

# Une méthode d'affectation des ressources humaines aux processus industriels

Meziane Bennour, Didier Crestani, Olivier Crespo

► **To cite this version:**

Meziane Bennour, Didier Crestani, Olivier Crespo. Une méthode d'affectation des ressources humaines aux processus industriels. *Journal Européen des Systèmes Automatisés (JESA)*, Lavoisier, 2008, 42 (5), pp.541-577. <lirmm-00296686>

**HAL Id: lirmm-00296686**

**<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00296686>**

Submitted on 15 Jul 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

---

# Une méthode d'affectation des ressources humaines aux processus industriels

Meziane Bennour\* – Didier Crestani\* – Olivier Crespo\*

\* Université Montpellier 2  
Laboratoire d'Informatique, de Robotique et de Microélectronique  
161 rue ada  
34392 Montpellier cedex 5  
{[bennour](mailto:bennour@lirmm.fr), [crestani](mailto:crestani@lirmm.fr), [crespo](mailto:crespo@lirmm.fr)}@lirmm.fr

---

*RÉSUMÉ. Le facteur humain, en raison de ses capacités cognitives, s'impose de plus en plus comme l'un des éléments clés de la flexibilité, de la réactivité et de la performance des processus industriels. Cet article s'intéresse à la recherche du personnel permettant d'assurer, lors de son affectation, les objectifs de performance recherchés. Il s'appuie sur une méthodologie d'estimation de performance des processus d'entreprise, développée en collaboration avec un partenaire industriel, incluant explicitement l'impact des compétences des individus employés. Cette méthodologie permet de proposer une démarche d'affectation rapide et performante des ressources humaines s'appuyant sur un algorithme dichotomique. Une comparaison avec des approches heuristiques classiques (recuit simulé, etc.) est réalisée sur des exemples de moyenne et grande complexité.*

*ABSTRACT. The human factor becomes a key element of the enterprise. Into a moving environment, it ensures timeless and competitiveness. It is now necessary to model and analyse enterprise processes by taking into account human resources as well as material resources. The proposed work is dedicated to decision making for human resource assignment. Human resources are needed for the execution of process activities and generate performance. A performance estimation methodology which links human capacities and process performance has been developed with an industrial. It underlies the human resource assignment approach proposed. It has been evaluated to determine the possible assignment guaranteeing the satisfaction a performance objective of a process. A fast and performant heuristic binary search algorithm for the human resources assignment is proposed. It is compared to other heuristic algorithms (Simulated Annealing, etc.) on large and medium complexity benchmarks.*

*MOTS-CLÉS : affectation des ressources humaines, estimation de performance, algorithme heuristique.*

*KEYWORDS: human resource assignment, performance estimation, heuristic algorithm.*

---

## 1. Introduction

La planification et l'ordonnement de la production industrielle ont été largement étudiés depuis les années 50. Ce domaine de recherche relève de plusieurs disciplines allant de la recherche opérationnelle aux sciences du management. A partir des années 80, le développement des technologies de l'information a conduit dans un premier temps à minimiser l'impact réel du facteur humain sur la performance des processus industriels. Il est maintenant reconnu qu'en plus de la dimension purement technologique, les facteurs sociaux et humains influencent profondément la productivité de l'entreprise (McCarthy et al., 2001) (Grabot, 2000). La modélisation du comportement humain et de son influence sur la performance des processus restent cependant actuellement peu étudiés et intégrés dans les travaux liés à la problématique d'affectation des ressources humaines.

Cet article propose et évalue un algorithme d'affectation du personnel à un unique processus industriel. Il s'appuie sur une approche intégrant explicitement l'impact des compétences humaines sur l'estimation de la performance de ces processus (Covès, 2000) (Bennour, 2004).

La seconde section rappelle les problématiques et les multiples contraintes d'allocation des ressources humaines. La troisième section résume la méthodologie d'estimation de performance qui supporte les algorithmes d'affectation développés et précise et formalise la problématique adressée. La quatrième section énonce les principes de la démarche dichotomique préconisée et de l'algorithme associé. La cinquième section décrit les exemples de moyenne et grande complexité utilisés ainsi que la démarche d'évaluation suivie. Les sections six et huit analysent expérimentalement, sur ces exemples, l'efficacité de l'algorithme Dichotomique et d'algorithmes heuristiques classiques (Descente, Recuit Simulé, Pisinger). Pour ce faire, le temps de calcul associé, le nombre de simulations nécessaires et le taux de couverture des solutions d'affectation trouvées sont calculés. Ce travail d'évaluation est mené en s'intéressant à deux types de problématique. La première relève de la minimisation des objectifs de performance comme par exemple (Temps d'exécution du processus  $\leq$  OB\_Durée). La deuxième s'intéresse à une logique de maximisation comme par exemple (Qualité associée au processus  $\geq$  OB\_Qualité). En conclusion les limitations de cette étude et les perspectives de recherches envisageables sont abordées.

## 2. Allocation des ressources humaines : de nombreuses contraintes à satisfaire

Les problématiques associées à l'allocation des ressources humaines dépendent de l'horizon temporel à l'intérieur duquel la décision doit être prise (Fanchini *et al.*, 1998). Sur le long terme, au niveau stratégique, différentes politiques d'affectation sont en général évaluées pour contrôler l'évolution socio-technique d'une entreprise en situation de changement et guider la gestion des compétences. L'analyse des

déséquilibres entre les besoins futurs et les ressources humaines déjà en place permet de mettre en lumière les dysfonctionnements qui perturberaient l'évolution de l'organisation (Lengnich-Hall et al., 1988) (Guerin, 1992). A moyen terme, on cherche à planifier l'affectation du personnel. Dans ce cas, l'objectif est plutôt d'ajuster la capacité de production à la charge prévisionnelle sans une affectation nominative des employés (Vairaktarakis et al., 2002) (Franchini et al., 2001) (Kane et al., 2001) (Kusiak et al., 1998). Enfin, le court terme permet d'ordonnancer et d'allouer effectivement les ressources en prenant en compte des contraintes plus opérationnelles comme leurs disponibilités, leurs temps de repos, leurs jours de congés et très rarement leurs compétences (Franchini et al., 2001) (Mosheiov, 2001) (Hung, 1999) (Jia, 1998). Une affectation intelligente en fonction de la forme de la demande permet aussi d'améliorer la flexibilité de l'outil de production.

Les contraintes d'affectation se déclinent au travers d'un ensemble de catégories. Tout d'abord les contraintes de compétence qui sont souvent négligées pour l'allocation des ressources humaines. Dans ce cas les compétences requises par une tâche se distinguent de celles acquises par les employés. La dimension quantitative associée à la constitution des équipes de travail nécessaires à l'exécution d'une tâche constitue une autre classe importante de contraintes. Elle accroît fortement la complexité du processus d'allocation en multipliant le nombre de possibilités à explorer. Elle est en général prise en compte dans la formulation générique du problème d'ordonnancement des ressources humaines (Soubeiga, 2003) comme dans la formulation de Dantzig (Dantzig, 1954) qui introduit la contrainte du nombre d'employés requis pour exécuter une tâche.

Les contraintes de disponibilité des entités humaines recouvrent de nombreux aspects. On distingue deux dimensions, l'une dynamique et l'autre statique. La première rend compte, à un instant donné, de l'affectation ou non d'un employé à une tâche lors du processus d'ordonnancement, ou de son absence en raison d'une indisponibilité prévue (congés) ou imprévue (maladie). La seconde limite la disponibilité des employés en raison de contraintes légales ou sociales liées aux durées de travail ou de repos imposées. On distingue alors plusieurs problématiques. Le "day of scheduling" (Alfares, 2000) vise à déterminer les jours de travail ou de repos tout en respectant les contraintes légales. Le "shift scheduling" (Aykin, 2000) (Narasimhan, 2000) prend en compte les quarts de travail et les pauses pour l'assignation des ouvriers sur un horizon journalier. Enfin, le "tour scheduling" (Bard et al., 2003) intègre les deux problématiques précédentes pour une planification hebdomadaire.

A ces contraintes classiques s'ajoutent aussi des contraintes organisationnelles permettant de spécifier à quelle(s) unité(s) organisationnelle(s) peut appartenir le personnel pouvant être utilisé (entreprise, unité, service, département, équipe, etc.).

Une dernière contrainte d'affectation majeure est celle qui doit assurer que la performance visée globalement au niveau du processus est bien satisfaite par les ressources allouées. Cette contrainte de performance est le plus souvent négligée car

on suppose que les employés affectés sont à même d'atteindre les objectifs imposés. Ainsi la problématique d'affectation se décline essentiellement en terme de disponibilité ou non du personnel. Cependant, de nombreuses études ont démontré l'importance du management des ressources humaines dans la performance des processus (Guest, 1997) (Mc Ewan et al., 1998). Parmi toutes les classes d'objectifs étudiés (profitabilité, productivité, coût de formation, etc.) la contrainte la plus souvent rencontrée en production est liée au temps (date de réalisation, durée des activités, du processus, etc.). Mais les objectifs de l'entreprise couvrent bien d'autres facettes de la performance tels que les aspects Financier et Qualité qui sont plus rarement pris en compte dans les travaux relevant de l'affectation des ressources.

En résumé, les principales contraintes liées à l'allocation des ressources humaines relèvent de la disponibilité des employés, de leur adéquation aux tâches devant être exécutées, de leur quantité, enfin de la performance qu'ils engendrent au niveau d'une activité et plus globalement du processus. Actuellement, l'ensemble des niveaux décisionnels et des contraintes énoncées ne peut être simultanément intégré dans les travaux menés sur l'affectation du personnel. De plus, peu d'études intègrent effectivement l'aspect humain lors de l'allocation des ressources (Franchini et al., 2000) (Tchommo et al., 2003) (Baptiste et al., 2006).

La suite de l'article se focalise sur le problème de l'affectation du personnel à un processus industriel d'entreprise. Cet article n'aborde pas directement les contraintes de compétences et de disponibilité. Il détaille comment satisfaire efficacement la contrainte de performance imposée au niveau d'un processus.

### **3. Contexte et positionnement du problème**

#### **3.1. Le problème d'affectation adressé**

L'affectation du personnel aux processus n'est pas neutre même si les employés sont considérés dans un premier temps comme apte à l'exécution de la tâche à réaliser. Cependant, même si les ressources humaines satisfont les contraintes définies en terme de compétence et de disponibilité, les choix retenus influenceront largement sur la performance finale du processus (Grabot, 2000). En effet, il est évident que l'on ne peut comparer l'efficacité d'un employé débutant et expérimenté. Cet état de fait est utilisé dans de nombreux travaux faisant appel à des niveaux de compétence (Franchini, 2000) (Drejer, 2001) (Norman et al., 2002) et influence fortement la durée des activités ainsi que le coût et la qualité associés. En plus de ces caractéristiques strictement individuelles, l'aptitude au travail collectif constitue aussi un paramètre très important qui ne peut être négligé lors de l'affectation du personnel tout au long de l'exécution d'un processus opérationnel d'une entreprise.

Dans un environnement industriel une réponse doit rapidement être trouvée pour faire face aux événements imprévus (machine en panne, retard de livraison, etc.) qui

viennent remettre en cause les plannings prévisionnels établis. Il faut alors déterminer très rapidement des solutions d'affectations acceptables en intégrant de nombreux paramètres dépendant de la connaissance de l'entreprise, du nombre de solutions alternatives et du temps disponible pour la prise de décision qu'imposent les contraintes de production (MacCarthy et al., 2001). La complexité du problème rencontré rend souvent impossible la détermination analytique, en un temps raisonnable, de la solution optimale satisfaisant l'ensemble des contraintes antinomiques de performance à intégrer (qualité, coût, délais). Il est possible de s'affranchir de cette condition d'optimalité de la solution en considérant qu'il est possible de rester dans une perspective opérationnelle en fixant un objectif de performance non optimal où l'on ne cherche pas la meilleure solution, qui est souvent impossible à imposer pratiquement, mais plus simplement une solution permettant seulement de livrer les produits aux coûts et délais imposés. Cela conduit donc à borner la performance en valeur maximale (durée, coût) ou minimale (qualité) de façon à éliminer les affectations non satisfaisantes, mais aussi à préserver un ensemble plus conséquent de solutions permettant d'atteindre les objectifs de performance imposés sans se restreindre aux seules solutions optimales.

Cet article vise à définir un algorithme simple et efficace permettant d'adresser le problème d'affectation des ressources humaines à un processus d'entreprise.

