps est totalement indépendante du nombre de commandes déjà présentes dans le système.

2. Les sigles anglo-saxons retenus sont ceux donnés dans le compte rendu des travaux de Conway, Maxwell et Miller.

3. Lorsque l'on est en univers certain, la référence utilisée pour juger la performance d'heuristiques (c'est-à-dire de règles de résolution numérique permettant d'obtenir rapidement une solution dont on ne peut garantir le caractère optimal) est, chaque fois que cela est possible, la solution optimale du problème posé. En univers aléatoire, la résolution d'un problème dynamique s'effectue progressivement sur la base d'un problème nécessairement incomplet (dans notre exemple, on n'attend pas d'avoir les 8700 commandes avant de commencer à les exécuter), sauf à postuler une inacceptable connaissance de l'avenir. Il s'ensuit que la solution de référence ne peut être la solution optimale.

indépendante du temps d'exécution, il ne faut pas s'étonner qu'elle ait des performances moyennes voisines de celles obtenues par la règle aléatoire.

- La règle d'ordonnancement **Temps Opérateur Minimum** (connue aussi sous son sigle **TOM** ou son équivalent anglo-saxon SPT, pour *Shortest Processing Time*) sélectionne celle des commandes en attente d'exécution devant un centre de production qui a le plus faible temps opératoire pour l'opération qui doit être exécutée dans ce centre de production.
- La règle d'ordonnancement de la plus faible durée de travail restant à accomplir dans l'ensemble des centres de production (règle **LWKR**, pour *Least Work Remaining*) est une variante de la règle précédente; elle prend en compte, pour chaque commande en compétition, le cumul des temps opératoires restant à exécuter sur le centre de production étudié et sur les centres suivants.
- La règle d'ordonnancement du quotient de la **marge** (= temps restant avant la livraison, diminué du cumul des temps opératoires restant à réaliser) par le nombre d'opérations restant à exécuter, est plus connue sous son sigle anglo-saxon **S/OPN** (pour *Slack/Operation*); elle préconise de sélectionner la commande ayant la valeur la plus faible de ce ratio.
- La règle **WINQ** (pour *Work in Next Queue*) se fonde sur la remarque qu'il n'est peut-être pas astucieux de privilégier une tâche dans un centre de production si, une fois terminée l'opération de cette tâche dans ce centre, la tâche part dans le centre de production où son opération suivante doit être exécutée, sachant que cette décision a pour conséquence d'allonger une file d'attente déjà importante. Cette règle calcule la priorité d'une tâche comme la somme des durées des opérations des tâches déjà en attente¹ au moment de la prise de décision, dans le centre de production dans lequel l'opération suivante de la tâche considérée sera exécutée; ce cumul est conventionnellement nul si cette opération est la dernière de la tâche. Pour cette règle, comme pour les précédentes (à l'exception des deux premières), la priorité la plus forte est donnée à la tâche qui a la valeur la plus faible de l'indicateur défini.

L'usage de ces différentes règles définit dynamiquement les priorités qui sont réévaluées à chaque décision de chargement de machine et non sur la base d'un calcul initial effectué en début de simulation et jamais remis en cause ultérieurement². Par ailleurs, elles n'imposent pas les mêmes contraintes en matière de systèmes d'information, selon qu'elles sont utilisées pour une prise de décisions opérationnelles ou dans le cadre de simulations:

- des règles, comme les règles S/OPN ou LWKR, s'appuient sur une connaissance correcte de ce qui se passe dans tout le système productif au moment où l'on prend la décision; elles supposent donc que l'entreprise fonctionne avec un système d'information en bases de données centralisées ou réparties avec mises à jour en temps réel;
- des règles, comme les règles PAPS ou TOM, ne reposent que sur des informations disponibles localement à l'endroit même où un choix est à opérer; elles

1. Si le système d'information le permet, il est judicieux d'inclure à ce cumul le temps résiduel de l'opération en cours.

2. On parle alors de **règles dynamiques** et de **règles statiques**. Certains logiciels d'ordonnancement s'appuient sur des règles; leurs performances sont meilleures lorsqu'ils utilisent des règles dynamiques.

constituent, au contraire, des procédures de gestion décentralisées faciles à implanter.

En résumé, moins une règle est «myope» dans l'espace du système productif (connaissance précise de l'état du système productif, au moment de la prise de décision) et dans le temps (prévision de l'évolution de l'état du système productif sur le très court terme et sur des bases «assez mécaniques»), plus la décision requiert d'informations et de capacités de stockage et de calculs.

Illustrons par un exemple ([tableau 100](#)) l'utilisation de ces règles, avec quatre commandes, notées *a*, *b*, *c* et *d*, qui sont candidates pour être exécutées sur la machine A qui se libère à l'instant $t = 90$ minutes.

TABLEAU 100
Jeu de données du problème d'ordonnancement en ateliers spécialisés
(durées et dates exprimées en minutes)

Tâches	Temps opératoire sur la machine A	Opération suivante		Cumul de <i>tous</i> les temps opératoires restant à exécuter à l'instant $t = 90$	Date de livraison demandée	Nombre d'opérations restant à exécuter
		à exécuter sur la machine	temps opératoire			
a	10	K	22	100	$t = 270$	4
b	20	K	10	37	$t = 170$	3
c	17	F	9	41	$t = 270$	2
d	15	G	4	29	$t = 170$	4

Au moment où se pose le problème, la charge totale de travail (travail en cours résiduel et travail en file d'attente) des centres de production F, G et K est: 65 minutes pour la machine F; 100 minutes pour la machine G; 0 minute pour la machine K. La tâche à exécuter sur la machine A est:

- la tâche a si l'on utilise la règle TOM;
- la tâche a ou la tâche b si l'on utilise la règle Winq puisque l'opération suivante de ces deux tâches s'effectue sur la machine K qui est celle qui a le volume de travail en cours et en attente le plus faible (ici 0);
- la tâche b si l'on utilise la règle de la marge ($a: 270 - 100 - 90 = 80$; $b: 170 - 37 - 90 = 43$; $c: 270 - 41 - 90 = 139$; $d: 170 - 29 - 90 = 51$);
- la tâche d si l'on utilise la règle S/OPN ($a: 80/4 = 20$; $b: 43/3 = 14,33$; $c: 139/2 = 69,5$; $d: 51/4 = 12,25$).

Les résultats fondateurs de cette approche amorcée en 1964 et 1965, sont donnés dans le [tableau 101 de la page 415](#). On peut y noter que la règle TOM conduit au temps d'achèvement moyen le plus faible mais qu'il n'en est pas de même pour la dispersion de ce temps. Une analyse complémentaire de ces données, s'appuyant sur le fait que le temps moyen total de la fabrication proprement dite est de 9 (puisque par hypothèse le temps moyen d'une opération dans un centre de production est de 1), conduit à un temps moyen d'attente devant chaque centre de production de 2,78 avec la règle TOM et 7,27 avec une règle du type FCFS, soit une durée d'attente multipliée par 2,6. Trois remarques complémentaires sont utiles à faire dans l'optique d'une implantation de la règle TOM:

- Une approximation à $\pm 10\%$ près des temps opératoires utilisés dans la décision, par rapport aux temps réels utilisés dans la simulation, n'accroît pas le

- nombre moyen instantané de tâches en attente. Si l'on pousse l'erreur jusqu'à $\pm 100\%$, on n'accroît que faiblement ce nombre moyen (qui passe à 27,13).
- On peut sans problème séparer les travaux en 2 classes, la première regroupant les tâches urgentes et la seconde, les autres tâches, et adopter la règle TOM dans chacune des classes ainsi que celle d'une priorité absolue des tâches de la classe urgente sur celles de l'autre classe. Dans ces conditions le nombre moyen instantané de tâches en attente ne s'accroît que faiblement (il passe à environ 27 pour un partage à 30 % / 70 % des tâches en urgent / normal); ces deux premières remarques attestent une certaine robustesse des performances de la règle TOM lorsqu'elle est utilisée dans des conditions dégradées (mais fort réalistes).
 - L'application stricte de la règle TOM peut avoir pour effet de retarder considérablement certaines opérations longues. Une solution efficace, semble-t-il, est alors de basculer périodiquement sur la règle FCFS; cela dit, la prise en compte explicite de la date de livraison constitue une alternative intéressante, comme nous allons le voir ci-après.

TABLEAU 101
Comparaison des performances des règles de priorité [104]

Règles		RANDOM	FCFS	TOM (SPT)	LWKR	WINQ	SOPN
Nombre moyen instantané de tâches en attente dans le système		59,42	58,87	23,25	47,52	40,43	Données non disponibles
Temps d'achèvement total d'une tâche	\bar{x}	74,70	74,43	34,02	Données non disponibles		66,10
	σ	Données non disponibles	41,06	53,65			16,31

Souvent le problème posé se complique par la nécessité de respecter, autant que faire se peut, des dates de livraison. La règle qui a les meilleures performances dans le respect des dates de livraison est la règle S/OPN. En contrepartie, elle accroît sensiblement le temps moyen d'achèvement d'une tâche mais elle réduit au maximum la dispersion de ces temps d'achèvement. Cela étant, dès que le taux d'utilisation du système productif devient élevé (disons supérieur à 90 %, pour fixer les idées), cette règle conduit à un pourcentage moyen de tâches terminées avec retard qui est supérieur à celui que l'on a en utilisant la règle TOM. Dans ces conditions, si l'objectif prioritaire est celui du respect des délais de livraison, la procédure préconisée consiste à utiliser la règle TOM en cas de forte «congestion» du système productif et, autrement, de faire appel à la règle S/OPN.

Ces travaux de Conway, Maxwell et Miller sont à l'origine d'un courant de recherche important qui a exploré la performance d'autres heuristiques de résolution de classes de problèmes d'ordonnancement extraordinairement variées. De nos jours, il est extrêmement facile (voir, [chapitre III, § I-2.1, page 148](#)) de déterminer la performance de règles

II-2.2.2 Cas d'une dépendance entre les centres de production

L'ordonnancement se complique, d'une part, par l'existence d'une certaine polyvalence de centres de production qui sont substituables dans l'exécution d'une opération, sans avoir pour autant les mêmes performances, et, d'autre part, par la possibilité d'inverser l'ordre de réalisation de certaines opérations (*open shop*, voir § I-1.5, page 384). Les investigations décrites dans Conway, Maxwell et Miller, montrent que le nombre moyen de tâches en attente est une fonction décroissante du nombre de centres de production polyvalents et que la règle TOM est plus performante que la règle FCFS, mais que sa supériorité relative décroît très sensiblement quand la polyvalence s'accroît. La possibilité d'inverser l'ordre de certaines opérations a des conséquences moins sensibles que dans le cas de la polyvalence; là encore, la règle TOM est plus performante que la règle FCFS. On peut indiquer, pour terminer, que cette étude aborde également le cas de la possibilité d'une structuration arborescente des différents centres de production, le premier nœud de l'arbre correspondant à la dernière étape du processus productif (assemblage par exemple).

SECTION III PERSPECTIVES ACTUELLES DE L'ORDONNANCEMENT EN ATELIERS SPÉCIALISÉS

Les entreprises industrielles qui ne sont pas orientées vers la série unitaire ou la fabrication en continu sont structurées en ateliers spécialisés, en îlots de fabrication, ou en lignes d'assemblage et/ou de fabrication. On ne s'intéressera ici qu'aux structures en ateliers spécialisés ou îlots de fabrication, largement prépondérantes dans les entreprises occidentales.

Les conditions requises pour passer en flux tirés et donc pouvoir gérer les flux sans préparation du travail par un ordonnancement «en bonne et due forme», font que le problème de l'ordonnancement se pose concrètement dans une majorité d'entreprises. Il est toujours possible de nier le problème et de se passer de toute préparation du travail (suppression de l'ordonnancement) en lançant les ordres de fabrication dès que ceux-ci arrivent dans le système productif et en gérant indépendamment les différents postes de travail avec l'aide de règles empiriques de décision. Cette absence de préparation du travail n'est pas sans inconvénients importants car:

- elle retarde inutilement certaines adaptations temporaires du système productif (appel momentané aux heures supplémentaires ou modifications du programme de maintenance, par exemple) ou des règles de pilotage de la production (d'où une modification des priorités des ordres de fabrication); elle conduit à supporter des problèmes au lieu de les anticiper, faute d'avoir détecté à temps certains dysfonctionnements (saturation excessive de certains postes de travail, retard dans les livraisons...);
- elle accentue le découplage entre la régulation à court terme et la régulation à moyen terme, ce qui peut rendre plus difficile la prise de certaines décisions (acceptation de marchés importants et contraignants, décision de sous-traitance...);
- elle laisse entier le problème de l'approvisionnement des composants et matières utilisés au cours de certaines opérations; celui-ci peut être

déclenché lors de l'introduction d'un ordre de fabrication dans le système productif mais tout défaut d'approvisionnement contribuera à engorger le système productif par des OF dont l'exécution est arrêtée, faute de composants ;

- elle pose le problème du contrôle d'un système productif fonctionnant sans prévision : l'absence de référentiel rend difficile l'appréciation de la pertinence des décisions prises.

Pour la majorité des gestionnaires, ces arguments militent en faveur du maintien d'une préparation du travail en ateliers. Dans ces entreprises, la qualité de l'ordonnancement est considérée comme un facteur important de compétitivité et les procédures manuelles laissent très progressivement place à des systèmes informatisés destinés à concevoir rapidement une proposition d'ordonnancement et, pour les plus sophistiqués d'entre eux, à réagir en temps réel aux inévitables aléas et perturbations qui caractérisent la production. L'investissement organisationnel que représentent de tels systèmes est encore peu répandu, pour deux raisons :

- de tels systèmes requièrent une formalisation préalable de ce que le système productif peut traiter (gammes, nomenclatures...), laquelle est loin d'être systématique et fiable dans les entreprises ; ils exigent en outre une mise à jour du système d'information qui est pratiquement de type « temps réel », ce qui est une condition encore plus restrictive ;
- de par leur formation, les ingénieurs privilégient le plus souvent la flexibilité physique (machines à commande numérique, convoyage automatique...) sur la flexibilité organisationnelle, dont l'une des formes est l'assistance informatique à l'ordonnancement¹.

Les préoccupations des établissements qui se dotent de tels systèmes d'assistance à l'ordonnancement varient selon le contexte dans lequel ils sont placés (production et/ou assemblage pour stock ou à la commande, le positionnement concurrentiel de l'entreprise en termes de coût, qualité, délai...) et la marge de manœuvre dont dispose le responsable de l'établissement.

Il est évident qu'un responsable, jugé uniquement sur le respect de dates de livraison des ordres de fabrication, n'a besoin que d'un système l'aidant à définir un ordonnancement respectant ces contraintes de livraison. Il sera conforté dans son attitude par le fait qu'en cas de difficulté majeure pour tenir ces délais, il pourra, presque toujours, assez facilement mobiliser les ressources additionnelles (heures supplémentaires, intérim, sous-traitance) qui lui permettront d'honorer ses obligations. Dans ce contexte, le problème essentiel est d'obtenir assez rapidement un ordonnancement respectant les contraintes².

Mais, dans de nombreuses entreprises, le problème est plus complexe car il s'agit à la fois de respecter, autant que faire se peut, les délais et d'obtenir une bonne utilisation du système productif, pour conserver une compétitivité par les coûts sur le long terme. Un tel contexte suppose que le respect des délais n'est plus un objectif intangible et pose un problème d'évaluation de la qualité de l'ordon-

1. Voir le § I-1.5, page 147, du chapitre III pour une discussion des relations de complémentarité entre flexibilité physique et flexibilité organisationnelle et de leurs impacts en matière de conception des systèmes productifs.

2. Dans ce type de situation, l'approche qui sera décrite au § III-2.2.1, page 437, s'avère tout à fait pertinente.

nancement, inconnu dans le cas précédent. La nécessité de ce jugement se justifie par deux observations :

- Si l'on a le choix entre deux ordonnancements respectant les contraintes de dates de livraison et mobilisant les mêmes ressources permanentes, il est difficile de considérer ces alternatives comme équivalentes lorsque l'un des deux ordonnancements permet des livraisons effectives plus précoces et donc, implicitement, une meilleure utilisation du système productif, parce que cela procure une marge de manœuvre accrue pour les commandes ultérieures qui permettra, peut-être même, d'accepter davantage de commandes.
- La décision à évaluer est très souvent compliquée par le fait que la marge de manœuvre du décideur porte non seulement sur l'ordonnement à retenir mais aussi sur une *éventuelle mobilisation de ressources additionnelles* (heures supplémentaires...). La question pratique qui se pose alors est de savoir :
 - si l'on a raison ou non de faire appel à un peu plus de ressources additionnelles, qui coûtent nécessairement de l'argent, pour effectuer un peu plus de production
 - ou s'il est plus judicieux de limiter cet appel en restreignant corrélativement la production.
- Le jugement peut s'opérer en faisant appel à une batterie d'indicateurs physiques (taux d'engagement des diverses ressources, pourcentage d'avancement des commandes, nombre de commandes en retard, dépenses de mobilisation momentanée de ressources additionnelles...) mais il est peu probable qu'entre deux solutions alternatives, tous les indicateurs soient en faveur de l'une d'entre elles. Un jugement d'ensemble passe donc par l'utilisation d'un indicateur synthétique qui doit être de nature économique, c'est-à-dire être le résultat d'un calcul faisant intervenir des grandeurs exprimées en valeur. D'autres logiques d'agrégation de points de vue existent (analyse multicritères...) mais sont difficiles à mettre en œuvre ou largement arbitraires. On reviendra concrètement sur ce problème au § III-2.5.1, page 442.

