

Le problème de dimensionnement des lots avec capacité avec les décisions d'heures supplémentaires et les temps de configuration

LINET O'ZDAMAR et MEHMET ALI BOZYEL

Département de génie industriel, Université Dog˘us, Gayrettepe Emekli Subay Evleri, 23/5 Istanbul, Turquie Courriel : lozdamar@hotmail.com

Reçu en septembre 1996 et accepté en septembre 1999

Le problème de capacité de dimensionnement des lots (CLSP) consiste à planifier les tailles de lots de plusieurs articles sur un horizon de planification dans le but de minimiser les coûts d'installation et de détention des stocks. À chaque période de production d'un article, un coût d'installation est engagé. La capacité est limitée et homogène. Ici, le CLSP est étendu pour inclure les décisions d'heures supplémentaires et les configurations consommatrices de capacité. La fonction objectif consiste à minimiser les coûts de détention des stocks et des heures supplémentaires. Les configurations entraînent implicitement des coûts via les coûts des heures supplémentaires, c'est-à-dire qu'elles entraînent des coûts d'heures supplémentaires supplémentaires lorsque les temps de configuration contribuent à l'utilisation de la capacité d'heures supplémentaires au cours d'une certaine période. Le problème qui en résulte devient plus compliqué que le CLSP standard et nécessite des méthodes différentes de celles proposées pour ce dernier. Par conséquent, de nouvelles approches heuristiques sont développées pour traiter ce problème. Parmi les approches heuristiques figurent l'approche HPP classique et ses modifications, une approche itérative omettant les variables binaires dans le modèle, une approche d'algorithme génétique basée sur la formulation de type transport du modèle de planification de la production d'un seul article avec une demande dynamique et une approche de recuit simulé basée sur le déplacement de la taille des lots familiaux au cours de périodes consécutives. Les résultats de calcul démontrent que l'approche de recuit simulé produit des programmes de haute qualité et est la plus efficace en termes de calcul.

1. Introduction

Le problème de dimensionnement de lot capacité (CLSP) à une étape est un problème NP-difficile (Bitran et Yanasse, 1982) dans la planification de la production qui suscite un intérêt continu parmi les chercheurs (Kirca, 1990 ; Kirca et Ko˘kten, 1994 ; Lambrecht et Vanderveken, 1979 ; Maes et van Wassenhove, 1986). Dans le CLSP, les demandes d'articles déterministes variant dans le temps sont supposées avoir lieu sur un horizon de planification fini, une capacité limitée est partagée par tous les articles produits au cours de chaque période et aucune commande en souffrance n'est autorisée. Chaque période de production d'un article entraîne un coût d'installation. La fonction objective minimise les coûts d'installation et d'inventaire. Dans certaines formulations, les configurations consomment également de la capacité (Kim et Mabert, 1995 ; Trigeiro et al., 1989). Les méthodes de résolution heuristique proposées pour ce problème impliquent généralement une approche période par période (Dixon et Silver, 1981 ; Dog˘ramaci et al., 1981 ; Gu˘nther, 1987 ; Lambrecht et Vanderveken, 1979) où la taille des lots pour les éléments sont déterminés sur la base d'un critère d'économie de coûts. Chaque période, la demande future est programmée pour être produite au cours de cette période jusqu'à ce qu'aucun coût supplémentaire ne puisse être économisé ou que la capacité de cette période soit épuisée. Il existe également des approches heuristiques alternatives basées sur la programmation mathématique (Cattrysse et al., 1990 ; Thizy et van Wassenhove, 1985 ; Trigeiro, 1987). Kirca et

Ko˘kten (1994) propose une approche itérative appelée l'approche article par article où une itération consiste à ordonnancer un sous-ensemble d'articles consommant progressivement la capacité disponible à chaque étape.

Le modèle proposé ici étend le CLSP à deux égards : (i) il suppose deux sources de capacité, le temps régulier et les heures supplémentaires, ce qui entraîne une perte d'homogénéité de la capacité ; Dixon et al. (1983) considèrent les décisions supplémentaires dans le problème de dimensionnement des lots avec capacité, mais leur cas est limité à un seul élément ; (ii) les installations consomment de la capacité et n'entraînent pas de coûts autres que les coûts de main-d'œuvre.

Les coûts supplémentaires liés aux configurations, tels que les coûts de mise au rebut ou de matériel de nettoyage, ne sont pas comptabilisés. Par conséquent, les coûts d'installation ne sont pas explicitement indiqués dans la fonction objectif. Au contraire, les configurations n'entraînent des coûts que si elles contribuent au fil du temps. Ainsi, la fonction objectif consiste en la tenue des stocks et les coûts des heures supplémentaires tandis que les contraintes garantissent que la demande est satisfaite à chaque période et que les capacités régulières de temps et d'heures supplémentaires ne sont pas dépassées. En fait, nous préférons appeler ce problème le problème de dimensionnement des lots multifamiliaux capacitifs avec des décisions supplémentaires, car si les éléments finaux avec les exigences de configuration similaires sont regroupés en familles, nous avons une formulation mathématique beaucoup plus compacte du modèle donné dans le tableau 1 et le nombre de variables binaires diminue considérablement. Le modèle proposé a un large domaine d'application à la fois dans les industries de transformation et

Tableau 1. Formulation mathématique du problème de dimensionnement des lots familiaux capacitaires avec décisions supplémentaires et temps d'installation

Modèle P : $\min \sum_t P_t - \sum_j l_j t_j + \sum_t P_{ot} t_j$

sujet à

$l_j t_j \leq Y_j t_j - l_j t_j \frac{1}{4} d_j t_j \quad 8 \leq j \leq t;$

$P_{ot} \leq \delta_j Y_j t_j + w_j t_j \leq C_t \leq O_t \quad 8 \leq t;$

$O_t \leq C_t \quad 8 \leq t$

$Y_j t_j \leq M w_j t_j \quad 8 \leq j \leq t; w_j t_j \leq \frac{1}{4}$

$0 \leq 1; Y_j t_j \leq l_j t_j; O_t \leq 8 \leq j \leq t;$

où

$d_j t_j$ ¼ demande de la famille j à la période t ; p_j ¼

temps de traitement par unité de famille j δ_j ¼ ; ov ¼

coût des heures supplémentaires par

heure ; h_j ¼ coût de possession par unité de la famille j par période ;

C_t ¼ capacité de temps régulier à la période t ;

C_{Ot} ¼ capacité d'heures supplémentaires en période t ;

$l_j t_j$ ¼ inventaire de la famille j détenu à la fin de la période t ;

$Y_j t_j$ ¼ taille du lot de production de la famille j à la période t ;

O_t ¼ capacité d'heures supplémentaires utilisée dans la

période t ; s_j ¼ de temps d'installation requis pour la famille j

$w_j t_j$ ¼variable binaire ¼ 1; si la famille j est produite à la

période t ; 0 ; sinon

M ¼ un grand nombre :

dans la fabrication discrète. Lorsque la fabrication discrète est considérée, les variables représentant les quantités de production doivent être intégrales. Cependant, dans le modèle P, nous ignorons ces contraintes d'intégralité, car le modèle devient beaucoup plus difficile à résoudre et de plus, puisque nous nous concentrons sur les approches heuristiques à utiliser en pratique, arrondir les quantités de production valorisées réelles semble être une approche de bon sens.

Le modèle implique un compromis entre les heures supplémentaires et les coûts de possession des stocks, tandis que les configurations peuvent entraîner ou non des heures supplémentaires. Ce compromis se produit parce que, pendant les périodes où les quantités demandées entraînent une capacité inutilisée, il peut être avantageux de produire plus et de conserver des stocks plutôt que de produire les quantités demandées en utilisant la capacité des heures supplémentaires pendant les périodes de pointe. Naturellement, ces décisions dépendent du rapport entre le coût unitaire de possession et le coût unitaire des heures supplémentaires. Le caractère implicite des coûts d'installation complique le processus d'optimisation, en particulier lorsque la capacité est modérément abondante. D'autre part, lorsque les contraintes de capacité sont assez fortes, il est difficile de trouver des solutions réalisables. En raison des configurations consommatrices de capacité, il n'est pas facile de concevoir des mécanismes de faisabilité de rétroaction/anticipation tels que ceux décrits dans Maes et van Wassenhove (1985) qui permettent des heuristiques période par période pour éviter les violations de capacité. En effet, la faisabilité

Le problème du CLSP avec des temps d'établissement est NP-Complet comme montré par Garey et Johnson (1979).

2. Méthodes de résolution

Les approches de solution proposées ici ne consistent pas en des règles de priorité période par période "remplir jusqu'à capacité", en raison de problèmes d'infaisabilité qui peuvent survenir dans les périodes futures en raison des temps de configuration et aussi parce qu'il est difficile de déduire des règles ce qui conduirait à des solutions quasi optimales lorsque la capacité n'est pas homogène. Tout en augmentant la taille des lots des familles, la décision dépend de la prévision des goulots d'étranglement de capacité qui se produiront dans les périodes futures en raison de l'augmentation de la demande et s'il est plus avantageux de conserver des stocks dans la période actuelle au lieu d'allouer (probablement des heures supplémentaires) la capacité à une configuration supplémentaire dans la future période de goulot d'étranglement. Dans l'existence de plusieurs familles à programmer sur l'horizon de planification, la décision devient assez compliquée. Ainsi, pour traiter ce problème, nous proposons un certain nombre d'approches différentes, y compris l'approche Hierarchical Production Planning (HPP), une approche de relaxation itérative, un algorithme génétique (GA) et une approche de recuit simulé (SA). Dans les sections suivantes, nous décrivons ces approches de résolution et comparons leurs résultats aux meilleurs (optimaux) sur trois ensembles de problèmes de test générés aléatoirement.