Cet algorithme doit :

- Déterminer quel personnel il est possible d'affecter aux différentes activités composant un processus de façon à ce que la performance estimée de ce dernier satisfasse un objectif de performance OB fixé. Ce dernier dépend bien évidemment du point de vue choisi. En général la durée du processus ou le coût associé doit être inférieur à une valeur donnée, alors que sa qualité doit être supérieure à un objectif imposé.
- Maximiser le nombre de solutions d'affectation déterminées. Cela revient donc à chercher à maximiser le taux de couverture de l'ensemble des solutions d'affectation réellement possibles. Dans une optique d'aide à la décision, cela permet de disposer d'un ensemble important de solutions envisageables au sein desquels le décideur peut finaliser son choix.
- Etre rapide. D'une part cette contrainte est imposée par un souci de réactivité dans un environnement industriel pour permettre au décideur de disposer d'une durée raisonnable pour sa prise de décision. D'autre part, la rapidité d'exécution de l'algorithme permet d'adresser efficacement la problématique générale de satisfaction simultanée de plusieurs objectifs de performance (Bennour, 2004) (Bennour et al., 2005) qui n'est pas l'objet du présent article.

L'algorithme suppose :

- Que le comportement  $\sigma = \{A1, \dots, An\}$  du processus d'entreprise est séquentiel c'est-à-dire qu'il est constitué par un enchaînement linéaire (potentiellement rebouclé) d'activités  $A_i$  à exécuter.
- Qu'au niveau de chacune des activités, les différents groupes d'employés éligibles (c'est-à-dire satisfaisant les contraintes qualitatives (compétence minimale, etc.) et quantitative (nombres d'employés, etc.)), ont été préalablement identifiés.
- Qu'il est possible d'évaluer l'impact des compétences de chacun des groupes d'employés à affecter pour estimer la performance de(s) activité(s) où ils sont impliqués ainsi que, plus globalement, la performance du processus étudié pour chaque configuration d'affectation choisie.

La prise en compte des compétences dans le calcul de la performance d'un processus impacte fortement le choix des employés à affecter pour satisfaire un objectif de performance. La démarche utilisée, qui sous-tend l'algorithme d'affectation, est maintenant rapidement présentée.

### ***3.2. Intégration des compétences dans le calcul de performance***

La démarche, développée dans (Covès, 2000) (Bennour, 2004) (Bennour et al., 2007), a été construite dans le cadre d'un projet avec l'industriel Merlin Gerin.

Dans un premier temps, elle permet d'estimer contextuellement, au niveau d'une activité donnée d'un processus industriel, l'influence (positive ou négative) que peuvent avoir les employés affectés sur la performance nominale de l'activité considérée. En plus de l'aspect individuel, la dimension collective du travail est intégrée partiellement en considérant contextuellement les capacités managériales et les facultés relationnelles du personnel impliqué. Dans ce projet, les points de vue de performance suivants ont été précisément analysés : la durée d'exécution du processus, le coût financier correspondant et enfin la qualité associée.

Dans un second temps, pour un comportement  $\sigma$  donné du processus, la démarche proposée permet de connaître la performance finale obtenue à l'issue de son exécution. Le comportement du processus est alors vu comme l'exécution séquentielle ou concurrente d'une succession d'activités pouvant être synchronisées. Cela signifie donc que l'on connaît, au préalable, dans quel ordre doivent être exécutées les activités composant le processus. La performance finale est obtenue en composant celle induite par les ressources humaines utilisées par chacune des activités traversées lors de l'exécution du processus pour le comportement  $\sigma$  choisi.

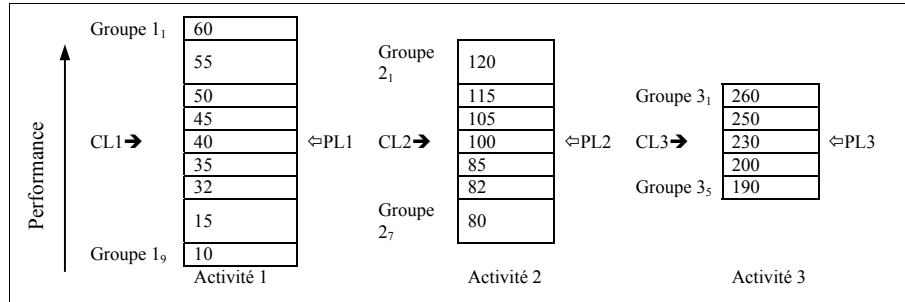
Les lois de composition suivantes ont été retenues pour les différents points de vue :

- **Durée** : Une loi de composition maximum est utilisée pour cumuler le temps associé à l'exécution des activités. En présence d'un comportement linéaire séquentiel cette loi se comporte comme une loi de composition additive. Après un comportement concurrent, la loi de composition maximum est utilisée pour traduire le fait que l'activité au niveau de laquelle s'effectue la synchronisation ne pourra commencer à s'exécuter que lorsque toutes les activités amont impliquées auront été achevées.
- **Financier** : Une loi de composition additive additionne les coûts associés à chacune des activités exécutées.
- **Qualité** : Afin de rendre compte cet aspect de la performance, notre partenaire industriel a retenu tout comme Norman (Norman et al., 2002) une loi de composition additive pour agréger la qualité estimée, bornée entre -100 et +100, des activités du processus.

En résumé, l'approche qui vient d'être décrite permet de connaître la performance engendrée par une activité en fonction des personnels affectés et indépendamment des autres activités impliquées dans l'exécution d'un processus. De plus, des lois d'agrégation simples et de bon sens sont employées pour agréger les performances associées à chacune des activités d'un processus afin d'obtenir, pour un point de vue donné, une estimation de la performance finale qu'engendrera le processus étudié pour le comportement choisi et les affectations retenues.

Ainsi, en utilisant cette démarche d'estimation de performance, il est possible de calculer pour chaque groupe de travail pouvant être affecté à une activité, la performance correspondante engendrée selon un point de vue donné. Les estimations obtenues peuvent alors être ordonnées localement, pour chaque activité de la plus petite à la plus grande. La figure 1 représente pour trois activités, un exemple de listes ordonnées de performance. Chaque valeur correspond à la performance d'un groupe de personnels satisfaisant des diverses contraintes d'affectation (organisationnelles, de compétences et quantitatives).





**Figure 1.** Exemple de configuration de performance (en unité de performance)

A partir de la configuration locale des données illustrée par la figure 1, il est fondamental de remarquer que les lois de composition retenues pour déterminer la performance d'un processus (maximum, addition) permettent de préserver globalement, au niveau du processus, la relation d'ordre locale définie après classement pour chacune des activités (figure 1). Cela revient à dire que si, pour chaque activité, le choix de l'affectation d'un groupe permet de satisfaire l'objectif de performance imposé au processus alors, nécessairement, tous les groupes situés au dessus ( $Performance\_Processus \geq OB$ ) ou au dessous ( $Performance\_Processus \leq OB$ ) dans les listes locales ordonnées (figure 1) vérifieront aussi la contrainte de performance qui doit être satisfaite globalement à la fin de l'exécution du processus. Cette propriété forte de dominance permet donc de conclure rapidement sur un grand nombre de configurations d'allocation.

Les objectifs et hypothèses de travail étant maintenant définis, la mise en équations et l'évaluation de la complexité du problème adressé peuvent être maintenant abordées. Cette partie présente la formulation du problème d'affectation qui peut être développée dans le cas où l'on impose de satisfaire, pour un processus donné, la contrainte  $Performance\_Processus \leq OB$  (durée, coût). Elle peut être adaptée sans difficulté au cas où la contrainte définie est de type  $Performance\_Processus \geq OB$  (qualité).

### 3.3. Formulation basique du problème d'affectation

Considérons dans un premier temps que l'on cherche à trouver une configuration d'affectation d'employés (un groupe à affecter pour chaque activité d'un processus) permettant d'optimiser la satisfaction de la contrainte globale d'objectif de performance imposée au niveau d'un processus  $Performance\_Processus \leq OB$ .

Le problème d'affectation qui nous intéresse peut alors se formuler de la façon suivante. Chacune des  $n$  activités composant un processus peut être assimilée à une classe regroupant un ensemble de  $j$  objets représentant les groupes d'employés

affectables. Chaque objet correspond à une configuration d'affectation d'employés, préalablement identifiée, satisfaisant aux contraintes de disponibilité (maladie, déplacement, présence) mais aussi pleinement aux contraintes organisationnelles (entreprise, unité, service, etc.), quantitatives (nombre) et qualitatives (compétence, certification, etc.) d'affectation imposées au niveau de l'activité considérée. La combinatoire induite par ces contraintes permet d'identifier pour chaque classe (activité) l'ensemble des  $ng_i$  objets (groupe d'employés) éligibles (affectables) sans prendre en compte la contrainte globale de performance imposée pour le processus. A chaque objet (groupe d'employés) d'une activité (classe), est associée la performance estimée du groupe d'employés correspondant pouvant être affecté. On cherche alors quels objets (groupe d'employés) il est possible de choisir (d'affecter), au niveau de chacune des classes (activités) composant le processus tout en satisfaisant au mieux l'objectif de performance imposé.

De façon générale, ce problème s'énonce comme suit : On dispose de  $n$  classes disjointes  $N_1, \dots, N_n$  d'objets à choisir. Chaque objet  $j \in N_i$  possède un poids  $W_{ij}$ . Comment choisir un objet appartenant à chacune de ces classes sans pour autant dépasser le poids total  $C$  imposé ? (équation 1)

$$\sum_{i=1}^n \sum_{j \in N_i} W_{ij} X_{ij} \leq C \quad \text{avec } X_{ij} \in \{0,1\} \quad i = 1, \dots, n, j \in N_i \quad [1]$$

Tous les coefficients  $W_{ij}$  et  $C$  sont des entiers positifs, et les classes sont mutuellement disjointes. Chaque classe  $N_i$  a une taille  $ng_i$ .

Pour le problème d'affectation de ressources posé les correspondances suivantes peuvent être établies : les classes correspondent aux activités du processus, le poids total  $C$  correspond à la contrainte à satisfaire au niveau du processus, le choix  $X_{ij}$  (variable bivalente) d'un objet dans chaque classe correspond au choix d'une configuration d'affectation pour chaque activité.  $W_{ij}$  représente la performance induite  $F_i(G_{A_i}^j)$  au niveau d'une activité  $A_i$  par le choix d'une configuration d'affectation  $G_{A_i}^j$ .

Si l'on considère les données suivantes :

- Soit un ensemble d'activités :  $A = \{A_i\} / 1 \leq i \leq n$
- Soit, pour chaque activité  $A_i$ , l'ensemble des groupes de personnel  $G_{A_i}$  candidats à l'affectation avec  $G_{A_i} = \{G_{A_i}^j\} / 1 \leq j \leq ng_i$
- Soit  $OB$  l'objectif de performance
- Soit, pour chaque activité  $A_i$ , une fonction d'évaluation  $F_i$  qui, pour un candidat  $G_{A_i}^j$ , calcule la valeur de performance  $F_i(G_{A_i}^j)$  associée.

Soit  $x_{A_i}^j$  la variable bivalente associée au groupe  $G_{A_i}^j$ . On a

$$x_{A_i}^j = \begin{cases} 1 & \text{si } G_{A_i}^j \text{ est choisi pour } A_i \\ 0 & \text{si } G_{A_i}^j \text{ n'est pas choisi pour } A_i \end{cases}$$

On cherche donc à satisfaire la fonction contrainte Z (équation 2) telle que :

$$Z = \sum_{i=1}^n \sum_{j=1}^{ng_i} x_{A_i}^j \cdot F_i(G_{A_i}^j) \leq OB \quad [2]$$

Avec comme contrainte supplémentaire, dans notre cas, au niveau de l'activité  $A_i$  de ne choisir qu'un seul groupe (équation 3).

$$\sum_{j=1}^{ng_i} x_{A_i}^j = 1 \quad [3]$$

La prise en compte d'un objectif de performance de type durée ou coût conduit naturellement à chercher à minimiser Z. En effet, au premier abord, on cherche en général à minimiser les coûts et la durée d'un processus.

Cette formulation peut être trivialement vérifiée avec les hypothèses retenues puisque nous disposons, au niveau de chacune des activités d'un processus, d'une liste ordonnée des performances engendrées par chaque affectation possible et que la loi de composition choisie est l'addition. Il suffit donc de choisir, au niveau de chacune des activités, la configuration d'affectation correspondant à la performance minimale. Cela conduit dans le cas de l'exemple de la figure 1 et si l'objectif de performance imposé peut être satisfait, à la configuration (CL1, CL2, CL3) = (10, 80, 190) pour une performance globale minimale de Performance\_Processus(10, 80, 190) = 10 + 80 + 190 = 280.

Si cette solution vérifie de façon optimale le problème posé elle n'est en revanche absolument pas satisfaisante du point de vue de l'objectif de maximisation du nombre de solutions d'affectation qui a aussi été fixé. En effet cette configuration est unique ce qui la rend nécessairement très contraignante et peu robuste puisque aucune solution alternative n'est identifiée.