On examinera donc les approches utilisées par les logiciels pour proposer un ordonnancement (§ III-1), avant de présenter les orientations des systèmes d'aide à la décision de lancement, actuellement en émergence et qui devraient rapidement se répandre et être en mesure d'apporter des éléments de réponse au problème dans toute sa complexité, telle qu'elle vient d'être décrite (§ III-2, page 432).

III-1 Les approches possibles

Pour être aussi concret que possible, on s'appuiera sur un exemple introductif (§ III-1.1) avant de présenter les démarches de résolution utilisées (§ III-1.2, page 421).

III-1.1 Exemple introductif

Pour comprendre la complexité du problème d'ordonnement et les différentes alternatives que l'on peut utiliser pour le résoudre, on s'appuiera sur un problème d'ordonnement simplifié, celui de la société **MÉCANO**.

Le système productif de **MÉCANO** comporte seulement quatre ateliers : un atelier de tour (avec un opérateur et une machine), un atelier de rectifieuse (avec un opérateur et une machine), un atelier de fours (avec un opérateur et deux fours) et un atelier de fraiseuses (avec trois opérateurs et trois postes de travail). Ce système est décrit par la **figure 123** (qui comporte des indications de coût qui seront utilisées ultérieurement, au § III-2.5.1, page 442).

FIGURE 123

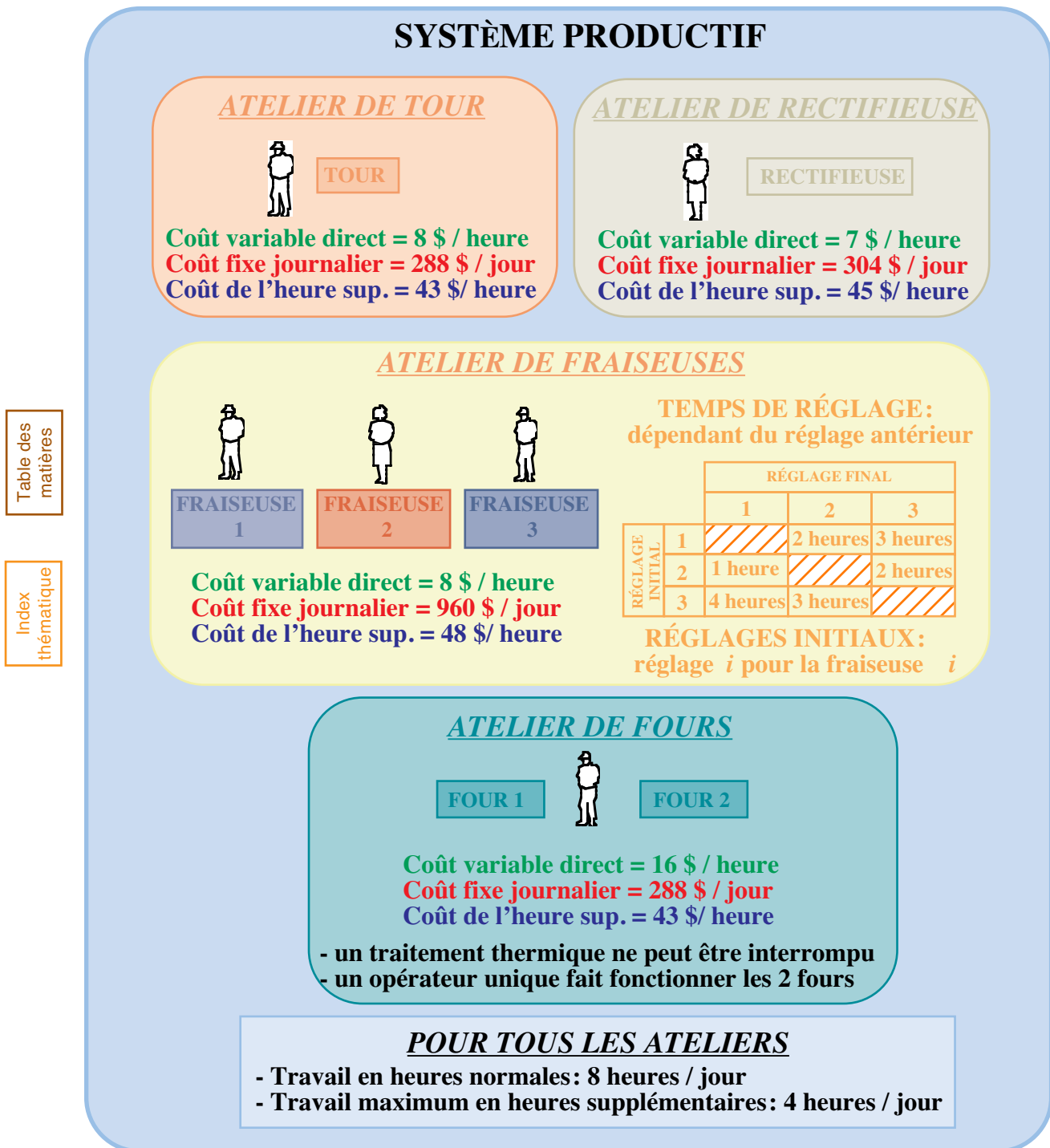
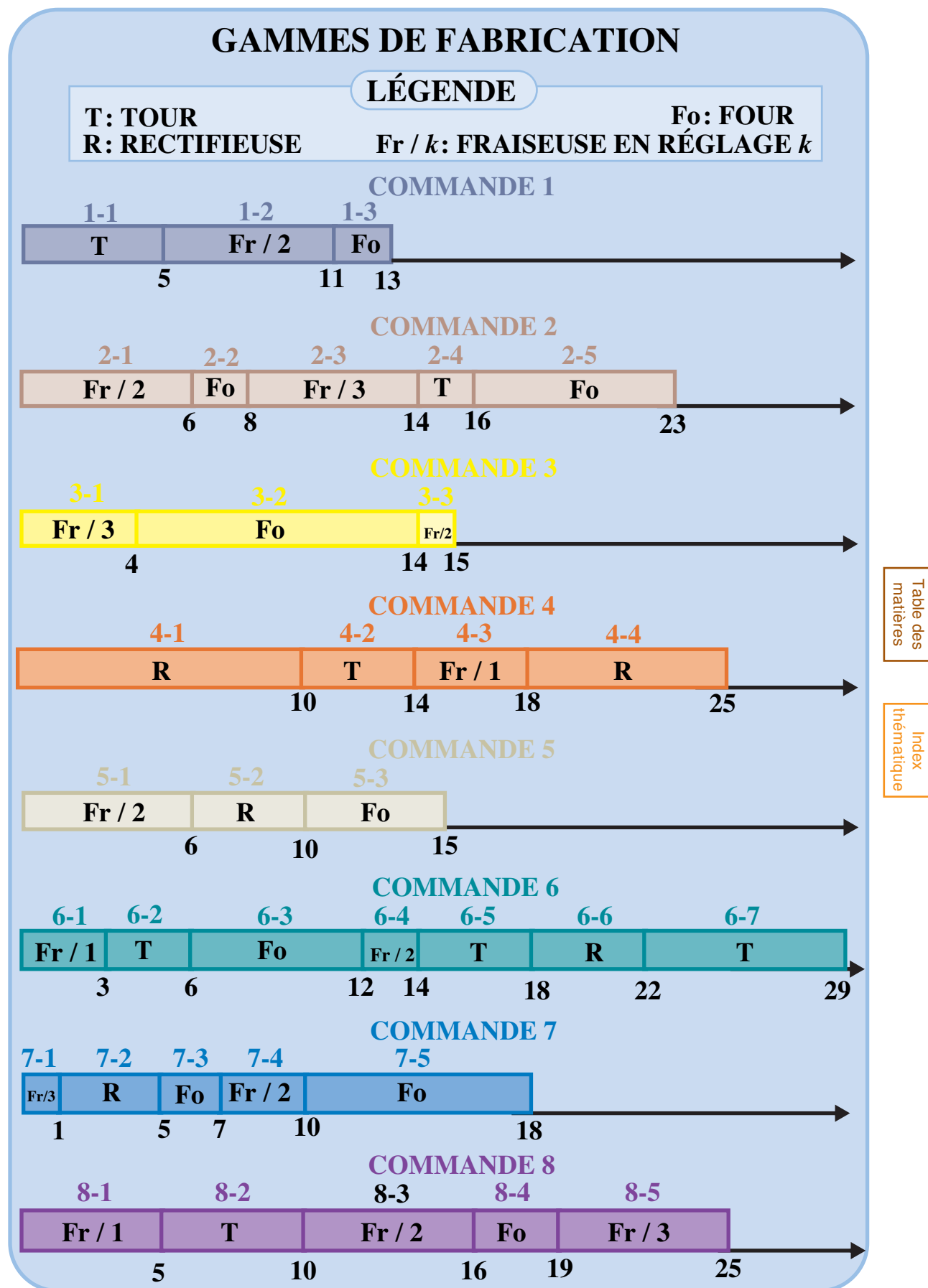
Description du système productif du cas **MÉCANO**

Table des matières

Index thématique

FIGURE 124

Description des gammes des ordres de fabrication du cas **MÉCANO**

L'un des ateliers, celui des fours, se caractérise par l'interdiction d'interrompre une opération commencée (préemption interdite, voir [page 361](#)); c'est généralement le cas des traitements chimiques ou thermiques. Un autre atelier, celui des fraiseuses, se caractérise par le fait que le temps de changement du réglage d'une machine n'est indépendant ni du réglage actuel, ni du réglage futur. Ce cas de figure est relativement fréquent pour les postes de travail dotés d'une certaine polyvalence. Il est évident que les temps de réglage utilisés sont fantaisistes (ils correspondent à un nombre entier d'heures, pour faciliter la présentation des résultats); en outre, pour simplifier le problème, le nombre de réglages possibles a été limité à trois¹. On notera enfin que la décision à prendre concerne non seulement la définition d'un ordonnancement mais aussi celle d'un appel momentané à des ressources additionnelles par le biais d'heures supplémentaires (sans prise en compte, ici, des contraintes imposées par la réglementation du travail).

On suppose que **MÉCANO** a huit commandes à exécuter, la première étant la seule à avoir déjà été commencée. Les gammes de ces commandes sont décrites sous forme de graphique de Gantt à la [figure 124, page 420](#). La $j^{\text{ème}}$ opération de la commande i est notée « $i-j$ »; la gamme est fournie par un graphique de Gantt où la mention de la machine requise est portée («Fo» pour un four, «R» pour la rectifieuse, «T» pour le tour et «Fr / k » pour une fraiseuse en réglage k). Ces gammes n'ont aucun réalisme: il s'agit seulement de travailler sur des gammes dans lesquelles non seulement l'ordre de passage sur les centres de production varie d'une commande à l'autre mais encore où un même poste de travail peut être utilisé plusieurs fois à des moments différents dans une même gamme. Pour la commande 1, seule la gamme des opérations restant à exécuter est portée sur la figure. Lorsqu'une opération doit être effectuée sur une fraiseuse, le numéro de réglage est indiqué sur le graphique. On supposera que ce type de réglage ne peut s'effectuer en temps masqué (c'est-à-dire pendant que s'achève l'opération précédente sur ce même poste).

Table des matières

Index thématique

III-1.2 Les solutions possibles

Deux grandes familles d'approches peuvent être rencontrées: le placement progressif d'ordres de fabrication (§ III-1.2.1) et le placement chronologiquement progressif d'opérations exécutables (§ III-1.2.2, [page 424](#)). Certains logiciels mélangent les deux approches.

III-1.2.1 Placement progressif d'ordres de fabrication

Cette approche, relativement facile à mettre en œuvre, a été utilisée manuellement par les premiers services de *planning* et est encore utilisée, malgré ses défauts, par quelques logiciels. Le principe retenu² pour le placement progressif d'ordres de fabrication est extrêmement simple pour ne pas dire simpliste:

- on commence par classer les ordres de fabrication (OF) selon un critère;

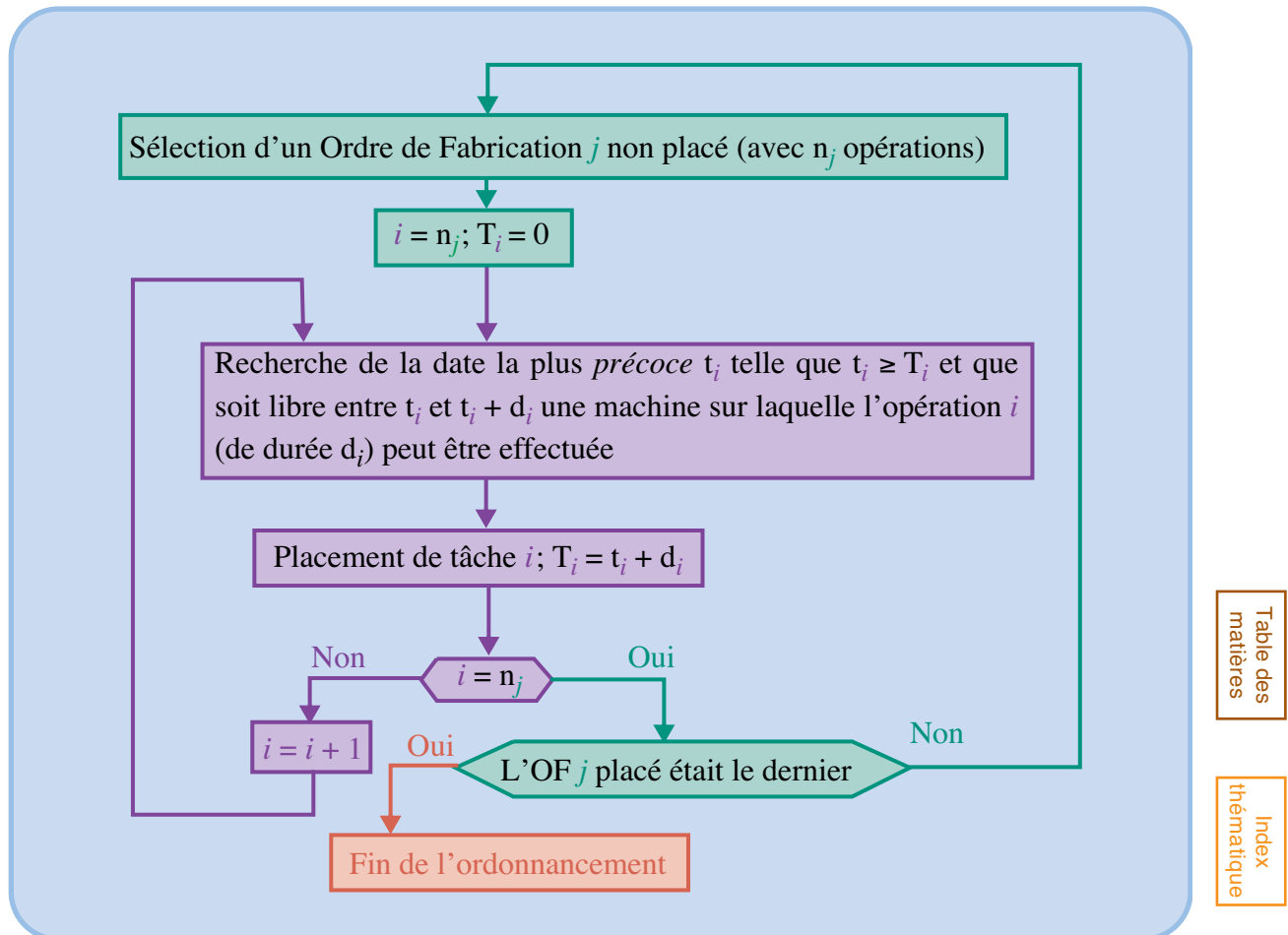
1. La spécialisation de chacune des machines à un réglage différent est une mauvaise stratégie si les demandes pour chacun des réglages ne sont pas «équilibrées». En tout état de cause, le problème des changements de réglage se pose nécessairement si le nombre de machines est différent du nombre de réglages.

2. Elle est connue sous le nom d'*horizontal loading*, dans la littérature anglo-saxonne et elle est parfois désignée comme la **procédure par placement des travaux** dans la littérature française spécialisée.