2.1. L'approche hiérarchique de la planification de la production

Dans l'approche HPP, nous agrégeons les demandes familiales de chaque période en une demande « type » sur l'horizon de planification et calculons la moyenne pondérée des temps de traitement des familles par rapport à leurs demandes pour aboutir à un temps de traitement agrégé unique. De plus, en réservant un pourcentage de la capacité de production régulière pour les temps d'installation, nous évitons les variables d'installation binaires explicites dans les contraintes de capacité et aboutissons au modèle P1 du tableau 2. Le pourcentage d'allocation d'installation, A, est calculé en multipliant le rapport du total du temps de préparation au temps de production total requis par le nombre prévu de saisons, N, dans l'horizon de planification qui est identifié au cours de la phase de prévision du processus de planification.

2.1.1. La procédure de remplissage

du modèle P1 est d'abord résolue pour déterminer les quantités de production de type agrégé (et par conséquent, les décisions en matière d'heures supplémentaires) sur l'horizon de planification. Ensuite, les tailles de lot des familles déclenchantes sont calculées pour chaque période avec leur somme toujours égale à la quantité optimale de production de type agrégé. Une famille se déclenche à la période t si sa borne inférieure l_{bjt} est positive. (l_{bjt} est égal à la demande nette (=max{0, d_{jt} - l_{jt-1} }) pour la famille j dans la période t.) La désagrégation de la famille est la faisabilité en utilisant la procédure de remplissage (FP) discutée

Tableau 2. Formulation mathématique agrégée utilisée dans l'approche HPP

Modèle P1 : $\min P \sum_t \delta h_l t \sum_j p_{ov} O_t p$

sujet à

$$I t_1 \sum_j p_{Xt} I t_1 \sum_j D t_8 t ;$$
$$P X t \sum_j \delta_1 A p C t \sum_j p O t_8 t ;$$
$$O t C O t_8 t ;$$
$$X t ; I I ; O t_0 8 t ;$$

où

$$D t \sum_j \frac{1}{4} \text{ demande de type agrégée à la période } t \sum_j \frac{1}{4} X d j t ;$$
$$P \sum_j \frac{1}{4} \text{ temps de traitement global par type d'unité}$$
$$\sum_j \frac{1}{4} X \sum_j X d j t p j X \sum_j \frac{1}{4} X d j t ;$$
$$\text{Un pourcentage d'allocation d'installation de } \frac{1}{4} \text{ pour la capacité}$$
$$\sum_j \frac{1}{4} N X s j X \sum_j X d j t p j \text{ où}$$
$$N \sum_j \frac{1}{4} \text{ nombre de saisons dans l'horizon de planification ;}$$
$$h \sum_j \frac{1}{4} \text{ coût de détention global } \sum_j \frac{1}{4} X \sum_j X d j t h j X \sum_j \frac{1}{4} X d j t ;$$
$$I t \sum_j \frac{1}{4} \text{ des stocks agrégés détenus à la fin de la période } t ;$$
$$X t \sum_j \frac{1}{4} \text{ quantité de production agrégée pendant la période } t ;$$

dans des travaux antérieurs (O'zdamar et al. 1996, 1998). Cependant, dans ces derniers articles, la fonction objectif consiste simplement à minimiser le nombre d'installations annuelles (= $\sum_j p_{sj} u_{bj}/y_j$ où u_{bj} est la borne supérieure familiale égale à la demande nette jusqu'à la fin de l'horizon de planification). Dans ce cas, aucune restriction de capacité n'existe et la désagrégation de la famille est effectuée une fois pour une seule période, par conséquent l'indice t est supprimé de u_{bj} et la taille du lot familial y_j .

Les contraintes sont utilisées pour faire correspondre la quantité de production totale à X^* et pour limiter la taille des lots familiaux individuels dans l'intervalle défini par l_{bj} et u_{bj} . Ainsi, FP est exécuté une fois pour une seule période et on suppose que X^* est connu d'une manière ou d'une autre. FP augmente la taille des lots familiaux un par un avec des incréments prédéfinis $d_j = (u_{bj} - l_{bj})/SN$, où SN est un numéro d'étape défini par l'utilisateur, jusqu'à ce que la somme de toutes les tailles de lots familiaux soit égale à X^* . Dans le but de réduire le nombre d'installations, les familles sont sélectionnées pour augmenter leurs tailles de lot à chaque itération selon une mesure de gradient approximative j représente l'économie sur les coûts d'installation et est donnée par $j = (s_j u_{bj}/y_j) - s_j u_{bj}/(y_j + 1)$.

Une solution optimale est obtenue par FP pour chaque instance de test générée aléatoirement dans O'zdamar et al. (1996) lorsque le Step Number, SN , qui contrôle l'ampleur des incréments de taille de lot, est fixé à 500. pas été mesuré par les auteurs.

De plus, FP garantit qu'aucune commande en souffrance n'a lieu en seulement deux périodes : la période actuelle t et la dernière période, T , de l'horizon de planification. Cela est dû au fait que les tailles de lot sont limitées entre leurs limites inférieure et supérieure, qui tiennent compte de la demande nette de la période en cours et de la somme de la demande nette jusqu'à la période T , respectivement. Dans FP, la plus grande taille de lot est généralement attribuée à la famille avec le plus grand j .

Si cette famille a une petite demande au cours de la prochaine période et qu'une famille avec une plus petite a une plus grande demande, alors, FP ne peut pas prévoir la situation et détient un inventaire pour la plus grande famille j et entraîne des commandes en souffrance pour l'autre famille au cours de la prochaine période. Par conséquent, FP n'a aucune garantie sur les commandes en souffrance dans les périodes comprises entre la commande en cours et la dernière.

Dans le but d'éliminer les commandes en souffrance tout au long de l'horizon de planification, dans cette étude, une procédure de recherche/couverture est ajoutée à FP. La procédure ajoutée vérifie s'il existe des reliquats dans la période de planification actuelle et, si c'est le cas, les quantités de production des familles qui ont un inventaire positif au cours de cette période sont transférées par ordre croissant de $s_j u_{bj}$ à celles avec des reliquats. Si les familles avec un stock positif dans la période en cours ne peuvent pas couvrir tous les reliquats, alors les familles avec un stock positif dans la période précédente sont utilisées pour couvrir le déficit et leurs tailles de lot sont transférées vers les tailles de lot des familles déficitaires de la période précédente. De cette manière, le déficit est couvert en recherchant les familles avec un inventaire positif dans les périodes antérieures jusqu'à ce que la première période soit atteinte. Puisqu'il n'y a pas de commandes en souffrance dans le plan global, la procédure de recherche/couverture garantit de couvrir toutes les commandes en souffrance.

Dans le contexte multi-période considéré ici, FP est exécuté une fois dans chaque période de l'horizon de planification (d_j, l_{bj}, u_{bj} gagnent tous un indice de t et sont recalculés à chaque période) et si des reliquats existent dans une période t , le Search / La procédure de couverture est exécutée, puis FP reprend la désagrégation au cours de la période suivante $t + 1$. Les pseudocodes du programme principal HPP et de la procédure FP se trouvent dans le tableau 3.

2.1.2. La procédure de remplissage modifiée

Étant donné que FP ne prend pas en compte les coûts d'inventaire et d'heures supplémentaires tout en augmentant la taille des lots, nous modifions FP pour aboutir à la FP modifiée, notée MFP, en modifiant la règle de sélection de la famille afin de prendre en compte explicitement le compromis entre l'inventaire et frais d'heures supplémentaires. Les familles sont sélectionnées pour incrémenter la taille de leurs lots dans l'ordre croissant de z_{jt} où

$$z_{jt} = \frac{1}{4} X t_{m/4t} + I_0 j m h_j = \delta o v \delta p j d j t \sum_j p s j p ;$$

où I_0 est le stock qui aurait été détenu au cours de la période m $j m$ si la taille du lot de la famille j avait été augmentée de d_{jt} au cours de la période courante t ; t est la première période de goulot d'étranglement où des heures supplémentaires seront nécessaires compte tenu du

Tableau 3. Pseudocodes pour le programme principal HPP, FP, MFP et Search/Cover

<div>Début du programme principal du HPP Résoudre le modèle agrégé P1 ; Lire X pour t ¼ 1::T pour t = 1 jusqu'à T do begin Calculer lbjt, ubjt, djt ; (lbjt = demande nette pendant la période t ; ubjt = demande nette de la période t à la période T) PF (MFP); Calculer l'inventaire ; Si X0 < 0 alors Recherche/Couverture ; Vérifier la faisabilité de la capacité ; fin; Calculer le coût ; fin;</div>	
<div>Début de la procédure de remplissage {Procédure de remplissage modifiée} Identifier JTt : ensemble de familles déclenchées ; {JTt :toutes les familles} ; X0 ¼ X t ; yjt = lbjt pour tout jeJTt ; Réduire la capacité ; Réduire la quantité distribuable X0 ¼ X0 t t P jeJT yjt Si X0 < 0 alors yjt = 0 pour tout jeJTt ; Redistribuer X > 0 puis t parmi jeJTt par ordre décroissant de j ; Si X0 t répéter Identifier JC :ensemble de familles dans JTt telles que (yjt+djt £ ubjt) et (djt £ X0)(et (il existe une capacité suffisante)) ; Trier JC dans l'ordre décroissant de j ; {Trier JC par ordre croissant de zjt} ; Incrémenter yjt, où jt est l'indice supérieur dans la liste JC, de djt ; réduire la capacité ; Réduire X0 ; jusqu'à JC = B; Si X0 > 0 alors répéter Identifier JC : ensemble de familles dans JTt tel que yjt<ubjt ; Trier JC dans l'ordre décroissant de j ; {Trier JC par ordre croissant de zjt} ; Incrémenter yjt, où jt est l'indice supérieur dans la liste JC, de min {ubjt) yjt, X0) yjt} ; Réduire X0 ; t jusqu'à la t ¼ 0 ; fin X0 ; Procédure Recherche/Couverture ; commencer Identifier le montant total de la commande en souffrance, B ¼ XQ ; Identifier JB : ensemble de familles avec ljt < 0 ; Identifier JI : ensemble de familles avec ljt > 0 ; r = t ; répéter répéter Identifier JJlr : ensemble de familles dans JI avec yjr et ljr > 0 ; Trier JJlr dans l'ordre croissant de j ; {Trier JJlr par ordre décroissant de zjr} ; Transférez min{yjr,)lkt, ljr} de la famille j en tête de liste aux familles ke JB avec lkt < 0 une à la fois par ordre d'indice ; {Réduire B ; Mettre à jour yjr, ljr, ykr, lkt pertinents } jusqu'à (B = 0) ou (JJlr = B); r = r) 1 ; jusqu'à B = 0 ; fin;</div>	

demandes dans les périodes futures. La règle de priorité représente le compromis entre le coût de détention causé par un incrément de taille de lot djt dans la période courante t, à supporter jusqu'à la période de goulot d'étranglement t ϕ , et le coût des heures supplémentaires qui serait supporté dans le cas où une quantité de production djt est

affecté à la famille j dans la période de goulot d'étranglement imposant un temps d'établissement.
Une caractéristique distinctive de MFP est que non seulement les tailles de lot des familles déclenchées augmentent à chaque période, mais que toutes les familles sont soumises à une attribution de taille de lot positive. Le

La raison en est qu'en vertu des restrictions de capacité, les familles non déclenchées peuvent également avoir besoin de continuer à constituer des stocks si leur demande dans un avenir proche est assez élevée et supérieure à la capacité disponible.

