

# Optimisation des tournées et dimensionnement des équipes : Modèle exploratoire

Dans ce chapitre, nous nous intéressons au problème d'optimisation des tournées des véhicules de la logistique du CHRU. L'étude de ce problème est abordée de manière générale et théorique. Nous présentons dans une première section la notation et la modélisation du problème. Puis nous proposons un modèle de Programmation Linéaire en Nombre Entier (PLNE) (Kergosien et al. [116]). Ce PLNE a surtout permis de poser clairement le problème. Nous proposons ensuite deux méthodes de résolution, deux méta-heuristiques. La première est un algorithme de population (algorithme génétique, Kergosien et al. [115] et [113]) alors que la seconde méthode est un algorithme de voisinage (recherche tabou). Des résultats expérimentaux sur ces trois méthodes sont présentés à la fin de ce chapitre.

## 3.1 Définition et notation du problème

Le problème est défini comme il a été présenté dans la section 2.1 du chapitre 2. L'objectif est de trouver une planification des tournées de véhicules entre les différents hôpitaux et des manutentionnaires entre les différents bâtiments de Bretonneau, ainsi que de déterminer le nombre de chauffeurs et de manutentionnaires nécessaires. Dans cette première étude, on suppose qu'à l'hôpital Bretonneau seul le grand quai est utilisé. Les camions viennent y charger ou décharger leur cargaison et de là les manutentionnaires se chargent de dispatcher les chariots dans les services récipiendaires, soit à pied, soit en utilisant les fenwickis intérieurs ou extérieurs, suivant le bâtiment à desservir. Les manutentionnaires se chargent également de collecter les chariots de linge sale et les déchets pour les amener au quai. Il faut donc tenir compte de tous les flux et les traiter conjointement. D'autre part, il faut aussi synchroniser les deux niveaux de tournées de véhicules au quai de Bretonneau. De nombreuses contraintes sont à prendre en compte dans ce premier modèle comme des contraintes de terrain, la séparation de certains flux comme le linge propre et le linge sale, des gestions spécifiques, notamment des procédures de livraisons au sein des services, le

### 3.1. DÉFINITION ET NOTATION DU PROBLÈME

---

nombre de places aux quais de déchargement, en particulier à l'hôpital Clocheville, la multiplicité des moyens de transports, etc. Seules les contraintes de temps de préparation ou production des chariots et de distinction de types de quais (hauts ou bas) n'ont pas été prises en comptes.

Nous notons  $\mathcal{A}$  un ensemble d'emplacements regroupant un ensemble de dépôts  $\mathcal{D}$ , un ensemble d'hôpitaux  $\mathcal{H}$  et un ensemble de bâtiments  $\mathcal{B}$  à l'intérieur d'un même hôpital  $T$  ( $T \in \mathcal{H}$ ).  $T$  représente l'hôpital Bretonneau. Il sert de zone de transfert entre les véhicules inter-hôpitaux et les véhicules intra-hôpitaux. L'ensemble des notations que nous utilisons est le suivant :

- $\tau$  : la période de planification et  $t$  l'indice du jour dans cette période. Les demandes qui ne sont pas préaffectées à un jour de la semaine, seront indicées par  $\tau + 1$ . La période de planification considérée est d'une semaine étant donné que la fréquence des demandes du CHRU sont au plus d'une semaine.
- $\mathcal{P}$  : l'ensemble des types de chariots (le type est associé au produit transporté donc au flux). Nous confondrons par la suite type de chariot et produit. Pour chaque type de chariot ou produit  $p$  nous posons :
  - $v_p$  : volume d'un chariot de type  $p$  (utile pour les camions).
  - $p_p$  : poids moyen d'un chariot de type  $p$ .
  - $tl_{i,p}$  : temps de livraison (hors transport) du chariot de type  $p$  depuis le point de livraison  $i \in \mathcal{H} \cup \mathcal{B}$ .
  - $tch_p$  : temps de chargement ou déchargement du chariot de type  $p$ .
  - $g_p$  : temps d'autonomie du chariot de type  $p$  s'il est limité en temps de transport (concerne l'UCPA).
- Pour chaque point de livraison à desservir  $i$ , hormis  $T$  ( $i \in \mathcal{H} \cup \mathcal{B} \setminus T$ ) :
  - $q_{i,p,t}$  : quantité de chariots de type  $p$  demandés le jour  $t$ .
  - $[e_{i,p,t}, l_{i,p,t}]$  : fenêtre de temps pendant laquelle les chariots en produit  $p$  sont attendus pour le jour  $t$ .
  - $Quai_i$  : le nombre de places sur le quai de chargement et déchargement du point  $i$ .
- Pour l'emplacement particulier  $T$  (zone de transfert des chariots à Bretonneau) :
  - $q_{T,p,t} = \sum_{i \in \mathcal{B}} q_{i,p,t}$  : quantité de chariots de type  $p$  demandés pour le jour  $t$  par l'ensemble des bâtiments de  $T$ .
  - $[e_{T,p,t}, l_{T,p,t}] = [\min_{i \in \mathcal{B}}(e_{i,p,t}), \max_{i \in \mathcal{B}}(l_{i,p,t})]$  : fenêtre de temps associée à la demande en chariots  $p$  pour le jour  $t$ .
  - $Quai_T$  : le nombre de places sur le quai de chargement et déchargement de Bretonneau.
- $d_{i,j}$  : distance (en temps) entre deux emplacements  $i$  et  $j \in \mathcal{A}$ .
- $\mathcal{V}$  : l'ensemble des véhicules se déplaçant entre les hôpitaux avec :
  - $Cv_v$  : la capacité en volume du véhicule  $v$ .
  - $Cp_v$  : la capacité en poids du véhicule  $v$ .
- $\mathcal{F}int \cup \mathcal{F}ext$  : l'ensemble des fenwick se déplaçant à l'intérieur des couloirs souterrain de Bretonneau et l'ensemble des véhicules fenwick se déplaçant en extérieur entre les bâtiments de Bretonneau :
  - $FCn_v$  : la capacité en nombre de chariots tractables par le fenwick  $v$ .
  - $FCp_v$  : la capacité en poids de chariots tractables par le fenwick  $v$ .

- Pour chaque moyen de transport  $v$  de l'ensemble  $\mathcal{V} \cup \mathcal{Fint} \cup \mathcal{Fext}$  :
  - $Vit_v$  : le facteur de vitesse du moyen de transport  $v$ .
  - $D_v$  : le dépôt du moyen de transport  $v$  (pour les fenwicks, le dépôt correspond au quai de Bretonneau).
  - $\mathcal{N}_v$  : l'ensemble des emplacements livrables par le moyen de transport  $v$ .
  - $\mathcal{B}_v$  : l'ensemble des produits transportables par le moyen de transport  $v$ .
- $\mathcal{R}$  : l'ensemble des personnes (chauffeurs/manutentionnaire  $r \in \mathcal{R}$ ), avec  $T_{max}$  la durée de travail maximale par jour avec pause,  $c_{max}$  le nombre maximal de chariots que peut tirer un manutentionnaire à pied,  $p_{max}$  le poids maximal qu'il peut tirer et  $Vit_{pieds}$  le facteur de vitesse du transport à pieds.

## 3.2 PLNE

Pour une première étude de ce problème, nous allons exposer un modèle de programmation linéaire en nombres entiers qui va permettre d'identifier et de formuler clairement toutes les contraintes du problème. Avant de présenter le modèle de PLNE, nous allons introduire les notations utilisées et quelques libertés qui ont été prises. Pour alléger la formulation en PLNE, nous avons fait l'hypothèse que le quai de Clocheville est limité à une place, les autres ne sont pas limités. De plus, nous considérons que toutes les demandes sont préaffectées à un jour, enfin les bâtiments de Bretonneau sont livrables uniquement par manutentionnaire. Chaque manutentionnaire se voit affecté un unique moyen de transport, certains se déplaceront toujours à pied tandis que d'autres utiliseront toujours le même fenwick.

Une des difficultés du modèle est qu'il ne s'agit pas simplement d'un problème de tournées avec plusieurs voyageurs. On doit en fait déterminer plusieurs tournées successives pour plusieurs voyageurs (véhicules), sachant que le nombre de tournées effectuées par chaque voyageur n'est pas connu à l'avance. On fixe arbitrairement que chaque véhicule effectuera au plus  $m$  tournées. On a un nombre total de moyens de transports égal à  $w$  (les camions, les fenwicks, les manutentionnaires à pied), donc on doit déterminer au plus  $w \times m$  routes. La formulation s'appuie donc sur un ensemble de  $w \times m$  routes parcourues dans une journée. Cela regroupe l'ensemble  $\mathcal{M}_{inter}$  des routes inter-hôpitaux et l'ensemble  $\mathcal{M}_{intra}$  des routes entre les bâtiments  $\mathcal{B}$  et  $T$ , l'hôpital Bretonneau. L'ensemble de toutes les routes est noté  $\mathcal{M} = \mathcal{M}_{inter} \cup \mathcal{M}_{intra}$ . A chaque moyen est associé un ensemble de  $m$  routes, une route pouvant éventuellement être vide (les routes de 1 à  $m$  correspondent au premier véhicule, de  $m + 1$  à  $2m$  au deuxième véhicule, etc.). Chaque route est donc caractérisée par un départ et un retour au même dépôt  $D_v$ , et un moyen de transport  $v$  (camions, fenwicks, pied). Nous noterons  $\Phi$  la fonction qui permet de passer d'un indice d'une route  $k$  à l'indice du moyen de transport correspondant  $v = \Phi(k)$ . Le nombre final de routes n'étant pas connu à l'avance, nous fixons le nombre  $m$  en fonction du système actuel et des quantités demandées par rapport aux capacités des moyens de transports. Chaque type de moyen de transport étant limité, nous imposerons une contrainte de succession d'utilisation de véhicules ou fenwicks entre deux routes utilisant le même véhicule ou fenwick. Enfin, nous noterons  $\mathcal{O}$  un ensemble d'indices de tournées, une même personne

### 3.2. PLNE

---

assure un maximum de  $|\mathcal{O}|$  tournées par jour, une tournée correspond à une route.

