

# PLANIFICATION ET AFFECTATION DES ACTIVITÉS DANS UNE PHARMACIE HOSPITALIÈRE PAR LES COLONIES DE FOURMIS

C. CRESPIY, V. AUGUSTO, X. XIE

H. BONTEMPS

Centre Ingénierie et Santé  
École Nationale Supérieure des Mines de Saint-Étienne  
158 cours Fauriel 42023 Saint-Étienne cedex 2 France  
ccrespy@messel.emse.fr, augusto@emse.fr, xie@emse.fr

Centre Hospitalier de Villefranche-sur-Saône  
Plateau d'Ouilly-Gleizé BP 80436  
69655 Villefranche-sur Saône cedex France  
hbontemps@ch-villefranche.fr

**RÉSUMÉ :** *Le problème de planification et d'affectation de ressources humaines dans une pharmacie hospitalière consiste à générer un emploi du temps faisable pour les préparateurs. Un grand nombre de contraintes de natures différentes entrent en ligne de compte : répétition de tâches, date d'exécution au plus tôt et/ou au plus tard, délai minimum entre deux occurrences, compétences des ressources, etc. Pour résoudre ce problème NP-difficile, nous proposons une nouvelle méta-heuristique pour la génération automatique d'emploi du temps pour le personnel d'une pharmacie hospitalière. La méthode proposée s'appuie sur une variation de l'algorithme d'optimisation par colonies de fourmis pour la planification des tâches et sur une heuristique spécifique pour l'affectation des tâches aux ressources en fonction de leurs compétences. Plusieurs tests numériques sont proposés pour démontrer l'efficacité de la méthode. Enfin, la méta-heuristique proposée est appliquée au problème de taille réelle. Un compte-rendu de l'application sur le terrain est également proposé.*

**MOTS-CLÉS :** *pharmacie, emploi du temps, planification, affectation, heuristique, colonie de fourmis.*

## 1 INTRODUCTION

L'environnement du secteur hospitalier a été sensiblement modifié ces dernières années, notamment au niveau financier avec un nouveau plan de redressement des comptes de l'assurance maladie imposant un effort particulier à l'hôpital. Dans ce contexte, la pharmacie se doit comme tous les autres services d'être plus attentive au travail de chacun et à son optimisation. Parallèlement à ce changement du mode de financement se développe la démarche de certification des établissements pour garantir la qualité des soins fournis aux patients pris en charge. De ce fait le management des hommes devient un véritable enjeu, notamment dans le développement d'une politique d'amélioration continue de la qualité.

Dans ce cadre, il nous a paru essentiel de mettre à disposition des pharmaciens responsables de pharmacie à usage intérieur (PUI) des outils permettant d'optimiser leur service pour répondre de manière plus efficace aux nouvelles normes et règles gérant le milieu hospitalier. De ce fait le projet consiste à créer un outil permettant d'optimiser l'emploi du temps des agents du service en mettant en place une répartition efficace et équilibrée des tâches pour optimiser le fonctionnement de la pharmacie.

Étant données les particularités du système étudié, nous proposons une nouvelle méthode basée sur un algorithme de colonies de fourmis pour la génération automatique d'un planning d'activité. Ces outils sont souvent utilisés pour la résolution du problème classique en recherche opérationnelle de génération d'emplois du temps d'université ; cependant le problème traité dans ce papier est plus complexe du fait de l'existence de nombreuses contraintes inhérentes aux systèmes hospitaliers, et plus particulièrement à la pharmacie. L'outil développé a été pensé pour le domaine hospitalier et donnera lieu à un outil informatique pour le personnel de la pharmacie.

Cet article est organisé comme suit : un bref état de l'art sur la planification de ressources et la gestion d'emplois du temps avec un focus sur les applications en milieu hospitalier est proposé section 2. Le système considéré ainsi que la méthodologie de travail sont présentés section 3. La définition formelle du problème traité et de ses contraintes est décrite section 4. Une méthode d'optimisation basée sur les colonies de fourmis est décrite et évaluée section 5. Enfin, les résultats proposés pour le problème grandeur nature sont présentés section 6. Nous conclurons sur la pertinence de cette étude et sur son implantation dans le système section 7.

## 2 REVUE DE LITTÉRATURE

Le problème traité dans cet article peut se ramener à un problème de génération d'emploi du temps en université (University Course Timetabling Problem, UCTP) (Socha, Sampels & Manfrin 2003). Il s'agit d'un problème typique de la recherche opérationnelle consistant à planifier un nombre fini de cours afin d'obtenir un emploi du temps faisable tout en maximisant un certain nombre de préférences. Ce problème est NP-difficile, tout comme les nombreux sous-problèmes qui en découlent : examination timetabling (Gaspero & Schaerf 2000), school timetabling (Eikelder & Willemen 2000), employee timetabling, etc. Le problème de génération d'emploi du temps peut se ramener à un problème de coloration de graphe, ce qui explique le grand nombre d'articles dans la littérature proposant des méthodes basées sur l'optimisation par colonies de fourmis (Dorigo & Blum 2005). En particulier, l'algorithme *MAX-MIN* (Stützle & Hoos 2000, Socha, Knowles & Sampels 2002) s'est révélé très efficace pour la résolution de l'UCTP. Finalement, (Socha et al. 2003) ont démontré l'efficacité des algorithmes de fourmis sur les problèmes hautement contraints tel l'UCTP et ont analysé l'impact lié au choix d'un type particulier d'algorithme.