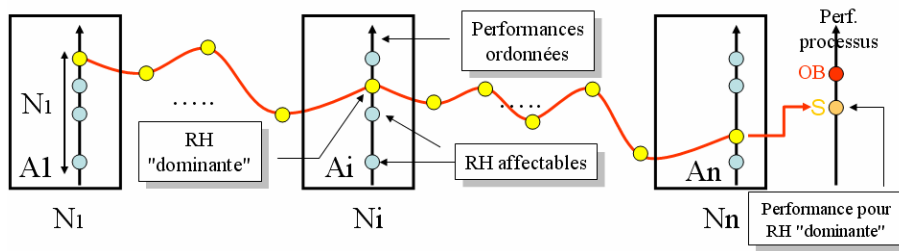
#### **3.4. Complexité générale du problème : maximisation du nombre de solutions**

La prise en compte de l'objectif de maximisation du nombre de solutions conduit donc à chercher à exprimer le nombre de combinaisons d'affectation pouvant être identifiées à partir d'une solution vérifiant l'équation 2.

En s'appuyant sur les deux propriétés fortes de la méthodologie d'évaluation de la performance énoncées (ordre local des performances des activités, ordre global préservé par les lois de composition de performance) la résolution de l'équation 2 permet de déterminer une combinaison d'affectation dominante dont l'optimalité, vis-à-vis du nombre de solutions identifiées, dépend de l'algorithme utilisé. Ainsi, dans

la figure 2, la solution S trouvée permet de satisfaire la contrainte imposée  $Performance\_Processus(S) \leq OB$ .

La propriété de dominance permet non seulement de conclure qu'une solution d'affectation dominante (RH dominante) est solution mais aussi que toute affectation locale dominée l'est aussi. Ainsi globalement, pour le processus, le choix, au niveau de chacune des activités, d'une affectation dominante ou dominée, permet de construire une combinaison d'affectations vérifiant nécessairement la contrainte de performance imposée.



**Figure 2.** Dominance et nombre de solutions d'affectation

Si  $N_k$  dénombre le nombre de configurations locales d'affectation identifiées à partir d'une solution S (figure 2) alors le nombre de solutions  $NB\_Sol(S)$  trouvées (combinaisons d'affectations valides au niveau du processus) peut s'exprimer par l'équation 4.

$$NB\_Sol(S) = \prod_{k=1}^n N_k \quad [4]$$

Il est évident que le nombre de solutions trouvées est alors très sensible à la position locale des affectations dominantes identifiées par l'algorithme utilisé pour satisfaire l'équation 2.

En utilisant les notations définies dans le paragraphe précédent, la maximisation du nombre de solutions déterminées à partir de l'identification d'une solution dominante revient à chercher à maximiser la fonction suivante (équation 5) :

$$NB\_Sol(S) = \prod_{i=1}^n \left( \sum_{k=1}^{j_i} j_i \cdot x_{A_i}^k \right) \text{ avec } \sum_{j=1}^{ng_i} x_{A_i}^j = 1 \text{ et } 1 \leq j \leq ng_i \quad [5]$$

Il est évident que cette fonction ne s'exprimant pas sous la forme d'une combinaison linéaire, ni même quadratique des variables du problème (terme produit présent dans l'équation 5), la recherche de la meilleure solution d'affectation

satisfaisant à la fois les contraintes de performance et de maximisation du nombre de solutions sera excessivement coûteuse en temps.

Face à l'importante complexité du problème posé, nous allons donc nous orienter vers une approche heuristique du problème d'affectation sans résoudre de façon optimale la contrainte de maximisation du nombre de solutions. En d'autres termes est-il possible d'identifier une solution de l'équation 2 qui permette, en général, d'identifier un nombre important de solutions d'affectation ?

### **3.5. Résolution heuristique : Formulation sous forme d'un sac-à-dos**

La formulation du problème d'affectation développée dans le paragraphe 3.3 suppose que l'on cherche à minimiser la fonction de coût associée à la performance comme il est d'usage pour la durée et les coûts. Cette hypothèse très contraignante ne conduit alors qu'à identifier une unique solution.

C'est pourquoi nous allons exploiter la latitude du cahier de charge qui spécifie que toute solution d'affectation permettant de satisfaire  $Performance\_Processus \leq OB$  reste satisfaisante pour l'industriel même si ce n'est pas la meilleure solution au sens de la performance. En se rapprochant par valeur inférieure, de la limite de performance imposée globalement, au niveau du processus, il est possible d'espérer obtenir une solution dominante permettant d'identifier un nombre nécessairement plus important de solutions d'affectation.

Supposons ainsi qu'au niveau du processus de la figure 1 on cherche à satisfaire la contrainte  $Performance\_Processus \leq 400$  à l'aide d'un algorithme permettant d'identifier les configurations dominantes. La configuration dominante  $(CL1, CL2, CL3) = (40, 100, 230)$  permet de satisfaire la contrainte  $Performance\_Processus(40, 100, 230) = 370 \leq 400$ . Cette configuration identifie  $NB\_Sol(40, 100, 230) = 5 * 4 * 3 = 60$  solutions d'affectation. Cependant la configuration dominante  $(45, 100, 230)$  conduit à  $Performance\_Processus(45, 100, 230) = 375$  qui satisfait elle aussi la contrainte mais identifie  $NB\_Sol(45, 105, 250) = 6 * 4 * 3 = 72$  solutions d'affectation. On constate donc que s'il est possible d'encore vérifier la contrainte de performance en intégrant de nouvelles configurations au niveau de certaines activités (ici l'activité A1) sans pour autant être conduit à en exclure par ailleurs (même positionnement des solutions dominantes pour les activités A2 et A3) alors le nombre de solutions identifiées sera lui aussi nécessairement plus important. Les algorithmes utilisés seront donc d'autant plus efficaces qu'ils parviendront à suivre ce mécanisme d'expansion lors de la détermination d'une solution dominante d'affectation.

En nous inscrivant non plus dans une logique de minimisation stricte d'une fonction de coût mais dans une logique de maximisation sous contrainte de la performance il est possible de formuler le problème d'affectation adressé sous la forme d'un problème de sac à dos à choix multiple. En effet la contrainte de coût a déjà été identifiée (équation 2) et correspond à la capacité du sac à dos. L'équation 3 qui impose de ne choisir qu'un seul objet (groupe d'employés) d'une classe (activité)

définie une problématique dite de choix multiple. Enfin, si l'on suppose que chaque objet  $j \in N_i$  possède un profit  $P_{ij}$  qui peut être identifié, tout comme le poids des objets, à la performance d'un groupe d'employé, alors nous cherchons maintenant à maximiser la somme  $Z$  des profits (équation 6) comme dans la formulation classique d'un problème de sac à dos. Ce type de problème est NP-difficile (Kulanoot, 2000) mais peut être résolu en temps pseudo-polynomial (Pisinger, 1995). Les algorithmes sont en général très sensibles aux valeurs numériques manipulées.

$$Z = \sum_{i=1}^n \sum_{j \in N_i} P_{ij} X_{ij} = \sum_{i=1}^n \sum_{j=1}^{ng_i} x_{A_i^j} \cdot F_i(G_{A_i^j}) \quad [6]$$

En d'autres termes, on cherche donc à maximiser le coût et la durée d'un processus sans pour autant dépasser la limite imposée par l'industriel.

### 3.6. Remarques et conclusion

La problématique d'affectation qui vient d'être présentée dépend donc de deux critères à la corrélation complexe : la satisfaction sous contrainte de la performance du processus et le nombre de solutions identifiées. Compte tenu de la complexité du problème de maximisation du nombre de solutions et des liens entre ces deux critères, la taille de la frontière de Pareto correspondante est nécessairement importante et difficile à déterminer.

L'heuristique adoptée de maximisation sous contrainte de la performance globale d'un processus formulée sous la forme d'un problème de sac à dos à choix multiple permet de montrer que la corrélation entre maximisation de la performance et du nombre de solutions est plus complexe qu'il n'y paraît. En effet supposons que l'on reprenne l'exemple précédent où l'on cherche à satisfaire la contrainte  $\text{Performance\_Processus} \leq 400$ . Une solution optimale du problème du sac à dos conduit à la configuration dominante  $(CL1, CL2, CL3) = (60, 80, 260)$ . Elle permet de satisfaire au mieux les contraintes en atteignant exactement l'objectif imposé avec  $\text{Performance\_Processus}(60, 80, 260) = 400$ . Cependant elle ne conduit qu'à l'identification de  $\text{NB\_Sol}(60, 80, 260) = 9 * 1 * 5 = 45$  solutions ce qui est bien inférieur aux nombres de solutions mises en évidence par les configurations moins optimales que nous avons citées plus haut. Ainsi l'utilisation d'un algorithme optimale de résolution du problème de sac à dos, comme celui proposé par Pisinger, ne permet pas de garantir une maximisation du nombre de solutions identifiées. Entre plusieurs solutions, l'heuristique ne définit donc pas une relation d'ordre globale au niveau du processus, mais seulement une relation d'ordre locale, au niveau des activités.

Dans la suite nous allons donc dans un premier temps utiliser un algorithme permettant de résoudre le problème de sac à dos à choix multiple posé pour identifier une solution dominante. Puis, dans un second temps, à partir cette solution

dominante, nous pourrions calculer le nombre de solutions d'affectation correspondantes identifiées.

En suivant cette démarche, nous allons évaluer, sur un ensemble de benchmarks, la capacité de plusieurs approches de résolution du problème de sac à dos à choix multiple, à maximiser le nombre de solutions d'affectation. Nous commençons par détailler l'approche dichotomique développée.

#### 4. Principes de l'algorithme Dichotomique proposé

##### 4.1. Démarche générale d'affectation

La figure 3 résume les principales étapes du processus d'affectation.

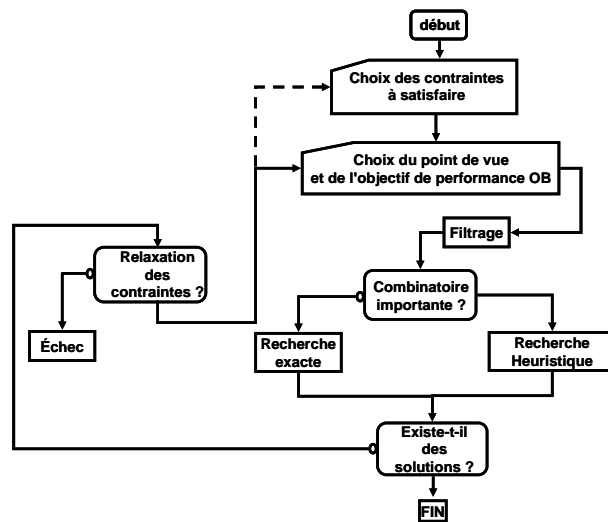


Figure 3. Démarche générale d'allocation des ressources humaines

Pour chaque activité, l'utilisateur définit tout d'abord les contraintes d'affectations imposées. Les ensembles de choix définis à partir de contraintes organisationnelles sont alors réduits en prenant en compte les contraintes de compétences et de disponibilité connues (congés, repos, maladie). A la suite de l'étape de filtrage, pouvant être mise en œuvre à l'aide d'un logiciel de programmation sous contraintes comme par exemple ECLIPSe (Eclipse, 2004), on dispose alors, au niveau de chaque activité, de la liste du personnel pouvant a priori être employé. Les contraintes quantitatives permettent alors de constituer les groupes de travail pouvant être potentiellement affectés à une activité. Ainsi, pour la  $i$ ème activité  $A_i$  d'un processus,  $n_{A_i}$  groupes  $G_{A_i}^j$  ( $1 \leq j \leq n_{A_i}$ ) regroupant un ensemble d'employés peuvent être constitués. Il est alors possible en utilisant

l'approche d'estimation de performance présentée, de calculer, pour chaque point de vue de performance considéré, et pour chaque groupe d'affectation, la performance d'une activité isolée. C'est à partir de ce point que sont effectivement disponibles les données de départ du problème à traiter (figure 1). La complexité du problème d'allocation peut alors être facilement calculée. Elle correspond, dans le pire des cas, à l'exploration exhaustive de toutes les combinaisons d'affectation envisageables et peut s'exprimer par l'équation [7].

$$\text{Nbre de configurations possibles} = \prod_{i=1}^n ng_i \text{ avec } n = \text{Nbre d'activités} \quad [7]$$

Deux cas sont à considérer. Soit la complexité combinatoire est faible et il est possible d'analyser rapidement et exhaustivement l'ensemble des configurations d'affectation envisageables pour déterminer celles permettant de satisfaire l'objectif de performance imposé. Soit, les données du problème font qu'il n'est pas possible d'explorer l'ensemble de l'espace solution et l'on doit avoir recours à une démarche heuristique pour déterminer les acteurs pouvant être employés. Enfin, si aucune solution n'a pu être trouvée l'utilisateur est amené à relâcher certaines contraintes d'affectation ou à être moins exigeant sur la performance attendue du processus.

