
Transport à la demande points à points en zone peu dense.

Proposition d'une méthode d'optimisation de tournées

Garaix Thierry^{*/}, Josselin Didier^{**}, Feillet Dominique^{*}, Artigues Christian^{*}, Castex Élodie^{**}**

** Laboratoire d'Informatique d'Avignon*

*** UMR ESPACE 6012 CNRS, Université d'Avignon et des Pays du Vaucluse*

RÉSUMÉ. Il s'agit dans cet article d'innover dans les transports à la demande (TAD) en développant une méthode efficace et robuste d'optimisation de tournées en zone rurale peu dense (Pays du Doubs Central, Franche-Comté). Après avoir replacé les TAD parmi les grands problèmes d'optimisation de tournées dans les transports, nous présentons le problème du TAD du Doubs Central, en précisant les fonctionnalités du service définies par et pour les collectivités et le formalisme mathématique qui en découle. Puis nous proposons et évaluons une méthode d'optimisation de tournées dédiée au problème étudié, et concluons sur les résultats et les perspectives possibles d'amélioration de la méthode.

ABSTRACT. This paper aims to develop innovative and efficient demand responsive transport in rural and low densities areas (Pays du Doubs Central, Franche-Comté, France). After having depicted the state of art of the main optimization methods for transport, we define precisely the parameters assigned to the TAD in the Doubs Central. We set the mathematical formalism to describe the problem. We then propose and evaluate a specific optimization problem able to find quickly rather good paths, according to the territorial and functional constraints. We conclude on discussing the results and opening on further improving works

MOTS-CLÉS: Transport à la Demande, TAD, Recherche Opérationnelle, optimisation de tournées, Doubs Central, zones rurales

KEYWORDS: Demand Responsive Transport, DRT, Operations Research, Vehicle Routing, Dial a Ride, Pickup and Delivery

1. Introduction

Les individus recherchent des services de transport toujours plus souples, plus proches de leurs besoins. Malgré de récents efforts, les transports publics ne répondent que partiellement à ces attentes. Si ceux-ci restent compétitifs sur certains segments (en milieu urbain dense sur sites propres), dans la majorité des cas la voiture reste la plus à même de répondre aux besoins de déplacements individuels ([WIE 02], [DUP 99]). Toutefois, sur ces segments où les véhicules privés sont les plus compétitifs (zones périurbaines, rurales...), de par leur disponibilité et leur efficacité, des solutions alternatives existent.

Relativement mal connus du grand public et plutôt négligés par les transporteurs, les transports à la demande (TAD), datant des années 1970, font leur réapparition depuis la fin des années 1990, à la faveur de l'avènement des technologies de l'information et de la communication ([AMB 04]). Les TAD peuvent se définir comme : des « transports terrestres collectifs individualisés de personnes, activés seulement à la demande ». En France, la LOTI les définit juridiquement comme « des services collectifs offerts à la place, déterminés en fonction de la demande des usagers, dont les règles générales de tarification sont établies à l'avance et qui sont exécutés avec des véhicules dont la capacité minimale est fixée par décret ».

En France, plus de 600 services de TAD existent, couvrant plus de 6000 communes. Ces services sont souvent développés pour maîtriser les dépenses de transport (remplacement des lignes de bus déficitaires), assurer le droit à la mobilité à certaines populations, captives, défavorisées (Personnes à Mobilité Réduite, par exemple) ou proposer un mode de déplacement adapté aux zones de faibles densités, aux heures de frange de la journée (matinées, soirées, nuits) ou aux événements localisés, occasionnels ou récurrents. Un dernier argument est celui de la lutte contre la pollution, puisque l'optimisation des tournées et le regroupement des voyageurs permettent en effet une économie potentiellement importante de véhicules roulant et donc d'émission de polluants atmosphériques ([HOU 04]).

Dans son fonctionnement, un TAD permet de répondre à des demandes (ou « requêtes ») de transport via une flotte de véhicules sous certaines contraintes de faisabilité et de fonctionnement. Les TAD posèrent immédiatement des problèmes d'optimisation de construction des tournées et d'aide à la décision nouveaux par rapport aux problèmes de transport classiques où la planification des tournées n'est pas quotidienne. Les enjeux de tels systèmes consistent à trouver des solutions optimisées de fonctionnement associant les objectifs, parfois contradictoires, des partenaires impliqués :

- rationaliser l'offre et rendre visible et attractif le service pour les Autorités Organisatrices de Transport (AOT, prescripteurs relayant la demande sociale via les élus),
- rentabiliser le service pour les transporteurs (sociétés auxquelles les AOT délèguent l'organisation des transports sur leur territoire),
- conquérir de nouveaux marchés pour les sous-traitants éventuels (taxis),

– améliorer le cadre de vie et l'accès aux facilités (associations d'usagers).