Si l'on s'intéresse au problème de génération d'emploi du temps dans le domaine hospitalier, le « nurse rostering problem » apparaît comme étant le plus proche de notre étude. Plusieurs états de l'art ont été proposés sur le sujet (Cheang, Li, Lim & Rodrigues 2003, Burke, Causmaecker, Berghe & Landeghem 2004). De nombreuses méthodes sont utilisées dans la littérature pour résoudre ce problème, essentiellement des méta-heuristiques : recherche tabou hybride (Burke, Causmaecker & Berghe 1999), algorithme génétique (Aickelin & Dowsland 2000), programmation par contraintes (Cheng, Lee & Wu 1997), etc. D'une manière générale, le « nurse rostering problem » est caractérisé par un grand nombre de contraintes fortes et de préférences à intégrer dans la génération de l'emploi du temps.

Nous proposons dans ce papier une nouvelle méthode pour la résolution du problème de génération d'emploi du temps également basée sur une méta-heuristique. Les méthodes présentées dans la suite peuvent être réutilisées pour la résolution du « nurse rostering problem ».

## 3 PRÉSENTATION DU SYSTÈME

### 3.1 Le CH de Villefranche-sur-Saône

Le centre hospitalier de Villefranche-sur-Saône est situé à une quarantaine de kilomètres au nord de Lyon. Il s'agit de l'un des principaux établissements

de santé de la région Rhône-Alpes, avec un grand nombre de lits pour les soins intensifs et les soins de longue durée, ainsi que des installations spécialisées pour les services de chirurgie et de médecine. On compte aussi dans le CH un service des urgences et une équipe de réanimation. Plus de mille employés travaillent sur le site, dont plus d'une centaine de médecins.

La pharmacie du centre hospitalier compte 21 employés dont 4 pharmaciens et 7 préparateurs. Leurs principales missions sont :

- **La gestion, l'approvisionnement et la dispensation des médicaments** (dont les stupéfiants, les MDS et les médicaments sous ATU), assurée par les préparateurs. Les médicaments sont stockés dans des armoires présentes dans les différents services. Celles-ci sont de deux types : (i) les armoires informatisées, dont le stock est géré par informatique (ces armoires restent dans le service et les préparateurs doivent se déplacer pour les réapprovisionner) ; (ii) les armoires non informatisées, dont le stock doit être déterminé par comptage (le réapprovisionnement est réalisé dans le service ou à la pharmacie).
- **Le guichet** : un préparateur est chargé de répondre aux commandes imprévues des infirmières ainsi qu'aux patients non hospitalisés qui se présentent pour des rétrocessions. Il doit également délivrer les poches de chimiothérapies. Ce guichet est ouvert toute la journée.
- **La réalisation de préparations magistrales ou hospitalières** : l'hôpital possède sa propre salle de fabrication de poches de chimiothérapies. Elle est assurée par deux des préparateurs sous la responsabilité d'un pharmacien.
- **La gestion des gaz médicaux**, uniquement en ce qui concerne la commande et le stockage.

### 3.2 Position du problème

À l'heure actuelle, la planification des activités de la pharmacie du CH de Villefranche-sur-Saône n'est pas réalisée de manière optimale : l'emploi du temps produit est peu robuste, et doit être entièrement remanié en cas d'absence d'un préparateur ou lors de périodes d'activité intenses. Nous proposons ainsi dans cet article une méthode pour la génération automatique d'emploi du temps pour la pharmacie.

Le problème considéré est similaire à un problème de génération d'emploi du temps d'université, comme cela a été illustré dans la section 2. Ce type de problème est difficile à résoudre (NP-complet). En outre, un grand nombre de contraintes inhérentes à

la particularité du système considéré entrent en ligne de compte et complexifient un peu plus ce problème classique de la recherche opérationnelle. Enfin, les natures et les durées des tâches à planifier varient énormément, ce qui les rend bien plus difficile à caser dans un planning en évitant d'éventuels chevauchements sur un horizon de temps discrétisé.

### 3.3 Méthodologie de travail

La méthodologie adoptée pour mener ce projet et résoudre le problème de planification se décompose en quatre étapes :

1. Inventaire des activités et des contraintes liées à la pharmacie.
2. Génération du planning par optimisation exacte ou par des heuristiques.
3. Validation du planning généré par le chef de service et le personnel de la pharmacie.
4. Test du planning en conditions réelles avec les préparateurs en particulier.