C'est dans ce cadre de la démarche heuristique qu'est proposé l'algorithme d'affectation qui va être maintenant décrit.

#### **4.2. Des données locales au contexte global**

Les capacités et propriétés de l'approche d'estimation de performance décrite et l'efficacité d'un algorithme de recherche dichotomique sont associées pour identifier le personnel permettant, en cas affectation, d'atteindre l'objectif de performance imposé. Pour cela, pour chaque activité  $A_i$ , le concept de Choix Limite ( $CL_i$ ) est introduit (figure 1). Il désigne l'un des  $ng_i$  groupes que l'on peut assigner. Il lui correspond une performance associée ou Performance Limite ( $PL_i$ ). Ce Choix Limite divise au niveau de chaque activité les ensembles ordonnés de choix en deux parties. D'une part l'ensemble des groupes de personnel générant une performance locale supérieure à  $PL_i$ , d'autre part celui composé de l'ensemble des affectations engendrant une performance inférieure ou égale à  $PL_i$ .

Comme cet article se focalise sur les processus séquentiels, c'est la loi de composition addition qui est employée, quelque soit la dimension de la performance étudiée, pour composer globalement les performances locales ( $CL_i$ ) de chacune des activités impliquées. Cette loi de composition additive préservant la relation d'ordre locale, il est possible de conclure, en une seule simulation numérique, et pour un grand nombre d'affectations envisageables, sur la satisfaction de l'objectif de performance ( $\leq OB$ ,  $\geq OB$ , etc.). Cela revient à dire que si, pour chaque activité, le choix de l'affectation d'un groupe permet de satisfaire l'objectif de performance imposé au processus alors, nécessairement, tous les groupes situés au dessus



(Performance Processus  $\geq$  OB) ou au dessous (Performance Processus  $\leq$  OB) dans les listes locales ordonnées (figure 1) vérifieront aussi la contrainte de performance qui doit être satisfaite globalement à la fin de l'exécution du processus.

Par exemple, dans la figure 1, la position des trois Choix limites conduit à une performance de  $CL1 + CL2 + CL3 = 40 + 100 + 230 = 370$ . Toutes les combinaisons d'affectations de groupes d'employés ayant une performance, au niveau des activités, située en dessous des  $CL_i$  conduiront à une performance globale du processus inférieure à 370. De la même façon, toutes les combinaisons de groupes d'employés ayant une performance située au dessus des  $CL_i$ , conduiront à une performance nécessairement supérieure à 370.

### 4.3. Algorithme Dichotomique proposé

Les principales étapes de notre algorithme Dichotomique (DIC) de recherche de solutions d'affectation sont explicitées dans l'organigramme de la figure 4.

La valeur initiale du pointeur Choix Limite associé à chaque activité est tout d'abord fixée et permet de connaître la performance correspondante du processus. Ces pointeurs sont alors positionnés localement, activité par activité, en utilisant un algorithme Dichotomique classique, de façon à satisfaire au mieux l'objectif de performance imposé globalement. La démarche dichotomique, utilisée localement au niveau de chacune des activités, est très efficace pour trouver un élément satisfaisant un critère donné à l'intérieur d'une liste ordonnée de N items. Sa complexité en  $O(\ln N)$  est faible.

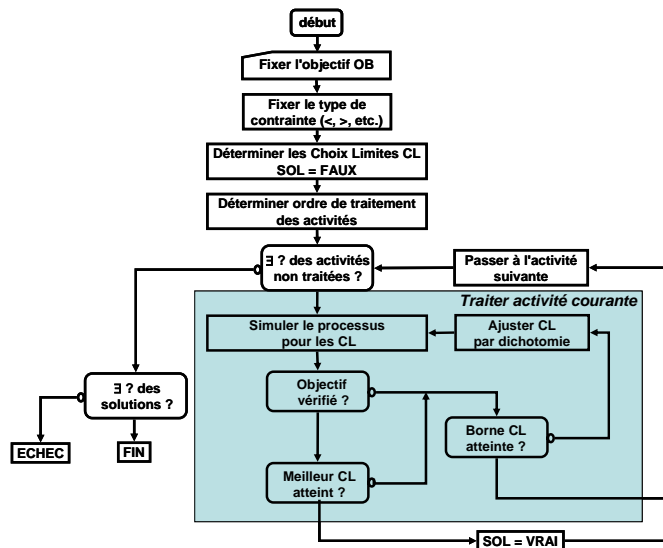


Figure 4. Algorithme d'affectation DIC basé sur une recherche locale Dichotomique

Le tableau 1 résume l'évolution de la démarche dichotomique sur l'exemple de la figure 1. Il permet d'identifier, pour chaque itération de l'algorithme, l'activité sur laquelle la progression dichotomique est envisagée, les valeurs de Choix Limite choisies pour chaque activité et la performance correspondante. Une simulation représente ici l'agrégation, pour un comportement donné du processus, des performances estimées associées à chacun des Choix Limites courants. Pour atteindre l'objectif de performance  $OB \leq 360$  à partir des Choix Limites initiaux (CL1, CL2, CL3) fixés comme précédemment à (40, 100, 230) 4 simulations sont nécessaires. La position des Choix Limites (CL1, CL2, CL3) est (15, 115, 230). Elle permet de déterminer  $2*6*3 = 36$  solutions globales d'affectation. Evidemment, dans le cas général l'égalité finale n'est pas toujours obtenue.

| N° itération | Activité Dicho | A1 | A2  | A3  | Perf | Commentaires        |
|--------------|----------------|----|-----|-----|------|---------------------|
| 1            | A1             | 40 | 100 | 230 | 370  | Init : Perf > OBJ   |
| 2            | A1             | 32 | 100 | 230 | 362  | Perf > OBJ          |
| 3            | A1             | 15 | 100 | 230 | 345  | Perf < OBJ : FIN A1 |
| 4            | A2             | 15 | 115 | 230 | 360  | Perf = OBJ : FIN    |

**Tableau 1.** Algorithme dichotomique : démarche basique

#### 4.4. Une heuristique pour l'ordre de traitement des activités

L'algorithme propose donc, une fois les Choix Limites initiaux fixés, de les ajuster localement un à un sans pour autant remettre en cause la valeur des autres. Il est donc évident que l'ordre de traitement des activités peut influencer notablement l'efficacité de l'algorithme. L'ordre de prise en compte des activités est donc fixé à l'aide d'une heuristique. Une fonction multicritère est employée. Elle utilise la performance moyenne, le gradient de performance (Perf. max – Perf. min) et le nombre de choix  $ng_i$  de chaque activité. En premier lieu sont traitées les activités possédant la performance moyenne la plus importante, présentant le moins de choix et le gradient le plus faible. Nous espérons ainsi fixer rapidement la valeur des Choix Limites pour les activités ayant la plus grande influence sur la performance (moyenne), mais un impact limité (gradient) et peu de choix envisageables.

En utilisant cette heuristique, l'exemple de la figure 1 conduit à traiter en premier l'activité 3 puis l'activité 2 et enfin l'activité 1. Cet ordre de traitement, illustré par le tableau 2, conduit pour l'objectif de performance  $OB \leq 360$ , aux Choix Limites finaux (CL3, CL2, CL1) de (200, 120, 40). Ils sont trouvés en 4 simulations et permettent de déterminer  $2*7*5 = 70$  solutions d'affectations possibles, soit près de deux fois plus que dans le cas où nous n'utilisons pas l'heuristique.

| N° itération | Activité Dicho | A3  | A2  | 1  | Perf | Commentaires        |
|--------------|----------------|-----|-----|----|------|---------------------|
| 1            | A3             | 230 | 100 | 40 | 370  | Init : Perf > OBJ   |
| 2            | A3             | 200 | 100 | 40 | 340  | Perf < OBJ : FIN A3 |
| 3            | A2             | 200 | 115 | 40 | 355  | Perf < OBJ          |
| 4            | A2             | 200 | 120 | 40 | 360  | Perf = OBJ : FIN    |

**Tableau 2.** Algorithme dichotomique : heuristique de traitement des activités

#### 4.5. Un backtrack pour améliorer l'efficacité de l'algorithme

La configuration de l'algorithme de recherche Dichotomique qui vient d'être proposée ne permet pas de reconsidérer la valeur finale d'un Choix Limite d'une activité une fois celle-ci traitée. Cependant, il peut être intéressant de revenir en arrière (DIC\_BK) si la première solution permettant de satisfaire l'objectif de performance imposé n'est trouvée qu'au niveau de la  $(k+1)^{ème}$  activité et donc après avoir considéré  $k$  activités. Dans ce cas, pour celles ci, les Choix Limites sont nécessairement positionnés au niveau de la valeur maximale, ou minimale d'une activité de façon à satisfaire au mieux l'objectif de performance recherché.

En reconsidérant ces  $k$  activités sans remettre en cause  $CL_{k+1}$  on peut alors espérer trouver de nouvelles positions finales pour leurs Choix Limites. Ce mécanisme limité de retour en arrière peut permettre de détecter plus de solutions que l'algorithme explicité plus haut, mais nécessite en contrepartie plus de simulations. Ce mécanisme n'est utilisé qu'une seule fois lorsque que la configuration spécifiée plus haut est identifiée. L'algorithme Dichotomique s'exécute ensuite normalement.

Considérons à nouveau l'exemple de la figure 1 pour un objectif de performance  $OB \leq 300$ . Le tableau 3 montre que l'utilisation de l'heuristique précédente.

| N° itération | Activité Dicho | A3  | A2  | A1 | Perf | Commentaires                |
|--------------|----------------|-----|-----|----|------|-----------------------------|
| 1            | A3             | 230 | 100 | 40 | 370  | Init : Perf > OBJ           |
| 2            | A3             | 200 | 100 | 40 | 340  | Perf > OBJ                  |
| 3            | A3             | 190 | 100 | 40 | 330  | Perf > OBJ : FIN A3         |
| 4            | A2             | 190 | 82  | 40 | 312  | Perf > OBJ                  |
| 5            | A2             | 190 | 80  | 40 | 310  | Perf > OBJ : FIN A2         |
| 6            | A1             | 190 | 80  | 32 | 302  | Perf > Obj                  |
| 7            | A1             | 190 | 80  | 15 | 285  | Perf < OBJ<br>FIN SANS BACK |
| 8            | A3             | 230 | 100 | 15 | 345  | Perf > OBJ                  |
| 9            | A3             | 200 | 100 | 15 | 315  | Perf > OBJ                  |
| 10           | A3             | 190 | 100 | 15 | 305  | Perf > OBJ                  |
| 11           | A2             | 190 | 82  | 15 | 287  | Perf < OBJ                  |
| 12           | A2             | 190 | 85  | 15 | 290  | Perf < OBJ : FIN BACK       |

**Tableau 3.** Algorithme dichotomique : démarche sans et avec backtrack

Cette heuristique conduit à la solution finale  $(CL3, CL2, CL1) = (190, 80, 15)$  ce qui correspond à seulement  $1*1*2 = 2$  configurations d'affectation calculées en 7 simulations. En faisant appel au mécanisme de retour en arrière on aboutit finalement aux valeurs  $(CL3, CL2, CL1) = (190, 85, 15)$  permettant d'atteindre  $1*3*2 = 6$  solutions en 12 simulations. Donc, dans cet exemple, l'algorithme DIC\_BK a permis d'obtenir trois fois plus de solutions que l'algorithme DIC, mais avec 5 simulations supplémentaires.

## 5. La démarche expérimentale

Cette partie décrit les benchmarks utilisés et la méthodologie d'évaluation suivie. Cette dernière permet d'étudier le comportement de l'algorithme Dichotomique présenté avec celui d'autres algorithmes classiques et de comparer, lorsque cela est possible, les résultats obtenus avec ceux trouvés par une démarche exacte exhaustive.

### 5.1. Description des benchmarks

Pour estimer l'efficacité de l'approche Dichotomique proposée, deux points de vue de la performance, relevant de classes d'objectifs différentes, sont étudiés. D'une part, l'aspect "Durée d'exécution du processus" est présenté. Il faut alors identifier les employés qui pourraient exécuter un processus au plus en un temps maximal imposé. Cela peut donc s'expliciter par  $Performance\_Processus \leq OB$ . D'autre part, l'aspect Qualité est étudié. Il faut dans ce cas déterminer les ressources humaines à même de garantir une performance au moins égale à un seuil donné. Ce qui peut être formulé sous la forme  $Performance\_Processus \geq OB$ .