Étant donné que les temps de réglage ne sont pas traités de manière exacte dans le modèle agrégé P1 (pour connaître le montant exact de la tolérance pour les temps de réglage à chaque période, nous devons d'abord déterminer la solution optimale !), mais par une estimation grossière de A, il peut y avoir des violations de capacité dans la solution familiale désagrégée fournie par FP et MFP lorsque la quantité de production agrégée, X est respectée \sum_t , chaque période. Par conséquent, nous incluons un contrôle de capacité dans MFP, mais pas dans FP, afin de préserver l'algorithme FP original suggéré dans Ozdamar et al. (1996). Dans la famille MFP, les tailles de lot sont incrémentées si la somme des capacités régulières et supplémentaires n'est pas dépassée. Cependant, si X^* ne peut pas être totalement réparti entre les familles en raison de restrictions de capacité, la taille des lots est alors augmentée sans activer le contrôle de capacité. Ainsi, le contrôle de capacité inclus dans le MFP ne garantit pas la faisabilité de la capacité tant que le plan global est strictement respecté. Dans le tableau 3, les modifications apportées sur FP sont indiquées en caractères gras entre parenthèses et celles-ci conduisent à MFP.

2.1.3. L'approche HPP itérative

Les performances de l'approche HPP ne dépendent pas seulement de la sélection des familles pour l'incrémentement de la taille des lots, mais aussi de la précision du modèle agrégé P1. Ce dernier est en partie basé sur la proximité du pourcentage d'allocation d'installation A par rapport au pourcentage optimal. La valeur de A peut être trop restrictive ou trop grande. Pour surmonter cette difficulté, une procédure itérative HPP est proposée, notée IHPP, où la valeur de A part de zéro et est progressivement augmentée à chaque itération. Une solution est obtenue en résolvant le modèle agrégé P1 avec la capacité de temps régulière d'origine, puis en utilisant MFP pour désagréger les quantités de production de type résultantes entre les familles à chaque période. Ensuite, les heures supplémentaires causées uniquement par les temps de réglage sont calculées pour chaque période, additionnées et divisées par le nombre de périodes dans l'horizon de planification, T, ce qui donne les heures supplémentaires moyennes causées par les réglages, RCD. Dans l'itération suivante, la capacité temporelle régulière de chaque période est diminuée de RCD. Le modèle P1 est résolu avec cette dernière capacité réduite et MFP trouve une autre solution. Ces itérations se poursuivent jusqu'à ce que le coût d'une solution soit supérieur au meilleur coût de l'opérateur historique, ou que le modèle agrégé P1 aboutisse à une solution irréalisable, ou que la solution trouvée par MFP viole la somme de la capacité régulière et des heures supplémentaires. L'objectif de la procédure de solution itérative est d'obtenir des pics d'utilisation de la capacité plus faibles et de lisser le profil d'utilisation de la capacité sur l'horizon de planification afin de réduire le nombre d'heures supplémentaires causées par les configurations. La réduction de la capacité de temps régulier permet non seulement d'économiser sur l'ampleur de la taille des lots, mais peut également réduire le nombre d'installations ayant lieu au cours de l'horizon de planification.

Mathématiquement, RCD s'exprime par :

$$RCD = \frac{1}{T} \sum_t (X_t - C_t) \quad \text{Ctg} = T ;$$

où

CRT = besoins en capacité y compris les temps de réglage résultant de la solution MFP en période t, = quantité optimale
 O_t d'heures supplémentaires en période t résultant de la solution du modèle P1.

Le pseudo-code du programme principal de l'approche IHPP intégrant le MFP est donné dans le tableau 4.

2.1.4. Un exemple

Un exemple illustratif avec quatre familles à programmer sur un horizon de planification de six périodes est donné dans le tableau 5. Le coût des heures supplémentaires est égal à 17/heure et les capacités régulières et supplémentaires sont toutes deux de 2800 heures par période. Le coût optimal pour cet exemple est identifié comme 59 708 par le package d'optimisation Lindo. Le modèle agrégé P1 est construit avec un temps de traitement agrégé de 2,7 heures/unité, et la capacité de temps régulier est réduite à 2 471,8 heures par l'allocation de configuration A qui est calculée à 11,75 % en utilisant une estimation de trois saisons dans l'horizon de planification. Les coûts totaux de détention et d'heures supplémentaires sont respectivement de 14,1 et 17. La demande est agrégée en 1488, 1103, 327, 904, 1306 et 924 unités, respectivement de la première à la sixième période. Le modèle agrégé P1 est résolu pour aboutir aux quantités de production et aux stocks trouvés dans le tableau 6.

Tableau 4. Pseudocode pour le programme principal de l'approche HPP itérative avec MFP

Début du programme principal de l'IHPP
C0 ← 1/4 C _t ;
CoûtMin ← 1 ;
Résoudre le modèle agrégé P1 avec C _t ;
répéter
Lire X _t , O _t pour t ← 1::T
pour t ← 1 à T
commencer
Calculer lbjt, ubjt, djt ;
MFP ;
Calculer l'inventaire ;
Si X ₀ < 0 alors Recherche/Couverture ;
Vérifier la faisabilité de la
capacité ; fin;
Calculer le coût, RCD ;
Si (Coût < CoûtMin) et (Capacité Réalisable) alors
CoûtMin ← Coût ;
C0 ← 1/4 C0 + RCD ;
Résoudre le modèle agrégé P1 avec C0 ; jusqu'à
ce que (le modèle P1 ne soit pas réalisable) ou (la solution MFP ne respecte pas
Capacité) ou (Coût > MinCost) fin ;

Tableau 5. Données de l'exemple

Famille j	sj	pj	hj	dj1	dj2	dj3	dj4	dj5	dj6
	148	3	11	481	279	185	355	132	322
	183	2	19	171	183	29	132	383	319
1	195	4	10	200	344	64	73	460	175
2									
3									
4									

Les itérations de l'approche HPP avec PF sont démontrées dans le tableau 7 pour la période 3 où SN = 3. Jusqu'à la période 3, la taille du lot de chaque famille est exactement égale à sa demande, puisqu'aucun inventaire n'est tenu pour la première et la deuxième période de la période. solution agrégée. Dans la période 3, la deuxième famille est augmentée de son pas à la deuxième itération puisqu'elle a le j le plus grand et le reste de est X_3 affecté à la quatrième famille à la troisième itération, puisque tous les pas sont supérieurs à $X_0 \frac{1}{4}$ 108:5 et devient 1,75 après y23 est fixé à 308. HPP + FP trouve une solution réalisable de capacité avec un coût de 69 587.

L'activation de la procédure de recherche/couverture est démontrée dans le tableau 8 où le paramètre SN est modifié à 100 et l'exemple est à nouveau résolu pour découvrir qu'à la cinquième période, la troisième famille a un inventaire négatif et les autres ont un inventaire positif. La procédure de recherche/couverture commence à transférer les tailles de lot par ordre croissant (sj ubjt). A savoir, les stocks des quatrième et deuxième familles sont ramenés à zéro en période 5 et leurs lots excédentaires sont réduits. Étant donné que la taille totale du lot transféré à la troisième famille (32 + 16,48) n'est pas suffisante pour couvrir le déficit de (53,44), et que la première famille

Tableau 6. Solution du modèle agrégé P1

Période	1	2	3	4	5	6
X t	1488	1103	714,55	915,480	915,48	0
je _t	0					

a une taille de lot de zéro à la cinquième période, la procédure réduit la taille de lot de la première famille à la quatrième période du montant nécessaire et la transfère à la troisième famille à la quatrième période alors qu'elle ne s'était pas déclenchée à ce moment-là.

La solution trouvée par la procédure HPP + MFP est illustrée dans le tableau 9 où SN = 100. La procédure de recherche/couverture n'a pas été activée pendant le processus de résolution. La période de goulot d'étranglement constatée par MFP est la cinquième période et les coûts de détention des stocks sont calculés en conséquence. Notez que bien que la troisième famille ne soit pas déclenchée dans la période 4, elle est produite. La solution a un coût de 71 517.

Les solutions agrégées et les capacités ajustées trouvées par l'approche IHPP à chaque itération sont présentées dans le tableau 10. Notez qu'à chaque itération, un niveau de production plus lisse est obtenu tout au long de l'horizon de planification. Le meilleur coût trouvé par l'IHPP est de 73 407.