Nous détaillerons essentiellement la deuxième phase de la méthodologie dans cet article. La première phase permet d'inventorier les activités prises en charge par le personnel de la pharmacie. Ces activités sont assujetties à un grand nombre de contraintes plus ou moins fortes, contribuant à complexifier la planification des activités et l'affectation des activités aux ressources.

## 4 DEFINITION DU PROBLÈME

Nous considérons un ensemble de tâche  $N = \{1 \dots i \dots n\}$  de durée  $p_i$  à planifier sur un horizon  $H$  discrétisé en journées et en créneaux de temps. Les contraintes suivantes doivent être considérées pour planifier une tâche  $i$  et sont toutes facultatives.

- La fréquence  $T_i$  de la tâche sur l'horizon de planification.
- Les contraintes de jour : durant quel(s) jour(s) doit être réalisée la tâche  $i$ .
- La date d'exécution  $t_i$  (si elle est fixe).
- La date d'exécution au plus tôt  $r_i$  (*ready date*).
- La date d'exécution au plus tard  $d_i$  (*dead-line*).
- L'écart minimum entre deux occurrences d'une même tâche.

Soit  $M = \{1 \dots j \dots m\}$  l'ensemble des ressources disponibles. Chacune de ces ressources possède un certain nombre de compétences, illustrées par le paramètre binaire  $b_{ij}$  qui vaut 1 si la ressource  $j$  est capable d'accomplir la tâche  $i$ , 0 sinon. En outre, la disponibilité de ces ressources peut varier dans le temps. Enfin nous devons considérer un certain nombre de contraintes molles telle le respect des pauses pour le personnel hospitalier et un équilibrage des charges correct entre les ressources disponibles.

Les variables de décisions suivantes sont définies :  $x_{ikl}$  vaut 1 si la tâche  $i$  est planifiée à partir du créneau  $l$  du jour  $k$ , 0 sinon ;  $s_i^1$  (resp.  $s_i^2$ ) est le jour durant lequel la tâche  $i$  est planifiée (resp. le créneau à partir duquel la tâche  $i$  débute) ;  $y_{ij}$  vaut 1 si la tâche  $i$  est réalisée par la ressource  $j$ , 0 sinon.

L'objectif consiste à minimiser le nombre de tâches non planifiées ou la durée totale des tâches non planifiées. L'analyse multi-objectifs n'est pas prise en compte dans cet article.

## 5 UNE MÉTA-HEURISTIQUE POUR LA GÉNÉRATION D'EMPLOI DU TEMPS

La méta-heuristique proposée construit une solution en trois temps :

1. **Pre-processing** : les tâches sont organisées afin de faciliter le processus de planification et d'affectation de ressources.
2. **Planification** : les tâches classées sont ensuite affectées à une certaine journée de l'horizon et à un certain créneau durant la journée choisie. Cette phase est réalisée grâce à un algorithme de colonies de fourmis.
3. **Affectation** : les tâches planifiées sont ensuite distribuées entre les ressources disponibles. Une heuristique gloutonne est proposée pour résoudre ce problème.

L'objectif consiste à minimiser le nombre de tâches non-planifiées et/ou la durée totale des tâches non-planifiées. Nous décomposons dans cette section la méthodologie adoptée.

### 5.1 Pre-processing

La phase de pre-processing permet de réorganiser les données avant de débiter l'optimisation. En effet, la construction de l'emploi du temps étant réalisée en plaçant les tâches successivement dans le planning, l'ordre d'apparition est important. La phase de pre-processing permet ainsi de classer les tâches à planifier en fonction d'un ou plusieurs critères. Dans ce problème, nous proposons un classement selon le

critère  $p_i T_i$  où  $p_i$  est la durée de la tâche et  $T_i$  est la fréquence de la tâche, i.e. le nombre de répétitions de cette tâche sur la semaine. En cas d'égalité, les contraintes horaires sont examinées : les tâches les plus contraintes (concernées par une date d'exécution fixée, au plus tôt et/ou au plus tard) devront être placées en priorité. Les tâches les plus courtes sont examinées en dernier.

## 5.2 Optimisation par colonies de fourmis

Les algorithmes de colonies de fourmis sont une technique probabiliste permettant la résolution de problèmes combinatoires pouvant être réduits à des problèmes de recherche de chemins dans un graphe (Coloni, Dorigo & Maniezzo 1991, Dorigo 1992). Ces algorithmes sont basés sur le comportement de fourmis cherchant un chemin entre leur colonie et une source de nourriture. L'idée originale a ensuite été adaptée à de nombreuses classes de problèmes d'optimisation. Ainsi, cette méthode peut également être appliquée à un grand nombre de problèmes combinatoires en ajustant le processus de construction de la solution.