Deux catégories d'exemples ont été employées. Pour le première (EX1, EX2 et EX3) un processus industriel simplifié nommé PAP (Processus de Mise en Production) a été utilisé (figure 5).

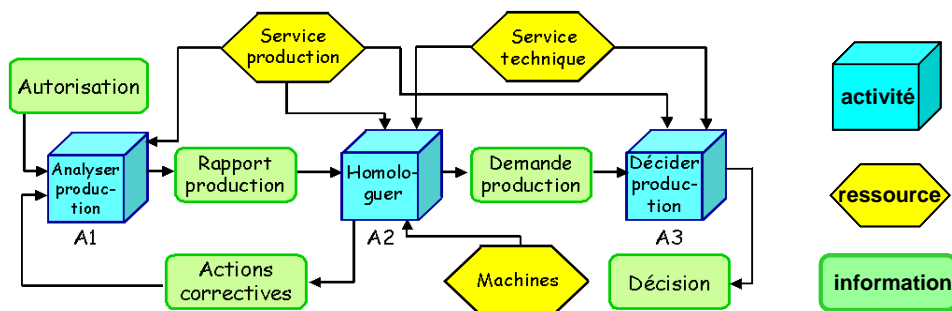


Figure 5. Processus de Mise en Production

Il est extrait du manuel qualité de notre partenaire industriel. Son but est d'autoriser, ou non, la mise en production d'un nouveau produit. Ce processus se décompose en 3 activités requérant des acteurs appartenant aux Services Technique et Production. Un ensemble de tests est défini au cours de l'activité A1. Ceux-ci sont ensuite réalisés par l'activité A2. Suivant les résultats obtenus, soit des actions correctives sont nécessaires avant homologation, soit une demande de mise en production est produite. Celle-ci est alors examinée durant l'activité A3 qui rend sa décision. Dans la suite, la séquence comportementale  $\sigma = \{A1, A2, A1, A2, A1, A2, A3\}$  où plusieurs itérations sont nécessaires avant la décision finale a été considérée.

Pour ce processus et ce comportement, les simulations numériques s'appuient sur une petite entreprise regroupant une trentaine de personnes réparties en plusieurs services. Les services Technique et Production sont composés de 3 équipes de 3 personnes relevant de qualification, d'expérience et de profils de compétence variés. De plus, les contraintes d'affectation définies permettent de générer, après la phase de filtrage du processus d'affectation (figure 3) un nombre important, mais calculable, de configurations d'allocation envisageables (tableau 4).

| Service | Activités |       |      |       |      |     | Nbre total de config.<br>Π config. |                      |
|---------|-----------|-------|------|-------|------|-----|------------------------------------|----------------------|
|         | A1        |       | A2   |       | A3   |     |                                    |                      |
|         | Prod.     | Prod. | Tech | Prod. | Tech |     |                                    |                      |
| EX1     | 9         | 9     | 3    | 9     | 9    |     | 19 683                             |                      |
| EX2     | 9         | 9     | 9    | 9     | 9    |     | 59 049                             |                      |
|         | A1        | A2    | A1   | A2    | A1   | A2  | A3                                 |                      |
| EX3     | 9         | 27    | 9    | 27    | 9    | 27  | 81                                 | 1 162 261 467        |
| EX4     | A1        | A2    | ...  | ...   | A14  | A15 |                                    |                      |
|         | 30        | 30    | ...  | ...   | 30   | 30  |                                    | $1.43 \cdot 10^{22}$ |

**Tableau 4.** Nombre d'allocations envisageables pour l'étude exhaustive

Ces contraintes d'affectations permettent de choisir explicitement les personnels des services Technique et Production pouvant être affectées à l'exécution des activités et précisent, si nécessaire, combien d'employés doivent être impliqués. Après vérification des contraintes de compétence et de qualification imposées il est possible d'identifier et de dénombrer, pour chacun des métiers relevant d'une activité, les groupes de personnes affectables. On connaît donc, en fonction des contraintes d'affectation imposées pour chacun des exemples développés (tableau 4), la liste de des personnels qui se substitueront aux ressources Service Production et Service Technique de la figure 5.

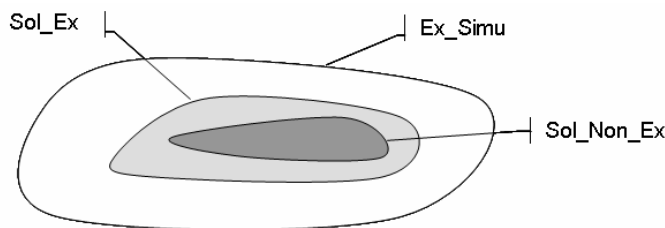
Pour la seconde catégorie d'exemple (EX4) un processus fictif composé de 15 activités exécutées séquentiellement a été construit. Chacune d'elle dispose de 30 possibilités différentes d'allocation dont les valeurs ont été attribuées aléatoirement mais de façon à rendre compte des distributions statistiques observées (distribution

uniforme ou non, variance, écart-type, etc.) sur le processus PAP selon des points de vue de performance Durée ou Qualité. Pour cette configuration il est impossible d'énumérer exhaustivement toutes les configurations d'affectation. Cet exemple correspond à notre avis à la limite de complexité qui peut être rencontrée pour un processus d'entreprise.

## 5.2. La méthodologie d'évaluation

### 5.2.1. Principes

Lorsqu'elle est possible, la simulation numérique exhaustive de toutes les combinaisons d'affectations envisageables permet de calculer, et de classer, selon un point de vue donné (Durée, Qualité), l'ensemble  $Ex\_Simu$  (figure 6) des performances pouvant être atteintes pour un comportement  $\sigma$  d'un processus. Dans cet ensemble, seules certaines d'entre elles satisfont l'objectif de performance  $OB$  imposé. Elles définissent alors l'ensemble exact des solutions  $Sol\_Ex$  (figure 6). Le recours à des algorithmes non exacts conduit alors nécessairement à un ensemble  $Sol\_Non\_Ex$  (figure 6) encore plus réduit de solutions. Cependant, cette réduction du nombre de solutions est compensée par une forte diminution du nombre de simulations nécessaires et donc du temps de calcul.



**Figure 6.** Les ensembles solution

Pour EX4, où l'exploration complète de l'espace des solutions n'est pas possible, les différents algorithmes testés pourront seulement être comparés entre eux. Pour les exemples de complexité limitée (EX1, EX2, EX3), la simulation exhaustive est utilisée pour évaluer l'efficacité des méthodes non exactes. Elle a été caractérisée par un indicateur : le taux de couverture.

### 5.2.2. Taux de couverture

Le taux de couverture représente, pour un objectif de performance donné  $OB$ , le pourcentage du nombre des affectations détectées par un algorithme donné par rapport au nombre exact de solutions. Il ne peut donc être défini que dans le cas où

la combinatoire du problème reste modérée. Le Taux de Couverture TC est défini par l'équation [8].

$$TC = \frac{\text{Nbre de solutions détectées par un algorithme}}{\text{Nbre total de solutions}} * 100 \quad [8]$$

Pour un ensemble de N objectifs on peut définir un Taux de Couverture Moyen  $TC_{Moyen}$  par l'équation [9] comme un indicateur pertinent du comportement de l'algorithme utilisé.

$$TC_{Moyen} = \frac{\sum_{i=1}^N TC_i}{N} \quad [9]$$

### 5.3. Description des algorithmes testés

Les algorithmes que nous avons comparés avec la démarche Dichotomique proposée sont maintenant rapidement décrits. Ils s'appuient tous sur la structure de données détaillée plus haut qui associe à chaque activité une liste ordonnée de couples (ressources humaines affectables, performance associée).

#### 5.3.1 Algorithme de Descente

L'algorithme de Descente (DES) est une méthode d'optimisation locale (Hao et al., 1999). La démarche suivante est utilisée :

- Initialisation pour chaque activité des CL initiaux. Afin d'assurer la satisfaction de l'objectif, on choisit le CL associé à la performance minimale (maximale) de l'activité pour un objectif de performance de type minimisation (maximisation).
- Calcul des performances du processus associées à tous les voisins de la configuration courante de CL. Un voisin ne diffère de la configuration courante que pour une activité où le CL suivant a été choisi.
- Le voisin pour lequel la performance globale associée améliore celle de la configuration courante en se rapprochant de la performance objectif sans la dépasser est sélectionné.

L'algorithme est itéré jusqu'à ce que la configuration courante ne puisse plus être améliorée. Ce type d'approche se bloque très facilement dans un minimum local.

### 5.3.2 Algorithme Recuit Simulé

Le Recuit Simulé (RS) est un algorithme méta-heuristique (Kirkpatrick et al., 1983). Il est très employé pour rechercher une solution approchée de problèmes à forte complexité combinatoire. En théorie, il évite à la fonction objectif de se trouver piégée dans un minima local. Il a été inspiré par la technique expérimentale de recuit utilisée par les métallurgistes pour simuler l'évolution d'un système physique instable vers un état d'équilibre thermique à une température  $T$  fixée (Metropolis et al., 53). Cet algorithme est très sensible au choix des paramètres de contrôle.

Les caractéristiques de l'algorithme utilisées sont les suivantes :

- Température initiale : 1
- Décroissance géométrique de la température :  $T_{K+1} = 0.88 T_K$
- Probabilité associée à la règle de Métropolis :  $P = \exp[-(\Delta f / T)]$ .  $\Delta f$  correspond à la différence d'évaluation de performance entre la configuration courante et celle du meilleur voisin.
- Génération des voisins : Au lieu d'utiliser la démarche employée pour l'algorithme de descente (choix du plus proche CL) une fonction dédiée a été développée de façon à accélérer le processus de recherche locale en fonction de la distance séparant la configuration courante de l'objectif.
- Solution initiale : CL minimaux ou maximaux des activités en fonction de l'objectif choisi (minimisation ou maximisation).
- Arrêt : Trois itérations sans évolution.

### 5.3.3 Algorithme de Pisinger

La section 3.3 a montré que le problème étudié dans cet article peut être formulé sous la forme classique d'un problème de sac à dos à choix multiple (Bennour et al., 2005). Pour résoudre ce problème NP-difficile, l'algorithme proposé par Pisinger (Pisinger, 1995) est utilisé pour déterminer très rapidement une solution optimale.

Il se décompose en deux parties pratiquement indépendantes. La première dérive des travaux de Dyer (Dyer, 1984) et Zemel (Zemel, 1984). Elle permet l'identification d'une solution approchée. Pour cela, les solutions dominées sont éliminées et un ensemble de techniques élaborées de tri vis-à-vis du poids des objets permettent d'obtenir un ensemble de solutions approchées.

A partir de ces dernières, la seconde partie de l'algorithme, développée par Pisinger à partir de considération sur la typologie de données, identifie des centres de classe (ou core) contenant les objets les plus prometteurs d'où est extraite la solution optimale. Celle-ci permet de satisfaire au plus près l'objectif fixé sans chercher à maximiser le nombre de solutions. Cela se traduit souvent par des CL placés aux positions extrémales des domaines de performance des activités.



Pratiquement le code développé par Pisinger dans ses travaux est disponible (Pisinger, 2006). Il a pu être directement employé sur les données associées aux différents exemples évalués.

#### 5.4. Approche exacte : Résultats

L'approche exacte permet d'évaluer avec précision l'efficacité des algorithmes heuristiques pour les benchmarks de complexité limitée. Les résultats de l'approche exhaustive sont résumés dans le tableau 5 ci-dessous. Ils ont été obtenus sur un processeur Pentium 4 à 2 GHz avec 1Go de RAM.

|     | Nombre de Simulations | Taux de Couverture (TC en %) | Temps d'Exécution (s) |
|-----|-----------------------|------------------------------|-----------------------|
| EX1 | 19 683                | 100                          | $4.16 \cdot 10^{-3}$  |
| EX2 | 59 049                | 100                          | $12.3 \cdot 10^{-3}$  |
| EX3 | 1 162 261 467         | 100                          | 240                   |

**Tableau 5.** Performances de l'approche exhaustive

L'approche exhaustive permet d'identifier l'ensemble complet de toutes les solutions satisfaisant un objectif de performance. Une couverture de 100 % est donc atteinte. Pour des contraintes d'affectation données, cette méthode permet de conclure sur toute la plage d'objectif de performance envisageable d'un processus. Evidemment, cette approche est coûteuse en temps puisque pour EX3 l'on atteint 4 minutes de temps CPU. Précisons ici que les valeurs indiquées dans le tableau 5 n'intègrent donc pas le temps nécessaire à la mémorisation et au classement de toutes les configurations d'affectations envisagées. Pratiquement le temps consacré à cette étape peut devenir prohibitif sur des exemples de grande combinatoire comme EX3. On peut considérer que pratiquement, pour résoudre des problèmes d'affectation apparaissant au niveau opérationnel, un temps de calcul supérieur à une dizaine de minutes semble rédhitoire. Par ailleurs, il est souhaitable de limiter au maximum l'attente de l'utilisateur du logiciel pour qui un temps de réponse excédant une dizaine de secondes peut aussi sembler trop important. EX3 était à la limite de nos capacités de calcul car il nécessite près de 5.5 Goctets pour mémoriser et classer l'ensemble de toutes les valeurs de performance générées exhaustivement. Sans avoir fait appel à des techniques efficaces de classement, le traitement de EX3 a pris pratiquement plusieurs heures.