2.2. L'approche itérative pour le problème original

Une deuxième approche consiste à traiter le modèle linéaire mixte entier original P en ignorant les expressions de consommation de configuration dans les contraintes de capacité et en excluant le dernier ensemble de contraintes liées aux variables binaires, re

Tableau 7. Itérations faites par HPP + FP en troisième période

Famille	j	lbj3	ubj3	djt3	yj3 (1ère itération)	yj3 (2ème itération)	yj3 (3ème itération)
	4,23	185	994	270	185	185	185,00
1	181,00	29	863	279	29	308	308,00
2	36,00	64	772	236	64	64	64,00
3	39,00	49	832	261	49	49	157,55

Tableau 8. L'activation de la procédure Search/Cover dans la cinquième période avec SN = 100

	Période 1	Période 2	Période 3	Période 4	Période 5	Inventaire à la période 5
yjt avant recherche/couverture						
Famille 1	481	279	185,00	500,48	0,00	13,48
Famille 2	171	183	191,00	0,00	369,48	
Famille 3	200	344	185,55	0,00	358	
Famille 4	636	297	153,00	415,00	188,00	16,48)53,44 32,00
yjt après recherche/couverture						
Famille 1	481	279	185,00	495,51	0,00	8,51
Famille 2	171	183	191,00	0,00	353,00	0,00
Famille 3	200	344	185,55	4,96	406,44	0,00
Famille 4	636	297	153,00	415,00	156,00	0,00

Tableau 9. Solution trouvée par HPP + MFP

	Période 1	Période 2	Période 3	Période 4	Période 5	Période 6
Famille 1 481		279 311,00	339,00	22,00	322,00	183 110,00 123,00
Famille 2 171		311,00	319,00	344 188,55	110,48	304,48 168,48
Famille 3 200		297 105,00	343,00	278,00	1 06.00	
Famille 4 636						

résultat dans le modèle linéaire P2. Le modèle P2 exclut complètement les temps de configuration et, par conséquent, les temps de configuration sont gérés via les approches itératives décrites dans l'IHPP. Le pseudo-code de l'approche itérative pour le problème original (IAOP) est donné dans le tableau 11. Une approche similaire à l'IAOP est utilisée avec succès par Özdamar et Yazgac, (1997) pour générer un plan directeur de production au niveau familial pour une famille. premier fabricant d'armoires de cuisine en Turquie.

En considérant l'exemple du tableau 5, la procédure IAOP aboutit aux couples réguliers capacité, coût suivants : (2801, 77 494), (2309, 67 233) et (1954, 71 469) et s'arrête en trois itérations. La valeur soulignée est le coût le plus bas.

2.3. L'approche de l'algorithm génétique

Les algorithmes génétiques (AG) sont des techniques de recherche robustes qui ont été appliquées à une variété de problèmes combinatoires en recherche opérationnelle (Michalewicz, 1994), en particulier dans le domaine de l'ordonnancement. Les principaux agents d'un GA sont le codage chromosomique, qui affecte les performances du GA en termes de qualité de la solution et d'efficacité de calcul, et les opérateurs de reproduction, de croisement et de mutation, qui permettent au GA de s'intensifier vers de bonnes solutions et de se diversifier pour éviter les problèmes locaux. opti ma. Un GA applique la reproduction, le croisement et la mutation à plusieurs reprises pendant un nombre prédéfini de générations après avoir généré de manière aléatoire une population initiale. La meilleure solution réalisable est toujours conservée.

Une nouvelle approche GA est développée ici pour le problème de dimensionnement des lots avec des décisions supplémentaires comme alternative à l'approche HPP et IAOP avec l'espoir qu'elle aboutira à des solutions beaucoup plus proches de l'optimum global. À notre connaissance, aucune approche GA n'a été proposée pour le problème de dimensionnement des lots.

La solution au problème de dimensionnement des lots familiaux capacitaires peut être représentée sous une forme tabulaire similaire à cellesatisfaites par les

type de présentation de transport trouvé dans le problème de planification de la production d'un seul article capacité avec demande dynamique (Bowman, 1956). Dans ce dernier contexte, la matrice de transport est de taille $T \cdot T$, et par exemple, une quantité de production positive dans la cellule (2,4) implique que la capacité à la période 2 est utilisée pour satisfaire une partie ou la totalité de la demande à la période 4. Ainsi, chaque colonne représente une période de demande et chaque ligne représente une période d'approvisionnement. Dans notre problème, cependant, il existe plusieurs familles et deux sources de capacité. Ainsi, chaque cellule de la matrice est constituée de $2F$ (F = nombre de familles) éléments (quantités de production), F d'entre eux représentant la production en temps régulier et le F restant représentant la production en heures supplémentaires. Chaque colonne a F demandes différentes pour F familles. Nous notons que si nous n'avions pas de configurations consommatrices de capacité et de familles multiples, alors une procédure simple qui satisfait la demande en utilisant la source de capacité la moins chère à partir de la première période conduirait à une solution optimale. Bien que notre modèle soit beaucoup plus compliqué, la représentation tabulaire ci-dessus est une manière valide de représenter une solution réalisable.

La manière dont la solution tabulaire est générée est la suivante : Une matrice $F \cdot T$, CAP, est construite. Chaque cellule (j,t) dans CAP consiste en un vecteur de taille T qui représente la priorité d'utilisation de la capacité entre les périodes avant et incluant la période t pour satisfaire la demande de la famille j pendant la période t . On note ce vecteur cap_{jt} . Par exemple, si cap_{24} est égal à $[3,1,4,2]$, alors la demande de la deuxième famille en quatrième période est satisfaite en premier en période 3 (le stock est conservé pendant une période) autant que la capacité disponible le permet . Ensuite, si toute la demande n'est pas satisfaite, la capacité de la période 1 est utilisée, etc. Ainsi, CAP est un cube FTT . Nous définissons également une deuxième matrice DEM de taille $F \cdot T$, où chaque colonne t , notée d_{emt} est un vecteur de taille F . d_{emt} représente la priorité entre différentes familles pour satisfaire leurs demandes à la période t .

Si, par exemple, $d_{em5} = [3,2,1,4]$, alors d'abord la demande (de la cinquième période) de la troisième famille est satisfaite en utilisant le vecteur de priorité de capacité correspondant cap_{35} .

Ainsi, une fois qu'une solution est encodée par les matrices DEM et CAP, il est très facile de générer heuristiquement la solution réalisable correspondante. Le pseudocode permettant de générer un horaire pour un codage donné est donné dans le tableau 12. A partir de la première période, la demande de chaque famille est satisfaite dans l'ordre du vecteur d_{emt} . Une fois sélectionnée, la demande de chaque famille est

Tableau 10. Solutions agrégées et coûts des solutions désagrégées résultant de l'approche IHPP

X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	Coût résultant de l'IHPP (disagg. sln.)	C _Q
1488	1103	462,18	1037,40	1037,40	924,00	78 773	2801
1488	1103	856,18	856,18	856,18	892,44	74 149	2311
1488	1103	788,70	788,70	959,50	924,00	73 407	2129
1488	1103	767,30	767,30	1001,80	924,00	73 760	2072

Tableau 11. Pseudocode pour la procédure IAOP

Procédure IAOP ;
commencer
C0 ¼ Ct ; CoûtMin =
1 ; Résoudre le modèle P2 avec
Ct ;
répéter Lire Y pour t = 1..T ; jt ,
Vérifier la faisabilité de la capacité de la solution, y compris les temps
de configuration ;
Calculer le coût, RCD ; Si
(Coût < MinCost) et (Capacité Réalisable) alors MinCost = Coût ; C0
¼ C0 différentiel ;
Résoudre le modèle
P2 avec C0 t ;
jusqu'à ce que (Modèle P2 Infaisable) ou (Solution Violates
Capacité) ou (Coût > MinCost) fin ;

capacités régulières et supplémentaires dans les périodes précédant et incluant la période en cours dans l'ordre du vecteur capjt. La capacité de chaque période est épuisée avant d'utiliser la capacité de la période suivante dans le vecteur capjt. Dans le tableau 13, nous démontrons la mécanique de générer un plan sur un chromosome partiel arbitraire pour l'exemple illustratif, c'est-à-dire que les vecteurs capj2 et dem2 sont donnés de sorte que les tailles de lot pour la deuxième période sont déterminées. Au cours de la première période, toutes les tailles de lots familiaux sont inévitablement égales à leurs demandes et, par conséquent, toute la capacité de temps régulier de la première période est épuisée et il reste 1099 heures de capacité d'heures supplémentaires. Le chromosome partiel a les gènes suivants : dem2 est [2,3,4,1] et cap22, cap32, cap42, cap12 sont [2,1], [1,2], [2,1], [1, 2], respectivement. Le tableau 13 montre comment la procédure Générer le plan est activée en fonction des vecteurs spécifiés. La seconde famille est la plus prioritaire et cap22 préfère utiliser la capacité de la seconde période. Ainsi, toute demande de la deuxième famille est satisfaite en utilisant la capacité de temps régulière de la deuxième période.

Ensuite, étant donné que la troisième famille préfère consommer la capacité de la première période, la capacité en heures supplémentaires de cette dernière période est consommée en premier sans encourir de temps de configuration, car cette famille se voit déjà attribuer une taille de lot au cours de cette période. La demande restante est satisfaite en consommant la capacité de temps régulière de la deuxième période, ce qui entraîne maintenant un temps de préparation. A noter que la première famille consomme

Tableau 12. La procédure Generate Plan utilisée pour développer un calendrier de production pour un chromosome donné

Procédure Générer un plan ; début
Infaisable = faux ; pour t = 1
à T commencer demset
=
{1..F} ; répéter
sélectionner la famille j* de priorité la plus élevée à partir de demt de sorte
que j* e {demset} ;
demset = demset) {j*} ; capset =
{1..T} ; répéter
sélectionner la période de priorité la plus élevée t* à partir de capj*t de sorte
que t* e {capset} ;
capset = capset) {t*} ; satisfaire
dj*t autant que possible à partir de Ct*, puis à partir de COT* ;
Réduire dj*t,
Ct* et COT* ; jusqu'à (capset = B) ou
(dj*t = 0) ; si dj*t > 0 alors Infaisable = vrai ;
jusqu'à (demset = B) ou (Infaisable) ; fin ; fin ;

capacité de la seconde période, puisque sa meilleure période préférée (période 1) voit sa capacité totalement épuisée par des familles de priorité plus élevée en dem2. Le coût minimum trouvé par l'AG est de 68 257 pour ce problème. Puisqu'un chromosome est représenté par des vecteurs prioritaires comme décrit ci-dessus, les opérateurs de croisement et de mutation sont faciles à appliquer car il n'y a pas de problèmes de faisabilité dans le cas où deux matrices prioritaires parentes sont croisées. Comme Storer et al. (1992) l'indiquent, il est assez pratique de représenter une solution par des priorités qui peuvent conduire à une solution heuristique. Nous appliquons un croisement à deux points appelé Linear Order Crossover décrit dans Falkenauer et Bouffoux (1991) pour les problèmes de séquençage car aucun élément ne doit être répété dans un vecteur de priorité et la séquence de priorité partielle doit être conservée afin d'intensifier la recherche vers la solution optimale.