- on prend ensuite le premier OF de la liste et on place les différentes opérations de l'ordre de fabrication choisi
- soit au plus tôt si on a retenu un placement progressif au plus tôt (voir algorithme à la figure 125),

FIGURE 125

Algorithme de placement au plus tôt des ordres de fabrication

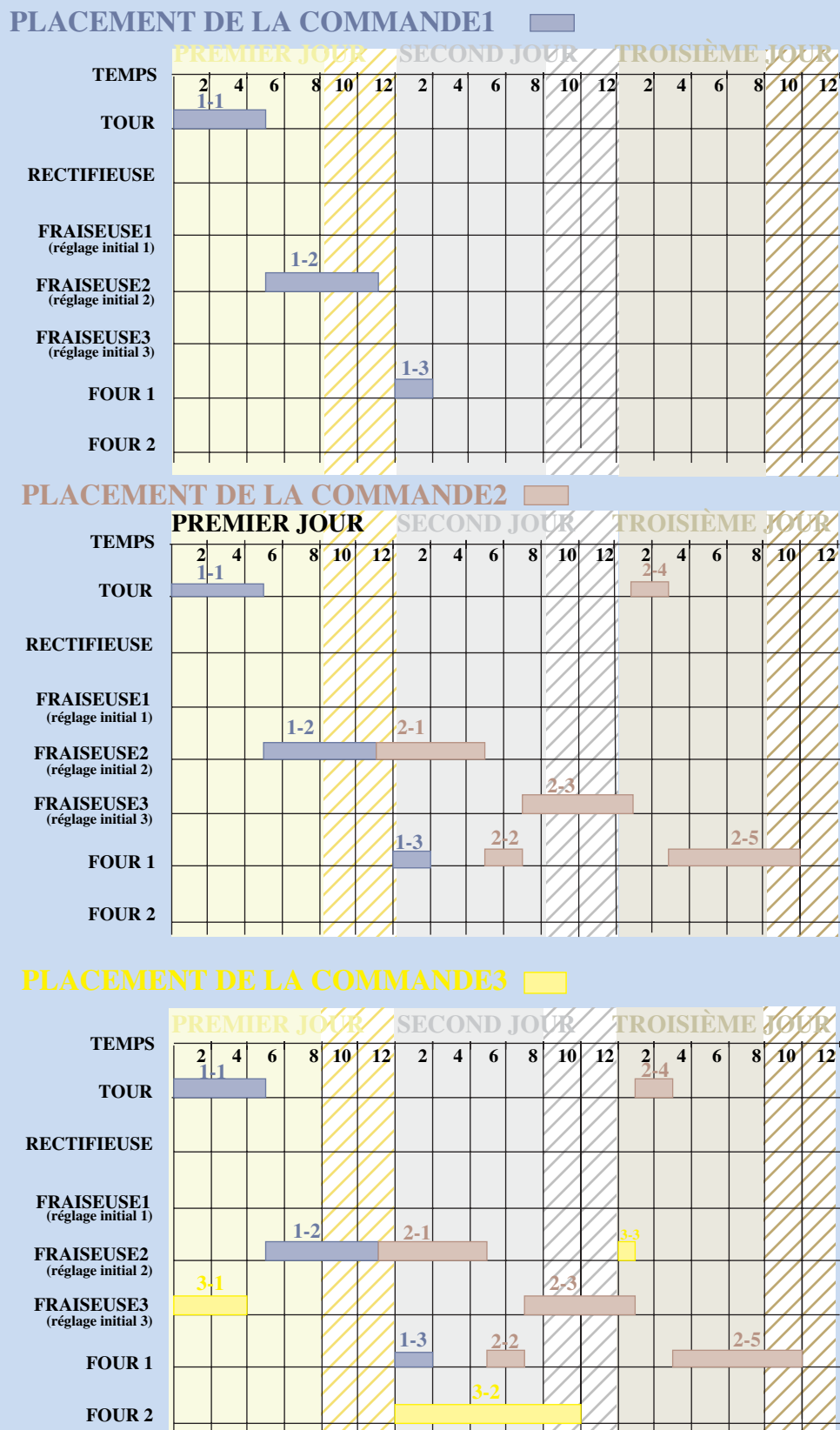


- soit au plus tard si l'on a retenu un placement progressif au plus tard (voir algorithme à la figure 127 de la page 425);
- ces placements se font en respectant la disponibilité des ressources devant être mobilisées et les contraintes d'antériorité des opérations à exécuter; une fois toutes les opérations placées, l'ordre de fabrication est rayé de la liste;
- s'il reste encore des ordres de fabrication dans la liste, on retourne à l'étape précédente, sinon, l'ordonnancement est terminé.

Le placement au plus tôt ne garantit pas le respect des dates de livraison et le placement au plus tard peut rendre impossible le placement des OF situés en queue de liste, tout en conduisant souvent à une médiocre utilisation des ressources en début de programmation.

La figure 126 illustre la technique du **placement au plus tôt** – on parle encore de **jalonement au plus tôt** (*front scheduling* dans la littérature anglo-saxonne) – avec une utilisation quasi systématique des heures supplémentaires. On suppose ici que l'ordre retenu est celui des numéros associés aux différentes commandes

FIGURE 126
*Placement au plus tôt des ordres de fabrication (cas **MÉCANO**)*

Table des
matièresIndex
thématique

(on verra au [tableau 103, page 447](#), que ce classement correspond à celui des dates de livraison).

Cette illustration numérique permet de mettre en évidence le dilemme majeur de ce type d'approche et la très grande difficulté qu'il y a à le résoudre :

- le *problème posé* : une fois la commande 1 programmée, on procède au placement de la commande 2 qui débute par l'opération 2-1¹ de fraiseuse en réglage 2 ; la fraiseuse 2 qui est en réglage 2, se trouve être occupée par l'opération 1-2, à partir du début de la sixième heure du premier jour ; l'opération 2-1 durant six heures, ne peut donc être exécutée avant l'opération 1-2 ;
- la *solution envisageable* : il serait possible de remettre en cause les décisions antérieurement prises et plus particulièrement ici, le placement de l'opération 1-2, en retardant le début d'une heure, afin d'utiliser les cinq heures disponibles en début de journée, pour l'opération 2-1 ;
- la *difficulté de mise en œuvre* de ce type de solution : cette solution, facile à mettre en œuvre lorsque l'on est au début du processus de placement, est particulièrement difficile à utiliser dès que plusieurs ordres de fabrication ont été programmés car il faut alors veiller au respect de toutes les contraintes d'antériorité des tâches déjà programmées, dont un déplacement partiel est envisagé directement (dans notre exemple, le déplacement de l'opération 1-2 implique celui de l'opération 1-3) ou par un effet de cascade ; on peut ajouter que les décalages de programmation induits peuvent annuler l'intérêt du décalage initial envisagé pour améliorer l'utilisation des ressources en début de simulation.

Cette interdépendance des décisions, rendant la remise en cause des décisions antérieurement prises à la fois difficile et d'un intérêt parfois contestable, explique que l'on s'en tienne généralement à l'algorithme initial (figure 125) et pourquoi ce type d'approches se traduit assez souvent par une utilisation médiocre des ressources disponibles.

On n'illustrera pas la logique du **placement au plus tard** (décrit par l'algorithme de la figure 127). Le principe consiste à partir de la dernière opération de l'ordre de fabrication retenu et à la programmer le plus tard possible, tout en respectant la date de livraison, puis à programmer l'opération précédente en respectant les contraintes d'antériorité et ainsi de suite. Cette seconde approche vise à limiter le montant des encours, puisque l'on attend le dernier moment possible pour programmer les opérations d'un OF. Cet objectif est imparfaitement atteint par cette méthode car, la solution retenue ne tendrait à minimiser le montant des encours que si la valeur des ordres de fabrication était prise en compte dans le classement des OF.

III-1.2.2 Placement chronologiquement progressif d'opérations exécutables

Cette seconde approche² s'inspire directement des travaux de simulation effectués à partir des années soixante. *Elle est utilisée par le logiciel de résolution du « mini-cas » proposé, avec la possibilité, à tout instant d'éclairer les décisions à prendre*

Table des
matières

Index
thématique



1. Le premier nombre désigne l'OF et le second, après le tiret, repère le numéro d'ordre de l'opération dans l'OF ; l'opération 2-1 est donc la première opération de la commande 2.

2. Elle est connue sous le nom de *vertical loading* dans la littérature anglo-saxonne. Elle est parfois désignée comme la **procédure par placement d'opérations** dans la littérature française spécialisée.

FIGURE 127

Algorithme de placement au plus tard des ordres de fabrication

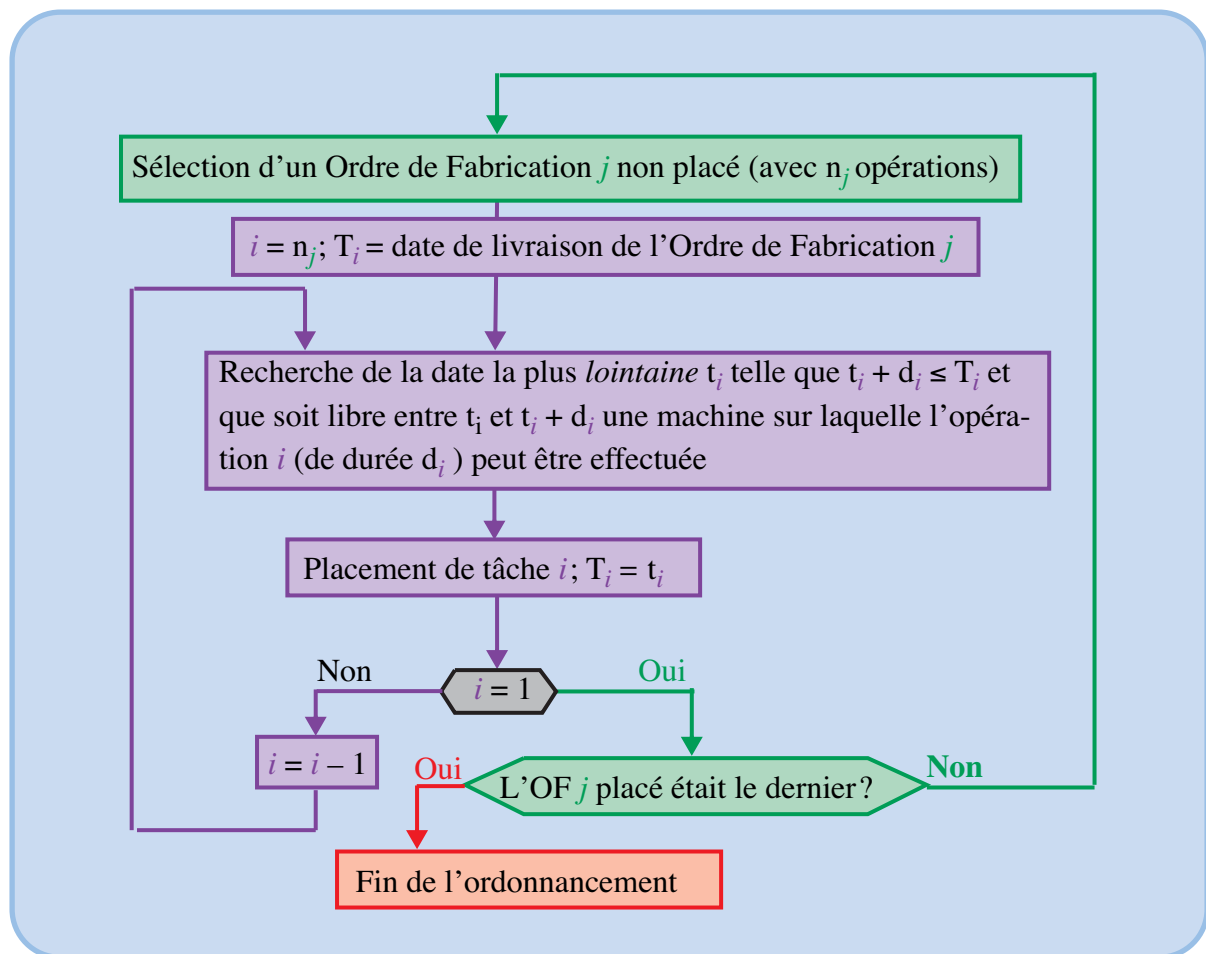


Table des matières

Index thématique

par un ensemble d'indicateurs qui seront évoqués ci-après. L'usage de ce logiciel permet de mieux comprendre les fondements des arbitrages à réaliser entre le point de vue commercial qui privilégie le respect des délais et le point de vue de la production qui cherche à utiliser de manière efficace les ressources disponibles. Il permet de revenir sur des décisions antérieurement prises, ce qui permet de mieux appréhender des mécanismes mis en œuvre dans certains SIAD¹.

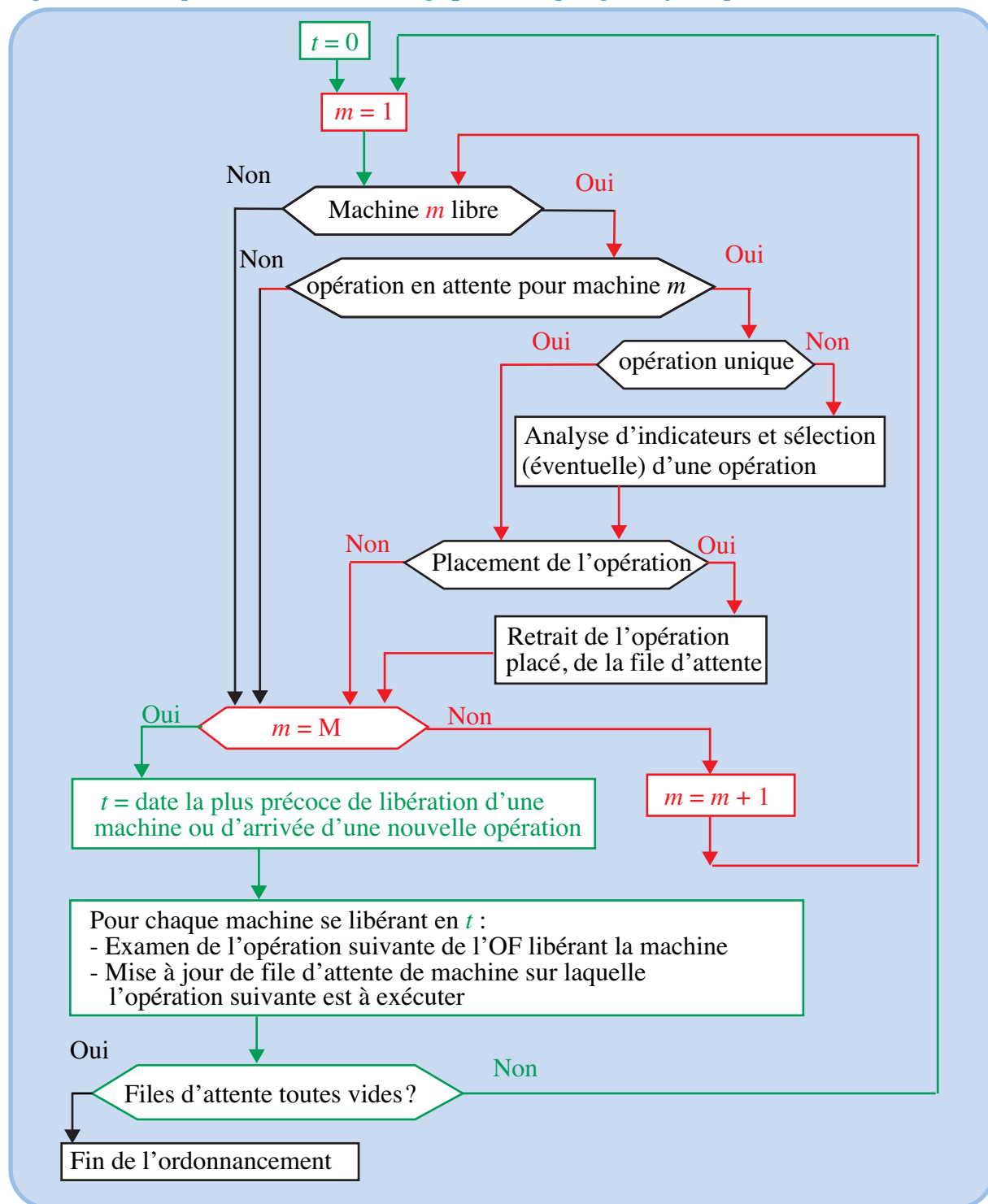
La démarche décrite par l'algorithme de la page 426, passe par deux grandes étapes que l'on va examiner. Préalablement il faut initialiser le processus en définissant, pour chaque atelier, la liste des opérations immédiatement exécutables dans cet atelier, à partir des premières opérations non encore exécutées des ordres de fabrication, ce qui donne, pour notre exemple, le premier tableau de file d'attente de la figure 129, page 428. Examinons l'application de l'algorithme.

- **Étape 1**: à une date t (initialement $t = 0$), on passe en revue successivement les différentes machines du système productif; si la machine n'est pas libre on passe à la machine suivante (initialement toutes les machines sont libres) sinon on examine s'il y a un ou plusieurs candidats pour être exécutés sur ce poste de travail:

1. Systèmes Interactifs d'Aide à la Décision (en anglais, DSS pour *Decision Support Systems*).

FIGURE 128

Algorithme de placement chronologiquement progressif d'opérations exécutables



- s'il n'y a qu'un seul candidat, celui-ci, sauf raison particulière liée à l'anticipation de l'arrivée d'une opération jugée prioritaire, est chargé sur la machine;
- s'il y a plusieurs candidats on sélectionne à l'aide d'une règle (ou d'un ensemble de règles) celui qui doit être privilégié; ces règles peuvent être celles introduites à la [page 412](#) ou être plus complexes;
- l'opération, une fois chargée, disparaît de la file d'attente.

