

# Optimisation de tournées de service réactives en temps réel

Sixtine Binart, Pierre Dejax, Michel Gendreau, Frédéric Semet

► **To cite this version:**

Sixtine Binart, Pierre Dejax, Michel Gendreau, Frédéric Semet. Optimisation de tournées de service réactives en temps réel. Anne Etien. 9ème édition de la conférence MANifestation des JEunes Chercheurs en Sciences et Technologies de l'Information et de la Communication - MajecSTIC 2012 (2012), Oct 2012, Villeneuve d'Ascq, France. 2012. <hal-00780231>

**HAL Id: hal-00780231**

**<https://hal.inria.fr/hal-00780231>**

Submitted on 23 Jan 2013

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

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

## Optimisation de tournées de service réactives en temps réel

Sixtine Binart<sup>1,3</sup>, Pierre Dejax<sup>2</sup>, Michel Gendreau<sup>3</sup>, Frédéric Semet<sup>4</sup>

1 : LAGIS, Université de Lille 1, Cité Scientifique, 59650 Villeneuve d'Ascq - France.

2 : IRCCyN, Ecole des Mines de Nantes, 4 rue Alfred Kastler, 44300 Nantes.

3 : CIRRELT, Ecole Polytechnique de Montréal, 2920 Chemin de la Tour, Montréal (QC) H3T 1J4 - Canada.

4 : LAGIS, Ecole Centrale de Lille, Cité Scientifique, 59650 Villeneuve d'Ascq - France.

Contact : [sixtine.binart@ed.univ-lille1.fr](mailto:sixtine.binart@ed.univ-lille1.fr)

---

### Résumé

On considère ici un problème de tournées de service dans lequel on distingue deux types de clients : les clients urgents (chez qui on doit intervenir suite à une panne) et des clients non urgents chez qui une opération de contrôle est planifiée (il peut s'agir d'un relevé de compteur, d'une opération de maintenance...). Ainsi, on dispose de clients que l'on doit visiter (les clients urgents) et de clients optionnels (les clients non urgents). Dans ce problème, on considère que les temps de parcours et de service sont stochastiques (ils sont soumis à des aléas), que les clients ont des rendez-vous (ou fenêtres de temps), et que l'on dispose de plusieurs dépôts (par exemple, les domiciles des techniciens). Pour traiter ce problème, on propose une méthode de résolution en trois étapes : tout d'abord, la construction du squelette des clients urgents (ensemble de routes allant desservir uniquement les clients urgents). Ensuite, l'insertion des clients non urgents dans ce squelette. Et enfin, une étape de programmation dynamique permettant de réagir en temps réel face aux aléas sur les temps de service et de parcours.

**Mots-clés :** tournées de service, gestion de flotte, temps réel

### 1. Introduction

A l'heure actuelle, les entreprises sont de plus en plus nombreuses à fournir des services à domicile à leur clientèle. Souvent, ces sociétés sont donc amenées à planifier des tournées de service pour aller effectuer des opérations d'installation, de contrôle, de maintenance, ou même de réparation. De surcroît, dans un contexte de compétitivité croissante, elles sont obligées de toujours chercher à optimiser leurs solutions afin de pouvoir proposer un service à la clientèle encore meilleur. Avec les technologies de géopositionnement et de communication, il est possible aujourd'hui de connaître à chaque instant la position de sa flotte de véhicules. Etant donné que les temps de parcours et de service sont soumis à de nombreux aléas (météorologie, pannes, accidents...), le fait de connaître la position exacte des véhicules à chaque instant permet de réagir en temps réel suite à de tels aléas et ainsi d'augmenter la qualité du service fourni aux clients. Il devient donc important pour les entreprises de service de planifier ces tournées en temps réel.

C'est pour ces raisons que nous avons choisi de nous intéresser au problème de tournées de service (ou tournées de techniciens) avec temps de parcours et temps de service stochastiques. Nous présenterons dans un premier temps une description détaillée de la problématique qui nous intéresse. Ensuite, nous donnerons un aperçu de la littérature ayant trait à des problèmes s'en rapprochant. Enfin, nous détaillerons la méthode de résolution proposée ainsi que quelques résultats préliminaires.