Les démarches heuristiques qui vont être maintenant étudiées devraient réduire le coût de calcul mais parviendront-elles à un taux de couverture acceptable ?

## 6. Benchmarks de complexité limitée : résultats

Dans la suite les résultats obtenus pour les classes d'objectifs ( $\text{Performance\_Processus} \leq \text{OB}$ ) et ( $\text{Performance\_Processus} \geq \text{OB}$ ) sont présentés. Ils utilisent les valeurs de performance obtenues, selon un point de vue Durée d'une part, et Qualité d'autre part, par l'approche d'estimation de performance (Covès, 2000) (Bennour, 2004) résumée au début de cet article. Dans chacun des cas, le comportement de l'algorithme Dichotomique proposé est d'abord globalement analysé. L'exemple EX2 permet d'illustrer l'évolution du taux de couverture et de comparer le comportement des algorithmes testés. L'efficacité de l'approche Dichotomique est ensuite analysée en comparant le taux de couverture et le temps d'exécution maximal des algorithmes. Les algorithmes sont employés pour déterminer les solutions d'affectation de 40 objectifs régulièrement espacés dans le domaine de performance du processus. Une courbe comparative du comportement (nombre de solutions) des algorithmes évalués est proposée. Pour chacun d'entre eux, les valeurs du taux de couverture et du temps de calcul sont détaillés.

### 6.1. Performance Durée : Critère d'affectation $\text{Performance\_Processus} \leq \text{OB}$ .

#### 6.1.1 Comportement global

La figure 7 présente le comportement du taux de couverture obtenu par l'approche Dichotomique sur EX2 selon un point de vue Durée.

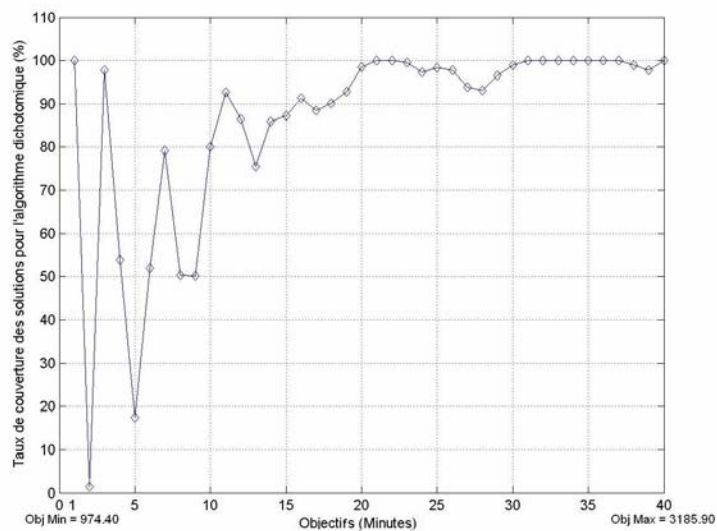
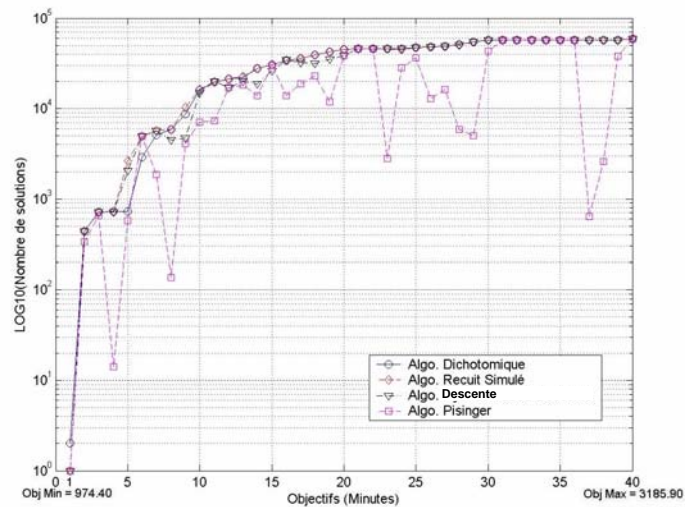


Figure 7. EX2 – Point de vue Durée – Taux de Couverture (TC)

L'axe des abscisses représente la plage d'objectifs sur laquelle l'algorithme a été utilisé. Nous avons donc cherché combien de solutions d'affectation étaient trouvées pour 40 objectifs différents allant d'une Durée  $\leq$  Objectif(1) = 974,4 minutes, à une Durée  $\leq$  Objectif(40) = 3186,9 minutes. Ce double indicage est indiqué au niveau de l'axe des abscisses. La connaissance du nombre exact de solutions calculé exhaustivement permet alors de déduire le taux de couverture atteint. Ce dernier est représenté en ordonnée de la figure 7 pour les 40 objectifs.

Pour un nombre moyen de simulations voisin de 10, la figure 7 montre que le taux de couverture est supérieur à 90% sur plus de 60% du domaine de recherche (partie haute de la figure 7). En revanche lorsque les contraintes temporelles pour l'exécution du processus deviennent plus exigeantes, l'algorithme a plus de difficultés pour trouver des solutions. Le taux de couverture présente alors un comportement plus erratique (partie basse de la figure 7).

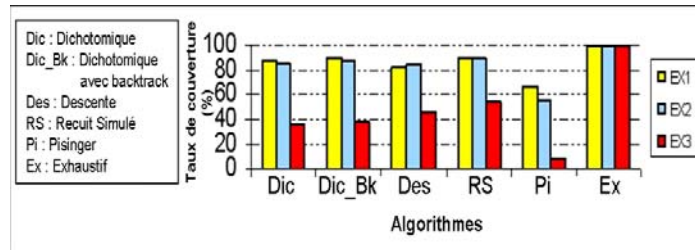
La figure 8 présente, pour tous les algorithmes étudiés, l'évolution du nombre de solutions trouvées en fonction de l'objectif imposé. Elle montre bien que plus ce dernier est exigeant, moins il existe de solutions le satisfaisant. On constate que tous les algorithmes, sauf celui de Pisinger, présentent sensiblement le même comportement.



**Figure 8.** EX2 – Point de vue Durée – Comportement des algorithmes

### 6.1.2 Comportement détaillé

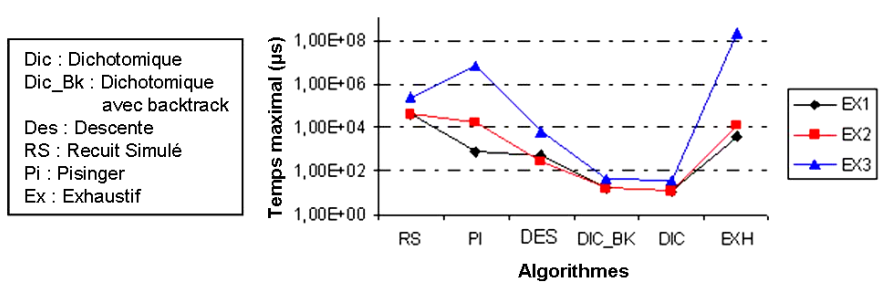
Les figures 9 et 10 comparent, pour l'ensemble des algorithmes étudiés, la valeur moyenne du taux de couverture et le temps d'exécution maximal observé pour trouver une solution.



**Figure 9.** EX2 – Point de vue Durée – Comparaison TC Moyen

Pour EX1 et EX2 on peut constater que les taux de couverture obtenus pour tous les algorithmes sauf celui de Pisinger sont voisins et supérieurs à 80%. En revanche pour EX3 qui présente une combinatoire plus élevée que les autres, les approches heuristiques négligent un volume plus grand de solutions et le taux de couverture chute nettement en devenant inférieur à 50%. L'utilisation du backtrack permet, pour l'algorithme Dichotomique de gagner près de 2% de taux de couverture au détriment de quelques simulations supplémentaires. Cependant les algorithmes trouvent toujours au moins une solution à proposer. L'algorithme de Pisinger, pour sa part est particulièrement inefficace et voit son taux de couverture chuter à 7%.

Par souci de clarté, pour comparer les temps d'exécution maximaux observés (figure 10) une échelle logarithmique est adoptée pour l'axe des ordonnées afin de représenter distinctement la différence entre les courbes. On constate que l'approche exhaustive présente évidemment le temps d'exécution le plus élevé. Il faut cependant noter que, pour les exemples EX1 et EX2, le temps de calcul de l'approche exhaustive reste du même ordre que ceux des algorithmes heuristiques à l'exception de l'exemple EX3. Pour ce dernier un fort décalage est observé en raison de la combinatoire importante. Ce constat est renforcé par le fait que les temps affichés n'intègrent pas, pour le calcul exact, le temps nécessaire à la mémorisation et au classement des données.



**Figure 10.** EX2 – Point de vue Durée – Comparaison Temps Max

### 6.1.3. Conclusion

L'analyse du comportement des différents algorithmes dans le cadre d'un objectif de performance de type  $\text{Performance\_Processus} \leq \text{OB}$  comme pour la Durée montre, qu'hormis Pisinger, toutes les approches heuristiques atteignent un taux de couverture moyen acceptable, compris entre 40% et 90%. Le temps de calcul maximal reste faible et nettement inférieur à la seconde.

En revanche l'algorithme de Pisinger ne satisfait ni l'un ni l'autre de ces critères en dépassant entre autre la seconde en durée de calcul. En effet, cet algorithme optimise globalement la performance d'un processus (fonction objectif) afin de remplir au mieux le sac à dos, en se positionnant au niveau de chaque activité, de façon à approcher, au plus près, la valeur limite de performance imposée pour le processus. Ce n'est qu'ensuite que nous déduisons le nombre de solutions détectées. Il ne conduit donc pas nécessairement à un grand nombre de solutions d'affectation.

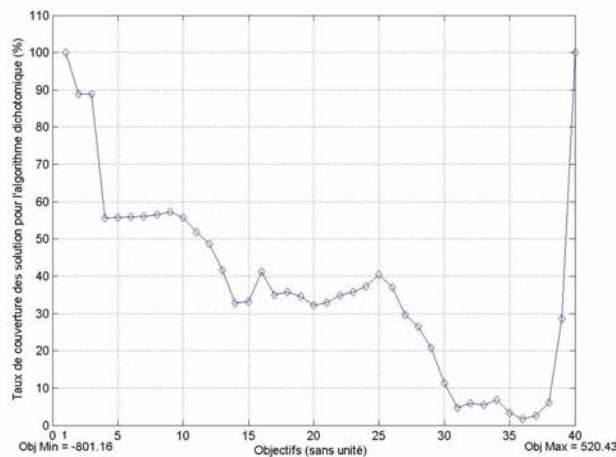
On constate qu'avec l'augmentation de la combinatoire (EX3), on observe une diminution du taux de couverture. Cela semble normal puisque des algorithmes heuristiques ne peuvent, en général, couvrir la totalité de l'espace solution.

Enfin, même si le taux de couverture de l'approche Dichotomique chute un peu plus nettement que les autres sur EX3, son temps d'exécution maximal reste le plus faible, et de l'ordre d'une quarantaine de microsecondes.

## 6.2. Performance Qualité : Critère d'affectation $\text{Performance\_Processus} \geq \text{OB}$

### 6.2.1. Comportement global

Le taux de couverture des solutions observé pour l'approche Dichotomique sur l'exemple EX2 selon un point de vue Qualité est illustré par la figure 11.