- **Étape 2** : lorsque toutes les machines ont été passées en revue pour cette date t , on détermine la prochaine date (qui deviendra la nouvelle date courante t dans la simulation) pour laquelle se produit une modification de l'état du système ; celle-ci consiste soit en une arrivée prévisionnelle d'un nouvel OF, soit par l'achèvement de l'exécution d'une (ou de plusieurs) opération(s) ; l'achèvement d'une opération implique que :
 - l'opération suivante de l'OF (si elle existe) est exécutable et sera donc immédiatement chargée dans la file d'attente concernée,
 - la machine libérée est disponible pour exécuter une opération se trouvant dans sa file d'attente ;
- en cas de libération de plusieurs machines, il est conseillé pour être en mesure de déceler la totalité des conflits potentiels à cette nouvelle date t :
 - d'abord de commencer par libérer toutes les machines sur lesquelles des opérations s'achèvent et de mettre aussitôt à jour des files d'attente
 - et seulement ensuite de procéder au chargement des machines libres suivant la logique décrite à l'étape 1.

Le début de la simulation de notre problème introductif est illustré par la [figure 129](#), pages [428](#) et [429](#) ; les arbitrages effectués pour le chargement des fraiseuses ne sont pas explicités ici. Le résultat final d'une simulation opérée à l'aide d'un ensemble de règles complexes mais n'aboutissant pas à une très bonne solution (afin de conserver un intérêt opérationnel à l'usage du logiciel) est consigné dans la [figure 130](#), pages [430](#) et [431](#). La solution proposée se caractérise par un large appel aux heures supplémentaires et le fait que la commande 3 n'est pas achevée à la fin du jour 3¹. On remarquera, par ailleurs, que l'ordonnancement conduit à des changements de réglage (par exemple, la fraiseuse 1 passe du réglage 1 au réglage 3 au cours du jour 1, ce changement étant noté « \rightarrow R3»).

Table des
matièresIndex
thématique

Ce type de simulation conduit généralement à une bonne utilisation des ressources des premières périodes. La principale difficulté réside dans le choix des règles à privilégier dans la sélection d'une opération lorsque plusieurs rentrent en compétition pour l'utilisation d'une ressource libre. Il faut en effet effectuer un arbitrage entre le point de vue commercial (respect des dates de livraison) et celui de la production (meilleure utilisation des ressources productives). Les règles disponibles (dont les principales ont été présentées à la [page 412](#)) tendent à favoriser soit l'un soit l'autre point de vue. *Le logiciel du CD-Rom permet d'éclairer chaque prise de décision par un calcul préalable des indicateurs associés à ces règles (puisque ceux-ci varient dynamiquement, ce qu'illustre la [figure 131 de la page 431](#) pour la première et la dernière des décisions décrites à la [figure de la page 429](#)) ; il donne également des éléments permettant de mieux évaluer la saturation prévisionnelle des ressources, suivant une démarche qui sera développée à la [page 433](#) et dans les graphiques de la [figure 132 de la page 436](#).*

La principale difficulté tient au fait qu'il est très difficile de prévoir comment se comportera exactement le système lorsqu'on utilise un ensemble de règles précises. Il s'ensuit que certaines décisions se traduisant par de bonnes performances sur le court terme peuvent masquer de mauvaises performances se produisant ultérieurement. Bien entendu, la réciproque est vraie.

1. On reviendra ultérieurement (§ III-2.4, [page 441](#)) sur l'horizon de planification à retenir.

FIGURE 129

Exemple de placement chronologiquement progressif d'opérations

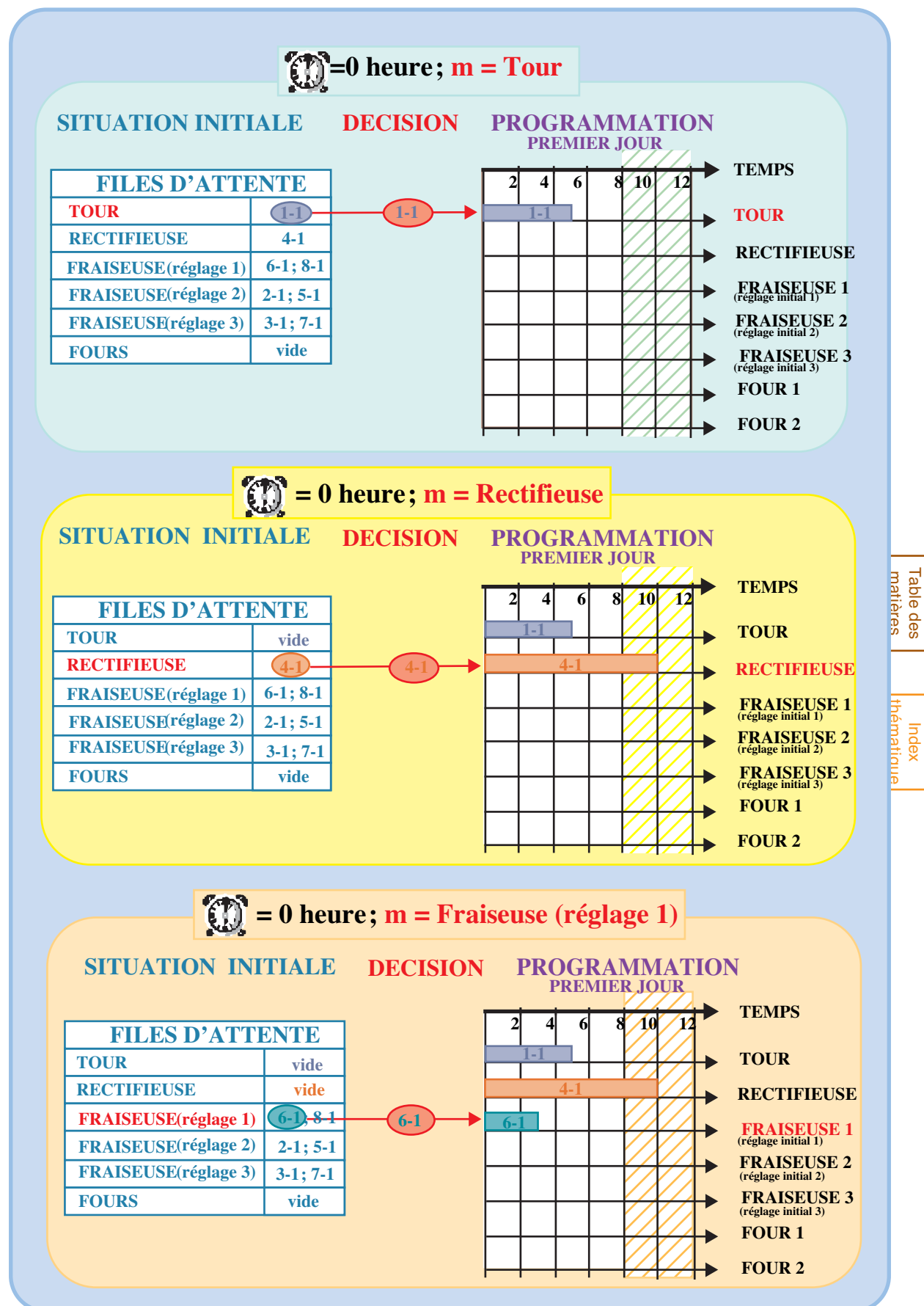


Figure 129 (suite)

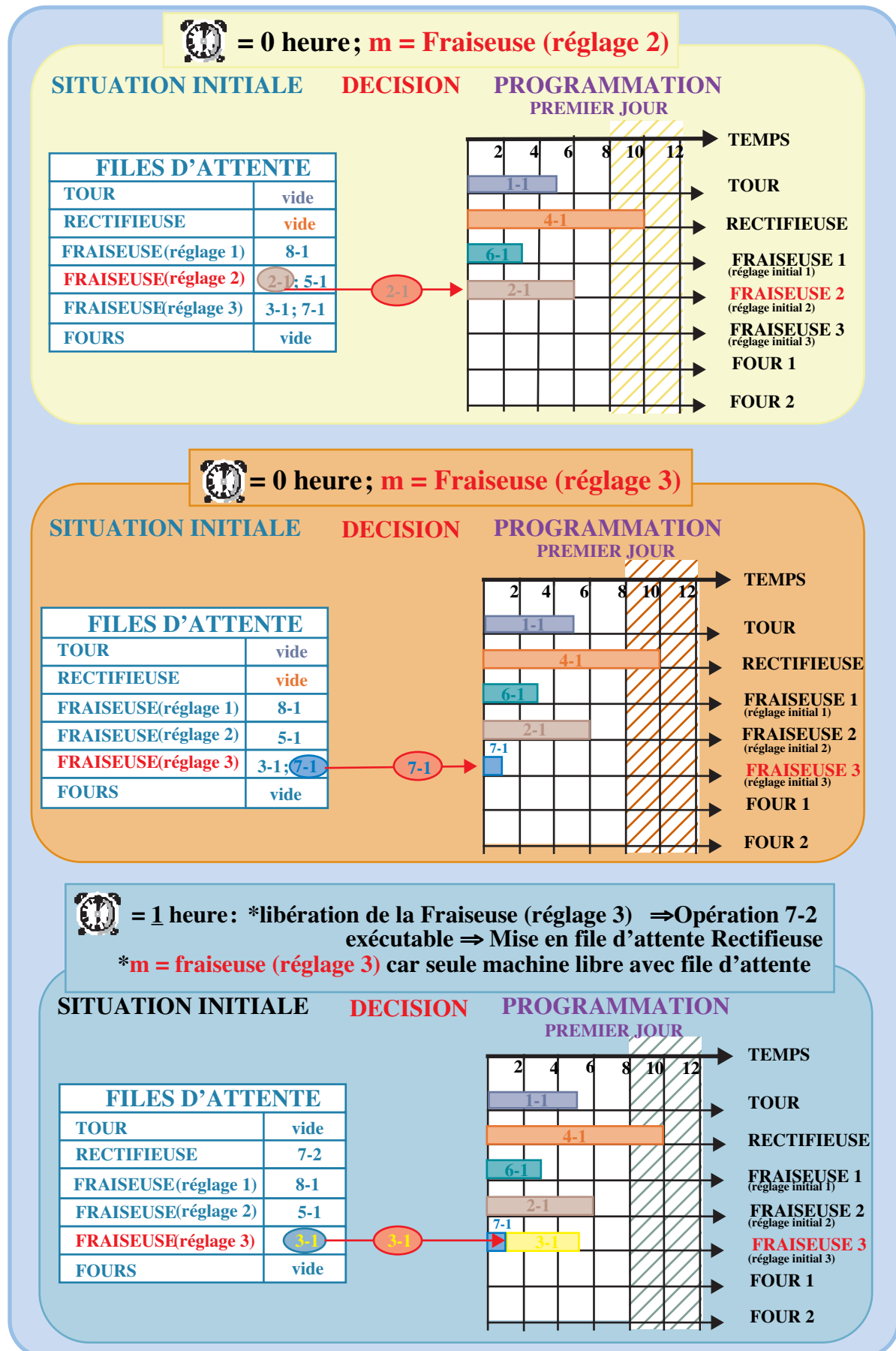


FIGURE 130
Proposition finale d'ordonnancement (cas **MÉCANO**)

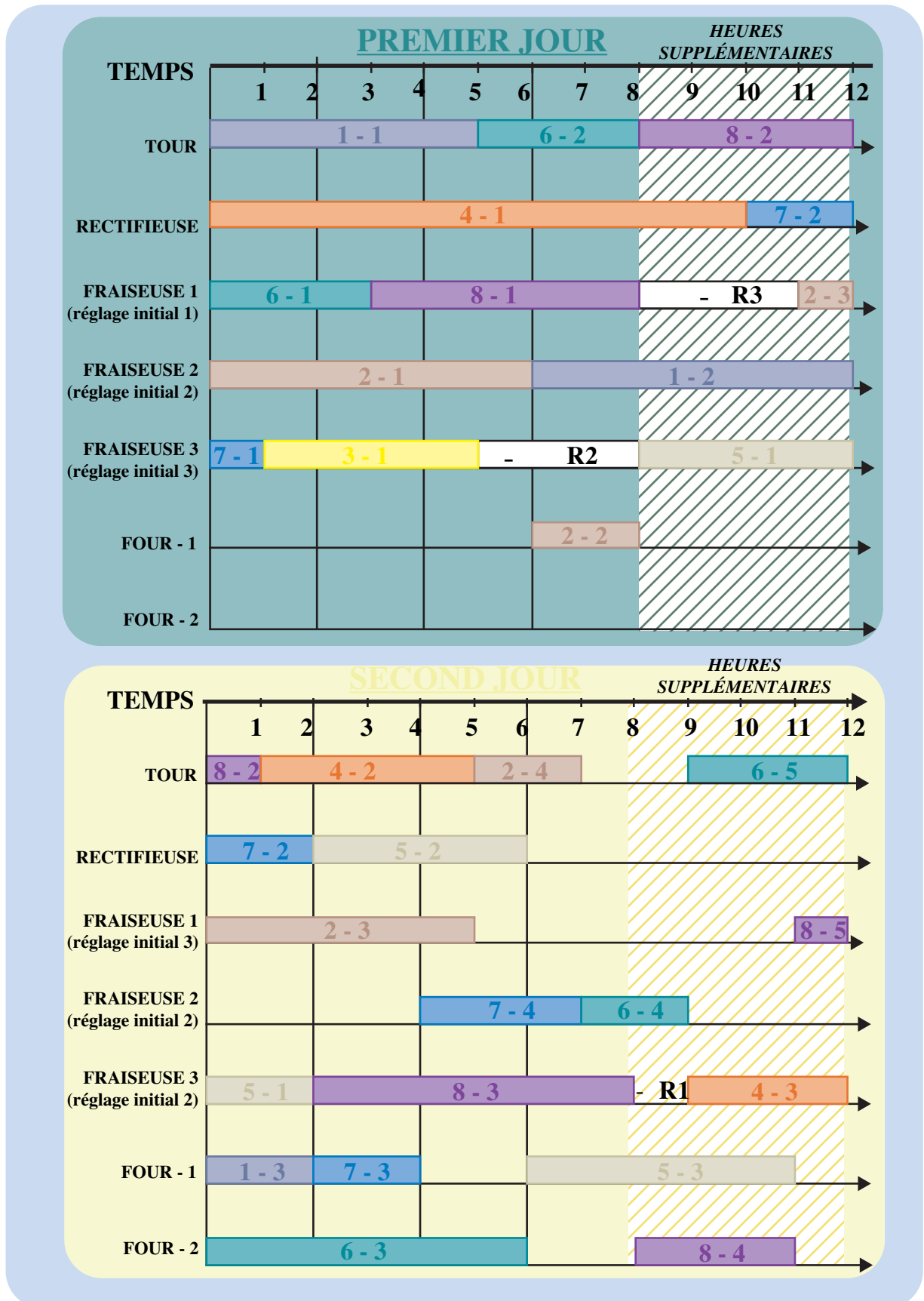


Figure 130 (suite)

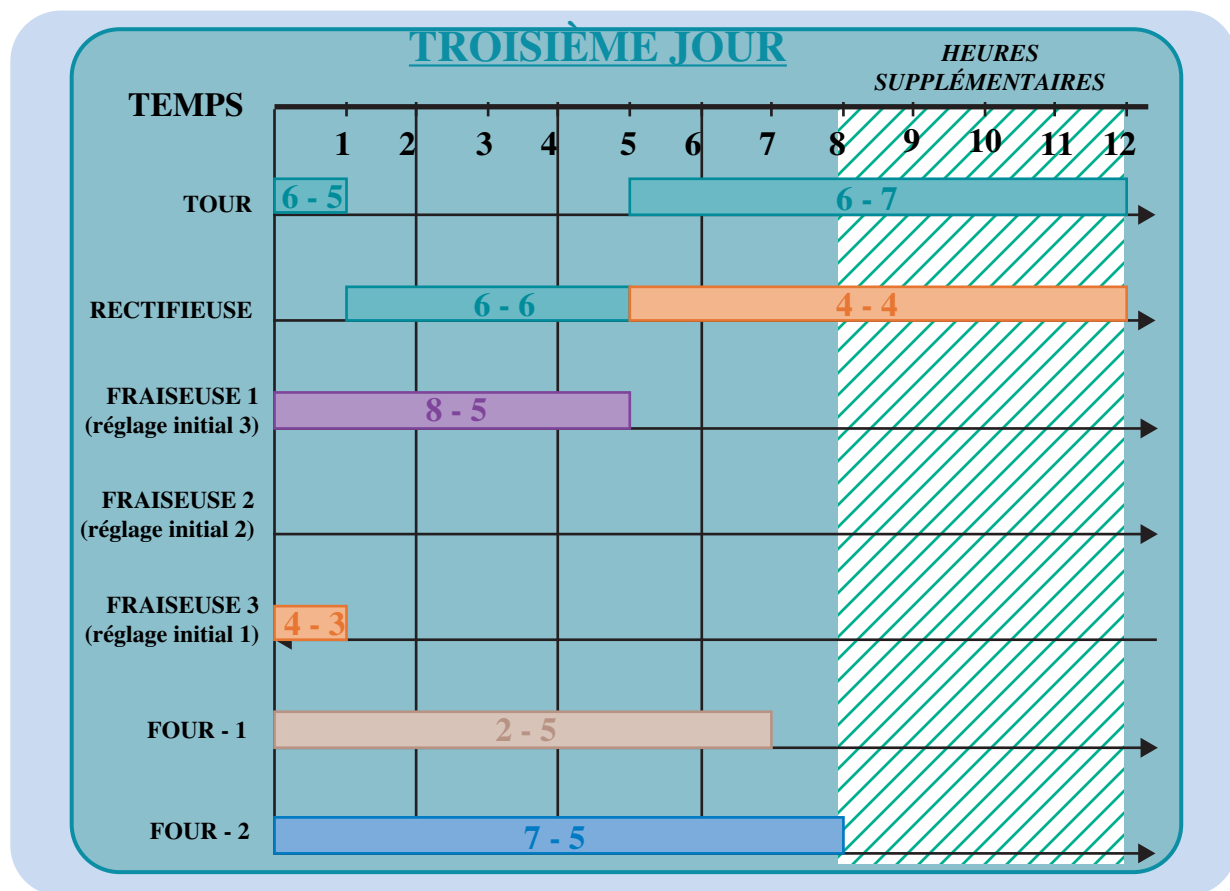


FIGURE 131

Calcul des indicateurs associés aux règles pour la file d'attente de la fraiseuse 2 à l'instant $t = 0$ et $t = 1$

$t = 0$

Tâche	PAPS Temps d'attente de la tâche en liste d'attente	TOM Temps opératoire de la tâche	Marge	S/OPN	WINQ	LWKR
3-1	0	4	9	3	0	15
7-1	0	1	14	2,8	0	18

$t = 1$

Tâche	PAPS Temps d'attente de la tâche en liste d'attente	TOM Temps opératoire de la tâche	Marge	S/OPN	WINQ	LWKR
3-1	1	4	8	2,666667	0	15

III-2 Définition d'un Système Interactif d'Aide à la Décision de Lancement (SIADL)

L'instabilité croissante de l'environnement des entreprises rend très difficile l'apprentissage des organisations par la «classique» méthode des essais et des erreurs. La recherche par tâtonnement de solutions de niveau satisfaisant exige, en effet, une stabilité suffisante des problèmes posés. Pour survivre, les entreprises sont amenées à remplacer en grande partie un *apprentissage par l'expérience* par un *apprentissage par la simulation*. C'est dans cette perspective que se situent les Systèmes Interactifs d'Aide à la Décision.