### 2. Description de la problématique

Avant de décrire notre problème, rappelons la définition du problème de tournées de véhicules (VRP) ainsi que celle du problème de tournées de véhicules avec fenêtres de temps (VRPTW).

Le VRP peut être formulé comme suit. Etant donné un dépôt, des véhicules de capacité limitée et des demandes clients, trouver des routes pour satisfaire les demandes, chacune de ces routes ayant pour origine et destination le dépôt donné. L'objectif est de minimiser la distance parcourue et parfois le nombre de véhicules utilisés.

Le VRPTW est une extension du VRP dans laquelle chaque client dispose d'une fenêtre de temps dans laquelle son service doit commencer.

Le problème auquel nous nous intéressons ici est une variante du problème de tournées de véhicules avec fenêtres de temps (VRPTW). On rajoute à ce problème plusieurs notions qui le rendent plus réaliste :

– *Priorité entre les clients*

Dans ce problème, on distingue deux types de clients : les clients urgents et les clients non urgents.

Les clients non urgents sont des clients chez qui une opération de service (contrôle, relevé de compteur, maintenance...) est prévue. Ils sont donc connus a priori. Si l'opération de service nécessite la présence du client, alors elle sera planifiée et une fenêtre de temps sera associée à celui-ci. Dans le cas contraire, la fenêtre de temps associée au client sera égale à la longueur d'une journée. En tout temps, on autorise la replanification d'un client non urgent (la période de validité d'un client non urgent correspond à l'horizon multi-périodes considéré).

Les clients urgents sont des clients chez qui une opération de type réparation est prévue. Lorsqu'un client urgent appelle pour signaler une panne, on lui attribue immédiatement une demi-journée au sein de laquelle son service aura lieu. Ils disposent donc d'une fenêtre de temps de quatre heures et d'une période de validité d'une journée. De plus, on suppose qu'un client urgent pourra être desservi au plus tôt le lendemain du jour de son appel (s'il appelle le jour J, il ne pourra être servi avant le jour J+1). Ainsi, en faisant cette hypothèse, les clients urgents sont connus a priori lors de la planification journalière.

– *Fenêtres de temps dures*

Les fenêtres de temps du dépôt (durée d'une journée de travail) et des clients urgents sont dures. Ce qui signifie qu'on n'autorise aucune heure supplémentaire ni aucun retard chez les clients urgents.

– *Aspect multi-dépôts*

On considère un problème de tournées de service au sein duquel des techniciens vont effectuer des opérations de service chez des clients. On peut donc supposer que ces premiers ne partent pas d'un dépôt central mais de leur domicile. Ainsi, chaque technicien a son propre « dépôt ».

– *Flotte de véhicules*

Contrairement au problème de tournées de véhicules avec fenêtres de temps, on suppose ici que le nombre de véhicules (ou de techniciens) est fixé et qu'on les utilise tous.

– *Capacité infinie*

Etant donné qu'il s'agit de tournées de service, les véhicules ne transportent que les techniciens et le matériel nécessaire à leurs interventions. On peut donc supposer que les véhicules ont la capacité nécessaire au transport de ces personnes et de ce matériel, ce qui revient à supposer que leur capacité est infinie.

– *Temps de transport et de service stochastiques*

Souvent, dans les problèmes de tournées de véhicules, des temps de transport et de service déterministes sont considérés. Cela ne reflète pourtant pas la réalité : les temps de transport sont soumis à des aléas tels que la météorologie, le trafic ainsi que les accidents. De même, comme il s'agit ici de tournées de service, les temps de service ne sont pas négligeables. Aussi, ces derniers sont soumis à des aléas (le client n'était pas là à l'heure prévue, la réparation prend plus de temps que prévu). Ces variations ne sont pas neutres. On a donc choisi de prendre en compte des temps de transport et de service stochastiques.

### 3. Revue de la littérature

Le problème de tournées de véhicules a fait l'objet de nombreux articles depuis la fin des années 50. Plus récemment, l'aspect temps réel du problème a aussi retenu l'attention de plusieurs chercheurs dès 1998 (soit 40 ans après). Depuis, de nombreuses variantes de ce problème en temps réel ont été proposées. Ici, nous nous intéressons tout particulièrement aux variantes du VRPTW présentant des caractéristiques communes avec notre problème.