**Figure 11.** EX2 – Point de vue Qualité – Taux de Couverture (TC)

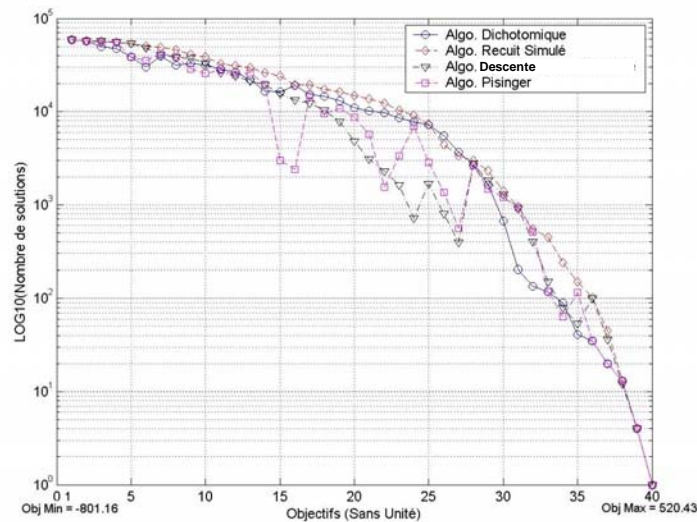
Le taux de couverture est plus bas que celui obtenu pour l'étude de performance Durée. Une réduction du taux de couverture est observée au fur et à mesure que l'objectif imposé devient plus difficile à satisfaire.

Trois zones peuvent être distinguées. La première où près de 60% des objectifs présentent un TC compris entre 30% et 60%. La deuxième où TC est inférieur à 30% et qui couvre près de 30% des objectifs. Enfin, pour la dernière, un taux supérieur à 60% est observé pour un nombre réduit d'objectifs.

Par rapport à un point de vue Durée, ces moins bons résultats peuvent être dus à la sensibilité des algorithmes vis-à-vis des valeurs numériques manipulées. En effet pour l'aspect Qualité on observe une répartition plutôt continue des valeurs de performance alors qu'elle évolue plutôt par paliers pour l'aspect Durée. Cette hypothèse demanderait à être confirmée par l'étude d'autres exemples.

Cette difficulté à trouver des solutions est aussi traduite par un nombre moyen de simulations voisin de 18 très supérieur à celui observé pour le point de vue Durée.

La figure 12 présente l'évolution du nombre de solutions détectées par les algorithmes sur la plage de performance Qualité sur EX2. On distingue deux parties. La première (des objectifs 1 à 15) où l'ensemble des algorithmes présentent globalement le même comportement. La deuxième (des objectifs 15 à 40) où les algorithmes ont plus de difficulté à trouver des solutions et où leurs comportements se différencient. Lorsque les solutions sont plus difficiles à rechercher, ils semblent donc plus ou moins adaptés à les identifier. Les algorithmes de Recuit Simulé et Dichotomique sont plus performants que les autres dans cette zone.



**Figure 12.** EX2 – Point de vue Qualité – Comportement des algorithmes

6.2.2. Comportement détaillé

Les figures 13 et 14 comparent, pour tous les algorithmes étudiés, la valeur du taux de couverture et du temps d'exécution maximal. Pour les exemples EX1 et EX2 le TC moyen est compris entre 33% et 54%. On constate que c'est l'algorithme du Recuit Simulé qui obtient les meilleurs résultats (figure 13) avec 54% avec EX1, 52% pour EX2, et 37% pour EX3. Pour l'algorithme Dichotomique, l'utilisation du backtrack est inefficace puisque l'augmentation du nombre de simulation ne produit aucun gain en taux de couverture. On note aussi, comme pour le point de vue Durée, que le TC moyen chute nettement pour l'exemple EX3 en raison de sa forte combinatoire. Le taux de couverture de l'algorithme de Descente est même si faible qu'il ne peut être visualisé sur l'histogramme de la figure 13.

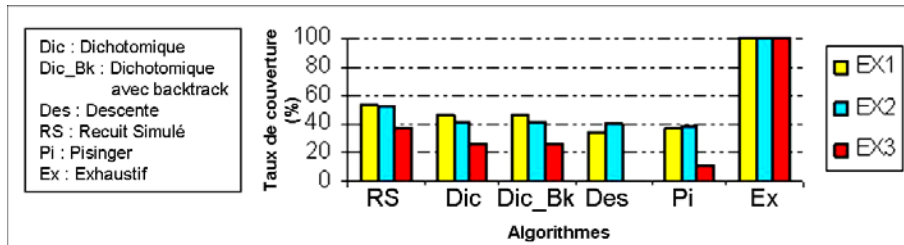


Figure 13. EX2 – Point de vue Qualité – Comparaison TC

Pour l'étude du temps d'exécution, comme pour le point de vue Durée, une échelle logarithmique est utilisée pour l'axe y afin de faciliter l'analyse de ce paramètre (figure 14).

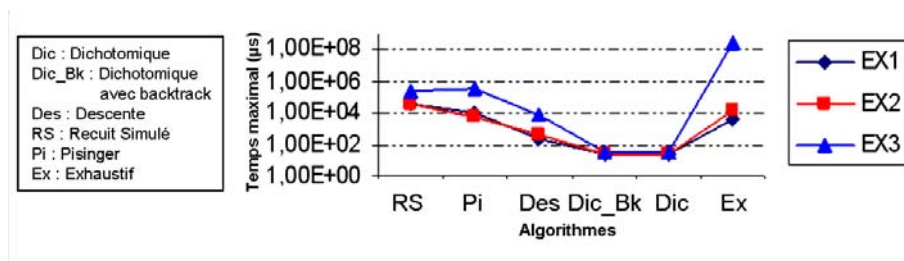


Figure 14. EX2 – Point de vue Qualité – Comparaison Temps Max

Sans même intégrer le temps nécessaire à la mémorisation et au classement des données, l'approche exhaustive présente évidemment le temps d'exécution le plus élevé. Cette observation est flagrante pour l'exemple EX3 où on atteint près de 4 minutes de temps CPU alors que pour les EX1 et EX2 les temps d'exécution restent proches de la milliseconde.

### 6.2.3. Conclusion

L'analyse du comportement des différents algorithmes pour un objectif de performance  $\text{Performance\_Processus} \geq \text{OB}$  comme la Qualité montre que tous les taux de couverture sont au-dessous de 60% pour les exemples traités. On note donc, la difficulté des algorithmes à trouver les solutions satisfaisant un objectif de performance pour ce type de problématique. Ce phénomène peut être lié pour l'axe Qualité à la distribution plus uniforme de performance constatée au niveau des activités.

Cependant tous les algorithmes heuristiques ont un temps de calcul maximal acceptable bien inférieur à la seconde. L'algorithme du Recuit Simulé présente les meilleurs résultats en terme de taux de couverture.

Après les exemples de complexité limitée, les comportements des différents algorithmes heuristiques vont maintenant être présentés et discutés sur l'exemple virtuel EX4 dont on ne peut explorer exhaustivement l'espace des solutions.

## 7. Benchmarks de grande complexité : Résultats

Nous ne cherchons ici qu'à évaluer la performance des algorithmes étudiés qu'en fonction du nombre de solutions détectées et de la rapidité de détection et non en fonction de leur pertinence. Il conviendrait de développer des heuristiques permettant d'identifier les solutions les plus pertinentes en ne retenant par exemple, pour un objectif donné, que les plus dominantes et les plus différenciées.

La figure 15 montre l'évolution du nombre de solutions trouvées par chaque algorithme dans le cas d'un objectif de type  $\text{Performance\_Processus} \leq \text{OB}$  (Durée).

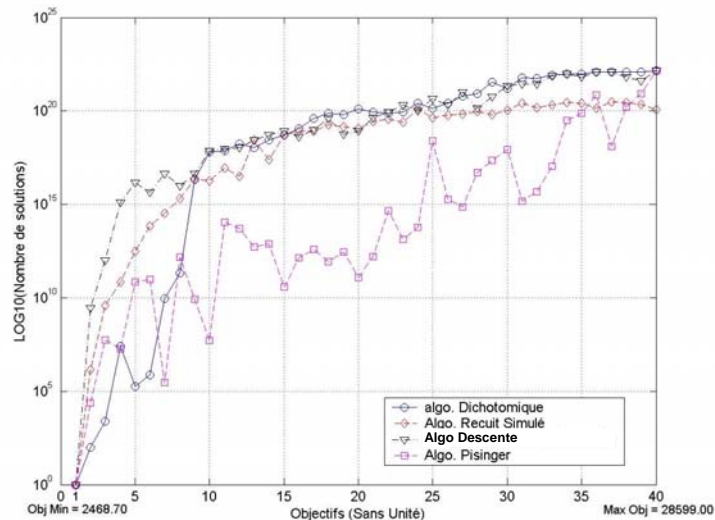
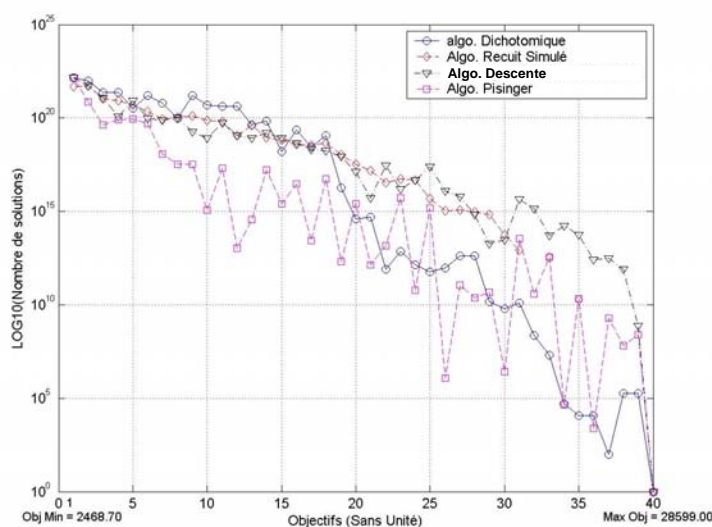


Figure 15. EX4 : Comparaison des algorithmes pour  $\text{Performance\_Processus} \leq \text{OB}$



Les algorithmes Dichotomique, Recuit Simulé et de Descente donnent des résultats équivalents pour la partie supérieure de la courbe (à partir du 10<sup>ème</sup> objectif). Dans sa partie inférieure, où les solutions sont plus difficiles à trouver, l'algorithme Dichotomique est moins efficace que le Recuit Simulé et la Descente. Néanmoins il reste meilleur que l'algorithme de Pisinger. Ce dernier donne très nettement les plus mauvais résultats.

La figure 16, illustre le comportement des algorithmes pour un objectif de type Performance\_Processus  $\geq$  OB (Qualité) où notre approche est plus en difficulté. Le plus mauvais résultat est encore obtenu avec l'algorithme de Pisinger.



**Figure 16. EX4 :** Comparaison des algorithmes pour Performance\_Processus  $\geq$  OB

Les figures 17 et 18 permettent de comparer les résultats obtenus sur EX4. Puisqu'il n'est pas possible de comparer l'approche Dichotomique à l'approche exhaustive, l'analyse est réduite à deux critères, le nombre de simulations et le temps d'exécution de chaque algorithme.

La figure 17 permet de constater que globalement le nombre de simulations moyen observé pour une problématique de minimisation (Durée) et de maximisation (Qualité) d'objectif est sensiblement identique.

Pour l'algorithme Dichotomique, le nombre de simulations nécessaires demeure limité et inférieur aux autres algorithmes excepté Pisinger.

Pour le Recuit Simulé, le nombre de simulations reste élevé et sensiblement identique à celui observé pour les exemples de complexité limitée. Il semble donc assez insensible à l'augmentation de la combinatoire.

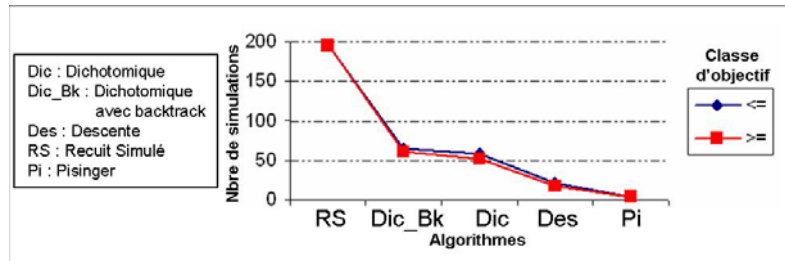


Figure 17. EX4 : Comparaison du nombre de simulations des algorithmes

La figure 18 montre que l'approche Dichotomique garde un temps d'exécution très faible (quelques microsecondes) par rapport à celui des autres algorithmes ( $> 0,1$  seconde), pour atteindre même 3 secondes pour l'algorithme DES. En revanche, une problématique de maximisation (Qualité) semble poser beaucoup de problèmes à l'algorithme de Pisinger en multipliant par 5 son temps de résolution par rapport à celui observé pour une logique de minimisation.