L'approche « Ant System » est la première méthode proposée dans la littérature (Dorigo, Maniezzo & Coloni 1996) et sera ajustée pour la résolution de ce problème. À chaque itération, la quantité de phéromone est mise à jour par toutes les fourmis qui ont construit la solution dans la même itération. Soit  $A$  le nombre total de fourmis dans la colonie.

La construction d'une solution réalisée par chaque fourmi est réalisée en deux temps :

1. Choix d'une journée sur l'horizon de planification durant laquelle la tâche aura lieu.
2. Choix du créneau horaire sur l'horizon d'une journée à partir duquel la tâche commencera.

Nous différencions ainsi la quantité de phéromone associée au choix d'un jour pour la planification d'une tâche, et la quantité de phéromone associée au choix d'un créneau horaire pour cette même tâche. La Figure 1 illustre cette méthode. Les probabilités  $p_{ik}$  et  $q_{il}$  permettant de déterminer le jour et le créneau sont décrites plus bas.

La quantité de phéromone  $\tau_{ik}$  est associée à la planification de la tâche  $i$  durant le jour  $k$ , tandis que la quantité  $\sigma_{il}$  est associée à la planification de la tâche  $i$  du créneau  $l$  au créneau  $l + p_i$  inclus. Ces quantités sont mises à jour de la manière suivante :

$$\tau_{ik} \leftarrow (1 - \rho)\tau_{ik} + \sum_{a=1}^A \Delta\tau_{ik}^a \quad (1)$$

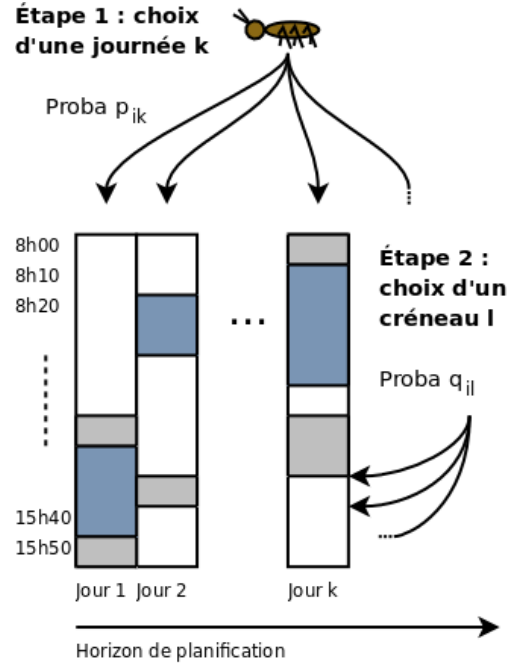


FIG. 1: Construction d'une solution en deux étapes

$$\sigma_{il} \leftarrow (1 - \rho)\sigma_{il} + \sum_{a=1}^A \Delta\sigma_{il}^a \quad (2)$$

où  $\rho$  est le taux d'évaporation et  $\Delta\sigma_{ik}^a$  (resp.  $\Delta\tau_{il}^a$ ) est la quantité de phéromone déposée pour le mouvement lié à la planification de la tâche  $i$  au jour  $k$  (resp. entre les créneaux  $l$  et  $l + p_i$ ) :

$$\Delta\tau_{ik}^a = \begin{cases} \frac{Q}{L_a} & \text{si la fourmi } a \text{ a planifié la tâche } i \\ & \text{au jour } k, \\ 0 & \text{sinon.} \end{cases} \quad (3)$$

$$\Delta\sigma_{il}^a = \begin{cases} \frac{R}{L_a} & \text{si la fourmi } a \text{ a planifié la tâche } i \\ & \text{entre les créneaux } l \text{ et } l + p_i, \\ 0 & \text{sinon.} \end{cases} \quad (4)$$

où  $Q$  et  $R$  sont deux constantes et  $L_a$  est la valeur globale de la fonction objectif pour la solution construite par la fourmi  $a$ . Les paramètres  $\rho$ ,  $Q$  et  $R$  doivent être choisis judicieusement en fonction de l'importance des traces de phéromone et de l'ordre de grandeur des valeurs de la fonction objectif.

Le mécanisme stochastique utilisé pour construire une solution est établi de la manière suivante. La probabilité de planifier la tâche  $i$  au jour  $k$  par la fourmi  $a$  est donnée par :

$$p_{ik}^a = \begin{cases} \frac{\tau_{ik}^\alpha \eta_{ik}^\beta}{\sum_{c_{il} \in N} \tau_{il}^\alpha \eta_{il}^\beta} & \text{si } c_{ij} \in N \\ 0 & \text{sinon.} \end{cases} \quad (5)$$

où  $N$  est l'ensemble des décisions possibles. Les paramètres  $\alpha$  et  $\beta$  contrôlent l'importance relative des traces de phéromones contre l'information heuristique  $\eta_{ik}$ , donnée par :

$$\eta_{ik} = \frac{1}{NT_k} \quad (6)$$

où  $NT_k$  est le nombre de tâches déjà affectées durant le jour  $k$ . Ainsi, l'information heuristique tient compte de la charge des journées candidates pour accueillir la tâche considérée.