Même si le VRPTW est un problème qui a fait l'objet de nombreux articles, peu sont ceux qui prennent en compte à la fois des temps de parcours et de service stochastiques. Pour résoudre ce problème, Wang *et al.* [1] formulent le problème comme un modèle d'affectation tandis que Li *et al.* [2] présentent une heuristique basée sur la recherche taboue. En 2007, Zeimpekis *et al.* [3] ajoutent à ce problème une notion de priorité entre les clients en se limitant toutefois à un seul véhicule. Dans ce contexte, ils proposent une méthode itérative consistant, à chaque étape, à évaluer la solution courante puis à insérer le meilleur client possible sur la route jusqu'à ce que tous les clients soient insérés.

Indépendamment de l'aspect stochastique, de nombreux auteurs se sont intéressés au problème de tournées de service avec fenêtres de temps, plusieurs dépôts et une priorité entre les clients (comprenant deux types de clients : les clients urgents et les non urgents) [4], [5] et [6]. Pour résoudre ce problème, ils proposent une génération de colonnes ainsi qu'un algorithme mémétique. En 2006, Dugardin *et al.* [7] s'intéresse au même problème mais propose de prendre en compte des aléas sur les temps de parcours. Toutefois, il ne les prend pas en compte lors de la planification. Il construit un planning de tournées et établit des politiques simples pour réagir aux différents aléas pouvant surgir. Enfin, plus récemment, Borenstein *et al.* [8] et Delage [9] proposent cette fois de prendre en compte des temps de service stochastiques. Pour résoudre ce problème, Borenstein *et al.* procèdent en plusieurs étapes : ils partitionnent d'abord les clients en plusieurs groupes, puis assignent les différents techniciens aux zones ainsi créées. Enfin, ils rendent les frontières de ces zones floues et attribuent les tâches des zones floues à toutes les zones limitrophes. Delage, quant à lui, propose une méthode en deux étapes : il établit dans un premier temps un planning de tournées puis utilise la programmation dynamique pour faire face aux aléas sur les temps de service. Cortes *et al.* [10], quant à eux, s'intéressent au VRPTW multi-période avec priorité entre les clients.

Personne ne s'est encore intéressé jusqu'ici au problème de tournées de service avec fenêtres de temps, plusieurs dépôts, une priorité entre les clients et, de surcroît, des temps de service et de parcours stochastiques.

### 4. Méthode de résolution du problème

Rappelons que le problème auquel nous nous intéressons ici est un problème de type VRPTW sans capacité, dans lequel on distingue deux types de clients : les clients urgents (avec fenêtres de temps) et les clients non urgents (avec période de validité de plusieurs jours). De plus, on considère que les temps de service et de parcours sont stochastiques.

Comme nous l'avons vu précédemment, on connaît tous les clients (urgents ou non) a priori. Les temps de service et de parcours sont stochastiques mais bornés. On suppose que l'on connaît les temps de service maximaux et minimaux et les temps de parcours minimaux et maximaux. Dans ce contexte, nous proposons une méthode en trois étapes, inspirée de Delage [9] : l'établissement du squelette des clients prioritaires, l'insertion des clients non urgents dans le squelette précédemment établi et enfin la programmation dynamique pour réagir face aux aléas sur les temps de service et de parcours. Cette méthode de résolution n'a de sens que si le nombre de clients urgents est non négligeable (sinon, l'établissement du squelette ne sert à rien).