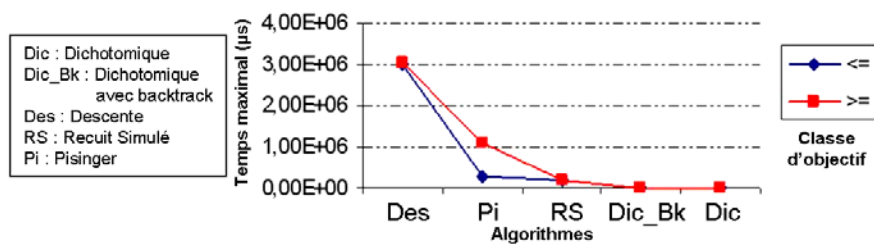


Figure 18. EX4 : Comparaison du temps d'exécution des algorithmes

Tous les algorithmes évalués sur cet exemple de très grande complexité ont permis l'identification d'un nombre important de solutions en un temps qui reste acceptable. Cependant, même si le taux de couverture moyen ne peut être expérimentalement calculé il est quand même possible de conclure que l'algorithme de Recuit Simulé reste globalement le plus performant mais que l'approche Dichotomique demeure cependant la plus rapide.

## 8. Conclusion

Cet article se focalise sur le problème d'allocation des ressources humaines à un processus d'entreprise. Ce dernier est considéré comme un ensemble ordonné d'activités dont les performances (durée, coût, qualité, etc.) ne sont pas préalablement connues. Quel que soit le point de vue considéré, la performance associée est déterminée en faisant appel à une méthodologie d'estimation de

performance, développée avec l'industriel Merlin Gerin, prenant en compte explicitement les compétences et caractéristiques des employés affectés. Cela permet, contrairement à la grande majorité des travaux du domaine, d'intégrer réellement l'impact de la ressource humaine sur le travail plutôt que de le réduire uniquement à une ressource disponible ou non. On cherche alors quel personnel peut être affecté pour satisfaire un objectif de performance préalablement fixé.

Une démarche Dichotomique locale a été proposée et comparée avec d'autres algorithmes heuristiques (Recuit Simulé, Pisinger, Descente) sur des benchmarks de moyenne et grande complexité.

Les algorithmes Dichotomique et du Recuit Simulé obtiennent les meilleurs résultats sur les exemples de complexité limitée. Les résultats les plus modestes sont ceux de l'algorithme Pisinger. Dans le cas de l'exemple de grande complexité, il est possible de conclure que les algorithmes Descente et Recuit Simulé sont plus efficaces que les algorithmes Dichotomique et Pisinger par rapport au nombre de solutions détectées. Néanmoins, l'approche Dichotomique a le plus court temps d'exécution. Globalement, elle présente donc de bons résultats. Elle est de plus facile à mettre en oeuvre et ne nécessite que peu de paramètres à fixer si on la compare au Recuit Simulé.

Ces approches heuristiques, de par leur faible temps de calcul, s'imposent d'elles mêmes pour traiter les processus présentant une forte complexité combinatoire puisque l'approche exacte nécessite un temps d'exécution CPU mais surtout de traitement des données prohibitif. Elles permettent, quelque soit la classe d'objectif considérée (maximisation, minimisation), l'identification d'un nombre significatif de solutions d'affectation, même si l'on constate une diminution sensible du taux de couverture pour le point de vue Qualité. Ce phénomène est, nous semble-t-il, liée à la répartition statistique des niveaux de performance pour les activités en fonction du point de vue retenu. Enfin, les mauvais résultats de l'algorithme de Pisinger, surtout en terme de taux de couverture, sont liés non pas à son inadéquation au problème du sac à dos, mais au fait qu'il cherche avant tout à satisfaire au mieux à l'objectif de performance imposé, ce qui ne garanti pas l'identification d'un nombre conséquent de solutions d'affectation.

De nombreuses pistes de travail peuvent prolonger cette étude.

Tout d'abord, il semble indispensable de poursuivre l'effort d'intégration de l'entité humaine dans les problématiques du génie industriel et particulièrement de l'affectation des employés. La modélisation qui est utilisée ici reste limitée et discutable selon certains aspects même si elle a été construite en collaboration avec un industriel. Il est évidemment illusoire, de par sa nature même, de vouloir modéliser l'entité humaine de façon totalement déterministe. Il faudrait donc étudier de quelle façon, et dans quelles limites, une fuzzification des comportements humains et de la performance pourrait être intéressante et les implications que cela aurait sur l'affectation du personnel.

La méthodologie d'affectation de ressources humaines développée ici doit, elle-aussi, être améliorée. Il est indispensable d'y intégrer la prise en compte explicite de la disponibilité des employés à travers leurs plannings d'affectations en considérant non plus un unique processus comme dans le cadre de cet article, mais l'ensemble des processus opérationnels de l'entreprise. Cela permettrait alors d'aborder les problématiques liées au pilotage opérationnel des processus. Nous avons déjà proposé dans (Froget, 2005) un algorithme d'affectation permettant d'adresser cette classe de problème. Il permet d'obtenir rapidement les plannings individuels d'affectation des employés en intégrant leurs contraintes de disponibilités (présence/absence, partage) mais aussi les contraintes légales (35 H). Il s'appuie sur la démarche d'affectation proposée dans cet article.

Il est aussi important de l'enrichir en permettant à l'utilisateur de définir des objectifs de performance intermédiaires à satisfaire. Ceux-ci permettront très certainement de réduire fortement le nombre de solutions envisageables. Enfin elle doit s'adresser simultanément à plusieurs classes de fonctions objectifs, associées à des points de vue complémentaires mais parfois antinomiques de la performance des processus industriels (Durée, Financier, Qualité, etc.) (Bennour et al., 2005).

Enfin, pour caractériser les algorithmes, des critères quantitatifs associés au volume de solutions trouvées et au nombre de simulations nécessaires ont été étudiés. Il faudrait aussi les comparer vis-à-vis de critères qualitatifs associés à la pertinence de solutions. Ceux-ci pourraient alors concourir à la définition de nouvelles heuristiques cherchant à favoriser, en plus du nombre de solutions recherchées, leur qualité. Dès lors, il serait intéressant de caractériser les algorithmes en identifiant ceux qui, pour une classe d'objectif donnée (performance maximale à ne pas dépasser ou minimale à satisfaire) et une plage de performance associée, donneraient les meilleurs résultats en termes tant quantitatifs que qualitatifs.

## 9. Bibliographie

- Alfares H. K., « Dual-based optimization of cyclic four-day workweek scheduling », *IMA Journal of Mathematics Applied in Business and Industry*, 11, 2000, p. 269-283.
- Aykin T., A « Comparative Evaluation of Modelling Approaches to the Labour Shift Scheduling Problem », *European Journal of Operational Research*, 125, 2000, p. 381-397.
- Baptiste P., Giard V., Hait A., Soumis F., *Gestion de production et ressources humaines, Méthodes de planification dans les systèmes productifs*, Presses Internationales Polytechniques, Québec, Canada, 2006.
- Bard J. F., Binici C., Desilva A. H., « Staff scheduling at the United States postal service », *Computers and Operations Research*, 30, 2003, p. 745-771.
- Bennour M., Crestani D., "Using competencies for performance estimation : From activity to the process", *Computer In Industry*, 58, 2007, p. 151-163.

- Bennour M., Crestani D., Crespo O., Prunet F., “Computer Aided Decision for Human Task Allocation with Mono and Multi Performance Evaluation”, *International Journal of Production Research*, 43(21), 2005, p. 4559-4588.
- Bennour M., Contribution à la modélisation et à l’affectation des ressources humaines dans les processus, Thèse de doctorat, Université de Montpellier 2, 2004.
- Chen H.G., « Heuristic for Operator Scheduling in Group Technology cells », *Computers and Operations Research*, 27, 1995, p. 261-276.
- Covès C., Analyse et Estimation de Performances de Processus d'Entreprise, Thèse de doctorat, Université de Montpellier 2, 2000.
- Dantzig G. B., « A comment on edie's traffic delays at tool booths », *Operation Research*, 2, 1954, p. 339-341.
- Drejer A., « How can we define and understand competencies and their development? », *Technovation*, 21, 2001, p. 135-146.
- Dyer M. E., « An O(n) algorithm for multiple-choice knapsack linear program » *Mathematical Programming*, 29, 1984, p. 57-63.
- Eclipse, Constraint Logic Programming System, Imperial college London, <http://www.icparc.ic.ac.uk/eclipse>, 2004.
- Franchini L., Caillaud E., Nguyen P., Lacoste G., « Planning and Scheduling Skills: a Case Study in an Agri-food Industry », *Proceedings of IEEE Conference of Systems, Man and Cybernetics*, 1998, p. 417- 422.
- Franchini L., « Aide à la décision pour la gestion des opérateurs de production », Thèse de doctorat, Institut National Polytechnique de Toulouse, 2000.
- Franchini L., Caillaud E., Nguyen, Lacoste G., « Workload Control of Human Resources to Improve Production Management », *International Journal of Production Research*, 39, 2001, p. 1385-1403.
- Froget N., Affectation des ressources humaines et pilotage opérationnel des processus d'entreprise, D.E.A., Université Montpellier, Juin 2005.
- Grabot, B. and Letouzy, A., « Short-term manpower management in manufacturing systems: new requirement and DSS prototyping », *Computer in Industry*, 2000, 43, p. 11-29.
- Guerin N, Drapeau C., Melançon S., « Planification stratégique des ressources humaines. Guide d'accompagnement », Québec (Canada), Mai 1992.
- Guest D. E., « Human Resource Management and Performance: A review and Research Agenda », *International Journal of Human Resource Management*, 8(3), 1997, p. 263-276.
- Hao J., Galinier P., Habib M., « Métaheuristiques pour l’optimisation combinatoire et l’affectation sous contraintes ». *Revue d'Intelligence Artificielle*, 1999.
- Hung R., « Scheduling a workforce under annualized hours » *International Journal of Production Research*, 37, 1999, p. 2419-2497.

- Jia T., Vers une meilleure gestion des performances d'un groupe autonome de fabrication, Thèse de doctorat, Université de Tours, 1998,
- Kane H., Baptiste P., « Un modèle d'ajustement de la capacité à la charge de travail basé sur les ressources humaines », *Actes de la troisième conférence francophone de MODélisation et SIMulation MOSIM'01*, Troyes (France), 25-27 Avril 2001, Vol. 2, p. 723-727.
- Kulanoot A., « Algorithms for some hard knapsack problems », Ph.D. Thesis, Curtin University of Technology, Australia, 2000.
- Kusiak A., Chen M., « Expert System for Planning and Scheduling Manufacturing Systems », *European Journal of Operational Research*, 34, 1988, p. 113-130.
- Kirkpatrick S., Gelatt, C. D., Vecchi, P.M., « Optimisation by simulated annealing », *Science* 220, 1983, p. 671-680.
- Lengnick-hall C. A., Lengnick-hall M. L., « Strategic Human Resources Management : A Review of the Literature and a Proposed Typology », *Academy of Management Review*, 13, 1988, p. 454-470.
- McCarthy B. L., Wilson J. R., Crawford S., « Human performance in industrial scheduling: a framework for understanding », *Human Factors and Ergonomics in Manufacturing*, 11(4), 2001, p. 299-320.
- McEwan A. M., Sackett P., « The Human Factor in CIM Systems : worker empowerment and control within a high-volume production environment », *Computer in Industry*, 1998, p. 39-47.
- Mosheiov G., « Scheduling Problem with a Learning Effect », *European Journal of Operational Research*, 132, 2001, p. 687-693.
- Narasimhan R., « Algorithm for a multiple shift scheduling of hierarchical workforce on four day or three-day workweeks. », *INFOR*, 38(1), 2000, p. 14-32.
- Norman B. A., Tharmmaphornphilas W., Needy K. L., Bidanda B., and Warner R. C., « Worker assignment in cellular manufacturing considering technical an human skill », *International Journal of Production Research*, 40(6), 2002, p. 1479-1492.
- Pisinger D., Algorithms for Knapsack Problems, Ph.D. Thesis, University of Copenhagen, Denmark, 1995.
- Pisinger D., Codes source en langage C, <http://www.diku.dk/~pisinger/codes.html>, 2006.
- Soubeiga E., « Development and application of hyperheuristics to personnel scheduling », PhD Thesis, University of Nottingham, 2003.
- Tchomo J. L., Baptiste P., Soumis F., « Etude bibliographique de l'ordonnancement simultané des moyens de production et des ressources humaines », Proceedings of the fifth French-Quebec International Congress of Industrial Engineering, Quebec, Canada, 2003.
- Vairaltarakis G. L., Cai X., Lee C.Y., « Workforce Planning in Synchronous production Systems », *European Journal of Operational Research*, 136, 2002, p. 551-572.
- Zemel E., « An O(n) algorithm for the linear multiple-choice knapsack program and related problems », *Information Processing Letters*, 18, 1984, p. 123-128.