Une fois le jour choisi, la probabilité  $q_{il}^a$  de planifier la tâche  $i$  entre les créneaux  $l$  et  $l + p_i$  est calculée en utilisant une équation similaire à (5). En revanche, l'information heuristique est évaluée de la manière suivante :

$$\eta_{il} = \frac{1}{\sum_{t=l}^{l+p_i} c_t + LT_t} \quad (7)$$

où  $LT_k$  est le nombre de tâches qui ont lieu durant le créneau  $t$  et  $c_t$  est un facteur de coût utilisé pour pénaliser certains créneaux (pause de midi par exemple).

Lorsqu'un jour et un créneau ont été choisis pour la tâche  $i$ , les variables  $x_{ikl}$ ,  $s_i^1$  et  $s_i^2$  sont mises à jour. La construction de la solution continue jusqu'à ce que toutes les tâches aient été planifiées.

L'algorithme 1 illustre la démarche utilisée pour la résolution du problème de génération d'emploi du temps. Un ensemble de  $A$  fourmis artificielles construisent des solutions à partir d'un ensemble fini de solutions partielles. À chaque étape, la solution partielle  $S^p$  est étendue en ajoutant une composante faisable depuis l'ensemble  $N(S^p)$ , défini comme l'ensemble de composantes pouvant être ajoutées à la solution partielle courante sans violer les contraintes du problème initial décrites section 4.

Le choix d'une composante depuis l'ensemble  $N(S^p)$  est lié à un mécanisme stochastique (lignes 4–7) biaisé par les traces de phéromones associées à chaque élément de  $N(S^p)$ . La règle utilisée pour le choix stochastique d'une solution est donnée dans les équations (5), (6) et (7).

Finalement, les traces de phéromone sont mises à jour. Les valeurs des traces associées à de bonnes solutions sont augmentées, tandis que les valeurs des traces associées à de mauvaises solutions diminuent (ligne 10). Toutes les traces diminuent sous l'effet de l'évaporation (équations (1) et (2)).

---

**Algorithme 1** Algorithme d'optimisation pour la planification de tâches

---

```

1  Initialiser les paramètres, initialiser les traces
   de phéromones
2  tant que la condition de fin n'est pas vérifiée
   faire
3      pour chaque tâche  $i$  à planifier faire
4          Calculer  $p_{ik}$  et  $q_{il} \forall k, l$ 
5          Planifier la tâche  $i$  au jour  $k$  au hasard
6          Planifier la tâche  $i$  au créneau  $l$  au hasard
7          Mettre à jour  $x_{ikl}$ ,  $s_i^1$  et  $s_i^2$ 
8      fin pour
9      Affecter les tâches aux ressources
10     Mettre à jour les traces de phéromones  $\tau_{ik}$ 
        et  $\sigma_{il}$ 
11 fin tant que

```

---

Les paramètres requis pour exécuter l'algorithme sont le nombre de fourmis (nombre d'itérations),  $\rho$ ,  $Q$  et  $R$  pour ajuster les valeurs des traces de phéromone, et  $\alpha$  et  $\beta$  pour ajuster l'importance relative de la phéromone contre l'information heuristique. Ces paramètres ont un rôle crucial dans la performance de l'algorithme et dépendent fortement des jeux de données en entrée.

### 5.3 Affectation des tâches aux ressources disponibles et compétentes

Une fois la planification des tâches terminée, celles-ci doivent être affectées aux ressources disponibles sur l'horizon considéré en respectant les contraintes de compétence : chaque ressource est compétente pour réaliser un certain nombre de tâches ; une tâche ne peut pas être réalisée par une ressource qui n'a pas les compétences nécessaires.

Afin de résoudre ce problème, nous adoptons une nouvelle représentation des solutions produites par l'algorithme d'optimisation de la phase précédente. Nous considérons les variables  $s_i^1$  et  $s_i^2$ , qui représentent respectivement le jour de la tâche  $i$  et le créneau de début de la tâche  $i$  dans le jour considéré. La Figure 2 illustre la représentation des solutions adoptées qui sera utilisée par l'heuristique d'affectation des tâches aux ressources disponibles.

$i$	1	2	3	...	$n$	← n° de la tâche
$s_i^1$	1	1	3	...	2	← jour de la semaine
$s_i^2$	12	25	18	...	30	← créneau du jour

FIG. 2: Représentation du planning généré

La méthode proposée dans cet article est basée sur une heuristique gloutonne. Nous privilégions volontairement une heuristique simple du fait du grand nombre de contraintes à prendre en compte lors de

l'affectation, afin de réduire les temps de calcul. Cette procédure affecte séquentiellement les tâches selon l'ordre donné par la phase de pre-processing aux ressources disponibles tout en respectant les contraintes de disponibilité. Deux types de contraintes sont également prises en compte : (i) les tâches possédant plus d'une occurrence devant être réalisée par une même ressource ; (ii) les tâches possédant plus d'une occurrence qui ne sont pas nécessairement réalisées par la même ressource.