Ces objectifs impliquent la prise en compte de critères économiques (coût de fonctionnement), sociaux (accessibilité, équité, coûts attractifs, convivialité), fonctionnels (réactivité, flexibilité, niveau de remplissage des véhicules), environnementaux (pollution, fluidité du trafic urbain, nuisances sonores et encombrement des parkings en zones urbaines).

Lorsqu'un système de transport ne peut plus être géré manuellement, notamment lorsque la quantité de demandes de mobilité autorise potentiellement le regroupement des passagers dans les véhicules, l'utilisation de systèmes automatisés pour la recherche de solutions optimisées s'impose. Ces systèmes permettent de gérer des services au fonctionnement complexe soumis à de nombreuses contraintes. La recherche opérationnelle¹ apporte ainsi une vision complémentaire à celle de l'expert opérationnel, grâce à l'introduction de théories et de concepts éprouvés tels que la recherche du plus court chemin basée sur la théorie des graphes, les méthodes d'ordonnancement ou de programmation linéaire sous contraintes. Ces approches peuvent intervenir pour aider au choix des services (simulation préalable de fonctionnement de système), pour gérer le TAD (optimisation des tournées effectives) ou pour suivre son fonctionnement (audit et retour sur expérience). Le traité IGAT rédigé sous la direction de G. Finke [FIN 02] détaille ces différents aspects.

Notre article s'inscrit dans ce contexte. Il s'agit d'innover dans les transports flexibles à la faveur d'une demande sociale émanant du Pays du Doubs Central (Franche-Comté), en faisant appel aux compétences scientifiques de géographes et d'informaticiens. Après avoir replacé les TAD parmi les grands problèmes d'optimisation de tournées dans les transports, nous présentons le problème du TAD du Doubs Central, en précisant les fonctionnalités du service définies par et pour les collectivités et le formalisme mathématique qui en découle. Puis nous proposons et évaluons une méthode d'optimisation de tournées dédiée au problème étudié, et concluons sur les résultats et les perspectives possibles d'amélioration de la méthode.

2. État de l'art

2.1. Les TAD parmi les problèmes de transport

Le calcul de tournées doit permettre, étant donné un ensemble de services à accomplir en différents lieux et une flotte de véhicules, de déterminer un ensemble optimisé (suivant certains critères) d'itinéraires de ces véhicules permettant de répondre aux différentes demandes de transport. La nature des services à accomplir conditionne fortement la structure des problèmes. On distingue ainsi des sous-familles de problèmes de calcul de tournées. L'état de l'art sur les problèmes de calcul de tournées de véhicules en recherche opérationnelle est le sujet d'un livre de Toth et Vigo [TOT 01].

1. science mariant mathématiques et informatique permettant de rationaliser les systèmes opérationnels.

Le problème général de calcul de tournées des véhicules est connu sous le nom de *Vehicle Routing Problem (VRP)* dont le représentant le plus célèbre est le *Traveling Salesman Problem*² (**TSP**). Pour le **VRP**, il s'agit d'aller visiter à l'aide de véhicules (un seul pour le **TSP**) différents lieux (pour la livraison de marchandises par exemple) en minimisant le coût global des trajets. Le **PDP** (*Pickup & Delivery Problem* ou *Ramassage & Livraison*) ajoute une contrainte, puisqu'il s'agit de transporter des marchandises des points d'embarquement vers des points de débarquement. Cela implique des contraintes de priorité (l'embarquement avant le débarquement) et de couplage (un même véhicule pour l'embarquement et le débarquement). Un **DARP** (*Dial-a-Ride Problem*) est une extension du **PDP** où les marchandises sont des personnes, d'où certaines contraintes techniques supplémentaires dues au transport de personnes comme par exemple, le fait d'avoir un seul point d'embarquement par point de débarquement et vice versa, et l'obligation de respecter des délais particuliers. Les transports à la demande (**TAD**) ou Demand Responsive Transport (**DRT**) au sens *transport en commun routier*, est un cas particulier de **DARP** où la qualité du service peut être mesurée en termes de confort, convivialité, ergonomie, formes des trajets par rapport aux origines et aux destinations, prise en compte des demandes en temps réel, etc.