#### 3.2.1 Les variables

Les variables du PLNE sont les suivantes :

- $x_{i,j}^{k,t} = 1$  si la route  $k$  passe par les emplacements  $i$  puis  $j$  le jour  $t$ ; 0 sinon ( $\forall k \in \mathcal{M}, \forall i, j \in D_{\Phi(k)} \cup \mathcal{N}_{\Phi(k)} (i \neq j), \forall t \in \tau$ )
- $dep_i^{k,t} =$  date de départ (ou de fin de livraison) de la route  $k$  le jour  $t$  à l'emplacement  $i$  si  $i \in \mathcal{N}_{\Phi(k)}$ ; date de départ du dépôt de la route  $k$  le jour  $t$  si  $i \in D_{\Phi(k)}$  ( $\forall k \in \mathcal{M}, \forall i \in D_{\Phi(k)} \cup \mathcal{N}_{\Phi(k)}, \forall t \in \tau$ )
- $fin^{k,t} =$  date de fin de la tournée  $k$  le jour  $t$  ( $\forall k \in \mathcal{M}, \forall t \in \tau$ )
- $IQc_{i,p}^{k,t} = 1$  si la route  $k$  livre au moins un chariot de produits  $p$  à l'emplacement  $i$  le jour  $t$ ; 0 sinon ( $\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P}$ )
- $Qc_{i,p}^{k,t} = \alpha$  si la route  $k$  livre  $\alpha$  chariots de produits  $p$  à l'emplacement  $i$  le jour  $t$ ; 0 sinon ( $\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P}$ )
- $IQtr_p^{k_1, k_2, t} = 1$  si la route  $k_1$  transporte au moins un chariot de produits  $p$  pour la route  $k_2$  le jour  $t$ ; 0 sinon ( $\forall k_1 \in \mathcal{M}_{inter}, k_2 \in \mathcal{M}_{intra}, \forall t \in \tau, \forall p \in \mathcal{P}$ )
- $Qtr_p^{k_1, k_2, t} = \alpha$  si la route  $k_1$  transporte  $\alpha$  chariots de produit  $p$  pour la route  $k_2$  le jour  $t$ ; 0 pour aucune livraison ( $\forall k_1 \in \mathcal{M}_{inter}, k_2 \in \mathcal{M}_{intra}, \forall t \in \tau, \forall p \in \mathcal{P}$ )
- $Liv_r^{k,o,t} = 1$  si la personne  $r$  est affectée à la route  $k$  lors de sa  $o^{ieme}$  tournée le jour  $t$ ; 0 sinon ( $\forall k \in \mathcal{M}, o \in \mathcal{O}, \forall t \in \tau, \forall r \in \mathcal{R}$ )
- $Ma_r = 1$  si la personne  $r$  est affectée à des routes de  $\mathcal{M}_{intra}$ ; 0 si la personne  $r$  est affectée à des routes de  $\mathcal{M}_{inter}$  ( $\forall r \in \mathcal{R}$ )
- $Ex_{k_1, k'_1}^t = 1$  si la route  $k_1$  passe avant  $k'_1$  à Clocheville; 0 si la route  $k'_1$  passe avant  $k_1$  à Clocheville ( $\forall k_1, k'_1 \in \mathcal{M}_{inter}, k_1 \neq k'_1$  et utilisant des véhicules différents,  $\forall t \in \tau$ ). Clocheville est l'hôpital ne possédant qu'une unique place de déchargement.

#### 3.2.2 Les contraintes

Les contraintes du PLNE sont les suivantes :

$$\forall k \in \mathcal{M}, \forall t \in \tau : \sum_{i \in \mathcal{N}_{\Phi(k)}} x_{D_{\Phi(k)}, i}^{k,t} \leq 1 \quad (3.1)$$

$$\forall k \in \mathcal{M}, \forall i \in D_{\Phi(k)} \cup \mathcal{N}_{\Phi(k)}, \forall t \in \tau : \sum_{j \in D_{\Phi(k)} \cup \mathcal{N}_{\Phi(k)} / i \neq j} x_{j,i}^{k,t} = \sum_{j \in D_{\Phi(k)} \cup \mathcal{N}_{\Phi(k)} / i \neq j} x_{i,j}^{k,t} \quad (3.2)$$

$$\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P} : q_{i,p,t} \sum_{j \in D_{\Phi(k)} \cup \mathcal{N}_{\Phi(k)} / i \neq j} x_{j,i}^{k,t} \geq Qc_{i,p}^{k,t} \quad (3.3)$$

$$\forall k_1 \in \mathcal{M}_{inter}, \forall t \in \tau : \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{N}_{\Phi(k_1)}} Qc_{i,p}^{k_1,t} v_p \leq Cv_{\Phi(k_1)} \quad (3.4)$$

$$\forall k_1 \in \mathcal{M}_{inter}, \forall t \in \tau : \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{N}_{\Phi(k_1)}} Qc_{i,p}^{k_1,t} p_p \leq Cp_{\Phi(k_1)} \quad (3.5)$$

$$\begin{aligned} \forall k_2 \in \mathcal{M}_{intra}, \forall t \in \tau : \text{ si } \Phi(k_2) \in \mathcal{Fint} \cup \mathcal{Fext} \text{ alors } \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{N}_{\Phi(k_2)}} Qc_{i,p}^{k_2,t} &\leq FCn_{\Phi(k_2)} \\ \text{ sinon } \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{N}_{\Phi(k_2)}} Qc_{i,p}^{k_2,t} &\leq c_{max} \end{aligned} \quad (3.6)$$

$$\begin{aligned} \forall k_2 \in \mathcal{M}_{intra}, \forall t \in \tau : \text{ si } \Phi(k_2) \in \mathcal{Fint} \cup \mathcal{Fext} \text{ alors } \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{N}_{\Phi(k_2)}} Qc_{i,p}^{k_2,t} p_p &\leq FCp_{\Phi(k_2)} \\ \text{ sinon } \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{N}_{\Phi(k_2)}} Qc_{i,p}^{k_2,t} p_p &\leq p_{max} \end{aligned} \quad (3.7)$$

$$\forall i \in \mathcal{H} \cup \mathcal{B}, \forall t \in \tau, \forall p \in \mathcal{P} : \sum_{k \in \mathcal{M}} Qc_{i,p}^{k,t} = q_{i,p,t} \quad (3.8)$$

Les contraintes (3.1) imposent à chaque route  $k$  de quitter au plus une fois son dépôt. Les contraintes (3.2) assurent la continuité des circuits : lorsqu'une route  $k$  arrive à un point  $i$ , elle repart de ce même point. Les contraintes (3.3) assurent la cohérence des variables de livraison : si une route  $k$  ne passe pas en un emplacement  $i$ , la quantité livrée en  $i$  en tout produit  $p$  est nulle. La quantité totale que livre une route à tous ses points de livraison ne doit pas excéder la capacité du véhicule correspondant en terme de nombre, poids et volume. Les contraintes (3.4), (3.5), (3.6) et (3.7) imposent cette restriction en fonction des types de routes (inter ou intra-hôpitaux). Enfin les contraintes (3.8) signifient que chaque client doit recevoir la quantité de chariots qu'il a demandé.

Les contraintes ci-dessous définissent l'ensemble des produits transportables par un véhicule, comme par exemple les véhicules qui ne transportent uniquement du linge sale.

$\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall p \in \mathcal{P}$  tel que la route  $k$  ne peut pas livrer de produit  $p, \forall t \in \tau :$

$$Qc_{i,p}^{k,t} = 0 \quad (3.9)$$

$$\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P} : Qc_{i,p}^{k,t} \geq IQc_{i,p}^{k,t} \quad (3.10)$$

$$\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P} : IQc_{i,p}^{k,t} \geq \frac{Qc_{i,p}^{k,t}}{q_{i,p,t}} \quad (3.11)$$

### 3.2. PLNE

---

Ces contraintes (3.10) et (3.11) définissent les variables d'état de livraison,  $IQc_{i,p}^{k,t}$ , pour une route  $k$  à un emplacement  $i$  en chariot  $p$  à un jour  $t$  en fonction de la variable de quantité livrée associée.

$$\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P} : \quad IQc_{i,p}^{k,t} e_{i,p,t} \leq dep_i^{k,t} \quad (3.12)$$

$$\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P} : \quad dep_i^{k,t} \leq l_{i,p,t} + HV(1 - IQc_{i,p}^{k,t}) \quad (3.13)$$

Les contraintes (3.12) et (3.13) assurent le respect des fenêtres de temps de livraison.

$$\begin{aligned} & \forall k \in \mathcal{M}, \forall i \in D_{\Phi(k)} \cup \mathcal{N}_{\Phi(k)}, \forall j \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau : \\ & dep_i^{k,t} + d_{i,j} Vit_{\Phi(k)} + \sum_{p \in \mathcal{P}} tl_{i,p} Qc_{i,p}^{k,t} \leq dep_j^{k,t} + HV(1 - x_{i,j}^{k,t}) \end{aligned} \quad (3.14)$$

Les contraintes (3.14) expriment les contraintes temporelles entre deux points de livraison  $i$  et  $j$  se succédant sur une même route  $k$ .

$$\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau : dep_i^{k,t} + d_{i,D_{\Phi(k)}} Vit_{\Phi(k)} \leq fin^{k,t} + HV(1 - x_{i,D_{\Phi(k)}}^{k,t}) \quad (3.15)$$

Les contraintes (3.15) permettent de définir les dates de retours aux dépôts.

$$\forall k_1 \in \mathcal{M}_{inter}, \forall t \in \tau, \forall p \in \mathcal{P} : Qc_{T,p}^{k_1,t} = \sum_{k_2 \in \mathcal{M}_{intra}} Qtr_p^{k_1,k_2,t} \quad (3.16)$$

$$\forall k_2 \in \mathcal{M}_{intra}, \forall t \in \tau, \forall p \in \mathcal{P} : \sum_{i \in \mathcal{B}} Qc_{i,p}^{k_2,t} = \sum_{k_1 \in \mathcal{M}_{inter}} Qtr_p^{k_1,k_2,t} \quad (3.17)$$

Les relations entre les quantités de chariots livrées au quai de Bretonneau et dans Bretonneau sont exprimées dans les contraintes (3.16) et (3.17). Pour un chariot de type  $p$  donné, la somme de tous les chariots livrés par les routes intra-Bretonneau, en provenance de la route inter-hopitaux  $k_1$  est égale au nombre de chariots livrés par  $k_1$  au quai (3.16). De plus tous les chariots de type  $p$  livrés par les routes intra-Bretonneau  $k_2$  proviennent des routes  $k_1$  de l'extérieur de Bretonneau (3.17).

$$\forall k_1 \in \mathcal{M}_{inter}, \forall k_2 \in \mathcal{M}_{intra}, \forall t \in \tau, \forall p \in \mathcal{P} : Qtr_p^{k_1,k_2,t} \geq IQtr_p^{k_1,k_2,t} \quad (3.18)$$

$$\forall k_1 \in \mathcal{M}_{inter}, \forall k_2 \in \mathcal{M}_{intra}, \forall t \in \tau, \forall p \in \mathcal{P} : IQtr_p^{k_1,k_2,t} \geq \frac{Qtr_p^{k_1,k_2,t}}{q_{T,p,t}} \quad (3.19)$$

De la même manière que (3.10) et (3.11), les contraintes (3.18) et (3.19) permettent de définir les variables d'état de transfert de chariots de type  $p$  entre une route  $k_2$  intra-Bretonneau et une route  $k_1$  extérieur à un jour  $t$  en fonction des variables de quantité transférées.