#### 4.1. Établissement du squelette

Dans cette étape, on ne considère que les clients urgents, connus a priori. Aussi, nous avons fait l'hypothèse dans notre problème qu'à chaque client urgent est associée une fenêtre de temps correspondant à une demi-journée de travail. Pour établir le squelette des clients prioritaires, il nous suffit donc de résoudre un m-TSPTW (ou VRPTW sans capacité) sur les clients prioritaires. Étant donné que l'on ne s'autorise aucun retard chez les clients urgents, on considère les temps

de service et de parcours maximaux (on se place dans le pire des cas). On utilisera les notations suivantes :

– Ensembles

- U l'ensemble des clients urgents (ou prioritaires)
- K l'ensemble des véhicules

– Paramètres

- $[e_i, l_i]$  fenêtre de temps du client  $i$  (le service doit débuter après  $e_i$  et avant  $l_i$ )
- $\bar{\tau}_{ij}$  temps de parcours maximum entre  $i$  et  $j$
- $\bar{\sigma}_i$  durée de service maximum chez le client  $i$

– Variables

- $x_{ik}$  booléen indiquant si le client urgent  $i$  est servi par le véhicule  $k$
- $y_{ijk}$  booléen indiquant si le client  $i$  est servi juste avant le client  $j$  par le véhicule  $k$
- $t_i$  heure de début de service du client urgent  $i$

Avec les notations ci-dessus, on peut alors formuler le problème comme suit :

$$\min . \sum_{i \in U} \sum_{j \in U} \bar{\tau}_{ij} \sum_{k \in K} y_{ijk}$$

sous contraintes :

$$\sum_{k \in K} x_{ik} = 1 \quad \forall i \in U \quad (1)$$

$$\sum_{j \in U} y_{ijk} = x_{ik} \quad \forall i \in U, k \in K \quad (2)$$

$$\sum_{i \in U} y_{ijk} = x_{jk} \quad \forall j \in U, k \in K \quad (3)$$

$$e_i \leq t_i \leq l_i \quad \forall i \in U \quad (4)$$

$$t_j \geq t_i + \bar{\sigma}_i + \bar{\tau}_{ij} + \sum_{k \in K} (y_{ijk} - 1)M \quad \forall i \in U, j \in U \quad (5)$$

$$x_{ik}, y_{ijk} \in \{0; 1\} \quad \forall i \in U, j \in U, k \in K \quad (6)$$

Les contraintes (1) garantissent que tous les clients urgents soient servis une et une seule fois. Les contraintes (2) et (3) sont les contraintes de tournée. Les contraintes (4) correspondent au respect des fenêtres de temps (le service chez le client  $i$  doit commencer entre  $e_i$  et  $l_i$ ). Enfin, les contraintes (5) sont des contraintes de précédence.

#### 4.2. Insertion des clients non urgents

Une fois le squelette de clients prioritaires établi, on dispose pour chaque véhicule d'une liste ordonnée de clients prioritaires à aller desservir. En se basant sur ce squelette, on redéfinit dans un premier temps les fenêtres de temps  $[e'_i, l'_i]$  associées aux clients urgents. Le début  $e'_i$  et la fin  $l'_i$  de la nouvelle fenêtre de temps correspondent à l'heure de début de service du client respectivement dans le meilleur et le pire des cas. On pose donc  $l'_i = t_i$  et on se base sur le squelette pour calculer les  $e'_i$  avec les temps de service et de parcours minimaux (en s'assurant que  $e'_i \geq e_i$ ).

Pour la suite de la résolution, on définit un nouveau concept :

– une phalange

Il s'agit de la portion de route entre deux clients prioritaires successifs du squelette (le dépôt de départ et le dépôt d'arrivée de chacun des véhicules sont considérés comme des clients prioritaires). Elle est définie par un client prioritaire d'origine  $o^p$  et un client prioritaire de destination  $d^p$  et par une longueur  $\Delta^p$  définie ci-dessous.

La longueur  $\Delta^p$  d'une phalange est donnée par la formule  $\Delta^p = l'_{d^p} - e'_{o^p} - \underline{\sigma}_{o^p}$  où  $l'_{d^p}$  est la fin de la fenêtre de temps au client prioritaire destination  $d^p$ ,  $\underline{\sigma}_{o^p}$  le temps de service minimal chez le client prioritaire origine  $o^p$  et  $e'_{o^p}$  le début de la fenêtre de temps du client  $o^p$ .

Dans cette étape, l'objectif est double : il s'agit d'abord de maximiser le profit associé à la desserte des clients non urgents insérés puis, dans un deuxième temps de minimiser les temps de parcours. On notera :

– Ensembles

- NU l'ensemble des clients non urgents
- P l'ensemble des phalanges sur tous les véhicules. Les phalanges sont ordonnées comme suit : la phalange  $p + 1$  a pour origine  $d^p$  (le client prioritaire destination de la phalange  $p$ ).