Nous proposons également plusieurs alternatives pour l'affectation des tâche en modifiant l'ordre de priorité sur les ressources. Dans la version de base, les ressources sont toujours examinées dans le même ordre : cette stratégie produit ainsi des solutions où les premières ressources seront plus chargées que les dernières. Dans une seconde version, les tâches sont affectées à des ressources tirées au hasard, afin de mieux répartir la charge. La variable  $y_{ij}$  est mise à jour à l'issu de cette phase.

#### 5.4 Tests numériques

Afin d'évaluer l'efficacité de la méthode proposée dans cette section, nous avons généré plusieurs instances de données de test fictives, présentées table 1. Cinq classes de problèmes de taille et de complexité croissante ont été générées. Chaque classe de problème contient dix jeux d'essai générés aléatoirement. La première colonne de la table 1 indique l'identifiant de la classe de problèmes. Les colonnes 2 et 3 indiquent respectivement le nombre de tâches à planifier et le nombre de ressources disponibles. Les colonnes 4 et 5 indiquent l'intervalle de valeurs où sont tirés la durée des tâche et leur fréquence.

Classe	$n$	$m$	$p_i$	$T_i$
1	100	5	[10;120]	[1;5]
2	150	7	[10;120]	[1;5]
3	200	10	[10;180]	[1;5]
4	500	25	[10;180]	[1;5]
5	1000	30	[10;180]	[1;5]

TAB. 1: Classes de données

Paramètre	Valeur
$\rho$	0,05
$Q$	20
$R$	10
$\alpha$	0,8
$\beta$	1,2
Nb. de fournis	10.000

TAB. 2: Paramétrage de la simulation

Afin de prendre en considération l'importance de la robustesse des résultats ainsi que leur signification statistique, 10 jeux d'essai sont générés pour chaque

instance. Les durées  $p_i$  sont générées selon une loi de poisson ( $\lambda = 1,67$ ) et les fréquences sont générées selon une loi exponentielle ( $\beta = 21,1$ ). Les lois et leurs arguments ont été estimés d'après les données relevées sur le terrain et présentées dans la section suivante. Contraintes jours et contraintes horaires sont générées aléatoirement de manière uniforme. Les classes de problèmes sont présentées par ordre de difficulté croissante ; en particulier, la classe 5 ne constitue pas un problème réaliste mais est présentée afin de tester la robustesse numérique de notre méthode. Toutes les autres données de test sont choisies en fonction des données réelles présentées dans la section 6. La table 2 résume le jeu de paramètre utilisé pour résoudre les problèmes.

Les résultats présentés dans la table 3 sont comparés avec deux heuristiques basiques : l'heuristique RAND construit une solution en planifiant les tâches et en les affectant aux ressources de manière aléatoire, tandis que l'heuristique RAND+PREPROC agit de la même manière en utilisant la procédure de pre-processing décrite plus haut. Notre méthode se décline en deux versions, la première (ANTOPT1) réalise l'affectation des ressources de manière séquentielle, tandis que la deuxième version (ANTOPT2) réalise l'affectation de manière aléatoire. Le critère considéré est le nombre de tâches non planifiées à l'issu de chaque heuristique. Chaque classe de problème a été testée avec 50.000 fournis (itérations).

Nous constatons que les deux variantes ANTOPT1 et ANTOPT2 sont meilleures que les heuristiques proposées pour comparaison sur les cinq classes proposées. L'effet du pre-processing est particulièrement appréciable si l'on compare les résultats de RAND et de RAND+PREPROC. En revanche, l'affectation aléatoire des tâches aux ressources dans ANTOPT2 entraîne une diminution des performances pour le même nombre d'itérations au profit de l'équilibrage des charges entre préparateurs (comme cela est illustré dans la section suivante). L'augmentation du nombre d'itérations permet de pallier à ce problème.

## 6 APPLICATION

L'approche présentée dans la section précédente est appliquée au cas d'étude représenté par la pharmacie de l'hôpital de Villefranche-sur-Saône. L'objectif de l'étude menée en collaboration avec les pharmaciens et les préparateurs du service consiste à fournir un outil d'optimisation permettant la génération automatique de plannings d'activité pour le personnel constitué de préparateurs. Cet outil doit être suffisamment flexible pour offrir une solution faisable en cas d'absence de l'une ou l'autre des ressources et/ou de semaine raccourcie (en raison d'un jour férié par exemple).