Les contraintes (3.20) et (3.21) définissent les relations de précédence entre routes intra et inter-hôpitaux transportant les mêmes produits. C'est-à-dire, le départ d'une route  $k_2$  intra-Bretonneau transportant des chariots de produits  $p$  provenant d'une route extérieure  $k_1$  ne peut s'effectuer que lorsque la route  $k_1$  a fini de décharger au quai tous ses chariots (3.20). Dans le cas contraire d'un ramassage de linge sale par exemple, la route  $k_1$  ne peut partir du quai de Bretonneau que lorsque les chariots de linges sale ont tous été collectés par l'équipe de manutention (3.21).

$\forall k_1 \in \mathcal{M}_{inter}, \forall k_2 \in \mathcal{M}_{intra} (T = D_{\Phi(k_2)}), \forall t \in \tau, \forall p \in \mathcal{P}$  p étant des chariots à livrer :

$$dep_T^{k_1,t} \leq dep_T^{k_2,t} + HV(1 - IQtr_p^{k_1,k_2,t}) \quad (3.20)$$

$\forall k_1 \in \mathcal{M}_{inter}, \forall k_2 \in \mathcal{M}_{intra}, \forall t \in \tau, \forall p \in \mathcal{P}$  p étant des chariots à récupérer :

$$dep_T^{k_1,t} \geq fin^{k_2,t} - HV(1 - IQtr_p^{k_1,k_2,t}) \quad (3.21)$$

Si une route  $k$  succède à une autre route  $k'$  avec le même véhicule alors la route  $k'$  ne peut démarrer que lorsque la première a fini son tour et que le véhicule est chargé. Nous définissons donc les contraintes suivantes :

$\forall k, k' \in \mathcal{M}$  tel que  $k$  précède la route  $k'$  avec le même véhicule,  $\forall t \in \tau$  :

$$fin^{k,t} + \sum_{p \in \mathcal{P}} \sum_{j \in \mathcal{N}_{\Phi(k')}} Qc_{j,p}^{k',t} tch_p \leq dep_{D_{Phi(k')}}^{k',t} \quad (3.22)$$

Certains chariots, en particulier les plateaux repas, ne peuvent pas être transportés au delà d'une certaine durée ( $g_p$ ). Les contraintes (3.23) et (3.24) permettent de tenir compte de l'autonomie de ces chariots.

$\forall k \in \mathcal{M}, \forall i \cup \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P}$  tel que  $p$  est limité dans la durée de transport :

$$dep_i^{k,t} - dep_{D_{\Phi(k)}}^{k,t} \leq g_p + HV(1 - IQc_{i,p}^{k,t}) \quad (3.23)$$

$\forall k_1 \in \mathcal{M}_{inter}, \forall k_2 \in \mathcal{M}_{intra}, \forall i \cup \mathcal{N}_{\Phi(k_2)}, \forall t \in \tau, \forall p \in \mathcal{P}$  :

$$dep_i^{k_2,t} - dep_{D_{\Phi(k_1)}}^{k_1,t} \leq g_p + HV(2 - IQc_{i,p}^{k_2,t} - IQtr_p^{k_1,k_2,t}) \quad (3.24)$$

Les contraintes (3.23) limitent la durée de transport de certains chariots dans une même route. Alors que les contraintes (3.24) limitent la durée de transport de certains chariots empruntant des routes inter-hôpitaux puis intra-Bretonneau.

Les contraintes (3.25) et (3.26) assurent qu'il n'y aura qu'une seule livraison à la fois à l'hôpital Clocheville (noté  $Hcl$ ). Ces contraintes sont valables pour des routes  $k_1$  et  $k'_1$  utilisant des véhicules différents.

$$\forall k_1, k'_1 \in \mathcal{M}_{inter}, k_1 \neq k'_1, \forall t \in \tau : Ex_{k_1,k'_1}^t = 1 - Ex_{k'_1,k_1}^t \quad (3.25)$$

$$\begin{aligned}
 & \forall k_1, k'_1 \in \mathcal{M}_{inter}, k_1 \neq k'_1, \forall t \in \tau : \\
 dep_{Hcl}^{k_1,t} & \leq dep_{Hcl}^{k'_1,t} - \sum_{p \in \mathcal{P}} tl_{Hcl,p} QC_{Hcl,p}^{k'_1,t} + HV(3 - Ex_{k_1,k'_1}^t - \sum_{i \in D_{\Phi(k_1)} \cup \mathcal{N}_{\Phi(k_1)}} x_{i,Hcl}^{k_1,t} \\
 & \quad - \sum_{i \in D_{\Phi(k'_1)} \cup \mathcal{N}_{\Phi(k'_1)}} x_{i,Hcl}^{k'_1,t}) \tag{3.26}
 \end{aligned}$$

Les contraintes (3.25) définissent les relations entre variables d'exclusions de routes sur cet hôpital Hcl. Les contraintes (3.26) expriment les précédences entre routes sur ce même hôpital.

Nous allons maintenant définir l'ensemble des contraintes servant à affecter un chauffeur ou un manutentionnaire à chaque route et à respecter les successions de tournées pour une même personne.

$$\forall k \in \mathcal{M}, \forall t \in \tau : \sum_{r \in \mathcal{R}} \sum_{o \in \mathcal{O}} Liv_r^{k,o,t} \leq 1 \tag{3.27}$$

$$\forall r \in \mathcal{R}, \forall t \in \tau : \sum_{k \in \mathcal{K}} Liv_r^{k,1,t} \leq 1 \tag{3.28}$$

$$\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P} : \sum_{r \in \mathcal{R}} \sum_{o \in \mathcal{O}} Liv_r^{k,o,t} \geq IQC_{i,p}^{k,t} \tag{3.29}$$

$$\forall r \in \mathcal{R}, \forall t \in \tau, \forall o \in \mathcal{O} \setminus |\mathcal{O}| : \sum_{k \in \mathcal{M}} Liv_r^{k,o,t} \geq \sum_{k \in \mathcal{M}} Liv_r^{k,o+1,t} \tag{3.30}$$

$$\forall r \in \mathcal{R} : HV(1 - Ma_r) \geq \sum_{k_1 \in \mathcal{M}_{inter}} \sum_{t \in \tau} \sum_{o \in \mathcal{O}} Liv_r^{k_1,o,t} \tag{3.31}$$

$$\forall r \in \mathcal{R} : HV Ma_r \geq \sum_{k_2 \in \mathcal{M}_{intra}} \sum_{t \in \tau} \sum_{o \in \mathcal{O}} Liv_r^{k_2,o,t} \tag{3.32}$$

$$\begin{aligned}
 & \forall k, k' \in \mathcal{M}, k \neq k', r \in \mathcal{R}, \forall o \in \mathcal{O}, \forall t \in \tau : \\
 fin^{k,t} - dep_{D_{\Phi(k')}}^{k',t} & \leq T_{max} + HV(2 - Liv_r^{k',1,t} - Liv_r^{k,o,t}) \tag{3.33}
 \end{aligned}$$

$$\begin{aligned}
 & \forall k, k' \in \mathcal{M}, (k \neq k'), \forall r \in \mathcal{R}, \forall o \in \mathcal{O} \setminus |\mathcal{O}|, \forall t \in \tau : \\
 fin^{k,t} + d_{D_k, D_{\Phi(k')}} & \leq dep_{D_{\Phi(k')}}^{k',t} + HV(2 - Liv_r^{k,o,t} - Liv_r^{k',o+1,t}) \tag{3.34}
 \end{aligned}$$

Les contraintes (3.27 et 3.28) imposent que chaque route est affectée à au plus une tournée d'une seule personne. Les contraintes (3.29) imposent qu'une route non vide  $k$  est obligatoirement affectée à une tournée d'une personne. Les contraintes (3.30) définissent l'ordre des

tournées des personnes, les premières tournées d'une personne sont affectées à des routes, les dernières tournées peuvent ne pas être affectées. La définition des variables d'affectation des personnes en tant que chauffeur ou manutentionnaire est réalisée par les contraintes (3.31) et (3.32). Les contraintes (3.33) permettent le respect du temps de travail maximal  $T_{max}$  pour chaque personne. Enfin, ce sont les contraintes (3.34) qui définissent l'affectation de deux routes successives entre deux tournées à la suite pour une même personne. Cette personne peut éventuellement voyager entre deux dépôts.

### 3.2.3 Fonction objectif

Nous avons choisi comme première fonction objectif pour le PLNE, de maximiser le temps libre global des manutentionnaires afin de leur permettre d'effectuer d'autres tâches dans l'hôpital Bretonneau pour alléger le travail du personnel de soins. Nous considérons donc ici que le nombre de personnes (chauffeurs et manutentionnaires) est constant. Par conséquent, il s'agit de maximiser :

$$T_{\max} \sum_{r \in \mathcal{R}} Ma_r - \sum_{t \in \tau} \sum_{k_2 \in \mathcal{M}_{intra}} \sum_{i, j \in D_{\Phi(k)} \cup \mathcal{N}_{\Phi(k)}/i \neq j} d_{i,j} \cdot x_{i,j}^{k_2,t} \quad (3.35)$$

Si nous avons choisi comme fonction objectif de minimiser la somme des retards des demandes à livrer ou collecter, il aurait fallu retirer les contraintes 3.13 et ajouter les contraintes suivantes :

$$\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P} : \quad dep_i^{k,t} - l_{i,p,t} \leq ret_{i,p,t} + HV(1 - IQc_{i,p}^{k,t}) \quad (3.36)$$

Avec  $ret_{i,p,t} \geq 0$  des variables continues précisant chacune le retard d'une demande au point  $i$  en flux  $p$  le jour  $t$ . Et il aurait fallu minimiser :

$$\sum_{i \in \mathcal{H} \cup \mathcal{B}} \sum_{p \in \mathcal{P}} \sum_{t \in \tau} ret_{i,p,t} \quad (3.37)$$

Si nous avons choisi de minimiser la somme des dépassements des temps d'autonomie des chariots des repas, il aurait fallu retirer les contraintes 3.23 et 3.24 et ajouter les contraintes suivantes :

$\forall k \in \mathcal{M}, \forall i \in \mathcal{N}_{\Phi(k)}, \forall t \in \tau, \forall p \in \mathcal{P}$  tel que  $p$  est limité dans la durée de transport :

$$dep_i^{k,t} - dep_{D_{\Phi(k)}}^{k,t} - g_p \leq aut_{i,p,t} + HV(1 - IQc_{i,p}^{k,t}) \quad (3.38)$$

$\forall k_1 \in \mathcal{M}_{inter}, \forall k_2 \in \mathcal{M}_{intra}, \forall i \in \mathcal{N}_{\Phi(k_2)}, \forall t \in \tau, \forall p \in \mathcal{P} :$

$$dep_i^{k_2,t} - dep_{D_{\Phi(k_1)}}^{k_1,t} - g_p \leq aut_{i,p,t} + HV(2 - IQc_{i,p}^{k_2,t} - IQtr_p^{k_1,k_2,t}) \quad (3.39)$$

Avec  $aut_{i,p,t} \geq 0$  des variables continues précisant chacune le dépassement du temps d'autonomie des chariots d'une demande au point  $i$  en flux  $p$  le jour  $t$ . Et il aurait fallu minimiser :

$$\sum_{i \in \mathcal{H} \cup \mathcal{B}} \sum_{p \in \mathcal{P}} \sum_{t \in \tau} aut_{i,p,t} \quad (3.40)$$

En revanche, ce modèle suppose un nombre fixe de personnes et se contente de répartir les rôles. Pour minimiser le nombre d'intervenants, les modifications du PLNE, bien que possibles, semblent moins triviales.