À savoir, étant donné deux vecteurs demt parents [3,2,4,1,5] et [1,4,5,2,3] et deux locus croisés, 2 et 4, la progéniture serait [3,4,5, 2, 1] et [5,2,4,1,3]. Tous les vecteurs dans CAP et DEM sont croisés de la même manière.

Tableau 13. L'application de Generate Plan basée sur un encodage partiel

Taille du lot attribuée à la famille	Taille du lot en période 1	Taille du lot en période 2	Capacité consommée de la période	Type de capacité	Capacité restante
Famille 2	171,00	183,00	2	heures	2261
Famille 3	200 + 274,75	69,25	1	supplémentaires	0
			2	régulières	1789
Famille 4	636,00	297,00	2	régulières	1075
Famille 1	481,00	279,00	2	régulières régulières	90

La probabilité de croisement d'un chromosome dépend à la fois de l'écart de la valeur de coût des chromosomes par rapport au coût maximum dans la population actuelle et de la fourchette de coûts des populations. Une probabilité de croisement adaptative similaire est décrite par Srinivas et Patnaik (1994) qui rapportent qu'elle fonctionne bien dans l'optimisation des fonctions multimodales. La probabilité de croisement d'un chromosome ne cesse d'évoluer en fonction de la convergence de la fourchette de coût des populations et augmente lorsque la fourchette de coût des populations tend à rester bloquée à un optimum local.

Le croisement est exécuté pour aboutir au remplacement complet de l'ancienne population.

Mathématiquement, la probabilité de croisement p_{ck} peut être définie comme suit :

$$p_{ck} = \frac{\max(\text{coût} - \text{coût}_{\min}, 0)}{\max(\text{coût} - \text{coût}_{\min}, 1)}$$

où \max et \min sont respectivement les valeurs de coût maximum et minimum dans la population actuelle, et coût_k est la valeur de coût du k ème chromosome trouvé par la procédure Generate Plan.

La probabilité de mutation est la même que la probabilité de croisement et la mutation est appliquée à une seule paire de gènes dans dem_t et cap_t et un seul vecteur est modifié dans les matrices DEM et CAP respectivement. À savoir, une paire d'éléments est sélectionnée au hasard dans l'un des vecteurs et échangée. Après le croisement et la mutation, la progéniture/le chromosome muté est/est évalué par la procédure Générer un plan.

2.4. Une approche de recuit simulé Le codage

du GA est indirect (Portmann, 1996) dans le sens où une solution à un chromosome est obtenue par l'algorithme constructif Generate Plan. Bien que les encodages indirects aient déjà été appliqués avec succès dans différents contextes (O' zdamar, 1999 ; Uckun et al., 1993), ici, l'AG nécessite une structure de données lourde qui limite les paramètres importants de l'AG tels que la taille de la population et peut nécessiter des calculs considérables. temps. Par conséquent, nous proposons une approche de recuit simulé (SA) (Kirkpatrick et al., 1983) qui s'attaque directement au problème sans ignorer les variables de configuration binaires. Puisque nous voudrions commencer la recherche à partir d'une bonne solution au problème d'origine, la solution initiale est obtenue à partir de la procédure IAOP. Ensuite, la solution est perturbée un nombre de fois prédéfini. Le schéma de perturbation est simple.

La table 14 est constituée du pseudocode de la procédure Perturb qui recherche les meilleures solutions en passant d'une solution voisine à la suivante. Une solution voisine est une solution où exactement une famille voit sa taille de lot modifiée d'un montant "DeltaLot" au cours de deux périodes consécutives. DeltaLot est délimité par le haut par les limites de capacité, les niveaux de stock actuels et les tailles de lot dans la paire de périodes sélectionnée.

Tableau 14. Pseudocode pour Perturb

```
Perturbation de la procédure ;
{CQ, OQ : capacité restante de temps réglementaire/d'heures
supplémentaires ; ubLot : limite supérieure de la taille du lot transférable ;
DeltaLot : taille du lot transféré}

début
  Sélectionnez t1, j*, décision d'augmentation/
  diminution ; si t1 = T, alors t2 =
  T ) 1 sinon t2 = t1 +
  1 ; si (diminuer) et (lj;t1 > 0) alors

  commencer si t1 < T alors ubLot ¼ min fC0 t1 b O0 t1p=pj ; yjt1; ljt1g
  minfC0 sinon ubLot ¼ min fC0 t1 b O0 t1p=pj ; yjt1g;
  (sj soustrait de C0 si yjt2 ¼ yjt1 b O0 t2
  DeltaLot = aléatoire (0, ubLot);
  Calculer CostDif ;
  Si CostDif > 0 alors Calculer PA, tSA sinon tSA
  = 1 ;
  Si ((CostDif ≠ 0) et (Accept)) ou (CostDif < 0) alors
    commencer
    Transférer DeltaLot de la période t1 à la période t2 ;
    {yjt1 ¼ yjt1 DeltaLot ; yjt2 ¼ yjt2 b DeltaLot ;
    Ajuster C0 t1, C0 t2 ; OQ t1, OQ t2 ; lj t1 ; lj t2} ;
    Coût = Coût + CoûtDif ;
    Si Coût < MinCost alors MinCost = Coût ; fin; fin;
  sinon
    si
    (augmenter) et (C0 t1 b O0 t1 > 0) alors

    commencent si t1 = T alors ubLot ¼ min fC0 t1 b O0 t1p=pj ; yjt2; ljt2g
    sinon ubLot ¼ min fC0 t1 b O0 t1p=pj ; yjt2g;
    (sj soustrait de C0 t1 b O0 t1 si yjt1 ¼ 0)
    DeltaLot = aléatoire (0, ubLot)
    Calculer CostDif ;
    Si CostDif > 0 alors Calculer PA, tSA sinon tSA
    = 1 ;
    Si ((CostDif ≠ 0) et (Accept)) ou (CostDif < 0) alors
      commencer
      Transférer DeltaLot de la période t2 à la période t1 ;
      {yjt1 ¼ yjt1 b DeltaLot ; yjt2 ¼ yjt2 DeltaLot ;
      Ajuster C0 t1, C0 t2, t1, O0 t2, lj t1, lj t2} ;
      Coût = Coût + CoûtDif ;
      Si Coût < MinCost alors MinCost = Coût ; fin; fin;
    fin;
```

La procédure Perturb sélectionne aléatoirement une période (t1) et une famille (j*) pour changer la taille des lots en deux périodes consécutives, t1 et t1 + 1. Elle décide aléatoirement si la taille du lot de la famille va être augmentée ou a diminué au cours de la période t1. Les deux opérations sont effectuées en considérant les tailles de lot et les niveaux de stock de la famille considérée dans les périodes consécutives sélectionnées. Les limites de capacité sont toujours surveillées lors du déplacement d'une taille de lot familial. Ainsi, une itération de Perturb consiste simplement à transférer des éléments aléatoires et réalisables

portion d'une taille de lot d'une période à l'autre. Il est généralement possible d'économiser sur les coûts d'inventaire et/ou d'heures supplémentaires au cours d'une itération lorsque, par exemple, un goulot d'étranglement de capacité est libéré en déplaçant un lot vers une autre période et/ou en réduisant les coûts d'inventaire en transférant un lot à la suivante (peut être sans goulot d'étranglement) période. Dans le cas où un coût détérioré résulte d'une itération, une probabilité d'acceptation, PA, est calculée. PA dépend de l'importance de la détérioration et de la température actuelle, tSA, qui à son tour dépend du nombre de fois qu'un coût détérioré a été obtenu consécutivement. Si un nombre généré aléatoirement entre zéro et un s'avère être inférieur à PA, alors l'itération détériorante est effectuée. PA est calculé comme suit :

$PA = \frac{CostDif}{MinCost - tSA}$ où CostDif est la différence de coût impliquée par l'itération et MinCost est le coût minimum obtenu jusqu'à l'itération courante. Le schéma de réduction de température est continu (Dowland, 1993). À savoir, après chaque mouvement non amélioré, la température, tSA, est réduite comme suit

$$tSA = tSA - b$$

où b est une constante non négative inférieure à un. Initialement, tSA est égal à un. Chaque fois qu'un mouvement d'amélioration est effectué, tSA est réinitialisé à un. Pour obtenir de bons résultats avec l'approche SA, b peut être déterminé en exerçant une analyse paramétrique ou il peut être ajusté dynamiquement pendant la recherche (Bozyel et O'zdamar, 1996). Dans le cas où b est dynamique, si les m derniers mouvements consécutifs ont entraîné une amélioration des valeurs de coût, alors b est diminué d'une taille de pas au choix de l'utilisateur (une taille de pas pratique est de 0,01) inférieure à un. D'autre part, si les m derniers mouvements consécutifs ont entraîné des valeurs de coût non améliorées, alors b est augmenté de Stepsize. Le schéma de réduction dynamique de la température sert à mettre à jour PA en fonction de la partie spécifique de la région réalisable vers laquelle la recherche se dirige. Si la recherche est dans une direction favorable, alors b est diminué pour encourager les mouvements non améliorés afin que les optima locaux puissent être ignorés. Les mouvements sans amélioration sont découragés lorsque la recherche visite une région défavorable de la région réalisable, de sorte que la recherche peut s'échapper d'une région qui est peu susceptible de stocker la solution optimale globale.