2.2. Caractéristiques discriminantes

Trois paramètres transversaux aux types d'applications mises en place apparaissent discriminants dans les problèmes de transport.

2.2.1. Les problèmes à un ou plusieurs véhicules

Lorsque les requêtes peuvent être servies par différents véhicules, le DARP est multi-véhicules. Les noms des problèmes à multiples véhicules peuvent être préfixés d'un *m-*. Par exemple pour le *Pickup & Delivery Problem*, on utilise les notations 1-PDP et *m-PDP*. Se pose alors le problème de l'attribution des requêtes aux différents véhicules. L'optimisation de ce type de système se complexifie fortement lorsque les véhicules peuvent coopérer pour prendre en compte les demandes de transport, en effectuant chacun une partie du transport du passager par exemple.

2.2.2. Les problèmes avec ou sans fenêtres de temps

A certains services, peuvent être associés des dates de réalisation précises. On en déduit des contraintes horaires appelées fenêtres de temps associant une date au plus tôt et une date au plus tard de service. L'abréviation courante pour spécifier la présence de fenêtres de temps consiste à utiliser le suffixe **-TW**. On parle ainsi de *Pickup and Delivery Vehicle Routing Problem with Time Windows (PDVRPTW)*, de PDPTW ou de VRPTW. Ces contraintes induisent un ordre partiel parmi les points de passage dans les tournées et tendent à réduire le nombre de solutions réalisables, donnant lieu à des modélisations et des méthodes de résolution spécifiques.

2. problème du voyageur de commerce.

2.2.3. *Les problèmes statiques ou dynamiques*

Un problème est dit statique si toutes les données sont connues sur l'horizon de temps où l'on doit calculer les tournées, c'est-à-dire de façon définitive préalablement au début du calcul des tournées de véhicules. Dans le cas dynamique, le système doit être capable d'intégrer des imprévus alors que des tournées ont déjà été planifiées et que leur exploitation a démarré. Ces imprévus peuvent être la panne ou le retard d'un véhicule, l'apparition ou la disparition d'une requête. ... Généralement, un premier algorithme calcule des tournées optimisées à partir des données connues. Un algorithme, généralement différent du premier, permet de prendre en compte les nouvelles données en respectant certains critères d'optimalité. Ceci se produit dans des applications à « temps réel », d'où une contrainte supplémentaire de rapidité d'exécution pour l'algorithme dynamique.

3. Transport à la demande à mettre en place dans le pays du Doubs Central

3.1. *Description générale du TAD à réaliser*

Le Pays du Doubs Central, regroupant quelques 25000 habitants répartis dans 99 communes dont 5 bourgs et associant 4 Communautés de Communes, se propose de mettre en place un TAD. La carte de la figure 1 montre la répartition des densités de population et la structure des flux domicile-travail alternants dans le pays. L'objectif, dans ce territoire rural à faible densité de population, consiste à regrouper des voyageurs dans des véhicules parcourant des trajets optimisés en fonction des demandes. L'équipe pluridisciplinaire, comprenant des géographes, des informaticiens, des ingénieurs en transport et des élus locaux souhaite activer deux leviers complémentaires pour favoriser le regroupement :

- par des réductions de coût octroyées aux clients qui souhaitent voyager ensemble, quitte à leur faire supporter quelques détours et pertes de temps raisonnables,
- par l'utilisation de méthodes d'optimisation de tournées visant à minimiser les coûts et, par conséquent, réduire les détours et les pertes de temps pour les passagers.

Pour le Doubs Central, les tournées des véhicules (horaires de passage et parcours) sont définies à partir des demandes de transport des usagers. Les véhicules utilisés (en sous-traitance) sont ceux des compagnies de taxis opérant sur le territoire du Doubs Central, soit une dizaine de véhicules potentiellement. Ce service étant fortement subventionné par les collectivités, son coût de fonctionnement doit être minimal. C'est avec cet objectif que nous nous proposons de calculer, une fois les demandes de transport recensées, des itinéraires de taxis optimisés en terme de coût pour la collectivité.