#### 3.2.4 Taille du PLNE

Le taille du PLNE est donné par :

- un nombre de variables binaires égal à :

$$|\mathcal{R}| + \tau \cdot \left( |\mathcal{M}_{inter}| \cdot |\mathcal{M}_{intra}| \cdot (1 + |\mathcal{P}|) + |\mathcal{M}| \cdot \left( |\mathcal{O}| \cdot |\mathcal{R}| + |\mathcal{A}| \cdot (|\mathcal{A}| + |\mathcal{P}|) \right) \right)$$

- un nombre de variables entières égal à :

$$\tau \cdot |\mathcal{P}| \cdot (|\mathcal{M}| \cdot |\mathcal{A}| + |\mathcal{M}_{inter}| \cdot |\mathcal{M}_{intra}|)$$

- un nombre de variables continues égal à :

$$\tau \cdot |\mathcal{M}| \cdot (1 + |\mathcal{A}|)$$

- un nombre de contraintes égal à :

$$2 \cdot |\mathcal{R}| + \tau \cdot \left( |\mathcal{A}| \cdot |\mathcal{P}| + |\mathcal{R}| \cdot (1 + |\mathcal{O}|) + |\mathcal{M}_{inter}| \cdot |\mathcal{M}_{intra}| \cdot \left( 2 + |\mathcal{P}| \cdot (4 + |\mathcal{A}|) \right) + |\mathcal{M}| \cdot \left( 5 + 2 \cdot |\mathcal{M}| \cdot |\mathcal{R}| \cdot |\mathcal{O}| + |\mathcal{P}| + |\mathcal{A}| \cdot (2 + 8 \cdot |\mathcal{P}| + |\mathcal{A}|) \right) \right)$$

La taille des instances testées avec deux produits est donnée par le tableau 3.1.

$\tau$	$ \mathcal{A} $	$ \mathcal{P} $	$ \mathcal{R} $	$ \mathcal{O} $	$ \mathcal{M} $	$ \mathcal{M}_{inter} $	$ \mathcal{M}_{intra} $
1	17	2	5	6	36	18	18

TAB. 3.1 – Taille des instances testées

Nous obtenons donc pour ces instances un PLNE avec :

- 13 685 variables binaires,
- 1 872 variables entières,
- 648 variables continues,
- 113 767 contraintes.

L'évaluation expérimentale du PLNE est donnée section 3.5.2.

### 3.3 Algorithme génétique

Etant donné la complexité du problème, nous avons choisi d'utiliser une méta-heuristique pour le résoudre. La première méta-heuristique testée est un algorithme génétique (Holland [99]), méthode fréquemment utilisée dans les problèmes de tournées de véhicules. A

### 3.3. ALGORITHME GÉNÉTIQUE

---

partir d'un ensemble de solutions initiales, ou population de  $N$  individus, elle consiste à faire évoluer cette population en utilisant des opérateurs de sélection, de croisement et de mutation. A chaque itération de l'algorithme, une nouvelle population de solutions ou d'individus est générée. Tout d'abord, un ensemble d'individus est sélectionné pour générer la population suivante. Ces individus sont ensuite croisés pour créer de nouveaux individus et compléter la nouvelle population. Certains de ces nouveaux individus peuvent subir une mutation. Le critère d'arrêt de l'algorithme dans notre cas est un nombre d'itérations sans amélioration de la meilleure solution trouvée ( $\#ite$ ).

---

**Algorithme 1** Structure générale de l'algorithme génétique

---

```
1:  $\mathcal{P}_{courant} \leftarrow$  Initialiser une population de  $N$  individus
2: Evaluer chaque individu de  $\mathcal{P}_{courant}$ 
3:  $S_{best} \leftarrow$  Le meilleur individu  $S \in \mathcal{P}_{courant}$ 
4:  $I \leftarrow 0$ 
5: Tant que  $I < \#ite$  faire
6:    $\mathcal{P}_{enfant} \leftarrow \emptyset$ 
7:   Pour  $j = 0$  à  $j = N/2$  faire
8:      $(P_1, P_2) \leftarrow$  Sélectionner deux individus parents de  $\mathcal{P}_{courant}$ 
9:      $(E_1, E_2) \leftarrow$  Croiser les deux parents  $(P_1, P_2)$  pour obtenir deux individus enfants
10:     $\mathcal{P}_{enfant} \leftarrow \mathcal{P}_{enfant} \cup \{E_1, E_2\}$ 
11:   Fin pour
12:   Muter aléatoirement des individus de la population  $\mathcal{P}_{enfant}$ 
13:    $\mathcal{P}_{courant} \leftarrow \mathcal{P}_{enfant}$ 
14:   Evaluer chaque individu de  $\mathcal{P}_{courant}$ 
15:   Si il existe un individu  $S \in \mathcal{P}_{courant}$  meilleur que  $S_{best}$  alors
16:      $S_{best} \leftarrow S$ 
17:      $I \leftarrow 0$ 
18:   Fin si
19:    $I \leftarrow I + 1$ 
20: Fin Tant que
```

---

Les éléments importants d'un algorithme génétique sont le codage et l'évaluation d'un individu (étapes 2 et 14), l'initialisation d'une population (étape 1), la sélection (étape 8), le croisement (étape 9) et la mutation des individus (étape 12). Ces éléments sont décrits dans les pages qui suivent, dans le cadre de l'adaptation que nous en avons faite au problème de la logistique. Mais avant cela, il est important de noter que l'algorithme génétique proposé prend en paramètre deux grandeurs :  $NbT_{inter}$  et  $NbT_{intra}$  qui correspondent respectivement aux nombres de tournées Inter-hôpitaux (les camions) et intra-Bretonneau (les manutentionnaires) qui peuvent se dérouler simultanément. Les deux paramètres bornent supérieurement le nombre de chauffeurs et de manutentionnaires à  $2.NbT_{inter}$  et  $2.NbT_{intra}$  sans pour autant interdire à l'algorithme génétique de trouver des solutions avec moins d'employés (voir section 3.3.1.2 pour plus de détails).

### 3.3.1 Codage et évaluation d'un individu

Le codage d'une solution est avant tout basé sur l'affectation des demandes aux véhicules. Un individu est représenté par une liste ordonnée de gènes constitués de deux éléments  $(vec_i; dmd_i)$  avec  $vec_i$  un numéro de véhicule et  $dmd_i$  un numéro de demande d'un gène  $i$  (cf. Figure 3.1 pour une illustration du codage). Cette liste de gènes est découpée en plusieurs segments, chaque segment représentant l'ensemble des demandes à traiter dans une même journée. A partir de ce codage, les tournées inter-hôpitaux des véhicules et intra-Bretonneau des manutentionnaires sont construites suivant deux procédures distinctes qui s'appliquent à chaque segment de manière identique. Un individu est évalué en fonction des retards des livraisons, des dépassements d'autonomies des chariots de l'UCPA et du nombre de chauffeurs et manutentionnaires nécessaires.

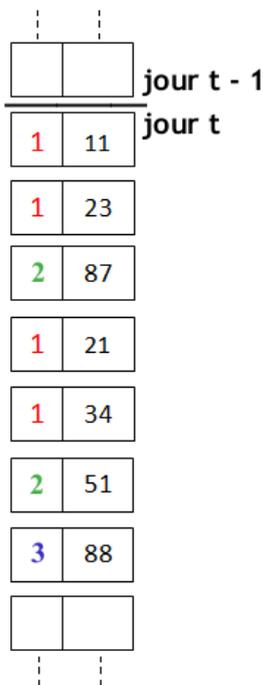


FIG. 3.1 – Illustration du codage

#### 3.3.1.1 Détermination des tournées des véhicules

La première procédure détermine pour chaque journée les tournées des véhicules entre les hôpitaux. Les étapes suivantes sont appliquées tant qu'il reste des gènes à traiter dans le segment. Les gènes sont traités dans leur ordre d'apparition dans le segment. Soit  $(vec_i; dmd_i)$  le gène courant :

- Si le véhicule  $vec_i$  n'est encore associé à aucune route, on en ouvre une nouvelle.
- On ajoute à la fin de la route associée au véhicule  $vec_i$  un sommet représentant l'hôpital dont est issue la demande  $dmd_i$ . Si la capacité restante du véhicule ne

### 3.3. ALGORITHME GÉNÉTIQUE

permet pas d'accueillir tous les chariots représentant la demande  $dmd_i$ , on scinde cette demande en deux et on ouvre une autre route associée au véhicule  $vec_i$  avec le reliquat de chariots n'ayant pas pu être emmenés sur la route précédente.

- S'il n'est pas encore présent, le dépôt du flux associé à la demande  $dmd_i$  est inséré en début et en fin de route pour le chargement des chariots pleins et le retour des chariots vide (ou l'inverse dans le cas du flux de linge sale). On s'arrange pour placer le dépôt du véhicule en tout début et toute fin de route.

La figure 3.2 illustre cette première partie de la procédure. Nous supposons dans cet exemple qu'à partir de la demande 21, la capacité du véhicule est atteinte, c'est-à-dire que la quantité de chariots de la demande 21 est trop importante pour qu'elle loge entièrement dans le véhicule. Cette demande sera donc divisée en deux, une première partie des chariots sera livrée à la fin de la première tournée puis la seconde partie sera livrée au début de la deuxième tournée.