Décrivons une itération de Perturb sur le petit exemple illustratif. Le tableau 15 montre comment Perturb effectue ses modifications. t1 = 2 et t2 = 3, j* = 2, la décision est d'augmenter la taille du lot de la deuxième famille en

période 2, et la solution historique a un coût de 70 311. Perturb transfère toute la taille du lot de 29 de la troisième période à la seconde. Les deux périodes ont des heures supplémentaires. Dans la période 3 (183 + 29 · 2 = 241) les heures supplémentaires sont enregistrées. Cependant 58 heures supplémentaires sont ajoutées à la période 2. Un inventaire de 29 unités est ajouté au coût de 19 · 29 = 551. Ainsi, les économies nettes sont (241 - 58) · 17 = 551 = 2560. L'approche SA trouve finalement une solution avec un coût de 60 480 pour ce problème.

3. Résultats de calcul

Les procédures de résolution décrites ci-dessus sont testées sur 198 instances de test générées aléatoirement. Les 90 premiers problèmes sont constitués de quatre familles à ordonnancer sur un horizon de planification de six périodes. Cet ensemble de petits problèmes est généré pour comparer les résultats aux solutions optimales. Les problèmes sont générés selon un plan factoriel complet basé sur trois caractéristiques majeures du problème : le rapport du coût de possession unitaire moyen (sur toutes les familles) au coût unitaire moyen des heures supplémentaires (H/O), l'écart type de la demande (STD) et le étanchéité de la capacité régulière et dans le temps (CS). Le coût/unité moyen des heures supplémentaires est estimé comme $ov \cdot \text{temps de traitement moyen}$, où le temps de traitement moyen est la moyenne des temps de traitement de la famille. Le facteur H/O est maintenu à trois niveaux (0,7, 1,1, 1,5) et le coût de possession unitaire par période se voit attribuer aléatoirement une valeur comprise entre 1 et 20 ; les temps de traitement des familles et les temps de configuration sont uniformément répartis entre [1–5] et [50–200], respectivement ; la demande est générée à partir d'une distribution normale de moyenne 100 unités par période avec un écart type STD maintenu à deux niveaux (10, 50) ; De plus, la demande a une tendance saisonnière sur l'horizon de planification et les modèles de demande de toutes les familles ne doivent pas nécessairement correspondre ; la capacité est calculée de manière à ce que chaque problème ait une solution réalisable, c'est-à-dire qu'une capacité totale suffisante (heures normales plus heures supplémentaires) soit mise à disposition pour satisfaire la demande cumulée à chaque période de l'horizon de planification ; un pourcentage d'allocation de 5% de la capacité demandée est également ajouté pour laisser un écart pour les temps de configuration, puis la moyenne calculée par période des capacités régulières et supplémentaires (qui sont maintenues constantes dans le temps pour plus de simplicité) est multipliée par un scalaire k pour aboutir à les trois niveaux du facteur CS : (k = 1,1, 1,3, 1,5). C'est-à-dire que la capacité est la plus restreinte au niveau CS de 1.1. La capacité totale est divisée à parts égales entre les capacités de temps régulier et les capacités d'heures supplémentaires. Ainsi, il en résulte $3 \cdot 2 \cdot 3 = 18$ combinaisons de facteurs. Pour chaque combinaison, nous générons cinq instances pour aboutir à

Tableau 15. Une itération de Perturb sur une solution pour l'exemple

	yj;2 (lj;2)	yj;3 (lj;3)	C ₂	C ₃	DeltaLot	CoûtDif
Avant de perturber	183 (0)	29 (0)	3819	3248		
Après perturbation	212 (29)	0 (0)	3877	3007	29	2560

Étonnamment, l'algorithme simple IAOP fonctionne aussi bien que le GA qui a été exécuté avec une taille de population de 80 et une limite de génération maximale de 1500 pour les petits problèmes. Malheureusement, bien que la procédure Generate Plan soit assez rapide, GA prend beaucoup de temps de calcul car le nombre d'opérations de croisement est élevé. En raison de la structure de données lourde et des exigences de mémoire excessives, GA n'a pas pu être testé dans des instances de test plus importantes.

	0	90	-	105	1.5°
2	86	-	-	150.4	
1	34	-	-	150.4	
1	39	-	-	150.4	
2	15	-	-	150.0	420
2	7	-	-	2-	
0	3	9	1	1	
0	3	dix	1	1	

2023	2022	2021	2020	2019	2018	2017	2016	2015	2014	2013	2012	2011	2010	2009	2008	2007	2006	2005	2004	2003	2002	2001	2000	1999	1998	1997	1996	1995	1994	1993	1992	1991	1990	1989	1988	1987	1986	1985	1984	1983	1982	1981	1980	1979	1978	1977	1976	1975	1974	1973	1972	1971	1970	1969	1968	1967	1966	1965	1964	1963	1962	1961	1960	1959	1958	1957	1956	1955	1954	1953	1952	1951	1950	1949	1948	1947	1946	1945	1944	1943	1942	1941	1940	1939	1938	1937	1936	1935	1934	1933	1932	1931	1930	1929	1928	1927	1926	1925	1924	1923	1922	1921	1920	1919	1918	1917	1916	1915	1914	1913	1912	1911	1910	1909	1908	1907	1906	1905	1904	1903	1902	1901	1900	1899	1898	1897	1896	1895	1894	1893	1892	1891	1890	1889	1888	1887	1886	1885	1884	1883	1882	1881	1880	1879	1878	1877	1876	1875	1874	1873	1872	1871	1870	1869	1868	1867	1866	1865	1864	1863	1862	1861	1860	1859	1858	1857	1856	1855	1854	1853	1852	1851	1850	1849	1848	1847	1846	1845	1844	1843	1842	1841	1840	1839	1838	1837	1836	1835	1834	1833	1832	1831	1830	1829	1828	1827	1826	1825	1824	1823	1822	1821	1820	1819	1818	1817	1816	1815	1814	1813	1812	1811	1810	1809	1808	1807	1806	1805	1804	1803	1802	1801	1800	1799	1798	1797	1796	1795	1794	1793	1792	1791	1790	1789	1788	1787	1786	1785	1784	1783	1782	1781	1780	1779	1778	1777	1776	1775	1774	1773	1772	1771	1770	1769	1768	1767	1766	1765	1764	1763	1762	1761	1760	1759	1758	1757	1756	1755	1754	1753	1752	1751	1750	1749	1748	1747	1746	1745	1744	1743	1742	1741	1740	1739	1738	1737	1736	1735	1734	1733	1732	1731	1730	1729	1728	1727	1726	1725	1724	1723	1722	1721	1720	1719	1718	1717	1716	1715	1714	1713	1712	1711	1710	1709	1708	1707	1706	1705	1704	1703	1702	1701	1700	1699	1698	1697	1696	1695	1694	1693	1692	1691	1690	1689	1688	1687	1686	1685	1684	1683	1682	1681	1680	1679	1678	1677	1676	1675	1674	1673	1672	1671	1670	1669	1668	1667	1666	1665	1664	1663	1662	1661	1660	1659	1658	1657	1656	1655	1654	1653	1652	1651	1650	1649	1648	1647	1646	1645	1644	1643	1642	1641	1640	1639	1638	1637	1636	1635	1634	1633	1632	1631	1630	1629	1628	1627	1626	1625	1624	1623	1622	1621	1620	1619	1618	1617	1616	1615	1614	1613	1612	1611	1610	1609	1608	1607	1606	1605	1604	1603	1602	1601	1600	1599	1598	1597	1596	1595	1594	1593	1592	1591	1590	1589	1588	1587	1586	1585	1584	1583	1582	1581	1580	1579	1578	1577	1576	1575	1574	1573	1572	1571	1570	1569	1568	1567	1566	1565	1564	1563	1562	1561	1560	1559	1558	1557	1556	1555	1554	1553	1552	1551	1550	1549	1548	1547	1546	1545	1544	1543	1542	1541	1540	1539	1538	1537	1536	1535	1534	1533	1532	1531	1530	1529	1528	1527	1526	1525	1524	1523	1522	1521	1520	1519	1518	1517	1516	1515	1514	1513	1512	1511	1510	1509	1508	1507	1506	1505	1504	1503	1502	1501	1500	1499	1498	1497	1496	1495	1494	1493	1492	1491	1490	1489	1488	1487	1486	1485	1484	1483	1482	1481	1480	1479	1478	1477	1476	1475	1474	1473	1472	1471	1470	1469	1468	1467	1466	1465	1464	1463	1462	1461	1460	1459	1458	1457	1456	1455	1454	1453	1452	1451	1450	1449	1448	1447	1446	1445	1444	1443	1442	1441	1440	1439	1438	1437	1436	1435	1434	1433	1432	1431	1430	1429	1428	1427	1426	1425	1424	1423	1422	1421	1420	1419	1418	1417	1416	1415	1414	1413	1412	1411	1410	1409	1408	1407	1406	1405	1404	1403	1402	1401	1400	1399	1398	1397	1396	1395	1394	1393	1392	1391	1390	1389	1388	1387	1386	1385	1384	1383	1382	1381	1380	1379	1378	1377	1376	1375	1374	1373	1372	1371	1370	1369	1368	1367	1366	1365	1364	1363	1362	1361	1360	1359	1358	1357	1356	1355	1354	1353	1352	1351	1350	1349	1348	1347	1346	1345	1344	1343	1342	1341	1340	1339	1338	1337	1336	1335	1334	1333	1332	1331	1330	1329	1328	1327	1326	1325	1324	1323	1322	1321	1320	1319	1318	1317	1316	1315	1314	1313	1312	1311	1310	1309	1308	1307	1306	1305	1304	1303	1302	1301	1300	1299	1298	1297	1296	1295	1294	1293	1292	1291	1290	1289	1288	1287	1286	1285	1284	1283	1282	1281	1280	1279	1278	1277	1276	1275	1274	1273	1272	1271	1270	1269	1268	1267	1266	1265	1264	1263	1262	1261	1260	1259	1258	1257	1256	1255	1254	1253	1252	1251	1250	1249	1248	1247	1246	1245	1244	1243	1242	1241	1240	1239	1238	1237	1236	1235	1234	1233	1232	1231	1230	1229	1228	1227	1226	1225	1224	1223	1222	1221	1220	1219	1218	1217	1216	1215	1214	1213	1212	1211	1210	1209	1208	1207	1206	1205	1204	1203	1202	1201	1200	1199	1198	1197	1196	1195	1194	1193	1192	1191	1190	1189	1188	1187	1186	1185	1184	1183	1182	1181	1180	1179	1178	1177	1176	1175	1174	1173	1172	1171	1170	1169	1168	1167	1166	1165	1164	1163	1162	1161	1160	1159	1158	1157	1156	1155	1154	1153	1152	1151	1150	1149	1148	1147	1146	1145	1144	1143	1142	1141	1140	1139	1138	1137	1136	1135	1134	1133	1132	1131	1130	1129	1128	1127	1126	1125	1124	1123	1122	1121	1120	1119	1118	1117	1116	1115	1114	1113	1112	1111	1110	1109	1108	1107	1106	1105	1104	1103	1102	1101	1100	1099	1098	1097	1096	1095	1094	1093	1092	1091	1090	1089	1088	1087	1086	1085	1084	1083	1082	1081	1080	1079	1078	1077	1076	1075	1074	1073	1072	1071	1070	1069	1068	1067	1066	1065	1064	1063	1062	1061	1060	1059	1058	1057	1056	1055	1054	1053	1052	1051	1050	1049	1048	1047	1046	1045	1044	1043	1042	1041	1040	1039	1038	1037	1036	1035	1034	1033	1032	1031	1030	1029	1028	1027	1026	1025	1024	1023	1022	1021	1020	1019	1018	1017	1016	1015	1014	1013	1012	1011	1010	1009	1008	1007	1006	1005	1004	1003	1002	1001	1000	999	998	997	996	995	994	993	992	991	990	989	988	987	986	985	984	983	982	981	980	979	978	977	976	975	974	973	972	971	970	969	968	967	966	965	964	963	962	961	960	959	958	957	956	955	954	953	952	951	950	949	948	947	946	945	944	943	942	941	940	939	938	937	936	935	934	933	932	931	930	929	928	927	926	925	924	923	922	921	920	919	918	917	916	915	914	913	912	911	910	909	908	907	906	905	904	903	902	901	900	899	898	897	896	895	894	893	892	891	890	889	888	887	886	885	884	883	882	881	880	879	878	877	876	875	874	873	872	871	870	869	868	867	866	865	864	863	862	861	860	859	858	857	856	855	854	853	852	851	850	849	848	847	846	845	844	843	842	841	840	839	838	837	836	835	834	833	832	831	830	829	828	827	826	825	824	823	822	821	820	819	818	817	816	815	814	813	812	811	810	809	808	807	806	805	804	803	802	801	800	799	798	797	796	795	794	793	792	791	790	789	788	787	786	785	784	783	782	781	780	779	778	777	776	775	774	773	772	771	770	769	768	767	766	765	764	763	762	761	760	759	758	757	756	755	754	753	752	751	750	749	748	747	746	745	744	743	742	741	740	739	738	737	736	735	734	733	732	731	730	729	728	727	726	725	724	723	722	721	720	719	718	717	716	715	714	713	712	711	710	709	708	707	706	705	704	703	702	701	700	699	698	697	696	695	694	693	692	691	690	689	688	687	686	685	684	683	682	681	680	679	678	677	676	675	674	673	672	671	670	669	668	667	666	665	664	663	662	661	660	659	658	657	656	655	654	653	652	651	650	649	648	647	646	645	644	643	642	641	640	639	638	637	636	635	634	633	632	631	630	629	628	627	626	625	624	623	622	621	620	619	618	617	616	615	614	613	612	611	610	609	608	607	606	605	60
------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	----