– Paramètres

- $\alpha$  pondération des temps de parcours dans l'objectif
- $\Delta_p$  longueur de la phalange  $p$
- $p_i$  le profit associé à la desserte du client  $i$
- $\tau_{ij}$  temps de parcours minimum entre  $i$  et  $j$
- $\underline{\sigma}_i$  durée de service minimum chez le client  $i$

– Variables

- $x_{ip}$  booléen indiquant si le client non urgent  $i$  est servi sur la phalange  $p$
- $y_{ijp}$  booléen indiquant si le client  $i$  est servi juste avant le client  $j$  sur la phalange  $p$

Avec les notations ci-dessus, on peut alors formuler le problème comme suit :

$$\max. \sum_{p \in P} \sum_{i \in NU} p_i x_i^p - \alpha \sum_{p \in P} \sum_{i \in NU} \sum_{j \in NU} \tau_{ij} y_{ij}^p$$

sous contraintes :

$$\sum_{p \in P} x_i^p \leq 1 \quad \forall i \in NU \quad (7)$$

$$\sum_{j \in NU \cup \{d^p\}} y_{ij}^p = x_i^p \quad \forall i \in NU, p \in P \quad (8)$$

$$\sum_{i \in NU \cup \{o^p\}} y_{ij}^p = x_j^p \quad \forall j \in NU, p \in P \quad (9)$$

$$\sum_{i \in NU \cup \{o^p\}} \sum_{j \in NU \cup \{d^p\}} \tau_{ij} y_{ij}^p + \sum_{i \in NU} \underline{\sigma}_i x_i^p \leq \Delta^p \quad \forall p \in P \quad (10)$$

$$\sum_{i \in S} \sum_{j \in S} y_{ijp} \leq \sum_{i \in S \setminus \{l\}} x_{ip} \quad \forall S \subset NU, |S| \geq 2, \forall l \in S \quad (11)$$

$$y_{ij}^p \in \{0; 1\} \quad \forall i \in NU, j \in NU, p \in P \quad (12)$$

$$x_i^p \in \{0; 1\} \quad \forall i \in NU, p \in P \quad (13)$$

Les contraintes (7) assurent que chaque client non urgent est desservi au plus une fois. Les contraintes (8) et (9) sont des contraintes dites de tournée. La contrainte (10) force la durée de la route sur une phalange à être inférieure à la longueur de la phalange. Les contraintes (11) sont des contraintes d'élimination de sous-tours. Elles sont nécessaires ici car les clients non urgents n'ont pas de fenêtre de temps.

### 4.3. Politique déroulée par programmation dynamique

Maintenant, on dispose d'une route pour chaque véhicule contenant à la fois des clients urgents et des clients non urgents. Jusqu'à présent, on avait supposé que les temps de service et de par-

cours étaient soit à leur borne inférieure, soit à leur borne supérieure. Dans cette étape, on prend en compte les aléas sur les temps de service et de parcours et on adapte le planning en temps réel face à ces aléas. Ici, on considère un véhicule à la fois et on suppose que les phalanges de celui-ci sont ordonnées comme suit : la phalange  $p + 1$  a pour client prioritaire origine  $o^{p+1} = d^p$  (le client prioritaire destination de la phalange  $p$ ). Chaque étape de la programmation dynamique correspondra ici à la fin de service chez un client. A chaque étape, on dispose d'une liste de clients non urgents non visités pouvant être servis avant le prochain client prioritaire. On peut alors prendre deux décisions distinctes : aller directement au prochain client prioritaire ou bien aller au client différé de cette liste qui maximise le profit.