La complexité des problèmes d'ordonnancement implique une assistance informatique importante et milite en faveur de Système Interactif d'Aide à la Décision de préférence à des systèmes fermés, fournissant une solution sur laquelle l'utilisateur ne peut intervenir. De tels SIAD appliqués aux décisions de lancement doivent accomplir deux grandes fonctions :

- la première est une *capacité à fournir plusieurs solutions alternatives* caractérisées par des propositions précises d'ordonnancement et, si nécessaire, de mobilisation temporaire de ressources additionnelles ; la recherche de ces solutions doit être largement automatique tout en autorisant l'intervention du responsable de l'ordonnancement pour modifier certains choix ainsi que certaines règles utilisées dans cette recherche ;
- la seconde fonction est *une certaine capacité de conseil* pour guider le choix final ; actuellement, cette fonction n'est au mieux satisfaite que par la fourniture d'une batterie d'indicateurs physiques relatifs à l'utilisation des ressources et à l'exécution des commandes ; une réflexion complémentaire sur la création d'indicateurs synthétiques s'impose donc et sera conduite ici.

Il semblerait que la première tentative dans ce sens soit due à Godin et Jones¹, qui ont mis au point pour la Western Electric en 1969, un système permettant à un opérateur de tester l'impact de combinaisons variées d'ordonnancements possibles et d'en simuler les effets avant de sélectionner un ordonnancement définitif. C'est à partir du début des années quatre-vingt que ces approches se développent vraiment, en raison des progrès réalisés dans la micro-informatique tant au niveau des performances que de celle des interfaces graphique.

Les caractéristiques communes de tels systèmes reposent sur quelques principes apportant des réponses aux principales préoccupations des utilisateurs² : prise en compte de l'instabilité des objectifs dans la formulation du problème (§ III-2.1), définition de mécanismes de proposition automatique d'une solution d'ordonnancement (§ III-2.2, page 436), organisation de l'interactivité du système (§ III-2.3, page 440), définition de la périodicité de l'ordonnancement (§ III-2.4, page 441) et définition de règles d'évaluation des solutions alternatives d'ordonnancement (§ III-2.5, page 442).

1. Décrit dans Buffa & Miller (1979, [74], p. 562 et suivantes).

2. Ces principes ont été appliqués dans la réalisation de SIADL dans de travaux de recherches contractuelles conduits sous notre direction, avec mise au point de prototypes. Il s'agit de la thèse de Riou-Long (1983, [364]) et de celle de r Raharison (1991, [356]) qui complète le travail antérieur par un couplage avec la base de données relationnelles de production présentée au chapitre XIV et avec un système-expert pilotant des règles par des méta-règles qui tiennent compte de l'état du système pour changer de stratégie d'ordonnancement et possédait une amorce de dialogue graphique.

III-2.1 Prise en compte de l'instabilité du problème d'ordonnancement

La recherche d'un ordonnancement est toujours guidée par l'objectif que l'on privilégie dans cette définition du problème (à l'exception du cas de figure étudié au § III-2.2.1, page 437) et un ensemble de contraintes. Cet objectif et ces contraintes, comme on va le voir, sont susceptibles de varier d'un jour à l'autre mais aussi au cours de la simulation.

Les interactions entre la fonction production et les autres fonctions de l'entreprise conduisent très naturellement à des modifications du point de vue à privilégier dans le pilotage du système productif : tantôt des problèmes de trésorerie conduiront à privilégier les tâches assurant les plus fortes rentrées d'argent, tantôt des préoccupations commerciales prévaudront (respect de délais...), tantôt des goulots d'étranglement se produiront sur certains postes de travail et tout sera mis en œuvre pour rétablir une situation « normale »... Entre deux ordonnancements successifs, il est normal que les objectifs et contraintes de l'ordonnancement changent. Un logiciel d'ordonnancement doit donc offrir une souplesse de formulation suffisante pour répondre à des besoins qui peuvent varier d'un jour à l'autre.

Tout aussi important, d'un point de vue opérationnel, est le fait que la hiérarchie des points de vue à prendre en compte dans le mécanisme de construction progressive d'un ordonnancement (sauf celui décrit au § III-2.2.1, page 437) repose sur une évaluation, mise à jour au cours de cette construction, d'indicateurs associés aux différents points de vue à prendre en compte. Ce mécanisme de construction progressive sera naturellement influencé par l'évolution de ces indicateurs qui pourra modifier les points de vue que l'on pouvait avoir antérieurement dans la construction de cet ordonnancement : des tâches qui n'étaient pas prioritaires, le deviennent à force de voir l'exécution de leurs opérations différées, certaines ressources deviennent ou cessent d'être des goulots d'étranglement. Si l'analyse dynamique du degré d'urgence de tâche s'effectue facilement à travers les indicateurs associés à certaines règles (Marge et S/OPN ; voir l'exemple de la figure 131, page 431), il n'en est pas de même sur l'analyse de la **saturation d'un centre productif** (atelier spécialisé ou poste de travail).

Dans notre exemple, on tire des gammes décrites dans la figure 124, page 420, les charges totales de travail des différents centres productifs : 12 heures pour les fraiseuses en réglage 1, 30 heures pour les fraiseuses en réglage 2, 17 heures pour les fraiseuses en réglage 3, 45 heures pour les fours, 29 heures pour la rectifieuse et 30 heures pour le tour. Ce constat amène à penser que la spécialisation des fraiseuses sur leurs réglages d'origine n'est pas efficace¹ et qu'il faudra sans doute changer le réglage des fraiseuses en réglage 1 ou 3. Par ailleurs, les 45 heures de charge des fours sont à répartir sur deux fours, ce qui conduit à une charge, a priori, comparable à celles des autres machines. Il est cependant difficile, sur la base de ces seules informations, d'apprécier la saturation prévisionnelle du système productif étudié.

- Une première raison réside dans le fait qu'il a été choisi arbitrairement de proposer un ordonnancement sur 3 jours et il n'y a aucune raison pour que la

1. Sans compter qu'elle n'est pas possible si le nombre de réglages est supérieur au nombre de machines.

charge du travail lancé en production coïncide avec la capacité offerte sur l'horizon d'ordonnancement retenu.

- Ensuite, les arrivées de travail sur un centre productif donné peuvent n'être pas régulières dans le temps. Il s'ensuit qu'un centre productif peut sembler peu critique parce que sa charge de travail est comparativement faible mais si celle-ci arrive massivement en fin de parcours, ce poste peut s'avérer plus critique que les autres, lorsque le travail commence à lui parvenir.

Pour répondre à cette dernière remarque, il faut apprécier la saturation autrement qu'au travers d'indicateurs simples. La saturation prévisionnelle d'un centre peut s'apprécier en comparant la courbe cumulée du travail disponible, avec une courbe cumulée de travail effectué. La courbe réelle de charge de travail disponible dépend de l'ordonnancement retenu mais on peut obtenir la courbe au plus tôt en faisant l'hypothèse que toutes les opérations antérieures à une opération d'un ordre de fabrication à exécuter sur ce centre productif ont toutes pu être ordonnancées au plus tôt (ce qui revient à considérer comme infinie la capacité de production des autres centres productifs). Ce calcul est effectué dans le [tableau 102](#), pour le cas **MÉCANO**. La [figure 132](#), [page 436](#), visualise l'évolution de la

TABLEAU 102
Détermination des charges cumulées disponibles au plus tôt

Machine(s)	Opération	Temps opératoire	Date t de disponibilité	Travail cumulé disponible en t	Machine(s)	Opération	Temps opératoire	Date t de disponibilité	Travail cumulé disponible en t
Fraiseuse (réglage 1)	6-1	3	0	3	Four	3-2	10	4	10
	8-1	5	0	8		7-3	2	5	12
	4-3	4	14	12		2-2	2	6	14
Fraiseuse (réglage 2)	2-1	6	0	6		6-3	6	6	20
	5-1	6	0	12		5-3	5	10	25
	1-2	6	5	18		7-5	8	10	33
	7-4	3	7	21		1-3	2	11	35
	8-3	6	10	27		2-5	7	16	42
	6-4	2	12	29		8-4	3	16	45
	3-3	1	14	30	Rectifieuse	4-1	10	0	10
Fraiseuse (réglage 3)	3-1	4	0	4		7-2	4	1	14
	7-1	1	0	5		5-2	4	6	18
	2-3	6	8	11		4-4	7	18	25
	8-5	6	19	17		6-6	4	18	29
Tour	1-1	5	0	5	Tour	2-4	2	14	19
	6-2	3	3	8		6-5	4	14	23
	8-2	5	5	13		6-7	7	22	30
	4-2	4	10	17					

Table des
matières

Index
thématique

charge de travail au plus tôt pour quelques postes de travail. L'ordonnancement réalisé ne pouvant que retarder l'arrivée de certaines tâches, amènera la courbe réelle à se déformer vers la droite. La courbe du travail cumulé au plus tôt, mise pointillés, est une ligne brisée composée de segments obliques, correspondant au fait qu'une tâche est en cours d'exécution, et (éventuellement) de segments horizontaux, correspondant à une inactivité du centre en raison d'une rupture de charge de travail¹. L'analyse conjointe de ces deux courbes permet d'affiner le concept de saturation prévisionnelle :

- l'existence de paliers implique, une possibilité de retard de certaines tâches en amont, sans provoquer de rupture de charge,
- une distance très faible entre les deux courbes à certaines dates implique que le retard de certaines tâches en amont peut amener une rupture de charge,
- l'absence totale de palier dans la courbe de production cumulée indique, si la charge de travail va jusqu'à la fin de l'horizon retenu, que la ressource considérée est un goulot d'étranglement et que l'on a intérêt à ce que les décisions d'ordonnancement ne viennent pas induire de ruptures de charge sur ce centre de production.

Ces courbes (établies automatiquement dans le logiciel) se peuvent se déformer entre deux décisions consécutives d'ordonnancement dès lors que plusieurs opérations étaient candidates pour passer sur le même centre de production².

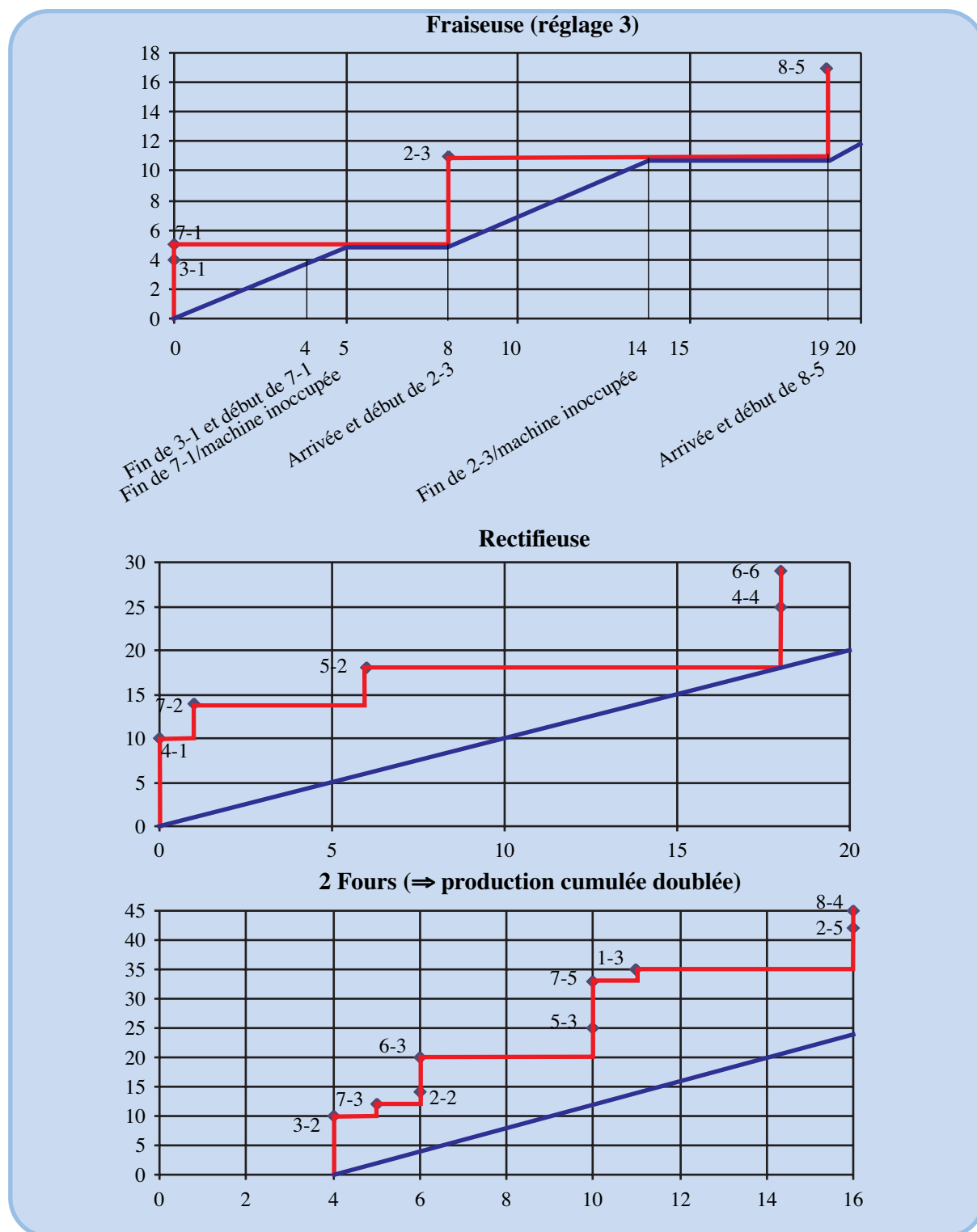
Pour en finir avec les raisons qui peuvent conduire à une modification du problème d'ordonnancement au cours de sa résolution, il faut intégrer le fait que la formulation initiale est une simplification d'un problème complexe, ce qui implique que :

- le fait d'avoir privilégié un point de vue n'autorise pas pour autant de retenir une solution où d'autres points de vue seraient trop « malmenés » ; dès lors, il est tout à fait admissible que l'on modifie la hiérarchie des points de vue en cours de résolution du problème (le plus souvent pour la partie du problème restant à résoudre) ;
- rien ne garantit que la formulation initiale du problème soit cohérente, dans la mesure où rien ne garantit a priori qu'il existe une solution respectant l'ensemble des contraintes à prendre en compte (dates de livraison, disponibilité des ressources...) ; dès lors, il est tout à fait normal que l'on soit amené à modifier certaines contraintes (dates de livraison, par exemple) en cours de résolution du problème ;
- la solution trouvée peut satisfaire l'ensemble des contraintes explicitement retenues et s'avérer inacceptable en pratique parce qu'elle viole des contraintes (comme, par exemple, la limite physique imposée à la file d'attente devant un poste de travail) qui ont été omises dans la formulation simplifiée (on parle de **relaxation de contraintes**) parce que l'on espérait

1. On peut observer deux paliers pour la fraiseuse en réglage 3, correspondant à une rupture d'alimentation : on dispose d'une marge indépendante (au sens donné en ordonnancement de projet) de 3 heures pour les opérations 3-1 et 7-1 et d'une marge totale de $3 + 5 = 8$ heures pour l'opération 8-5.

2. En effet, la date théorique d'exécution de l'opération non retenue, qui était la même que celle de l'opération retenue (avec l'hypothèse implicite de capacité infinie pour calculer les courbes de charges cumulées au plus tôt), est retardée, avec les incidences que cela implique pour les opérations suivantes de la même tâche.