Des résultats nettement meilleurs sont obtenus avec SA qui commence par la solution IAOP. Dans le cas où la solution IAOP est irréalisable, SA génère au hasard sa propre solution de départ en attribuant au hasard des tailles de lot aux familles à chaque période dans les limites de capacité. Si un goulot d'étranglement de capacité se produit, les tailles de lot sont ajoutées à celles allouées dans les périodes précédentes et l'inventaire est ainsi accumulé. Afin de montrer l'effet de commencer avec une bonne solution, nous fournissons également des résultats obtenus avec SA en utilisant des solutions de départ générées aléatoirement sous l'en-tête SA-I. Dans SA et SA-I, le paramètre b a été sélectionné à 0,01 après qu'une analyse paramétrique préliminaire a été effectuée pour les problèmes apparemment difficiles. Nous testons également les effets de l'utilisation d'un paramètre dynamique b qui change au cours de la recherche en utilisant un pas de 0,01. Les résultats correspondants sont fournis sous SA-II. Les trois variantes de SA sont autorisées à effectuer 10 000 itérations et elles sont réexécutées 15 fois avec une nouvelle graine aléatoire dans les petits problèmes, et dans les problèmes plus importants, la limite d'itération et le nombre de réexécutions sont fixés à 100. 000 et 10, respectivement. Les meilleures solutions sont signalées. Notez que la performance de SA-II est considérablement meilleure que celle de SA-I dans les petits problèmes, mais dans les grands problèmes, l'effet de la dynamique b est perdu. Dans les problèmes plus importants, les résultats deviennent plutôt insensibles aux autres valeurs de b lorsque SA est utilisé avec une valeur b fixe. Cet effet pourrait être dû au nombre accru d'itérations autorisées et SA converge vers la solution finale indépendamment de b . Le temps de calcul pour SA est considérablement inférieur à celui de GA parce qu'une itération dans SA ne consiste qu'en quelques opérations mathématiques de base. L'approche SA est la plus performante parmi toutes les approches testées en ce qui concerne la qualité de la solution et l'efficacité de calcul et ses besoins en mémoire sont minimes. La performance relativement supérieure de SA est due à son traitement simple du CLSP dans le sens où les variables du modèle original P sont traitées lors des mouvements qui sont exécutés dans les limites des contraintes données. De plus, une solution voisine est très proche de la solution actuelle, car la taille des lots

d'une famille sont perturbés dans des périodes consécutives plutôt qu'éloignées. Par conséquent, SA est moins susceptible de sauter la solution optimale si la solution actuelle en est déjà proche. D'autre part, les heuristiques basées sur HPP aboutissent à des solutions inférieures, car elles conduisent inévitablement à une sous-optimalité due à l'agrégation des familles. La sous-optimalité des solutions augmente à mesure que le nombre de familles augmente. Par conséquent, la méthode IAOP qui traite le problème au niveau de la famille tout en assouplissant les variables de configuration binaires est nettement supérieure à l'heuristique basée sur HPP. Lorsque les deux méta-heuristiques GA et SA sont comparées, nous observons que dans le GA, à la fois le nombre d'opérations mathématiques et les besoins en mémoire sont excessivement élevés. Par conséquent, les performances de GA sont inférieures à SA et de toute façon le GA est difficile à utiliser avec des problèmes de plus grande taille.

Quant à l'effet du plan factoriel sur les performances heuristiques, les résultats de l'ANOVA montrent que seuls deux facteurs principaux (CS et H/O) sont statistiquement significatifs. Cependant, l'effet de CS est beaucoup plus significatif que H/O et toutes les heuristiques sauf SA(IAOP) et IAOP se détériorent en termes de performances lorsque CS est à son niveau élevé. Ce résultat est dû au fait naturel que l'ensemble des solutions réalisables devient plus grand lorsque la capacité est abondante. Lorsque la capacité est restreinte, il est plus difficile de trouver une solution optimale entière et ce fait est cohérent avec les résultats de l'heuristique basée sur la programmation mathématique fournie par Trigeiro et al. (1989) pour le CLSP avec des temps d'installation. Par exemple, pour des problèmes plus importants de 15 familles, GAMS pouvait résoudre un problème avec CS = 1,5 en 5 minutes sur l'ordinateur Pentium 200 MHz et résultait en un écart d'intégralité de 0,1 %, mais il a convergé vers un écart d'intégralité de 63 % pour un problème similaire avec CS = 1,1 dans les 41 minutes sur le même ordinateur. La raison pour laquelle SA est efficace pour résoudre les problèmes de capacité restreinte est qu'elle part d'une solution faisable et reste dans la région faisable qui est relativement étroite.