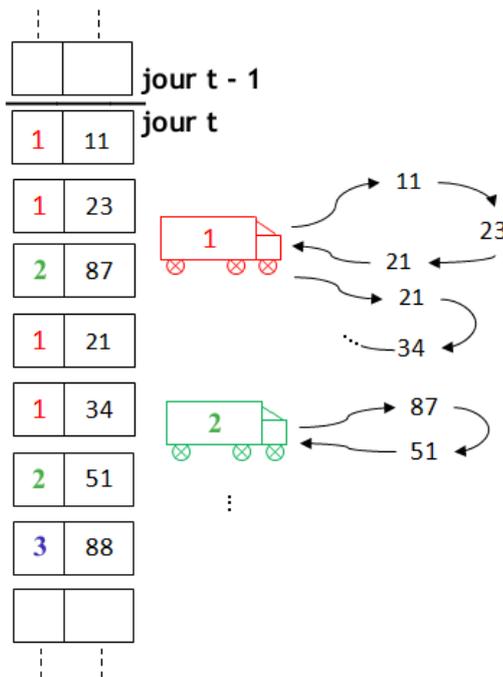


FIG. 3.2 – Illustration du codage

Les routes ainsi construites sont ensuite exploitées dans l'ordre dans lequel elles ont été créées :

- La date de départ d'un véhicule d'une route est calculée en fonction de sa précédente tournée, de la fenêtre de temps de la première demande (inutile de partir trop tôt), du temps de chargement des chariots à livrer ou collecter et du nombre de véhicules circulants au même moment. Une tournée peut être retardée si le nombre de véhicules circulants est égal à  $NbT_{inter}$ .

### 3.3. ALGORITHME GÉNÉTIQUE

---

- les dates de livraisons/collectes de chaque demande sont ensuite déterminées en fonction de la date de départ du véhicule, des fenêtres de temps associées aux demandes, des durées de déchargement et chargement, des durées de livraison et des places disponibles sur les quais, dépendantes des précédentes routes calculées. Cette étape permet notamment de construire une liste des demandes à livrer à Bretonneau, triée par dates d'arrivées, ainsi qu'une autre liste pour les demandes à collecter.
- Une optimisation gloutonne est ensuite appliquée. Elle consiste à tester le déplacement de chaque sommet de la route vers toutes les places possibles. Un sommet est déplacé uniquement s'il améliore un des critères ci-dessous sans détériorer les autres. Les critères sont la durée totale de la tournée, les dates d'arrivée des demandes à Bretonneau (puisque les dates de livraison dans les bâtiments de Bretonneau ne sont pas connus à cette étape) et les retards des livraisons/collectes dans les hôpitaux autres que Bretonneau. Le meilleur déplacement pour chaque sommet est choisi en fonction d'abord de la diminution du retard engendré, puis par rapport aux dates d'arrivées des chariots sur le quai de Bretonneau et enfin en fonction de la diminution de la durée totale de la tournée.

#### 3.3.1.2 Détermination de tournées des manutentionnaires

La seconde procédure détermine jour par jour les tournées des manutentionnaires au sein de l'hôpital Bretonneau. Rappelons que le nombre de manutentionnaires présent à un instant donné de la journée est borné par  $2 \times NbT_{intra}$ .

La liste des tournées des manutentionnaires est construite itérativement en utilisant prioritairement les fenwick. Le principe général, basé sur la liste précédemment construite  $\mathcal{L}$  des demandes à livrer triée par dates d'arrivées, est le suivant :

- Si un manutentionnaire est à quai, on crée la liste des chariots également à quai dont la date de livraison au plus tôt dans les services n'est pas trop éloignée de l'heure courante (moins de 45 minutes) et on trie cette liste par date de livraison au plus tard dans les services croissante et en cas d'égalité en regroupant par bâtiments.
- Si dans les minutes qui viennent on sait qu'un véhicule doit arriver au quai avec des chariots plus prioritaires, i.e. dont la date de livraison au plus tard dans les services est inférieure à celles des chariots de la liste, le manutentionnaire attend et les nouveaux chariots sont insérés en tête de liste. Le délai d'attente a été fixé à 10 minutes au maximum.
- Si la liste n'est pas vide, on affecte au manutentionnaire les premiers chariots (en fonction de la capacité et du moyen de transport qu'il peut prendre) et il débute sa tournée en livrant les chariots par ordre de priorité et en livrant en une seule fois tous les chariots d'un même bâtiment qu'il a emmené.

La collecte des chariots, notamment du linge sale et de la salubrité, pose un problème spécifique puisqu'il faut que les manutentionnaires aillent collecter les chariots dans les services pour les amener sur le quai avant le passage du véhicule chargé de les évacuer. La difficulté est de déterminer l'heure de départ des tournées des manutentionnaires. Pour résoudre ce

problème, les demandes à collecter sont converties en demandes à livrer fictives et ajoutées au préalable dans la liste  $\mathcal{L}$ . Les fenêtres de livraison dans les services de ces demandes fictives sont de la forme  $[A; l_d]$ , où  $l_d$  est l'heure d'arrivée du véhicule sur les quais moins le temps de trajet nécessaire pour aller du service au quai, et  $A$  est l'heure d'embauche des manutentionnaires. Ces fenêtres de temps sont considérées comme dures, aucune livraison ne doit être en retard. De plus, nous interdisons qu'une tournée de collecte de manutentionnaire se termine après le départ du véhicule de collecte. Cependant, l'heure d'embauche des manutentionnaires n'est pas une donnée du problème. Il faut donc calculer la valeur  $A$  en fonction de la liste  $\mathcal{L}$  afin de déterminer toutes les routes intra sachant que toutes les demandes à collecter doivent être réalisées en temps et en heure. La valeur de  $A$  peut être bornée par la date au plus tôt d'embauche possible d'un manutentionnaire en réel et la date d'arrivée du premier chariot sur le quai. Grâce à cet encadrement, nous procédons par une recherche dichotomique pour trouver la meilleure valeur de  $A$  possible (une valeur trop petite nécessitera un grand nombre de manutentionnaires). Pour chaque valeur de  $A$  testée, nous calculons toutes les tournées des manutentionnaires, si les contraintes sur les chariots à collecter sont violées alors nous testons une valeur dans l'encadrement inférieur, sinon dans l'encadrement supérieur.

#### 3.3.1.3 Dimensionnement des équipes

La dernière étape pour évaluer une solution est de déterminer le nombre de manutentionnaires et de chauffeurs nécessaires pour effectuer toutes les tournées précédemment calculées. Si les tournées sont considérées comme des tâches fixées (dates de début et de fin fixées), et les personnes comme des ressources, l'objectif est d'exécuter toutes les tâches en minimisant la quantité de ressources utilisées. De plus l'écart entre la date de fin de la dernière tâche d'une ressource et la date de début de sa première tâche doit être inférieur au temps de travail maximal d'une personne par jour ( $T_{max}$ ). En considérant une unique journée, le problème est un problème NP-difficile intitulé "*The fixed job schedule problem with spread-time constraint*" déjà étudié dans la littérature (Fischetti et al. [72]). Mais ce problème doit en plus ici être résolu sur les cinq jours de la semaine, et chaque personne doit avoir les mêmes horaires chaque jour. Ce problème devant être résolu pour chaque individu généré, nous avons implémenté deux algorithmes rapides afin de déterminer une borne inférieure et supérieure du nombre de personnes nécessaires pour effectuer les tournées. Ces algorithmes sont exécutés indépendamment pour déterminer le nombre de manutentionnaires en fonction des tournées intra-Bretonneau, puis pour déterminer le nombre de chauffeurs en fonction des tournées inter-hôpitaux.

La borne supérieure est calculée par une heuristique gloutonne d'affectation des tournées. On dispose de 5 listes de tournées (une par jour) ordonnées par date de début croissante. La plus petite date de début, toutes journées confondues, définit la date de début de travail d'une personne, sa date de fin de travail étant donc  $T_{max}$  unité de temps plus tard. On affecte ensuite pour chaque jour des tournées à cette personne, pour cela, jour par jour on prend les tournées dans l'ordre de la liste correspondante et si la personne peut effectuer la tournée on lui affecte et on la retire de la liste sinon on passe à la tournée suivante. On

### 3.3. ALGORITHME GÉNÉTIQUE

---

recommence le processus jusqu'à ce que toutes les listes soient vides. Le nombre d'itérations est une borne supérieure du nombre de personnes nécessaires.

La borne inférieure est calculée à partir d'une courbe de charge maximale qui recouvre toutes les courbes de charges de chaque journée. L'idée directrice est d'annuler cette courbe de charge en lui retirant itérativement des rectangles correspondants à la charge de travail d'une personne sur une journée ( $T_{max}$ ). Soit  $C_0(t)$  cette courbe de charge maximale à l'étape 0, une nouvelle courbe de charge  $C_i(t)$  est calculée à chaque étape  $i$  de l'algorithme suivant :

$$\begin{aligned} 1 \quad \delta_i &= \min\{t/C_{i-1}(t) \neq 0\} \\ 2 \quad C_i(t) &= \begin{cases} 0 & \text{si } t < \delta_i \\ \max(0; C_{i-1}(t) - 1) & \text{si } \delta_i \leq t < \delta_i + T_{max} \\ C_{i-1}(t) & \text{si } \delta_i \geq \delta_i + T_{max} \end{cases} \end{aligned}$$

L'algorithme se termine lorsque  $\forall t, C_i(t) = 0$  et la borne inférieure est égale au nombre d'itérations effectuées. Cet algorithme permet de trouver des solutions en considérant que les tournées peuvent être préemptées.

#### 3.3.1.4 Evaluation d'un individu

Comme pour la plupart des problèmes de tournées de véhicules avec des fenêtres de temps, il est difficile de générer uniquement des solutions respectant ces fenêtres (sans retard). L'une des méthodes classiques pour palier cette difficulté est de relâcher cette contrainte et de considérer le retard comme un critère dans la fonction objectif. De plus la contrainte des fenêtres de temps dépend de chaque service de l'hôpital, si une bonne solution est trouvée mais qu'elle possède un retard important sur une demande d'un service, il est envisageable de revoir la fenêtre de temps de cette demande en discutant avec le service concerné. D'autant plus que les fenêtres de livraisons dont nous disposons sont celles actuellement utilisées, il n'y a pas été possible de se renseigner auprès de chaque service du CHRU pour connaître leurs souhaits.

L'évaluation d'un individu doit donc tenir compte de plusieurs critères : (1) la somme des retards des demandes à livrer ou collecter, (2) la somme des dépassement des temps d'autonomie des chariots des repas et (3) l'estimation du nombre de manutentionnaires et de chauffeurs nécessaires (moyennes entre les bornes inférieures et supérieures). Le choix qui a été fait pour tenir compte de ces critères a été de considérer une combinaison linéaire. Nous noterons  $Eval_i$  l'évaluation d'un individu  $i$ .