FIGURE 132
Charges cumulées au plus tôt et production cumulée au plus tôt

Table des
matièresIndex
thématique

qu'elles seraient respectées par la solution obtenue ; dès lors, il faudra recommencer le processus de recherche, en intégrant de nouvelles contraintes.

III-2.2 Mécanismes de proposition automatique d'un ordonnancement

Le problème à résoudre est celui de l'établissement d'une proposition d'ordonnancement (éventuellement modifiable, comme on le verra au § III-2.3, page 440)

pour exécuter un ensemble réel de tâches (ou commandes ou ordres de fabrication). Pour ce faire, trois familles d'approche sont envisageables : la première est orientée vers la recherche d'une organisation de la réactivité aux aléas par la détermination d'ensembles de solutions alternatives respectant les contraintes du problème (§ III-2.2.1), la seconde est orientée vers l'utilisation d'heuristiques (§ III-2.2.2, page 437).

III-2.2.1 Recherche d'un ensemble d'ordonnancements admissibles

Une approche originale, amorcée à la fin des années soixante-dix¹ vise à organiser la réaction en temps réel aux inévitables incidents qui viendront perturber la mise en œuvre d'un ordonnancement. Il s'agit alors, non pas de rechercher de ce que l'on pourrait considérer comme étant la « bonne solution », mais de trouver un ensemble d'**ordonnancements admissibles**, c'est-à-dire de solutions d'ordonnancement respectant les contraintes de dates de livraison et celles de disponibilité de ressources. Cette approche ne se situe donc pas exactement dans la lignée de celle développée précédemment et qui visait à proposer un bon ordonnancement (mais avec une approche compatible avec une certaine réactivité en temps réel, aux aléas).

Concrètement, on recherche, pour chaque ressource, une **séquence de groupes d'opérations permutables** qui sont tels que les contraintes seront satisfaites quel que soit l'ordre d'exécution des opérations de chaque groupe, étant entendu que l'on n'exécute pas d'opération d'un groupe tant que toutes celles du groupe précédent n'ont pas été achevées. L'opérateur dispose donc d'indications souples qui, par rapport à un ordonnancement unique, lui permettent de faire face à des aléas en sachant qu'il respectera les contraintes initiales. Certaines perturbations importantes peuvent amener à prendre localement une décision n'appartenant pas à la séquence de groupe d'opérations permutables en cours et une aide peut être apportée pour choisir une solution permettant de retrouver rapidement un ensemble de solution admissibles, sur la base de l'état d'avancement observé sur l'ensemble du système productif au moment de la prise de décision.

Cette organisation de la réactivité est un atout très important de la démarche. Par contre, en ne visant qu'à respecter des contraintes, cette approche considère tous les ordonnancements admissibles comme équivalents. Cette position est d'autant plus acceptable que le problème est fortement contraint, c'est-à-dire que les degrés de liberté sont faibles. Si le problème est trop contraint, il n'y aura pas de solution admissible et cette approche se prêtera assez mal à une négociation des contraintes. Si le problème n'est pas assez contraint, il sera assez contestable de considérer tous les ordonnancements admissibles comme équivalents.

III-2.2.2 Utilisation d'heuristiques

La seconde approche consiste à proposer un ordonnancement en faisant appel à l'une des approches décrites au § III-1.2, page 421 (placement progressif d'ordre

1. Ces travaux de recherche ont été conduits au LAAS (Laboratoire d'Automatique et d'Analyse des Systèmes du CNRS). Les nombreux travaux qui alimentent ce courant théorique sont décrits au chapitre V (approche par propagation de contraintes) et au chapitre XIII de l'ouvrage coordonné par Lopez et Roubellat (2001, [286]). Un logiciel utilisé par plus d'une soixantaine d'entreprises exploite cette approche avec succès.

de fabrication), de préférence celles décrites au § III-1.2.2, page 424 (placement chronologiquement progressif d'opérations exécutables).

Le choix de règles à retenir peut s'appuyer, au départ, sur les enseignements généraux tirés de la simulation de systèmes fictifs (cf. § II-2.2, page 411) mais on a vu que la performance des règles dépendait de la configuration de l'atelier où le système doit être implanté, des caractéristiques des flux rentrant dans ce système et des objectifs assignés au pilotage du système. Il sera alors nécessaire, au départ, de procéder à un certain nombre de simulations pour étalonner les performances des heuristiques utilisées, soit à partir de jeux de données observées, soit à partir de données générées aléatoirement et dont les caractéristiques sont définies à partir des stabilités constatées sur le passé. Les conclusions tirées ne devront pas être définitives : il est vivement conseillé de refaire ce type d'analyse comparative de performances des règles, lorsque l'évolution de certaines caractéristiques de la demande ou du système productif risque d'invalidier les performances relatives trouvées précédemment.

Une règle ayant tendance à privilégier un point de vue (le point de vue commercial ou celui de la production), des travaux visant à concilier des points de vue antagonistes ont été conduits dès la fin des années soixante. L'ordonnancement s'effectue alors sur la base d'un indicateur calculé comme une moyenne pondérée de différents indicateurs retenus dans la définition des règles élémentaires¹. La légère supériorité de cette approche se fonde sur des études rétrospectives qui ne permettent pas de garantir le maintien de cet avantage sur de nouvelles données. Cette approche a donc été abandonnée au profit d'une utilisation astucieuse d'un ensemble de règles pour résoudre des problèmes réels d'ordonnancement.

On a vu à la page 415 qu'il n'est pas forcément judicieux de travailler avec la même règle. Cette remarque faite dans les travaux pionniers des années soixante peut être généralisée dans l'espace et le temps. Il ne semble pas judicieux :

- d'utiliser les mêmes règles pour tous les postes de travail à un moment donné, parce que les problèmes ont peu de chance d'être identiques dans tout le système productif (certains postes sont plus « critiques » que d'autres parce qu'ils sont davantage sollicités comparativement et constituent, temporairement ou non, des goulots d'étranglement) ;
- d'utiliser la même règle au cours du temps pour un même poste de travail parce que la saturation de ce poste pourra varier au cours du temps et que les caractéristiques « moyennes » des tâches à traiter (degré d'urgence...) peuvent varier de manière significative.

Si ces remarques paraissent assez évidentes, leur traduction opérationnelle est loin d'être facile. Les premiers travaux entrepris sérieusement dans cette direction sont dus à Holloway et Nelson (1973, [233]), qui ont combiné implicitement une

1. Conway, Maxwell et Miller testent en 1963-1965 une moyenne pondérée de deux règles utilisant la même unité (le temps) et montrent que les performances obtenues sont meilleures mais que les résultats sont très sensibles au choix des coefficients de pondération. Plus tard, Hershauer et Ebert (1975, [229]) généralisent la démarche et proposent une somme pondérée d'un nombre quelconque d'indicateurs obtenus en application de règles élémentaires, où les coefficients de pondération varient entre - 1 et + 1 et où les valeurs prises par les différents indicateurs sont « normées » afin que l'hétérogénéité des échelles n'induisse pas de distorsion. Ultérieurement Cliffe et Mac Mannus (1980, [97]) poursuivent dans cette voie, sans apporter de preuves déterminantes de l'intérêt de ce surcroît de sophistication.

table de décision et l'utilisation de plusieurs règles sélectionnées par la table de décision. Cette approche amène les remarques suivantes :

- Si l'intérêt de cette modulation des règles dans l'espace et dans le temps semble assez évident, il reste à choisir ces règles, les moments et les endroits où elles peuvent être utilisées et à apprécier leur influence sur le comportement du système. La difficulté de la mise au point de ce pilotage fin, est atténuée s'il est possible de créer et modifier facilement une base de règles composites s'appuyant sur des indicateurs sur l'état du système au moment de la prise de décision et utilisant de manière conditionnelle plusieurs règles élémentaires¹, ce qui suppose de pouvoir tester commodément l'impact de nouvelles règles sur des jeux de données représentatifs. Un arbitrage peu évident doit être opéré ensuite entre l'application automatique de règles complexes relevant d'une approche de type système-expert (voir ci-après) et une approche dans laquelle on rend la main à un opérateur chaque fois que le problème à résoudre est considéré comme non trivial. Dans ce dernier cas, on est confronté à deux problèmes : celui de l'expertise de l'opérateur à traiter des cas complexes et celui du déclenchement judicieux de son intervention. L'introduction de cette approche dans un site productif implique un investissement qui dépasse celui de la simple acquisition d'un logiciel.
- L'usage d'heuristiques n'exclut pas la possibilité d'appliquer localement la solution optimale à un problème de dimension restreinte (centre de production doté de machines identiques, poste de production critique, flux unidirectionnels sur quelques postes de travail, non indépendance des temps de lancement des opérations sur une machine critique...). Dans cette perspective, on pourra utiliser les résultats de travaux évoqués en [section I](#).
- Une difficulté inhérente à ce type d'approche tient à l'hypothèse implicite d'indépendance des tâches à exécuter. Lorsque les tâches sont des ordres de fabrication (OF) établis dans le cadre d'un processus de MRP (voir [chapitre VI, § IV-3, page 496](#)), on ne rencontre pas de problème particulier si les dates de livraison sont respectées. Dans le cas contraire, le retard de certains OF peut empêcher l'exécution d'autres OF alors que le centre de production est disponible, en raison de l'indisponibilité de certains composants et rendre sans intérêt le respect de la date de livraison d'autres OF². S'il est assez facile de prendre en compte dans le logiciel la mise à jour de dates de livraison de certains OF consécutivement au retard d'un autre OF, il est plus difficile de prendre en compte ces interdépendances dans la gestion des heuristiques.

L'usage possible de systèmes-experts pour résoudre des problèmes d'ordonnancement mérite quelques commentaires. Les premières applications des

1. Par exemple, en supposant que l'on ait défini un indicateur synthétique de la saturation prévisionnelle d'un centre de production (sur la base des réflexions conduites en [page 433](#)), on peut imaginer d'utiliser la règle suivante qui combine des indicateurs sur l'état du système productif et des indicateurs sur les tâches à exécuter : « *si* plusieurs opérations candidates à l'utilisation d'une machine venant de se libérer et ayant une valeur positive du critère S/OPN inférieure à 2 heures, *alors si* la charge de travail sur la machine suivante est supérieure à 5 heures *alors* utiliser le critère WINQ pour départager ce sous-ensemble de candidats, *sinon* retenir l'opération suivie d'une opération à exécuter sur le centre de production ayant l'indicateur de saturation prévisionnel le plus élevé »

2. L'impossibilité de respecter une date de livraison pour un OF d'une référence de niveau i peut déclencher mécaniquement un retard pour d'autres OF de référence de niveau $i - 1$ (ou plus en cascade) et, ce faisant, rendre inutile le respect de la date de livraison d'autres OF portant sur des références de niveau i utilisées également pour les références de niveau $i - 1$ sus-visées.

systèmes-experts¹ dans la résolution de problème d'ordonnancement remontent au début des années 1980 : une équipe de la Carnegie Mellon University, sous la conduite de Fox explore l'approche « Intelligence Artificielle » dans le cadre du projet ISIS², suivi d'autres équipes de recherche un peu partout dans le monde (en France notamment avec les travaux du CERT aboutissant au logiciel OPAL). Il est préférable de considérer le système-expert comme une composante importante mais pas exclusive d'un système s'appuyant également sur une base de données (décrivant le système productif, son utilisation prévisionnelle, les gammes et nomenclatures, ainsi que les commandes à exécuter) et un programme élémentaire de simulation, chargé d'exécuter les décisions prises par le système-expert. En effet, les tentatives du « tout système-expert » ont très rapidement montré leurs limites. L'intérêt principal du système-expert est de permettre une modulation facile dans l'espace et dans le temps des heuristiques à utiliser : des règles de niveau haut (méta-règles) surveillent l'évolution de l'état du système et de l'exécution des OF et provoquent, si nécessaire, le basculement d'un ensemble de règles de décision à un autre ensemble jugé plus pertinent dans le nouveau contexte.

III-2.3 Système interactif

L'instabilité des problèmes et le pilotage des heuristiques nécessitent l'usage d'une interface graphique permettant à la fois une compréhension rapide de problèmes et de propositions de solutions alternatives et une facilité de prise en compte de décisions prises.

L'usage de tels SIAD repose sur l'idée que l'usage d'heuristiques et de systèmes-experts doit être considéré comme une amélioration des procédures de tâtonnement utilisées par les agents de planning à la recherche d'une solution de niveau satisfaisant mais qu'il n'en reste pas moins nécessaire d'éviter tout automatisme en la matière en permettant au responsable de modifier ponctuellement certains arbitrages pour améliorer la proposition d'ordonnancement (l'analyse de ces modifications pouvant, du reste, conduire à de nouvelles heuristiques). En 1978, une expérience conduite par Hurriion³ montrait que l'utilisation de simulation interactive avec aide graphique donnait des résultats meilleurs que ceux obtenus par un traitement global et en temps différé du problème par les principaux logiciels industriels. Depuis, un certain nombre de travaux ont confirmé l'intérêt de cette vision. Pour tirer partie de cette approche, de nombreuses idées

1. *Note de la page précédente.* Le principal atout de cette approche est de permettre de travailler avec une liste de règles du type « *si* condition satisfaite *alors* conclusion vraie », d'autant plus facile à mettre à jour qu'une adjonction ou une suppression de règle ne nécessite pas de modification des programmes existants. Une condition peut être relative à l'état actuel ou prévisionnel de ressource(s) ou d'ordre(s) de fabrication. Une conclusion peut être la modification de l'état actuel ou prévisionnel de ressource(s) ou d'ordre(s) de fabrication, consécutive ou non à la réalisation d'un calcul ou d'une prise de décision.
2. Voir Fox, Allen, Smith & Strohm (1983, [158]). Voir également le numéro spécial consacré à l'Intelligence Artificielle en gestion de production de la revue *International Journal of Production Research* (volume XXVI, n° 5, mai 1988).
3. Voir Hurriion (1978, [236]). Ce travail académique reposait sur une vision assez simple de la réalité : les heuristiques se limitent à quelques règles locales, système productif simple (pas de machines identiques, pas de contraintes de non-interruption de certaines tâches, temps continu...), utilisation d'un seul critère d'évaluation des performances, absence de rétroaction de la résolution sur la définition des problèmes. Par contre, le pilotage de chaque machine était indépendant de celui des autres machines et l'opérateur, qui voyait s'afficher progressivement les propositions d'ordonnancement, pouvait intervenir à tout instant pour imposer son choix.

sont exploitables pour obtenir une bonne synergie entre le système et son utilisateur :

- proposition de solutions sous forme de graphiques de Gantt, affiché sur un moniteur graphique « haute définition » de grand format avec possibilité de déplacement facile (scrolling) dans toutes les directions ;
- utilisation de la couleur pour assurer une meilleure perception des informations à privilégier à un moment donné : par exemple, utilisation d'une couleur choisie par l'opérateur lui permet de mieux visualiser la progression prévisionnelle d'une tâche donnée ou, encore, pour faciliter la détection de goulot d'étranglement, coloration automatique du graphique Gantt, différenciée en fonction de l'importance de la file d'attente, devant chaque poste de travail, à chaque période ;
- pointage par souris d'une opération programmée :
 - pour demander des informations sur l'état prévisionnel du système à ce moment (usage de graphiques et d'indicateurs de synthèse),
 - pour demander une explicitation du raisonnement ayant conduit à la programmation de cette opération de préférence aux autres,
 - pour modifier la décision et relancer la simulation,
 - etc. ;
- représentations de l'évolution prévisionnelle de caractéristique du système productif (saturation, par exemple) ou de la production (retards, par exemple)
- inclusion de mécanismes de capitalisation d'expérience pour faire évoluer la base des règles (voir les remarques faites sur ce point à la [page 439](#)) ; c'est très certainement le problème le plus difficile à résoudre et l'un des grands enjeux des années à venir.

Table des
matières

Index
thématique

III-2.4 Périodicité et horizon de l'ordonnancement

Si l'on n'a pas adopté la démarche consistant à organiser la réactivité par l'établissement d'un ensemble d'ordonnements admissibles (§ III-2.2.1, [page 437](#)), on est confronté au fait que les aléas rencontrés dans l'exécution d'un ordonnancement rendent peu probable son respect au-delà d'un (ou, au maximum, deux jours). Pour cette raison, il est souhaitable de régénérer la proposition tous les jours (ou, au maximum, tous les deux jours), pour tenir compte de l'exécution effective de l'ordonnancement et de l'arrivée de nouveaux OF. On obtient alors un « ordonnancement glissant » dont la philosophie est voisine de celle qui sous-tend les techniques de planification glissante de type MRP.

Cette périodicité relativement élevée du recalcul de l'ordonnancement n'implique pas qu'il soit inutile de proposer un ordonnancement au-delà de ce qui a des chances raisonnables d'être réalisé. En effet, une programmation sur plusieurs jours permet d'obtenir une visibilité de certains problèmes à venir que l'on ne peut pas déceler à temps avec un « comportement myope ». Cet éclairage de l'avenir moins immédiat permet éventuellement d'effectuer des arbitrages différents pour des décisions qui ont toutes les chances d'être exécutées et, parfois même, prendre des décisions qui ne semblent pas se justifier sur le très court terme (modification de réglage de machine, appel à des heures supplémentaires...). Cet horizon d'ordonnancement sera donc supérieur à l'intervalle de temps séparant deux régénérations successives d'ordonnancement. Cela étant, il est inutile de

l'allonger au point que le système productif soit globalement en sous-charge au cours des dernières périodes d'ordonnancement.