On définit les paramètres suivants :

- $\alpha_j$  pénalité de retard chez le client prioritaire  $j$
- $l_j$  la fin de la fenêtre de temps du client prioritaire  $j$
- $\tau_{ij}$  le temps de parcours du client  $i$  au client  $j$  ( $\underline{\tau}_{ij} \leq \tau_{ij} \leq \bar{\tau}_{ij}$ )
- $\sigma_i$  le temps de service du client  $i$  ( $\underline{\sigma}_i \leq \sigma_i \leq \bar{\sigma}_i$ )

A l'étape  $k$ , on définit :

- $v_k$  client où se trouve le technicien à l'étape  $k$
- $t_k$  heure de fin de service chez  $v_k$
- $V^p$  la liste ordonnée des clients non urgents associée à la phalange  $p$
- $\tilde{V}^p = V^p \cup o^p$  la liste des clients non urgents associée à la phalange  $p$  + le client prioritaire origine de la phalange  $p$
- $\tilde{V}_k^p$  la liste des clients non urgents de la phalange  $p$  pas encore visités à l'étape  $k$

On peut proposer deux types de programmation dynamique. Dans le premier algorithme, on ne considère qu'une phalange  $p$ . Dans le second, on considère le reste de la route. Le premier a pour fonction de revenu :

$$f(v_k, t_k, \tilde{V}_k^p) = \max(E[f(d^p, t_k + \tau_{v_k d^p}, \emptyset)], 1 + \max_{\bar{v} \in \tilde{V}_k^p} E[f(\bar{v}, t_k + \tau_{v_k \bar{v}} + \sigma_{\bar{v}}, \tilde{V}_k^p \setminus \{\bar{v}\})])$$

$$\text{avec } f(d^p, t, \emptyset) = -\alpha_{d^p} \max(t - l_{d^p}, 0)$$

Le second a une fonction de revenu légèrement différente, à savoir :

$$f(v_k, t_k, \tilde{V}_k^p) = \max(E[\hat{f}(d^p, t_k + \tau_{v_k d^p})], 1 + \max_{\bar{v} \in \tilde{V}_k^p} E[f(\bar{v}, t_k + \tau_{v_k \bar{v}}, \tilde{V}_k^p \setminus \{\bar{v}\})])$$

$$\text{avec } \hat{f}(d^p, t) = -\alpha_{d^p} \max(t - l_{d^p}, 0) + f(d^p, t + \sigma_{d^p}, V^{p+1})$$

## 5. Résultats préliminaires

### 5.1. Génération des instances

Pour tester la méthode, nous avons choisi d'utiliser les instances proposées par Tricoire [4] car elles correspondent au problème de tournées de service multi-dépôts, avec fenêtres de temps et priorité entre les clients dans un contexte multi-périodique.

Dans ses instances, Tricoire considère 3 techniciens et un horizon de 5 jours (soit une semaine de travail). Il propose de prendre en compte deux types de clients : des rendez-vous avec une fenêtre de temps de 2 ou 4 heures et une période de validité d'une journée (urgents), ainsi que des différables sans fenêtre de temps avec une période de validité de plusieurs jours (non urgents).

Comme, dans un premier temps, nous résolvons le problème sur un horizon de temps d'une journée, nous avons généré 5 instances journalières par instance originale. L'instance générée pour le jour  $j$  ( $j=1, \dots, 5$ ) de l'horizon correspond à l'instance originale dans laquelle on a gardé seulement les clients dont la période de validité comprend le jour  $j$ .

Toutefois, en procédant comme indiqué ci-dessus, nous nous retrouvons avec des instances dans lesquelles le pourcentage de clients urgents est très faible (moins de 10% du nombre de clients

total). Comme nous l'avons mentionné précédemment, il est crucial que la proportion de clients urgents soit non négligeable. Nous avons donc modifié les instances journalières générées ci-dessus afin d'assurer un pourcentage de clients urgents suffisant (25%).

Aussi avons-nous modifié les fenêtres de temps associées aux clients urgents afin de n'avoir que des fenêtres de temps de 4h. Nous avons donc attribué à la première moitié des clients urgents la matinée et au reste des clients urgents l'après-midi.

## 5.2. Contexte expérimental

Les deux premières étapes sont résolues de façon exacte sous Cplex 12.1 avec l'interface OSI. Jusqu'à présent, dans la dernière étape, les aléas sont fournis manuellement au programme. Dans un futur proche, un simulateur sera utilisé pour générer des temps de parcours et des temps de service aléatoires suivant la loi de distribution Gamma (cf. section 1).