#### 3.3.2 Initialisation d'une population

La population initiale est constituée de  $N$  individus générés aléatoirement selon la procédure suivante :

- Découper la période considérée  $\tau$  (5 jours de la semaine) en demi-journées.
- Pour les demandes attribuées à un jour particulier, affecter chaque demande dont l'intersection entre la fenêtre de temps correspondante et la demi-journée est la plus

### 3.3. ALGORITHME GÉNÉTIQUE

---

- grande parmi les deux demi-journées concernées.
- Affecter aléatoirement les autres demandes (non attribuées à un jour) aux demi-périodes.
- Enfin, pour chaque demi-journée :
  - Trier aléatoirement les demandes affectées à cette demi-journée.
  - Et affecter un numéro de véhicule pour chaque demande et de manière équiprobable.

#### 3.3.3 Sélection des individus

Trois types d'opérateurs de sélection ont été implémentés pour choisir dans la population courante les individus qui donneront par croisement la population suivante :

- Le premier opérateur sélectionne les 50% meilleurs individus de la population. Des couples d'individus parents sont ensuite constitués aléatoirement de manière à ce que chaque individu soit sélectionné exactement deux fois.
- Le deuxième opérateur est une sélection probabiliste de type roulette avec la possibilité de sélectionner plusieurs fois le même individu. Dans ce cas, les meilleurs individus ont plus de chance d'être sélectionnés par rapport au moins bon individus. Chaque individu  $i$  a une probabilité de sélection  $P_i^{select}$  telle que :

$$P_i^{select} = \frac{\max_{j \in \mathcal{P}}(Eval_j) - Eval_i}{N \times \max_{j \in \mathcal{P}}(Eval_j) - \sum_{j \in \mathcal{P}} Eval_j}$$

- Le dernier opérateur sélectionne les individus par tournois deux à deux. Le meilleur des deux individus choisis aléatoirement est sélectionné. Comme pour le précédent opérateur de sélection, un individu peut être sélectionné plusieurs fois.

#### 3.3.4 Croisement des individus

Une fois les deux individus parents sélectionnés, deux types d'opérateurs de croisement sont utilisés pour créer deux individus enfants. Ces opérateurs portent uniquement sur des segments de chromosomes pris aléatoirement mais appartenant à la même journée. Le premier type de croisement est effectué à partir d'un opérateur de recopie (cf. figure 3.3), aussi appelé *Partially Mapped Crossover* (Goldberg [89]). Cet opérateur consiste à recopier un segment d'un parent P2 dans un individu enfant E1 et à compléter cet individu E1 par les demandes du parent P1 dans l'ordre de leurs apparitions dans le chromosome et en évitant les doublons. Lorsqu'une demande  $d$  apparaît en double, elle est remplacée par la demande  $d'$  telle que la demande  $d$  du segment P2 correspond à l'emplacement de  $d'$  dans P1. Par exemple, dans la figure 3.3, la demande 11 a été remplacée par la demande 21.

Le deuxième opérateur est un opérateur de type insertion (cf. exemple Figure 3.4). Pour générer le premier enfant, un groupe de gènes consécutifs du premier parent est d'abord sélectionné. Puis un point d'insertion dans le deuxième parent est choisi. Et le premier enfant correspond à l'insertion du groupe de gènes au point d'insertion et à la suppression des gènes en doublon dans le deuxième parent. Le second enfant est créé de la même manière

### 3.3. ALGORITHME GÉNÉTIQUE

---

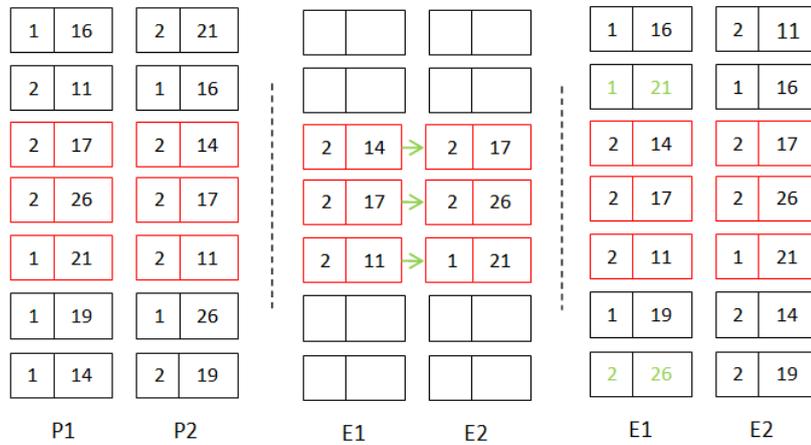


FIG. 3.3 – Partially Mapped Crossover

en inversant les parents.

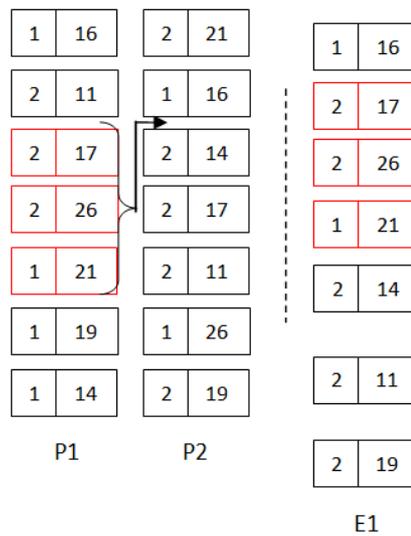


FIG. 3.4 – Opérateur insertion

#### 3.3.5 Mutation des individus

Chaque individu a une probabilité faible  $P_{mut}$  de subir une mutation. Dans ce cas, il existe deux possibilités de mutation :

- La mutation de véhicule qui modifie le numéro de véhicule d'un gène tiré au hasard.
- La mutation de place d'un gène qui avance ou recule dans un segment un gène sélectionné aléatoirement suivant que la demande associée au gène est en retard ou non.

Les résultats de l'algorithme génétique sont présentés en section 3.5.3.1.

## 3.4 Recherche tabou

La deuxième méta-heuristique développée pour résoudre notre problème est une recherche tabou ([88] et [95]). La structure générale de la méthode est décrite par l'algorithme 2. Le critère d'arrêt de l'algorithme est à nouveau un nombre maximal d'itérations sans amélioration de la meilleure solution connue.

---

### Algorithme 2 Structure générale de l'algorithme tabou

---

```

1:  $S_{courant} \leftarrow$  Générer aléatoirement une solution initiale
2:  $S_{best} \leftarrow S_{courant}$ 
3:  $I \leftarrow 0$ 
4: Tant que  $I < \#ite$  faire
5:    $\Psi \leftarrow$  Calculer le voisinage non tabou de la solution  $S_{courant}$ 
6:    $S_{courant} \leftarrow$  Meilleure solution du voisinage  $\Psi$ 
7:   Stocker le mouvement dans la liste tabou
8:   Si  $S_{courant}$  est meilleur que  $S_{best}$  alors
9:      $S_{best} \leftarrow S$ 
10:     $I \leftarrow 0$ 
11:   Fin si
12:    $I \leftarrow I + 1$ 
13: Fin Tant que

```

---

Les éléments importants de l'algorithme tabou sont : le codage et l'évaluation d'une solution, la construction d'une solution initiale, l'opérateur de voisinage et la liste tabou. Nous allons décrire par la suite chaque élément, à l'exception du codage et de l'évaluation d'une solution, qui restent inchangés par rapport à l'algorithme précédent.

#### 3.4.1 Construction d'une solution initiale

La méthode tabou commence la recherche par une solution initiale générée selon les étapes suivantes :

- Pour toutes les demandes qui ne sont pas affectées à un jour fixe, le jour est choisi de manière aléatoire en essayant d'équilibrer la charge des chariots à livrer ou collecter chaque jour. Ensuite, pour chaque jour  $t$ , les demandes sont triées par  $l_{i,p,t}$  croissant (date de livraison souhaitée au plus tard).
- L'affectation de chaque demande au véhicule est réalisée comme suit :
  - Les véhicules sont triés dans une liste par capacité décroissante.
  - Tant qu'il reste des demandes non encore affectées à un véhicule :
    - Le premier véhicule  $k$  de la liste est retiré.
    - Les premières demandes non affectées sont attribuées au véhicule  $k$  jusqu'à atteindre la capacité maximale du véhicule.
    - Enfin, le véhicule  $k$  est réinséré à la fin de la liste des véhicules, pour une utilisation ultérieure dans la journée.

### 3.4.2 Opérateur de voisinage

Le voisinage d'une solution courante est construit avec deux types d'opérateurs de voisinages. Le premier consiste à changer le numéro du véhicule chargé du transport d'une demande dans la solution considérée, par un autre véhicule qui peut réaliser cette demande. Le second opérateur de voisinage consiste à déplacer une demande de la solution vers le haut. Dans ce cas, le nombre de mouvements testés est limité à une valeur préfixée, notée  $Op_{mov}$ . Si la demande n'est pas assignée à un jour particulier, les différents tests des mouvements porteront aussi sur les autres jours.

Pour construire le voisinage d'une solution, l'opérateur de voisinage est tout d'abord sélectionné pour être appliqué à la demande qui souffre du plus grand retard de livraison. L'opérateur est sélectionné de manière probabiliste de sorte que lorsqu'il améliore la solution courante, sa probabilité de sélection augmente, et dans le cas contraire sa probabilité diminue. Soit  $Suc_1$  le nombre de fois que l'opérateur 1 a amélioré la solution courante plus le nombre de fois que l'opérateur 2 n'a pas amélioré la solution courante, et respectivement  $Suc_2$  pour l'opérateur 2. L'opérateur de voisinage 1 a une probabilité d'être sélectionné égale à  $Suc_1/(Suc_1 + Suc_2)$  (pour l'opérateur 2, elle est de  $Suc_2/(Suc_1 + Suc_2)$ ). Au départ de l'algorithme,  $Suc_1 = Suc_2 = 1$ .

Si aucune des demandes qui ne sont pas interdites par la liste tabou n'est en retard, alors la demande dont les chariots arrivent le plus en avance au point de livraison est sélectionnée. Dans ce cas, l'opérateur de mouvement déplace la demande sélectionnée vers le bas.

### 3.4.3 Liste tabou

La liste tabou est constituée des couples : (type d'opérateur de voisinage ; numéro de demande sélectionnée). Tout mouvement d'une solution courante vers une autre utilisant ce type d'opérateur sur cette demande est interdit pendant un nombre d'itérations égal à la taille de la liste tabou notée  $T_{lt}$ . Cette liste tabou permet d'éviter de toujours se focaliser sur la même demande en retard.