Cela étant, l'occurrence d'aléas pose la question de savoir ce qu'il convient de faire sur le terrain (par exemple, comment prendre en compte une nouvelle commande urgente ou faut-il inverser le séquençement de deux opérations sur une machine, en raison du retard pris par la première des deux tâches concernées). Laisser les opérationnels sans instruction, c'est courir le risque d'une amplification de l'impact d'un aléa. On peut alors recourir à une démarche proposée par Holloway, Nelson et Suraphongschai (1979, [233]). Parmi les méthodes heuristiques qu'ils ont testées, celle qui donne les meilleurs résultats consiste à créer deux listes de priorité: la première est générée lors du calcul centralisé de l'ordonnancement et la seconde est générée localement lors des arrivées des tâches nouvelles ou d'aléas. Lorsqu'une machine se libère, l'arbitrage entre les deux tâches qui ont la plus forte priorité dans chacune des deux listes, se fait sur la base de la plus faible marge disponible (différence entre le temps restant avant la date de livraison et le cumul des temps opératoires des opérations restant à faire), ce qui ne conduit pas systématiquement à une utilisation de la machine (cas d'une tâche de la première liste, non encore arrivée à la machine au moment où le calcul est fait, et ayant la plus faible marge disponible). Il ne s'agit là, bien entendu que d'un exemple de solutions utilisables pour faire face aux aléas. Cela étant, la tendance d'un suivi en temps réel de la production et les progrès des moyens informatiques mobilisés permettent de relancer beaucoup plus fréquemment les calculs d'ordonnancement.

III-2.5 Critères d'évaluation de l'ordonnancement

L'évaluation économique d'un ordonnancement s'impose à terme pour les sites de production de biens ou services où le respect des délais n'est pas le critère exclusif de jugement et qui sont dotés de systèmes offrant à leurs responsables de véritables alternatives décisionnelles. Dans les autres cas de figure, ce qui suit ne présente qu'un intérêt académique. Nous examinerons tout d'abord la problématique des critères de jugement et les fondements d'un critère économique (§ III-2.5.1), avant d'appliquer ces principes sur notre exemple numérique (§ III-2.5.2, page 447).

III-2.5.1 La problématique des critères d'évaluation

III-2.5.1.1 Les indicateurs traditionnels

On a vu que la formulation du problème d'ordonnancement était presque toujours simplifiée. Dès lors, il est indispensable que le décideur ait connaissance de toutes les conséquences des choix proposés. Cela implique qu'en plus du planning prévisionnel sous forme de Gantt et de tableaux, un certain nombre d'indicateurs soient disponibles pour faciliter la détection d'éventuels problèmes ainsi que la recherche d'améliorations de la solution.

Les indicateurs recensés dans la «littérature» ne sont pas tous suffisants pour décrire toutes les conséquences des décisions à prendre et leur pertinence dépend du type d'usage qui en est fait. Certains indicateurs ont pour vocation de mesurer des performances «globales» en régime de croisière (durée moyenne d'achèvement des tâches par exemple) et sont sans signification sur courte période. Leur

intérêt est autre : ils permettent de comparer les performances d'heuristiques dans des simulations périodiques de grande envergure. Par contre, les indicateurs associés à des ressources individuelles ou des OF particuliers (taux d'occupation des postes de travail, temps perdus en réglage, indicateur d'avancement des différentes tâches...) qui sont délaissés dans la littérature, présentent un intérêt évident pour les opérationnels.

Il est cependant insuffisant de s'en tenir là. La multiplicité des informations fournies par les indicateurs portant sur des données physiques rend difficile tout jugement synthétique. L'introduction d'un indicateur économique donne, de ce point de vue, un éclairage complémentaire dont l'intérêt est indéniable.

III-2.5.1.2 Les bases d'un indicateur d'évaluation économique

Les approches traditionnelles de la comptabilité de gestion sont inefficaces pour résoudre le problème posé. On fera appel à un mécano de quatre concepts (introduits au § II-1.1, page 74, du chapitre I) pour définir un indicateur économique d'évaluation globale d'une décision complexe d'ordonnancement et de mobilisation temporaire de ressources additionnelles, qui soit utilisable de manière pertinente dans la comparaison de décisions alternatives et facilement acceptables par les gestionnaires.

III-2.5.12a) Le concept de marge

La notion de marge, différence entre un cumul de produits et un cumul de charges est ancienne¹ et a été successivement appliquée au concept de charges directes et à celui de charges variables pour mesurer la contribution d'une production à la couverture de charges non prises en compte et à la sécrétion de l'excédent brut d'exploitation. L'indicateur économique d'évaluation de la décision d'ordonnancement doit être une marge reflétant les conséquences des décisions prises, tant au niveau des charges que des produits. Il s'ensuit que plus grande sera la valeur de cet indicateur, meilleure sera la décision. La légitimité de cet indicateur aux yeux des gestionnaires repose sur le fait (voir § III-2.5.12e, page 446) que si l'on se situe dans le cadre fictif d'une programmation en univers certain, couvrant un exercice budgétaire, il suffirait de retrancher à cette marge les charges non prises en compte (lesquelles sont indépendantes des décisions à évaluer) et d'effectuer quelques corrections mineures, pour retrouver l'excédent brut d'exploitation de l'exercice.

Le choix de cette caractéristique amène deux commentaires sur certains problèmes qui peuvent surgir lors de son usage.

- La détermination d'une marge implique la connaissance de recettes. De nombreux sites industriels travaillent pour d'autres entités d'un même groupe, ce qui implique qu'il n'y a pas de recette associée à l'exécution d'ordre de fabrication. Le problème posé est alors celui beaucoup plus général des prix de cession dont la solution, moyennant certaines précautions méthodologiques, permet de se ramener au problème précédent.

1. Voir Johnson & Kaplan (1987, [247]) et Berliner & Brimson (1988, [47]), pour une mise en perspective historique de ces concepts; voir également l'article de Giard, *Une comptabilité de gestion en crise*, dans l'ouvrage collectif d'ECOSIP (1993, [135]).

- Il n'est pas gênant d'intégrer dans ce calcul, des charges indépendantes de ces décisions à évaluer, à condition que ces charges soient identiques pour toutes les alternatives étudiées, puisque la comparaison d'alternatives passera par l'analyse d'une différence de marges (on reviendra sur ce point au § III-2.5.12c, page 445).

III-2.5.12b) Distinction entre charges fixes et charges variables

La distinction économique entre charges fixes (ou de structure) et charges variables (ou opérationnelles) avec le volume d'activité, remonte au XIXe siècle mais son utilisation en gestion est plus récente. La notion de variabilité se réfère au niveau d'activité. Celle de charges fixes s'apprécie à la fois en référence à une période au cours de laquelle une production doit être réalisée et à une plage d'activité.

- La référence implicite au découpage temporel retenu s'explique par le fait que plus la période de production considérée est grande, plus il est facile d'adapter le niveau des ressources (principalement hommes et machines) à celui juste requis par le volume de production désiré pour cette période de production.
- La référence implicite à une plage d'activité s'explique par le fait que les charges fixes varient de manière discontinue. Un «palier» de charges fixes permet donc une production susceptible de varier dans une certaine plage de volume d'activité (qu'il n'est du reste pas toujours aisé de déterminer avec précision).

Les applications de ce concept restent assez limitées :

- le raisonnement bien connu du point mort remonte¹ aux années 1903-1904 et constitue un premier modèle permettant d'analyser l'incidence de la séparation des charges fixes et variables sur la rentabilité de l'entreprise ;
- l'utilisation systématique de cette distinction s'est répandue à partir des années cinquante, dans le cadre de la méthode des coûts variables (ou *direct costing*).

La notion de marge, introduite dans l'approche des coûts directs, s'adapte sans difficulté à l'approche du coût variable : dans une logique de point mort généralisé, le cumul des marges sur coûts variables des différents produits contribue à la couverture des charges fixes et à la constitution d'un excédent d'exploitation. La principale difficulté rencontrée dans la mise en œuvre de la méthode du coût variable réside dans la dissociation des charges fixes et des charges variables.

Dans le problème de l'évaluation d'un ordonnancement sur un horizon de quelques jours, on peut considérer comme charges fixes les charges que l'entreprise supportera si elle décide de ne rien produire. Dans cette optique :

- le personnel présent en heures normales constitue à l'évidence une charge fixe sur l'horizon retenu (que l'on peut déterminer en faisant appel à la technique classique d'abonnement de charges) ;
- les charges dont l'importance découle directement de la programmation retenue, comme la consommation de fluides ou le coût des heures supplé-

1. Voir sur ces points, l'excellent article de Kaplan (1984, [254]).

- mentaires, seront considérées comme variables puisque leur montant varie bien avec la production décidée;
- l'amortissement des équipements sera considéré comme une charge fixe de période ou comme une charge variable (ou comme un panachage des deux solutions), selon la conception retenue par les spécialistes de comptabilité de gestion de l'entreprise.

III-2.5.12c) Le concept de charge différentielle

D'autres implications du concept de charge fixe ont été analysées par Clark en 1923 [95], notamment avec l'introduction de la notion de **charge différentielle** ou **charge incrémentale**, qui associe à la mise en œuvre d'une décision, la variation de charge qui en découle. Historiquement, ce concept a présenté un intérêt tout particulier dans l'analyse des décisions relatives à l'investissement (dans l'établissement de bilans différentiels) ou de désinvestissement, ce dernier cas ayant conduit à la notion de charge écludable.

On a indiqué que l'indicateur économique retenu pour synthétiser l'ensemble des conséquences d'une décision complexe d'ordonnancement et de mobilisation temporaire de ressources additionnelles s'analysait comme une marge. La logique différentielle implique que la comparaison de deux décisions complexes alternatives passe par l'analyse de la différence de leurs marges associées et que celle-ci ne doit intégrer que les conséquences de l'usage des degrés de liberté laissés au décideur, tant au niveau des charges que des produits.

Pour qu'il en soit bien ainsi et puisque la marge est une différence entre des produits et certaines charges, il faut, d'une part, que la différence des produits reflète bien la différence effective de valeur de production sortie entre les alternatives et, d'autre part, que la différence effective de charges entre les alternatives étudiées corresponde à des variations de flux de trésorerie observables. On considérera donc comme charge associée à une décision complexe :

- le décaissement effectif qui résulte de l'appel à des heures supplémentaires ou de toute ressource supplémentaire dont on décide la mobilisation dans le cadre décisionnel étudié ici; en effet, l'accroissement de coût, par rapport à celui des heures normales, est totalement inapproprié dans la mesure où cette mobilisation de ressources additionnelles est une décision qui n'a rien de fatal et que le gestionnaire prend dans le cadre de la marge de manœuvre dont il dispose;
- les pénalités de retard qui doivent être comptabilisées dès qu'il apparaît qu'un retard est devenu inévitable (et non lorsqu'elles seront effectivement supportées).

En ce qui concerne les coûts variables directs, tels qu'ils ont été définis précédemment, l'usage du concept « différentiel » conduit aux mêmes conclusions (puisque l'on se base implicitement sur une solution de référence qui consiste à ne pas produire et donc à ne pas engager ce type de dépenses).

Deux remarques complémentaires peuvent être faites :

- L'approche par un calcul de coûts d'exécution des OF ne peut être retenue. Il faut calculer un coût global d'utilisation des ressources mobilisées par les décisions prises, toute autre solution est arbitraire et s'avère impropre à

l'évaluation de décisions complexes. On peut ajouter que le coût d'un OF ne saurait être influencé par le fait qu'il est réalisé partiellement en heures supplémentaires ou qu'il occasionne un changement de réglage de machine. Cette solution serait en effet, parfaitement arbitraire puisque ces surcoûts seraient à imputer à d'autres OF si l'on modifiait l'ordonnancement. Il faut donc considérer que l'appel à des ressources excédentaires et la non utilisation de ressources à des fins productives (réglages, arrêts par manque de travail...) sont imputables à la décision globalement prise et non à un OF particulier.

- Si la tension sur le système productif est trop grande, suite à des retards antérieurement pris ou à de mauvaises négociations des délais, il sera difficile d'éviter certaines charges, comme celles résultant de l'usage d'heures supplémentaires ou de l'application de pénalités. Cela étant, la différence des marges des alternatives étudiées ne reflétera que les conséquences de l'usage des degrés de liberté, puisque les retards inévitables conduiront aux mêmes pénalités.

III-2.5.12d) Le concept de valeur ajoutée directe

La notion de **valeur ajoutée directe** à laquelle nous faisons appel combine deux concepts. Celui de valeur ajoutée est, grosso modo pour les comptables, la différence entre la valeur de la production d'un exercice et la consommation en provenance de tiers (notamment achats de composants et matières premières) requises par cette production. Sur le court terme retenu et compte tenu de l'objectif assigné à l'indicateur, la valeur ajoutée mesurée est strictement associée à la production programmée sur l'horizon retenu et doit donc exclure des consommations qui ne sont pas liées à la décision étudiée (prestation de nettoyage par des tiers, par exemple). C'est la raison pour laquelle le qualificatif de direct a été rajouté pour signifier que ne sont prises en compte, que les consommations liées à la seule production du court terme, à l'exclusion de consommations indirectes.

Cette valeur ajoutée directe sera donc utilisée pour mesurer la valeur de la production réalisée consécutivement à la décision complexe d'ordonnancement et de mobilisation temporaire de ressources additionnelles. Le seul problème qui se pose est celui d'OF qui ne sont pas exécutés en totalité durant la période d'ordonnancement étudiée. Une partie de cette valeur ajoutée directe sera à porter au crédit de cette décision complexe mais laquelle? À ce niveau, il est impossible d'éviter une certaine part d'arbitraire. Une solution possible consiste, à retenir au titre de la période, une quote-part de la valeur ajoutée de l'OF; cette quote-part peut être le quotient du temps d'exécution de l'OF durant la période, par le temps total d'exécution de l'OF. Cette solution est contestable si le coût horaire des ressources mobilisées pour une opération diffère fortement des coûts correspondants pour les autres opérations; dans ce cas, la parade consiste à travailler avec des heures pondérées par les coûts.

III-2.5.12e) L'établissement d'un compte de résultats associé à la décision d'ordonnancement

L'utilisation de ces concepts permet d'établir un *compte de résultat prévisionnel du court terme* dont le solde est une marge qui reflète les conséquences prévisionnelles des décisions d'ordonnancement et de mobilisation

momentanée de ressources additionnelles. Son acceptation par les gestionnaires est facilitée par le fait que si l'on se situe dans le cadre fictif d'une programmation en univers certain, couvrant un exercice budgétaire, il suffirait de retrancher à cette marge les charges non prises en compte et d'effectuer quelques corrections mineures pour retrouver l'excédent brut d'exploitation.

Cela étant, l'objectif poursuivi étant de permettre la comparaison de décisions alternatives complexes, c'est donc la différence de leurs marges associées qui permet d'indiquer la meilleure alternative d'un point de vue économique (ce qui ne signifie pas que d'autres points de vue ne soient pas à prendre en compte dans la décision finale). On notera que, dans cette différence, les charges fixes disparaissent ce qui élimine les critiques que l'on pourrait faire à l'usage de certaines conventions les concernant. Par ailleurs, l'arbitraire de la solution retenue pour définir les quotes-parts de valeurs ajoutées directes imputables aux décisions prises, a une incidence d'autant plus faible que l'horizon d'ordonnancement retenu est grand (ce qui accroît le nombre de commandes exécutées en totalité). Dans le cas contraire, les «effets de bord» peuvent être importants et la valeur des encours de début et de fin de période, forte par rapport à la valeur de la production.

III-2.5.2 Exemple d'évaluation économique d'un ordonnancement

On illustrera concrètement l'application de ces principes en prenant appui sur l'exemple numérique du cas **MÉCANO**, introduit au § III-1.1, page 418, pour lequel une solution a été proposée (figure 130, pages 430 et 431). Des informations complémentaires pour permettre le calcul de la valeur ajoutée directe associée à chaque commande ainsi que les dates de livraison (implicitement définies avec une convention de fin de journée) sont fournies dans le tableau 103. Il faut ajouter que la commande 1 est la seule à avoir été commencée antérieurement et l'on supposera que 50 % du travail à effectuer sur cette commande a déjà été réalisé antérieurement.

TABLEAU 103
Caractéristiques des commandes à exécuter

Commandes	Date de livraison	Coût des composants achetés	Prix de vente	Valeur ajoutée directe
Commande 1	Jour 1	20	1 380	1 360
Commande 2	Jour 3	30	1 830	1 800
Commande 3	Jour 3	25	980	955
Commande 4	Jour 3	80	2 130	2 050
Commande 5	Jour 3	25	1 380	1 355
Commande 6	Jour 3	30	2 080	2 050
Commande 7	Jour 4	25	1 515	1 490
Commande 8	Jour 5	30	1 965	1 935

Par ailleurs, la figure 123, page 419, donne pour chaque atelier :

- un montant de charges fixes quotidiennes qui correspondent au coût des heures normales disponibles et à une quote-part des amortissements,

- un taux horaire des heures supplémentaires,
- un coût horaire variable direct qui correspond à une consommation d'énergie et fluides (y compris pendant les réglages, pour les fraiseuses).