La machine utilisée est une machine de type 4CPU, 3.2Ghz, 1Go de RAM.

Pour le réglage de la plupart des paramètres, nous nous sommes inspirés des valeurs utilisées par Tricoire. Ainsi, il proposait de prendre en compte une vitesse de 35km/h. Nous avons donc choisi de considérer une vitesse minimale de 20km/h et une vitesse maximale de 50km/h (pour garder une vitesse moyenne de 35km/h). En ce qui concerne les temps de service, nous avons considéré que les clients non urgents avaient des temps de service compris entre 15 et 30 minutes et les clients urgents entre 30 et 60 minutes. Tricoire considère qu'une journée de travail comprend 8 heures, soit 480 minutes. Les fenêtres de temps des dépôts seront donc  $[0; 480]$  et celles des clients urgents seront soit  $[0; 240]$ , soit  $[240; 480]$ . Enfin, pour régler le paramètre  $\alpha$  pour la pondération des temps de parcours dans la fonction objectif, nous avons regardé les résultats obtenus lors de l'insertion des non urgents dans le squelette. En choisissant  $\alpha = 0.01$ , on assure notre préférence à insérer des clients non urgents plutôt que de parcourir une moins grande distance.

## 5.3. Résultats obtenus

La méthode s'avère très efficace sur des instances contenant une vingtaine de clients (les temps de résolution ne dépassent pas quelques secondes). Voyons concrètement les résultats obtenus à chaque étape de la méthode de résolution sur une instance à 20 clients dont 5 urgents, à titre d'exemple :

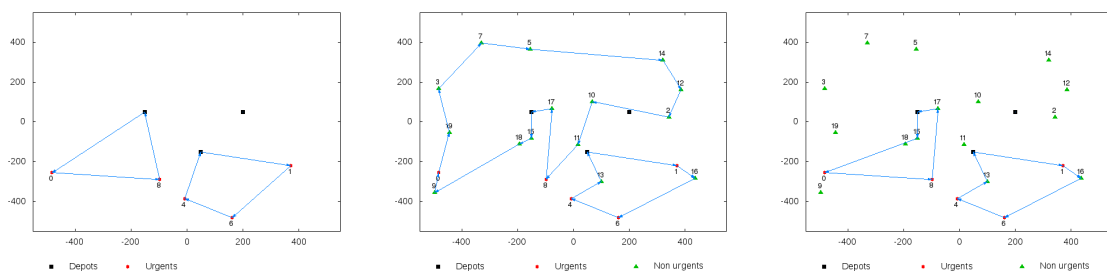


FIGURE 1 – Établissement du squelette

FIGURE 2 – Insertion des clients non urgents

FIGURE 3 – Programmation Dynamique

On constate qu'à la dernière étape, après prise en compte des aléas, le nombre de clients non desservis est très élevé alors même qu'un technicien est resté chez lui. Cette solution n'est pas acceptable. Rappelons que, dans la deuxième étape de résolution, on a supposé que les temps de service et de parcours étaient minimaux afin d'insérer le maximum de clients non urgents. On remarque qu'en faisant cette hypothèse, on a permis au technicien qui n'avait pas de client urgent à desservir après l'établissement du squelette de rester chez lui après l'insertion des non urgents.



Pendant ce temps, un autre technicien doit aller desservir 15 clients dans une même journée. Ce déséquilibre de charge de travail entre les techniciens est envisageable dans le meilleur des cas mais pas dans la réalité (avec des aléas sur les temps de service et de parcours). Il nous faut donc modifier la méthodologie d'insertion des clients non urgents dans le squelette afin de rééquilibrer les tournées.

Ci-dessous un récapitulatif des résultats obtenus par cette méthode.