## 3.5 Résultats expérimentaux

Avant de tester les algorithmes développés pour le problème de logistique sur des instances réelles, nous avons généré aléatoirement des instances en respectant les hypothèses de départ (cf. section 3.1) et dont les caractéristiques sont inspirées de l'activité réelle de la logistique au sein du CHRU, voire identique pour tous les éléments concernant la structure du CHRU (hôpitaux, bâtiments, dépôts, quais et services), les moyens de transports (types, quantités et capacités) et les demandes sur lesquelles nous avons des informations précises. Le langage utilisé pour l'implémentation est le  $C++$ . L'ordinateur utilisé est un Pentium(R) 4 CPU 3.00 GHz, 1.00 Go de RAM et le solveur utilisé pour le PLNE est CPLEX version 10.

### 3.5.1 Génération d'instances

Etant donné l'ampleur de la collecte de données à réaliser (pour chaque service, relever les quantités des chariots commandées par produit, les temps de livraison, etc.), le CHRU n'a pas encore fini de fournir toutes les données réelles. Les données les plus délicates à collecter sont les quantités des chariots demandées et les fenêtres de temps des demandes. Cependant, pour certaines logistiques (Logistique Hôtelière et pharmacie), nous avons une estimation très fine des quantités de chariots livrées par service. Pour les autres logistiques, les quantités de chariots ont été données "à la louche" par le personnel concerné du CHRU mais peuvent être retrouvées ou confirmées en analysant les tournées actuelles. Les fenêtres de livraison des services sont plus difficiles à connaître. Elles dépendent fortement des services sauf pour certains flux où elles sont clairement connues, comme les repas. Soit nous avons pu obtenir des estimations de ces fenêtres de temps soit nous nous sommes basés sur les tournées actuelles.

En se basant sur quelques données réelles fournies par le CHRU et l'étude complète des flux (cf. Annexe A), nous avons créé dans un premier temps une instance théorique aussi réaliste que possible. Les caractéristiques principales de cette instance sont les suivantes :

- L'ensemble  $\mathcal{D}$  est constitué de 4 dépôts (logipôle, Blanchisserie, Salubrité à Bretonneau et les Archives près de Trousseau).
- L'ensemble  $\mathcal{H}$  est composé non pas de 6 mais de 9 hôpitaux (Trousseau 1, Trousseau 2, Trousseau 3, Bretonneau, Clocheville 1, Clocheville 2, Ermitage, CPTS, et CPU). En effet, dans la résolution du problème, les demandes des hôpitaux autres que Bretonneau sont groupées car seuls les horaires de passages des camions en ces lieux nous intéressent. Il est indispensable de connaître ces dates de passages pour assurer la synchronisation des tournées des véhicules et des manutentionnaires au quai de Bretonneau. Ces demandes groupées sont trop volumineuses par rapport aux capacités des véhicules. C'est pour cela que les hôpitaux Trousseau et Clocheville ont été divisés en plusieurs points de livraison (hôpitaux virtuels). Cela permet d'améliorer la granularité de l'approche.
- L'ensemble  $\mathcal{B}$  est constitué de 8 bâtiments à Bretonneau.
- la période de planification  $\tau$  est de 5 jours, les demandes affectées au sixième jour sont celles qui ne sont pas assignées à jour fixe dans la semaine.
- L'ensemble  $\mathcal{P}$  est composé de 11 types de chariots ou flux (LH, Pharmacie, repas du midi, repas du soir, linge propre 1, linge propre 2, linge sale 1, linge sale 2, la stérilisation, les dossiers patients, et les déchets). Pour la même raison que les hôpitaux, le flux de linge propre et de linge sale sont divisés de manière à obtenir une granularité plus fine (la quantité des chariots demandée en linge étant plus élevée par rapport aux autres flux). Il y a donc 3 types de chariots à collecter et 8 à livrer.
- L'ensemble des véhicules  $\mathcal{V}$  est constitué de 16 camions dont la capacité peut varier entre 12 et 21 chariots (selon le volume et le poids des chariots et le type de véhicules).
- Il y a au total 3 fenwicks ( $\mathcal{F}_{int} = 2$  et  $\mathcal{F}_{ext} = 1$ ). Nous avons l'estimation du poids moyen d'un chariot par flux. Suivant les chariots à emmener, les fenwicks intérieurs peuvent tracter en moyenne 5 chariots, alors que le fenwick extérieur peut en tracter en moyenne 6. Les fenwicks permettent de se déplacer deux fois plus vite qu'à pied.

### 3.5. RÉSULTATS EXPÉRIMENTAUX

---

- L'ensemble  $\mathcal{R}$  est composé de 25 personnes à répartir entre l'équipe des chauffeurs et celle des manutentionnaires. Un manutentionnaire peut tracter en moyenne 2 chariots.

Cette instance compte au total 631 demandes. Les temps de chargement et déchargement, le temps de livraison/collecte suivant les flux et le point de livraison, le nombre de places par quais d'hôpital, l'ensemble des produits transportables suivant le moyen de transport, ainsi que les durées de déplacements entre hôpitaux et au sein de Bretonneau sont celles fournies par le CHRU. Pour les quelques flux dont le CHRU nous a fourni les distributions empiriques du nombre de chariots demandés, nous avons retenu pour chaque service une valeur haute du nombre de chariots (correspondant à une probabilité cumulée de l'ordre de 90%). Pour les autres flux, nous nous sommes basés sur les tournées existantes pour estimer le nombre de chariots constituant chaque demande. De même, les fenêtres de livraison ont été définies à partir des tournées actuelles, elles sont de l'ordre de la demi-journée pour la plupart des flux et de l'ordre de deux heures pour les plateaux repas. Les collectes de linge sale et des déchets sont prévues en début de matinée.

Pour réaliser les tests, cent instances ont été construites à partir de l'instance théorique précédemment décrite, en faisant varier aléatoirement les quantités de chariots demandés de plus ou moins 1 ou plus ou moins 2, suivant que le nombre théorique est inférieur ou supérieur à 8. Un maximum de 2182 chariots peut être à livrer ou à collecter dans une même instance. Les fenêtres de temps sont également modifiées aléatoirement, la date de livraison au plus tôt est retardée d'au plus une heure et demie et la date de livraison au plus tard est décalée de plus ou moins une heure au maximum, en conservant bien sûr la cohérence entre les deux dates.

#### 3.5.2 Résultats pour le PLNE

Nous avons voulu tester dans un premier temps le PLNE avec Cplex. Cependant, le nombre de variables et de contraintes est beaucoup trop important pour tester sur des instances détaillées précédemment. De manière à tester sur des instances raisonnables pour le solveur, nous avons limité la période de planification à une journée, le nombre de véhicules à 9, le nombre de personnes à 5 et le nombre de types de produits à 2 maximum. Nous considérons que les produits sont (a) livrés ou (b) collectés ou (c) livrés avec une autonomie de transport.

Pour les instances considérant un seul produit (à livrer uniquement), sur 6 instances, le modèle ne trouve pas de solution réalisable pour une instance et retourne une solution réalisable en moyenne en 211 secondes (au plus en 729 secondes). Pour les instances à deux produits (cas (a), (b) ou (c)), 18 instances ont été testées. 3 instances demeurent non résolues au bout de 2500 secondes. Pour les autres instances, une solution réalisable est trouvée en moyenne en 635 secondes (au plus en 1522 secondes). Nous avons été également confrontés au problème de l'occupation de l'espace mémoire vive lorsque nous avons voulu augmenter la taille des instances. Très vraisemblablement, ce modèle ne pourra pas résoudre des instances sur plus d'une journée. En revanche, le modèle semble bien se com-

### 3.5. RÉSULTATS EXPÉRIMENTAUX

---

porter si on augmente le nombre de demandes.

Nombre d'instances	Nombre de produits			Temps moyen (s)	Nombre d'instances sans solution réalisable
	cas (a)	cas (b)	cas (c)		
6	1	0	0	211,2	1
6	2	0	0	512,6	1
6	1	0	1	726,2	1
6	1	1	0	667,0	0

TAB. 3.2 – Résultat pour le PLNE

#### 3.5.3 Résultats pour les méta-heuristiques

##### 3.5.3.1 L'algorithme génétique

Des tests préliminaires ont montré qu'une suite de cent itérations sans amélioration ( $\#ite$ ) constitue un critère d'arrêt satisfaisant, la taille de population a été fixée à  $N = 100$  et la probabilité de mutation  $P_{mut}$  à 25%, un des deux opérateurs de mutation étant alors choisi de façon équiprobable. Nous nous sommes ensuite surtout préoccupés de l'influence de l'opérateur de sélection (sélection des 50% meilleurs (M), sélection par roulette (R) et sélection par tournois (T)) et de l'influence de l'opérateur de croisement (par recopie (C) ou par insertion (I)). Un couple (sélection, croisement) est appelé une configuration. Deux autres paramètres importants sont le nombre de tournées inter-hôpitaux  $NbT_{inter}$  et intra-hôpital  $NbT_{intra}$  pouvant se dérouler au même moment. Ils ont été fixés à  $NbT_{inter} = 5$  et  $NbT_{intra} = 7$ .

Chacune des six configurations sélection-croisement a été testée sur les 100 instances générées afin de les comparer entre elles. Les résultats sont résumés dans le tableau 3.3.  $\Delta_i$  est la moyenne sur toutes les instances des écarts relatifs de la solution trouvée par la configuration  $i$  à la meilleure solution trouvée par l'ensemble des 6 configurations, l'écart type est indiqué par  $\sigma(\Delta_i)$ .  $Nb_{min}$  est le nombre de fois où la configuration  $i$  trouve la meilleure solution.  $Tps$  indique le temps moyen de résolution en secondes et  $Nb_{eval}$  représente le nombre moyen en milliers d'appels à la fonction d'évaluation d'un individu.

L'opérateur de croisement I semble être plus performant que l'opérateur C quelque soit l'opérateur de sélection. Tandis que l'opérateur de sélection R est le moins performant. De plus, les deux combinaisons de cet opérateur I avec les opérateurs M et T dominent les autres configurations.

Le tableau 3.4 confirme la dominance des configurations (I,M) et (I,T) sur les autres configurations. Il présente une moyenne sur les 100 instances de la fonction objectif, de la somme des retards en minutes, de la somme des dépassements des autonomies des chariots repas, des bornes inférieure et supérieure du nombre de manutentionnaires, des bornes in-

### 3.5. RÉSULTATS EXPÉRIMENTAUX

---

Crois.	C	I	C	I	C	I
Sélec.	M	M	R	R	T	T
$\Delta_i$ (%)	46,1	29,0	67,2	66,3	40,9	29,4
$\sigma(\Delta_i)$	25,7	28,1	20,6	18,7	26,2	29,8
$Nb_{min}$	10	31	5	2	19	33
$Tps$ (s)	606	736	371	411	585	803
$Nb_{eval}$	33,1	40,1	20,3	22,7	32,1	43,9

TAB. 3.3 – Résultat pour l'AG

férieure et supérieure du nombre de chauffeurs, du nombre de demandes en retard et du nombre de demandes en chariot repas pour lesquelles l'autonomie d'au moins un chariot a été dépassée. Nous pouvons également noter que lors de l'évaluation d'une solution, la différence entre la borne supérieure et la borne inférieure est égale dans la plupart des cas à une personne seulement pour le calcul du nombre de manutentionnaires, et à deux personnes pour le calcul du nombre de chauffeurs.