Il faut ajouter qu'une pénalité de 200 dollars lidurien est supportée par **MÉCANO** pour chaque jour de retard subi par une commande, et ce quelle que soit la commande¹.

La solution de la [figure 130](#), page 430 et 431 (que l'on a dit ne pas être le meilleur), fait un large appel aux possibilités offertes d'heures supplémentaires. L'indicateur économique est calculé dans la perspective implicite d'une comparaison avec celui d'une solution alternative (non explicitée ici mais qui, par exemple, éviterait de faire appel aux heures supplémentaires).

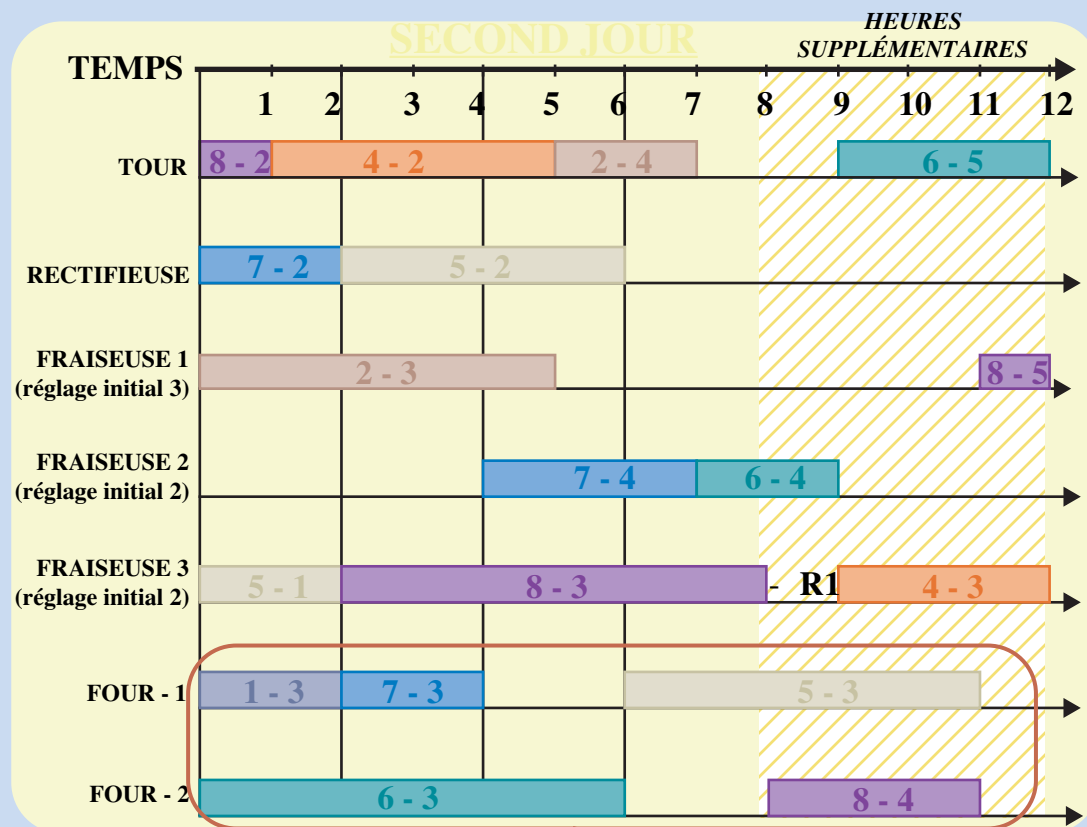
La détermination des *coûts variables directs* nécessite de calculer à partir de la solution de la [figure 130 de la page 430](#), le nombre d'heure de fonctionnement pour chaque machine et chaque jour. Un extrait de cette [figure 130](#) (ordonnancement du jour 2) est repris à la [figure 133](#), pour expliquer la constitution du tableau des heures de machines utilisées. La valorisation de ces heures implique l'utilisation d'un coût donné à la [figure 123 de la page 419](#) que la [figure 133](#) reprend en extrait. Cet exemple détaille le calcul des coûts variables directs de l'atelier de fours sur chacun des 3 jours constituant l'horizon d'ordonnancement.

Le calcul du coût des *heures supplémentaires* décidées, qui s'analyse comme une charge différentielle liée aux décisions analysées, nécessite de calculer sur la [figure 130](#) le nombre d'heures supplémentaires pour chaque atelier et chaque jour. Un extrait de la [figure 130](#) (ordonnancement du jour 2) est repris à la [figure 134 de la page 450](#), pour expliquer la constitution du tableau des heures supplémentaires. Un extrait de la [figure 123 de la page 419](#) (description de l'atelier de fours) est repris à la [figure 134 de la page 450](#), pour expliquer le processus de valorisation aboutissant au calcul des coûts additionnels d'heures supplémentaires de chacun des 3 jours constituant l'horizon d'ordonnancement.

L'autre poste à prendre en compte dans les charges additionnelles est la *pénalité de retard*. À la fin du premier jour, la commande 1 aura au moins un jour de retard, ce qui conduit à une pénalité de 200 au titre du jour 1. Le jour suivant, l'analyse de l'ordonnancement montre que, pour toutes les commandes devant être livrées le lendemain, il reste moins de 12 heures de travail, ce qui laisse a priori la possibilité de les achever en heures supplémentaires sans avoir à payer de pénalité. Le lendemain, la commande 3 n'est pas achevée, ce qui conduit à devoir supporter au titre de cette troisième journée, une pénalité de 200 \$ (qui aurait été évitée si le four 2 avait traité l'opération 3-2, au lieu de l'opération 7-5). Ces informations sont reprises directement dans le tableau de synthèse de la [figure 136, page 453](#).

1. L'usage de pénalités fictives peut être retenu pour permettre la prise en compte d'une incidence économique des retards de livraison; en effet, en l'absence de telles pénalités, l'intérêt économique de la mobilisation de ressources additionnelles peut devenir nul et le jugement s'effectuer sans prise en compte d'une évaluation économique; la pénalité fictive devient alors un paramètre de pilotage du système dont la valeur sanctionne plus ou moins fortement sa défaillance.

FIGURE 133
Exemple de calcul des coûts variables directs



	Heures-machines utilisées			Coût variable direct		
	Jour 1	Jour 2	Jour 3	Jour 1	Jour 2	Jour 3
Tours	12	10	8	96	80	64
Rectifieuse	12	6	11	84	42	77
Fraiseuses	36 [†]	23	6	288	184	48
Fours	2	18	15	32	288	240
Total	-	-	-	500	594	429

[†]. y compris 6 heures de réglage

$$288 \$ = 18 \text{ h} \times 16 \$$$

ATELIER DE FOURS

FOUR 1



FOUR 2

Coût variable direct = 16 \$ / heure

Coût fixe journalier = 288 \$ / jour

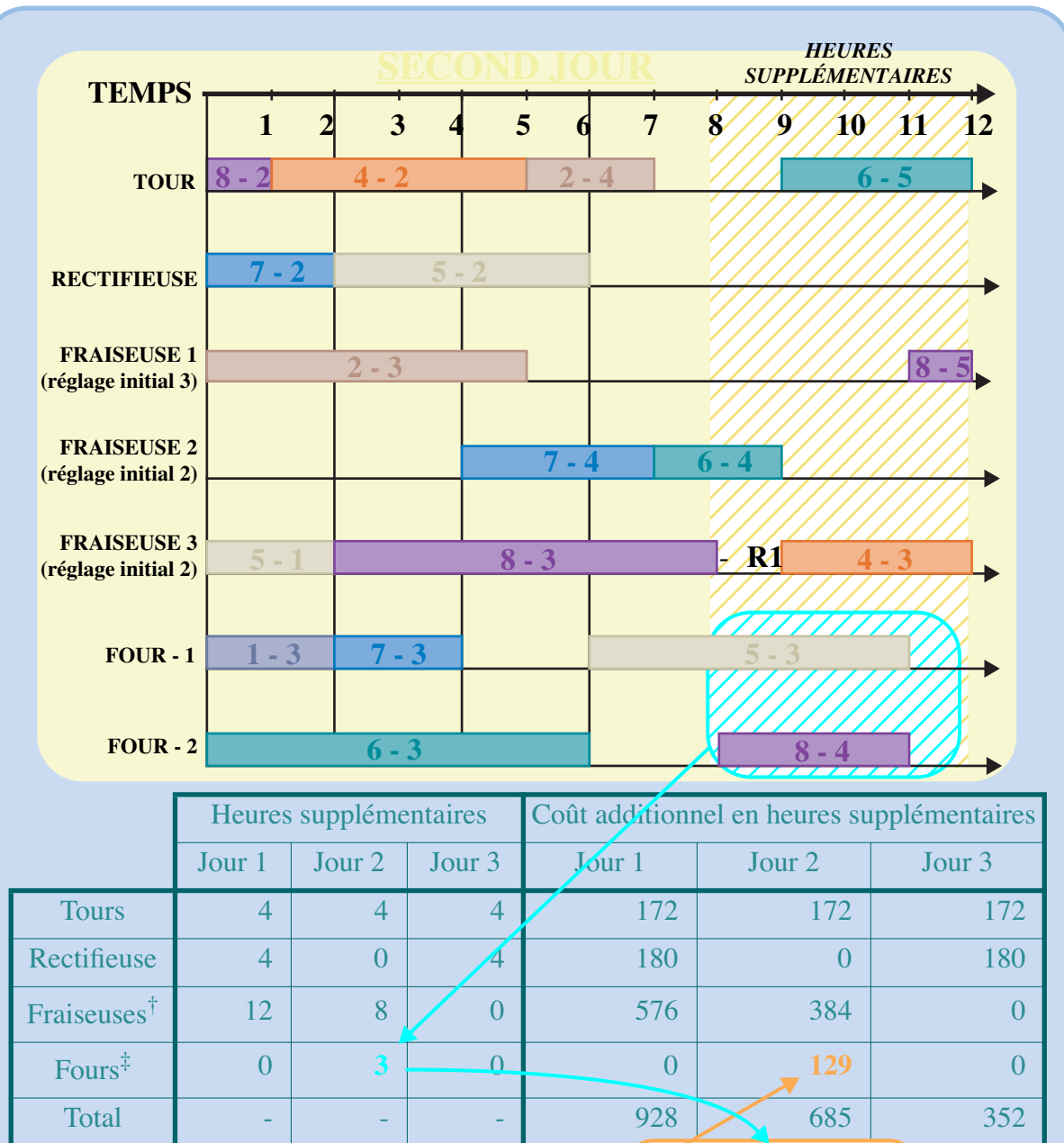
Coût de l'heure sup. = 43 \$ / heure

- un traitement thermique ne peut être interrompu
- un opérateur unique fait fonctionner les 2 fours

Table des matières

Index thématique

FIGURE 134
Exemple de calcul de coûts additionnels



[†]. Le même opérateur conduit les fraiseuses 1 et 2 pendant les h

[‡]. Opérateur unique pour les deux fours

$$\$ 129 = 3 \text{ h} \times \$ 43$$

ATELIER DE FOURS

FOUR 1



FOUR 2

Coût variable direct = 16 \$ / heure

Coût fixe journalier = 288 \$ / jour

Coût de l'heure sup. = 43 \$ / heure

- un traitement thermique ne peut être interrompu
- un opérateur unique fait fonctionner les 2 fours

La figure 135 de la page 452 explicite le calcul de la *valeur ajoutée directe* produite chaque jour. Il reprend dans sa partie supérieure la figure 130 ; le calcul est explicité pour la commande 2 qui est achevée dans les 3 jours. La répartition de la valeur ajoutée directe est calculée au prorata du temps exécuté au cours d'une journée.

L'établissement du compte de résultats prévisionnel est donné à la figure 136 de la page 453. Cette figure reprend les informations établies dans les figures 133 à 133 ; s'y ajoutent les charges fixes quotidiennes données dans la figure 123, page 419 (reprise partiellement dans la figure 134) et les pénalités de retard dont le calcul a été expliqué précédemment.

Le résultat n'est pas à évaluer indépendamment pour chaque jour, mais pour le premier jour ou les deux premiers jours ou les trois premiers jours (etc.) afin de tenir compte de l'interdépendance des décisions. Il convient de ne pas omettre, dans la comparaison de deux alternatives, cette modulation du jugement dans le temps pour éviter un comportement trop myope. En effet, la solution la meilleure sur le premier jour peut s'avérer la plus mauvaise sur trois jours.

Une dernière remarque mérite d'être faite sur un autre usage possible de ce type d'indicateur. Il est facile de calculer la variation de marge générée par l'introduction d'une nouvelle tâche dans le système productif. Ce calcul différentiel présente un intérêt opérationnel pour les industries travaillant à la commande qui peuvent y trouver :

- une aide précieuse dans l'élaboration des devis,
- une autre base pour le calcul de la partie variable de la rémunération des commerciaux (au lieu du chiffre d'affaires¹) qui améliore naturellement la coordination entre le Commercial et la Production ; en effet, en période de surcharge il est plus intéressant pour le commercial et l'entreprise de proposer au client l'alternative «prix élevé – exécution rapide» ou «prix réduit – exécution différée», alors qu'en période de sous-charge, l'intérêt commun est alors de proposer au client si possible une prestation de type «prix élevé – exécution rapide», avant de se rabattre sur une prestation de type «prix réduit – exécution rapide».

1. qui n'induit aucune incitation à tenir compte de la charge du système

FIGURE 135
Exemple de calcul de valeur ajoutée directe

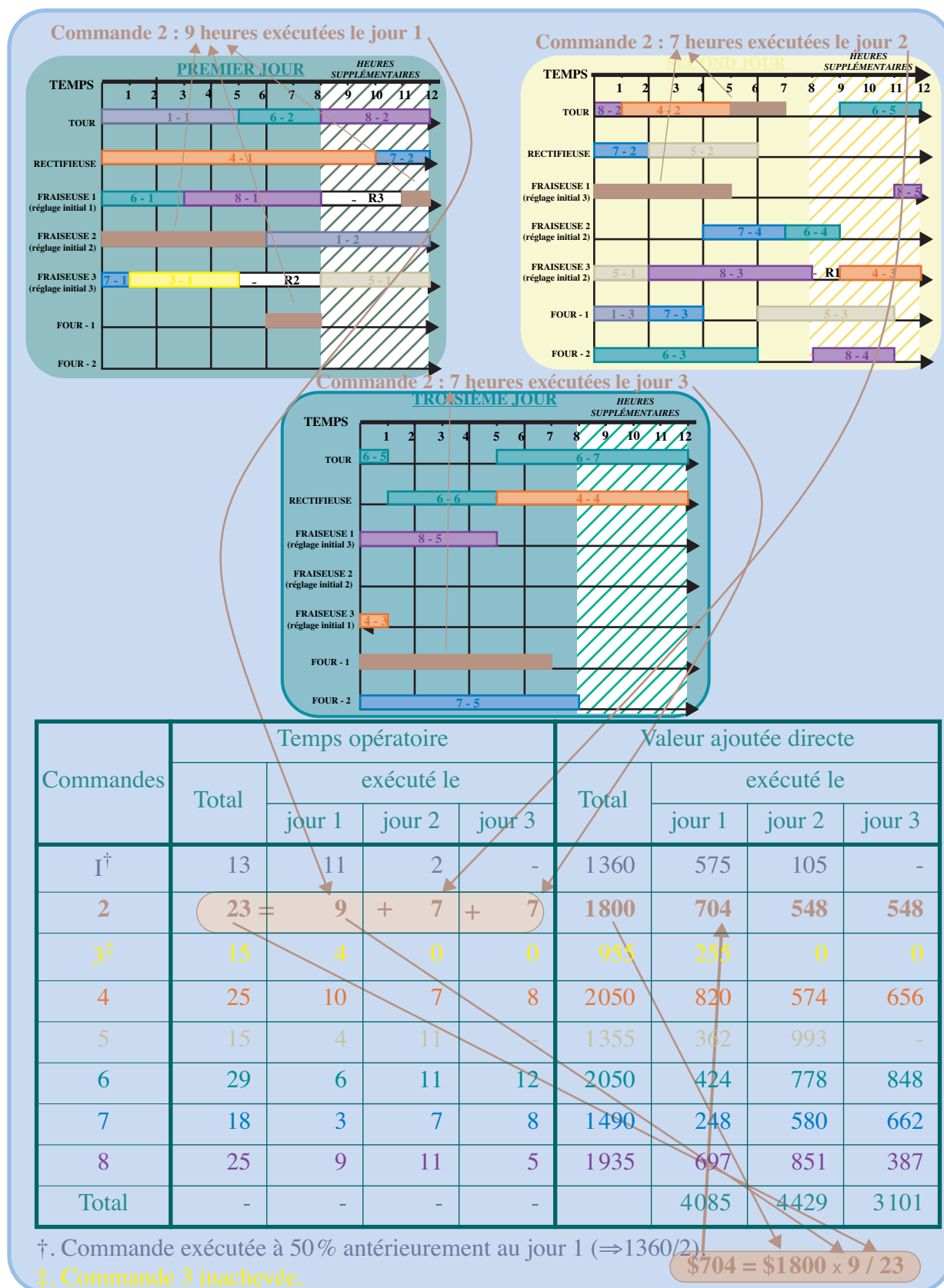


FIGURE 136

Exemple de calcul des marges associées aux décisions prises