Classe	RAND	RAND+PREPROC	ANTOPT1		ANTOPT2	
	Critère (%)	Critère (%)	Critère (%)	CPU (s)	Critère (%)	CPU (s)
1	0,20	0,30	0,10	125	1,90	135
2	3,50	1,20	0,90	191	5,90	198
3	5,40	4,40	2,50	259	4,70	287
4	8,90	3,90	3,10	653	5,80	677
5	9,40	5,30	2,00	1531	6,10	2099

TAB. 3: Performance de la méta-heuristique proposée

Id	Code	Durée (min)	Fréquence $T_i$	Contraintes jour					Contraintes horaires			Écarts		
				L	M	M	J	V	$t_i$	$r_i$	$d_i$	1	2	3
1	CHIMIOS	140	5	X	X	X	X	X	10:00	-	-	-	-	-
2	CHIMIOS (STÉ)	60	2	X	-	X	-	-	14:00	-	-	-	-	-
3	CHIMIOS (URG)	30	1	-	-	X	-	-	08:10	-	-	-	-	-
4	GAZ MED	20	1	-	-	-	-	-	-	-	-	-	-	-
5	STUPEFIANTS	20	5	X	X	X	X	X	-	13:00	-	-	-	-
...														
171	UDR (URG)	10	3	-	-	-	-	-	-	-	-	1	2	-
172	NEO-NAT (URG)	10	2	-	-	-	-	-	-	-	-	3	-	-

TAB. 4: Extrait du jeu de données issu du cas d'application

## 6.1 Données

Un travail d'observation et de collecte de données préliminaire a permis d'établir la liste des tâches réalisées par les préparateurs de la pharmacie durant une semaine. Bien que cette phase ne sera pas développée plus avant dans cet article, elle mérite une attention toute particulière en raison de l'absence de système d'information informatique performant au sein de la pharmacie. Le relevé des activités et de leurs durées est le fruit d'un travail intensif de plusieurs semaines. La table 4 donne un aperçu des 172 tâches recensées. Par exemple, la tâche 2 qui dure 60 minutes a lieu deux fois par semaine, en l'occurrence le lundi et le mercredi à 14h00 précises ; la tâche 171 qui dure 10 minutes a lieu trois fois par semaine : les occurrences 1 et 2 ont lieu un certain jour et le lendemain, tandis que les occurrences 2 et 3 doivent laisser passer une journée.

Concernant les ressources humaines, la pharmacie dispose d'une équipe de 7 préparateurs dont les compétences sont connues. Les emplois du temps sont générés sur un horizon d'une semaine, à raison de 5 jours de travail par semaine. L'horizon est discrétisé en intervalles de temps de 10 minutes chacun.

## 6.2 Résultats

La méthode proposée permet la génération de plannings en un temps raisonnable (moins de cinq minutes pour l'obtention d'un emploi du temps optimal, dans la mesure où toutes les tâches sont planifiées). La rapidité de la méthode permet aux utilisateurs de lancer un grand nombre d'itérations afin

de s'assurer d'obtenir une solution convenable selon leurs besoins et les ressources disponibles. La planification des tâches offre rapidement de très bon résultats car l'optimum est approché peu de temps après l'exécution du programme. Les tests ont été volontairement réalisés sur une machine de puissance moyenne, afin de reproduire les conditions de la pharmacie : processeur de type Pentium 4, 1 Go RAM sous MS Windows ou Linux.

L'affectation des tâches aux préparateurs constitue le point le moins satisfaisant étant donné le déséquilibre de charge observé table 5 : en effet, l'affectation séquentielle des tâches aux ressources permet l'obtention plus rapide de plannings faisables, mais déséquilibre fortement la charge de ces mêmes ressources. Dans ces conditions, la charge varie de 26,25 % à 92,92 %. Si l'affectation est réalisée de manière aléatoire, la charge est bien plus équilibrée et ne varie que de 60,83 % à 74,17 % ; mais dans ce cas, l'exécution requiert un nombre d'itérations plus grand pour atteindre une solution faisable. Ce problème peut être résolu en remplaçant la méthode d'affectation gloutonne par une méthode plus élaborée basée sur un recuit-simulé : cela permettrait le test de solutions voisines de la solution proposée et notamment d'éviter les chevauchements de tâches. La charge moyenne aux alentours de 75 % est correcte car elle tient compte des temps de pause et de repas.

## 7 CONCLUSIONS ET PERSPECTIVES

Nous avons proposé dans cet article une nouvelle méthode hybride combinant pre-processing et méta-heuristique pour la génération de plannings de tra-

Préparateur	Affect. séquentielle Charge (%)	Affect. aléatoire Charge (%)
1	87,08	62,50
2	92,92	70,83
3	90,00	71,25
4	80,00	74,17
5	70,42	72,50
6	50,41	70,00
7	26,25	60,83
Durée (s)	128	438

TAB. 5: Charge des préparateurs

vail dans un contexte hospitalier. Cet outil est centré sur l'usage d'algorithmes de colonies de fourmis, outil particulièrement adapté au problème fondamental de génération d'emploi du temps. Ce dernier a été adapté aux particularités rencontrées dans le système et enrichi de techniques de pre-processing et d'affectation de ressources en fonction de compétences. Les résultats produits sont satisfaisants et parfaitement adaptés aux conditions de travail des préparateurs en pharmacie.