Le prochain facteur statistiquement significatif est H/O, mais son effet est beaucoup moins détectable que CS. Lorsque H/O augmente,

Tableau 19. Résultats pour différentes valeurs des facteurs CS et H/O (54 problèmes avec 15 familles/10 périodes)

		HPP \bar{p} FP HPP \bar{p} MFP IHPP IAOP SA(IAOP)					SA-I		GAMS SA-II	
SC = 1,1	moy	0,24	0,14	0,13	0,05	0,00	0,02	0,01	0,02	
	std	0,22	0,08	0,07	0,02	0,01	0,02	0,01	0,02	
SC = 1,3	moy	0,46	0,34	0,29	0,05	0,01	0,03	0,02	0,01	
	std	0,21	0,18	0,18	0,03	0,02	0,04	0,01	0,01	
SC = 1,5	moy	0,88	0,60	0,49	0,04	0,03	0,11	0,03	0,04	
	std	0,43	0,23	0,24	0,03	0,02	0,09	0,02	0,16	
H/O = 0,7	moy	0,48	0,35	0,27	0,06	0,02	0,06	0,02	0,01	
	std	0,30	0,19	0,16	0,05	0,02	0,04	0,02	0,02	
H/O = 1,1	moy	0,54	0,42	0,35	0,05	0,02	0,05	0,02	0,01	
	std	0,37	0,30	0,29	0,04	0,02	0,05	0,02	0,02	
H/O = 1,5	moy	0,47	0,40	0,38	0,09	0,02	0,05	0,02	0,02	
	std	0,40	0,36	0,34	0,11	0,03	0,04	0,03	0,03	

la sélection entre différentes familles devient plus importante pour constituer des inventaires. La fluctuation de la demande, STD, ne joue pas un rôle important dans la performance heuristique, car toutes les heuristiques agissent de toute façon en fonction des prévisions de la demande future. Ce dernier résultat est à nouveau cohérent avec les conclusions tirées par Trigeiro et al. (1989). Dans le tableau 19, les performances des heuristiques sont fournies en termes d'écart moyen en pourcentage (écart type) par rapport à la meilleure solution étant donné différentes valeurs de CS et H/O dans les problèmes de plus grande taille avec 15 familles et 10 périodes.

4. Conclusion

Le problème de dimensionnement des lots avec capacité avec des décisions supplémentaires et des temps de configuration est pertinent dans de nombreuses situations pratiques où les tailles de lots familiaux doivent être déterminées dans un service goulot d'étranglement sur un horizon de planification. Dans de nombreux cas, les configurations n'impliquent aucun coût supplémentaire à l'exception des heures de main-d'œuvre supplémentaires qui entraînent des coûts si la capacité est limitée et que des heures supplémentaires ont lieu.

Le problème considéré étant NP-difficile, il est nécessaire de développer des approches heuristiques pour le résoudre. Dans les résultats de calcul, nous démontrons que l'approche classique de planification hiérarchique de la production fournit des solutions considérablement sous-optimales au problème. Par conséquent, nous développons une approche itérative qui omet les variables binaires des contraintes de capacité dans le problème, un Algorithme Génétique (GA) dont le codage est basé sur la formule bien connue de type transport linéaire de la planification de la production d'un seul article capacité. problème, et une approche de recuit simulé (SA) basée sur le déplacement partiel ou total de la taille des lots entre des périodes consécutives. Parmi toutes les heuristiques, l'approche itérative et la SA sont les procédures fournissant les meilleures solutions. Les bonnes performances de l'AG sont limitées à des problèmes de petite taille en raison de la structure de données lourde de l'encodage.

Les références

Bitran, GR et Yanasse, HH (1982) Complexité computationnelle du problème de taille de lot capacité. *Sciences de gestion*, 28, 1174–1186.

Bowman, EH (1956) Ordonnancement de la production par la méthode de transport de la programmation linéaire. *Recherche opérationnelle*, 3, 100–103.

Bozyel, MA et O`zdamar, L. (1996) Une approche heuristique du problème capacité de dimensionnement et d'ordonnancement des lots (avec installations parallèles). Document de travail, Département d'ingénierie des systèmes, Université Yeditepe, Istanbul.

Catrysse, D., Maes, J. et van Wassenhove, LN (1990) Set partitioning and column generation heuristics for capacitated lotsizing. *Journal européen de recherche opérationnelle*, 46, 38–47.

Dixon, PS, Elder, MD, Rand, GK et Silver, EA (1983) Un algorithme heuristique pour déterminer la taille des lots d'un article soumis à des capacités de production régulières et supplémentaires. *Journal de gestion des opérations*, 3, 121–130.

Dixon, PS et Silver, EA (1981) Une procédure de solution heuristique pour le problème de dimensionnement de lot à capacité limitée à plusieurs éléments et à un seul niveau. *Journal de gestion des opérations*, 2, 23–39.

Dog`ramaci, A., Panayiotopoulos, JC et Adam, NR (1981) Le problème de dimensionnement dynamique des lots pour plusieurs articles sous capacité limitée. *IIE Transactions*, 13, 294–303.

Dowland, KA (1993) Quelques expériences avec des techniques de recuit simulé pour des problèmes d'emballage. *Journal européen de recherche opérationnelle*, 68, 389–399.

Falkenauer, E. et Bouffoix, S. (1991) Un algorithme génétique pour l'atelier, dans *Actes de la Conférence internationale IEEE 1991 sur la robotique et l'automatisation*, Sacramento, CA.

Garey, M. et Johnson, D. (1979) *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman and Co., San Francisco, CA.

Gu`nther, HO (1987) Planification de la taille des lots et des besoins en capacité dans un système de production en une seule étape. *Journal européen de recherche opérationnelle*, 31, 223–231.

Hax, AC et Candea, D. (1984) *Gestion de la production et des stocks* ment, Prentice-Hall Englewood Cliffs, NJ.

Kim, D. et Mabert, VA (1995) Heuristique d'ordonnancement de cycle intégrative versus séparée pour les problèmes de dimensionnement et de séquençage de lots discrets capacitifs. *Journal international de recherche sur la production*, 33, 2007–2021.

Kirca, O`. (1990) Un algorithme efficace pour le problème de taille de lot dynamique à un seul élément capacitif. *Journal européen de recherche opérationnelle*, 45, 15–24.

Kirca, O`. et Ko`kten, M. (1994) Une nouvelle approche heuristique pour le problème de dimensionnement dynamique des lots multi-items. *Journal européen de recherche opérationnelle*, 75, 332–341.

Kirkpatrick, A., Gelatt, Jr., CD et Vecchi, MP (1983) Optimisation par recuit simulé. *Sciences*, 220, 671–680.

Lambrecht, MR et Vanderveken, H. (1979) Procédure heuristique pour le problème de chargement multi-items à opération unique. *IIE Transactions*, 11, 319–326.

Maes, J. et van Wassenhove, LN (1985) Heuristique de lotsizing dynamique capacité multi-items à un seul niveau : une comparaison computationnelle, partie II : horizon glissant. Document de travail 85-08, Kath olieke Universiteit Leuven.

Maes, J. et van Wassenhove, LN (1986) Une heuristique simple pour le problème de dimensionnement de lot capacité multi-items à un seul niveau. *Lettres de recherche opérationnelle*, 4, 265–273.

Michalewicz, Z. (1994) *Algorithmes génétiques + Structures de données = Programmes d'évolution*, Springer Verlag, Berlin.

O`zdamar, L. (1999) Une approche d'algorithme génétique à un problème d'ordonnancement de projet de catégorie générale. *Transactions IEEE sur les systèmes, l'homme et la cybernétique*, 29, 44–69.

O`zdamar, L., Atli, AO`. et Bozyel, MA (1996) Techniques de désagrégation familiale heuristique dans les systèmes de planification de production hiérarchiques. *Journal international de recherche sur la production*, 34, 2613–2628.

O`zdamar, L., Bozyel, MA et Birbil, I. (1998) Un système d'aide à la décision hiérarchique pour la planification de la production (avec étude de cas). *European Journal of Operations Research*, 104, 403–422.

O`zdamar, L. et Yazgac, , T. (1997) Paramètres de date d'échéance basés sur la capacité dans les systèmes de production sur commande. *Journal international d'économie de la production*, 49, 29–44.

Portmann, MC (1996) Algorithmes génétiques et ordonnancement : un état de l'art et quelques propositions, dans *Actes de l'atelier sur la planification et le contrôle de la production*, FUCAM, Mons, Belgique, pp. 1–24.

Srinivas, M. et Patnaik, LM (1994) Probabilités adaptatives de croisement et de mutation dans les algorithmes génétiques. *Transactions IEEE sur les systèmes, l'homme et la cybernétique*, 24, 656–667.

Storer, RH, Wu, SD et Vaccari, R. (1992) Nouveaux espaces de recherche pour les problèmes de séquençage avec application à l'ordonnancement en atelier. *Sciences de gestion* 38, 1495–1509.

Thizy, JM et van Wassenhove, LN (1985) Relaxation lagrangienne du problème de dimensionnement de lots capacités multi-items : une implémentation heuristique. IIE Transactions, 17, 308–313.

Trigeiro, WW (1987) Une heuristique à double coût pour le problème de dimensionnement de lot capacité. IIE Transactions, 19, 67–72.

Trigeiro, WW, Thomas, LJ et McLain, JO (1989) Dimensionnement de lot capacité avec temps de configuration. Sciences de gestion, 35, 353–366.

Uckun, S., Bagchi, S., Kawamura, K. et Miyabe, Y. (1993) Gestion de la recherche génétique dans la planification d'emplois. Expert IEEE, 8, 15–24.

planification et gestion de projet, modèles de dimensionnement des lots, planification hiérarchique de la production et méthodologies de solution basées sur des méta-heuristiques. Des domaines d'intérêt plus récents sont le développement de méthodes d'optimisation globale et d'approximation de fonctions utilisant la logique floue. Le Dr O'zdamar a publié des articles connexes dans des revues telles que International Journal in Production Research, European Journal of Operations Research, IEEE Transactions on Systems, Man and Cybernetics, IEEE Transactions on Fuzzy Systems et Journal of Global Optimization.

Biographies

Le Dr Linet O'zdamar travaille actuellement au département de génie industriel de l'Université Dogus, à Istanbul, en Turquie. Dr. O'zdamar a obtenu son B.Sc., M.Sc. et doctorat. diplômes de l'Université Bogazici, à Istanbul. Ses intérêts de recherche passés et présents portent sur les domaines de

Mehmet Ali Bozyel a obtenu un B.Sc. et M.Sc. Diplômes du Département de Génie Industriel de l'Université de Marmara, Istanbul. Il dirige actuellement une entreprise de taille moyenne fabriquant des machines de climatisation industrielle et est responsable des tâches de planification de la production. M. Bozyel a publié des travaux connexes dans International Journal in Production Research, IIE Transactions et European Journal of Operations Research.