Crois.	C	I	C	I	C	I
Sélec.	M	M	R	R	T	T
Fonction objectif	5831	4153	9700	9098	5524	4182
Somme des retards	4920	3405	8434	7968	4629	3444
Somme des dépassements des autonomies	655	494	1003	868	637	483
Borne inférieure (manutentionnaires)	14,42	14,19	14,71	14,72	14,35	14,37
Borne supérieure (manutentionnaires)	14,89	14,93	15,12	15,18	15	14,92
Borne inférieure (chauffeurs)	9,78	9,76	10,04	9,98	9,88	9,72
Borne supérieure (chauffeurs)	12,12	11,9	12,69	12,48	12,35	11,9
Nombre de demandes en retard	60,79	52,49	82,55	80,21	56,89	52,44
Nombre de demandes avec autonomie dépassée	7,8	7,03	9,7	8,96	7,84	7,07

TAB. 3.4 – Résultats des six configurations de l'AG

Sur l'instance théorique, l'algorithme génétique (N=100, I, M) a fourni une solution constituée de 154 tournées inter-hôpitaux avec 10 chauffeurs et 14 manutentionnaires, pour un cumul de 25 autorisé. Il y a 68 demandes en retard sur 631 pour une somme totale de 1468 minutes de retard. Néanmoins, la configuration (I,M) conduit probablement à une convergence de la population trop rapide. Pour éviter le phénomène de "super individu",

### 3.5. RÉSULTATS EXPÉRIMENTAUX

$Op_{mov}$	20	40	60	20	40	60	20	40	60
$T_{lt}$	20	20	20	15	15	15	10	10	10
$\Delta_i(\%)$	46,9	49,2	48,7	45,9	44,7	42,4	47,1	42,2	46,8
$\sigma(\Delta_i)(\%)$	22,4	21,5	22,2	22,6	25,9	22,8	22,6	24,7	22,1
$Nb_{min}$	6	4	8	8	11	8	5	12	6
$Tps(s)$	66,3	107,9	146,5	61,1	107,7	137,6	68,1	104,8	142,4
$Nb_{\Delta \leq 10}$	9	8	8	10	16	9	10	16	6
$Nb_{\Delta \leq 20}$	14	14	14	16	21	20	14	23	14

$Op_{mov}$	20	40	60	20	40	60
$T_{lt}$	5	5	5	0	0	0
$\Delta_i(\%)$	45,9	46,2	44,2	53,8	54,5	53,4
$\sigma(\Delta_i)(\%)$	23,1	23,7	26,5	21,9	21,3	22,4
$Nb_{min}$	7	6	11	1	2	5
$Tps(s)$	66,7	102,8	139,6	79,0	136,1	168,1
$Nb_{\Delta \leq 10}$	9	9	20	5	6	7
$Nb_{\Delta \leq 20}$	19	16	24	8	8	10

TAB. 3.5 – Résultats pour la recherche tabou

d'autres tests seront effectués pour tous ces opérateurs de sélection sans en écarter pour l'étude suivante ( voir chapitre 4 : Modèle appliqué).

#### 3.5.3.2 La recherche tabou

Après quelques tests préliminaires, le nombre maximum d'itérations sans amélioration de la meilleure solution a été fixé à 100. Pour les paramètres : taille de la liste tabou ( $T_{lt}$ ) et taille de l'opérateur de voisinage de type déplacement d'une demande ( $Op_{mov}$ ), nous avons testé différentes combinaisons. Le tableau 3.5 présente pour les 100 instances générées les mêmes indicateurs que pour les tests de l'algorithme génétique : la moyenne  $\Delta_i$  des écarts relatifs entre la meilleure solution trouvée pour toutes les combinaisons et la meilleure solution trouvée par la configuration  $i$ , l'écart type  $\sigma(\Delta_i)$ , le nombre d'instances pour lesquelles la combinaison  $i$  trouve la meilleure solution ( $Nb_{min}$ ) et le temps moyen de résolution en secondes ( $Tps$ ). Pour une comparaison plus fine, nous avons ajouté le nombre d'instances pour lesquelles l'écart relatif est inférieur à 10% ( $Nb_{\Delta \leq 10}$ ) et 20% ( $Nb_{\Delta \leq 20}$ ).

Les résultats ont en moyenne des écarts relatifs importants car les meilleures solutions trouvées parmi toutes les meilleures combinaisons ont une petite valeur. Les résultats montrent globalement peu de différences. Néanmoins, nous pouvons remarquer que trois combinaisons semblent dominer les autres : ( $Op_{mov} = 40, T_{lt} = 15$ ), ( $Op_{mov} = 40, T_{lt} = 10$ ) et ( $Op_{mov} = 60, T_{lt} = 5$ ), pour un temps de résolution à peu près similaire. Nous nous sommes donc intéressé à ces trois combinaisons et avons relevé la moyenne sur 100 instances de la fonction objectif, de la somme des retards en minutes, de la somme des dépassements des autonomies des chariots repas, des bornes inférieure et supérieure du nombre de manu-

### 3.5. RÉSULTATS EXPÉRIMENTAUX

---

tentionnaires, des bornes inférieure et supérieure du nombre de chauffeurs, du nombre de demandes en retard et du nombre de demandes en chariot repas pour lesquelles l'autonomie d'au moins un chariot a été dépassée. Le tableau 3.6 présente ces résultats. Ces trois combinaisons permettent de trouver des solutions avec en moyenne très peu de demandes en retard, entre 4 et 7. Le nombre de personnes total varie en moyenne entre 23 et 26. Parmi ces trois combinaisons, celle où  $Op_{mov} = 40$  et  $T_{lt} = 10$  paraît le meilleur compromis puisqu'elle possède des valeurs moyennes de ces indicateurs sensiblement inférieures ou proche des minimums par rapport aux deux autres. Nous pouvons remarquer également que la somme des retards de livraisons/collectes est nettement inférieure à la somme des dépassements d'autonomie des transports de chariots. Ce phénomène s'explique facilement par le choix de la demande sur laquelle l'opérateur de voisinage va travailler. Il s'agit la demande la plus en retard sans tenir compte des autonomies des chariots. L'une des améliorations possibles de cet algorithme tabou est de prendre en compte ces dépassements dans le choix de la demande.

$Op_{mov}$	40	40	60
$T_{lt}$	15	10	5
Fonction objectif	709,57	639,17	682,72
Somme des retards	102,47	65,17	60,96
Somme des dépassements des autonomies	359,4	326,65	374,81
Borne inférieure (manutentionnaires)	13,75	13,69	13,78
Borne supérieure (manutentionnaires)	14,57	14,63	14,75
Borne inférieure (chauffeurs)	9,28	9,48	9,29
Borne supérieure (chauffeurs)	11,94	11,67	11,57
Nombre de demandes en retard	6,86	5,69	5,68
Nombre de demandes avec autonomie dépassée	4,98	4,87	5,57

TAB. 3.6 – Résultats des trois combinaisons

#### 3.5.3.3 Comparaison des deux méta-heuristiques

En comparant les tableaux 3.4 et 3.6, la recherche tabou trouve en moyenne des solutions cinq fois meilleures que l'algorithme génétique en un temps de résolution six fois plus petit. Elle permet aussi de trouver des solutions avec une personne en moins par rapport à l'algorithme génétique. La recherche tabou est donc incontestablement la meilleure méthode pour converger rapidement vers une très bonne solution. Néanmoins, l'algorithme génétique

permet de trouver  $N$  solutions différentes alors que la recherche tabou n'en trouve qu'une. Ce point n'est pas à négliger puisque l'objectif de l'étude du projet est aussi de tester par la suite des solutions différentes trouvées par les algorithmes avec un simulateur pour étudier leurs robustesses dans un environnement aléatoire (quantité de chariots variables, distance en temps non constant, etc.). L'une des perspectives de ce travail a donc été d'hybrider ces deux méthodes de manière à obtenir plusieurs bonnes solutions différentes. De plus, contrairement à l'algorithme pour le problème de transports de patients, nous ne sommes pas contraints par le temps de résolution. Une autre perspective est donc d'augmenter le nombre d'itérations sans amélioration pour les deux méthodes et d'observer laquelle des deux méthodes continu à converger vers de meilleures solutions. Enfin, dans cette première modélisation, toutes les contraintes ne sont pas prises en compte, ces perspectives ont donc été réalisées pour le modèle appliqué.

### 3.6 Conclusion

Pour cette première étude de la logistique, le problème a été abordé de manière théorique sans prendre en compte les contraintes de temps de préparation ou production des chariots ni de distinction des types de quais. Un modèle de PLNE a été présenté mais il ne permet pas de résoudre des instances réelles, étant donné la complexité importante du problème. Deux autres méthodes de résolution se basant sur le même codage d'une solution ont été développées : un algorithme génétique et une recherche tabou. Des résultats expérimentaux préliminaires sur ces deux méthodes ont montré que la recherche tabou est plus performante que l'algorithme génétique.

Suite à cette première étude outre le fait d'intégrer des contraintes représentant plus finement la réalité, un ensemble de perspectives ont été établies. Une des premières perspectives est de tester une hybridation des deux méthodes. Une seconde est d'approfondir les tests en augmentant le nombre d'itérations sans amélioration. Enfin, une dernière amélioration possible concerne l'opérateur de voisinage de l'algorithme tabou, afin de prendre en compte les dépassements des autonomies des chariots dans le choix de la demande.

Un certain nombre d'hypothèses posées au tout début de cette étude exploratoire sont à revoir. En premier lieu, l'hôpital Bretonneau subit d'importants travaux de transformation (destruction de certains bâtiments et construction de nouveaux). Ce n'est qu'à la fin de ces travaux que l'on pourra effectivement s'appuyer sur le quai unique. Pendant cette période transitoire, d'autres quais, liés à des bâtiments isolés doivent être utilisés par les véhicules. D'autre part les manutentionnaires, à terme, ne pourront utiliser que les fenwickss souterrains ou se déplacer à pied dans les bâtiments (il n'y aura pas de déplacements en extérieur). Actuellement, les collectes de linge sale et de déchets sont assurées par une équipe de salubrité déjà existante et le CHRU ne souhaite pas désorganiser cette équipe, les manutentionnaires ne s'occuperont donc pas de ces flux, par contre, les véhicules devront prendre en charge le linge sale. Ces considérations, ainsi que quelques autres présentées dans le chapitre suivant nous ont amené à enrichir la modélisation et à proposer un modèle plus appliqué.