L'outil d'optimisation conçu pour générer automatiquement des plannings pour les employés de la pharmacie répond à toutes les contraintes posées : l'emploi du temps respecte les contraintes sur les tâches, sur le personnel et ses heures de travail, sur les contraintes spécifiques imposées par le pharmacien pour la bonne marche du service, sur les demandes de la cadre du service pour la simplicité d'utilisation. De ce point de vue, les techniques utilisées semblent répondre de manière satisfaisante à la demande formulée par le personnel de la pharmacie.

Un certain nombre de perspectives s'ouvrent à l'issue de ce travail :

- Le test de variations de l'algorithme d'optimisation par colonies de fourmis (*MIN-MAX* par exemple) pourrait donner de meilleurs résultats et diminuer les chevauchements intempestifs de tâches lors de la planification des activités : en particulier, les pénalisations induites lors de l'affectation des tâches sur une journée pourrait tenir compte de la probabilité de chevauchement avec d'autres tâches sur de courtes périodes (10 minutes par exemple).
- L'heuristique gloutonne proposée pour l'affectation des tâches aux ressources doit être remplacée par une méthode de meilleure qualité. Une méta-heuristique de type recuit-simulé est actuellement en cours de développement.
- L'implémentation de contraintes supplémentaires permettrait d'améliorer la pertinence du modèle dans le contexte hospitalier.

L'investigation de méthodes utilisées pour le « nurse rostering problem » constitue une perspective intéressante.

Enfin, la poursuite de l'implantation et l'amélioration du logiciel développé pour la pharmacie constitue un travail de valorisation essentiel qui s'inscrit dans une collaboration à long terme avec la pharmacie du CH de Villefranche-sur-Saône.

## REMERCIEMENTS

Les auteurs remercient Mme Patricia Cinquin, cadre du service de la pharmacie, Mmes Chatillon, Coursier et Hellot, pharmaciennes, ainsi que l'ensemble des membres de l'équipe de la pharmacie pour leur précieuse collaboration.

## REFERENCES

- Aickelin, U. & Dowsland, K. A. (2000). Exploiting problem structure in a genetic algorithm approach to a nurse rostering problem, *Journal of Scheduling* **3**(3): 139–153.
- Burke, E. K., Causmaecker, P. D. & Berghe, G. V. (1999). A hybrid tabu search algorithm for the nurse rostering problem, *Lecture Notes in Computer Science* **1585**: 187–194.
- Burke, E. K., Causmaecker, P. D., Berghe, G. V. & Landeghem, H. V. (2004). The state of the art of nurse rostering, *Journal of Scheduling* **7**(6): 441–499.
- Cheang, B., Li, H., Lim, A. & Rodrigues, B. (2003). Nurse rostering problems - a bibliographic survey, *European Journal of Operational Research* **151**(3): 447–460.
- Cheng, B. M., Lee, J. H. & Wu, J. C. (1997). A nurse rostering system using constraint programming and redundant modeling, *IEEE Trans Inf Technol Biomed* **1**(1): 44–54.
- Coloni, A., Dorigo, M. & Maniezzo, V. (1991). Distributed optimization by ant colonies, *Actes de la première conférence européenne sur la vie artificielle, Paris, France*, Elsevier Publishing, pp. 134–142.
- Dorigo, M. (1992). *Optimization, Learning and Natural Algorithm*, PhD thesis, Politecnico di Milano, Italy.
- Dorigo, M. & Blum, C. (2005). Ant colony optimization theory: A survey, *Theoretical Computer Science* **344**: 243–278.
- Dorigo, M., Maniezzo, V. & Coloni, A. (1996). Ant system: Optimization by a colony of cooperating



agents, *IEEE Transactions on Systems, Man, and Cybernetics - Part B* **26**(1): 29–41.

Eikelder, H. M. M. & Willemsen, R. J. (2000). Some complexity aspects of secondary timetabling problems, *Proceedings of the 3rd International Conference on Practice and Theory of Automated Timetabling (PATAT 2000)*, pp. 18–29.

Gaspero, L. D. & Schaerf, A. (2000). Tabu search techniques for examination timetabling, *Proceedings of the 3rd International Conference on Practice and Theory of Automated Timetabling (PATAT 2000)*, pp. 104–117.

Socha, K., Knowles, J. & Sampels, M. (2002). A MAX-MIN ant system for the university timetabling problem, *Proceedings of ANTS 2002, Berlin, Germany*.

Socha, K., Sampels, M. & Manfrin, M. (2003). Ant algorithms for the university timetabling problem with regard to the state-of-the-art, *Proceedings of the EvoCOP Workshop, April 14-16, Essex, UK*.

Stützle, T. & Hoos, H. H. (2000). MAX-MIN ant system, *Future Generation Computer Systems* **16**: 889–914.