Instances	Clients non desservis	Temps de calcul		Clients non desservis	Temps de calcul
C4_10	1	2s	C4_33	7	15s
C4_11	5	20s	C4_34	7	16s
C4_12	4	2s	C4_40	4	2s
C4_13	4	2s	C4_41	6	9s
C4_14	7	6s	C4_42	7	4s
C4_20	2	18min48s	C4_43	6	2s
C4_21	4	4min2s	C4_44	7	4s
C4_22	6	5min4s	C4_50	5	1min31s
C4_23	6	5min13s	C4_51	6	24s
C4_24	4	13min47s	C4_52	6	3s
C4_30	9	6s	C4_53	7	6s
C4_31	9	8s	C4_54	9	37s
C4_32	6	7s			

On remarque dans les résultats ci-dessus que, sur l'ensemble des instances de type C4, le nombre moyen de clients non desservis est 7. Or, quand on regarde les routes obtenues graphiquement, on voit clairement qu'on aurait pu obtenir de meilleurs résultats. Il faut donc modifier la méthode d'insertion des non urgents dans le squelette.

#### 5.4. Amélioration de la méthode

Nous avons pu voir ci-dessus que, lors de l'insertion des non urgents, en prenant en compte des temps de service et de parcours minimaux, on peut se retrouver avec des tournées très déséquilibrées. Pour remédier à ce problème, on propose une insertion des non urgents en trois étapes :

1. On résout le problème en considérant les temps de service et de parcours maximaux (forçant ainsi l'équilibrage des tournées).
2. On considère des temps de service et de parcours moyens et on réoptimise en conservant l'affectation des clients non urgents aux véhicules obtenue en 1.
3. On considère des temps de service et de parcours minimaux et on réoptimise en conservant l'affectation des clients non urgents aux véhicules obtenue en 2.

Cette méthodologie est en cours d'expérimentation et des résultats devraient être obtenus sous peu.

## 6. Conclusion

Comme nous avons pu le voir, le problème que nous proposons de traiter est original. En effet, le problème de VRPTW avec priorité entre les clients et temps de parcours et de service stochastiques n'a encore fait l'objet d'aucun article à ce jour. La méthodologie en trois étapes que nous proposons (établissement du squelette des clients urgents, insertion des clients non urgents, et programmation dynamique) mérite encore quelques modifications mais semble prometteuse. Toutefois, il reste encore du travail, à savoir l'amélioration de notre méthode de résolution, la prise en compte des temps de service et de parcours stochastiques.

**Bibliographie**

1. Xiubin Wang and Amelia C. Regan. Assignment models for local truckload trucking problems with stochastic service times and time window constraints. *Transportation Network Modeling*, 1771 :61–68, 2001.
2. Xiangyong Li, Peng Tian, and Stephen C.H. Leung. Vehicle routing problems with time windows and stochastic travel and service times : Models and algorithm. *International Journal of Production Economics*, 125(1) :137–145, May 2010.
3. V Zeimpekis, I Minis, K Mamassis, and G M Giaglis. Dynamic management of a delayed delivery vehicle in a city logistics environment. In *Dynamic Fleet Management*, chapter 9, pages 197–217. Springer edition, 2007.
4. Fabien Tricoire. *Optimisation des tournées de véhicules et de personnels de maintenance : application à la distribution et au traitement des eaux*. PhD thesis, Ecole des Mines de Nantes, 2006.
5. Nathalie Bostel, Pierre Dejax, and Pierre Guez. Multiperiod planning and routing on a rolling horizon for field force optimization logistics. *Routing Problem : Latest Advances*, 2008.
6. Fabien Tricoire, Nathalie Bostel, Pierre Dejax, and Pierre Guez. Exact and hybrid methods for the multiperiod field service routing problem. *Central European Journal of Operations Research*, Online Fir, September 2011.
7. Frédéric Dugardin. Optimisation réactive de tournées de service en environnement dynamique. Technical report, Ecole des Mines de Nantes, 2006.
8. Yossi Borenstein, Nazaraf Shah, Edward Tsang, Raphael Dorne, Abdullah Alsheddy, and Christos Voudouris. On the partitioning of dynamic workforce scheduling problems. *Journal of Scheduling*, 13(4) :411–425, December 2009.
9. Erwann Delage. Re-optimization of technician tours in dynamic environments with stochastic service time. Technical report, Ecole des Mines de Nantes, 2010.
10. C E Cortés, M Gendreau, D Leng, and A Weintraub. A simulation-based approach for fleet design in a technician dispatch problem with stochastic demand. *Journal of the Operational Research Society*, pages 1–14, August 2010.