Les tournées des taxis sont calculées jour après jour, pour une journée entière. Un système centralisé de réservation accueille jusqu'à la veille au soir les requêtes des utilisateurs. Chaque usager propose un horaire de prise en charge ou d'arrivée plus ou moins précis, par tranche horaire. En fonction des détours maximaux possibles, on lui indique à la réservation l'horaire théorique d'arrivée ou de prise en charge le cas

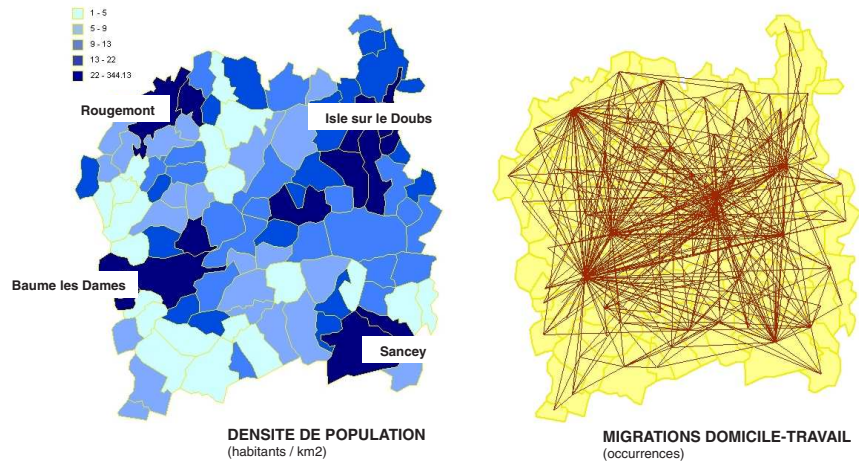


Figure 1. Densités de populations et migrations alternantes dans le Pays du Doubs Central

échéant (selon une fourchette dont la borne supérieure constitue la limite maximale à ne pas dépasser par le service). Le calcul des tournées est lancé une fois bloquées les réservations pour le lendemain. Les usagers sont ensuite rappelés afin de leur spécifier leur heure effective de prise en charge. Enfin, les feuilles de routes correspondantes sont communiquées aux taxis. Ces derniers restent libres de s'échanger les courses par la suite. Un taxi ne peut pas regrouper ses propres clients avec ceux du TAD, à cause des risques de non conformité aux engagements vis à vis du client. Le système est de plus défini par :

- le niveau de demandes de mobilité (lié à la structure des semis de populations sur le territoire),
- le nombre de véhicules disponibles (non extensible),
- le nombre de dépôts de ces véhicules (les adresses des compagnies de taxis généralement),
- le calcul du coût préconisé pour rémunérer les compagnies de taxis (prise en compte du haut-le-pied³, fixation du facteur kilométrique et du montant de la prise en charge),

3. trajet à vide du taxi pour aller chercher un client.

- de quatre forfaits tarifaires (selon zone et regroupement),
- des détours autorisés (valeurs temporelles maximales à ne pas dépasser par trajet).

3.2. Illustration du fonctionnement du TAD

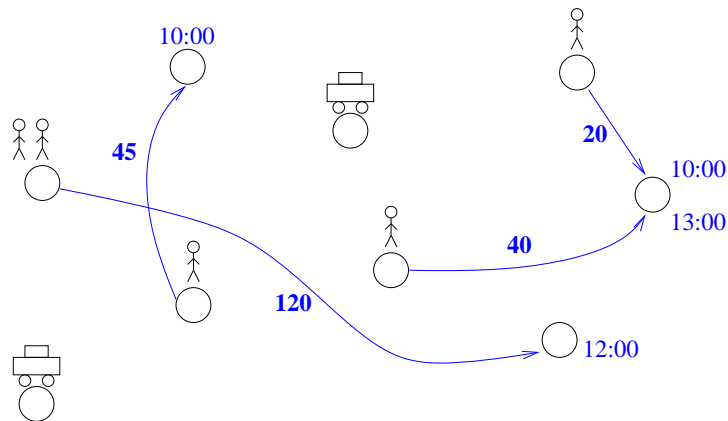


Figure 2. Représentation du Transport à la Demande dans le Doubs Central - Demandes

La figure 2 représente les requêtes (flèches), les temps de trajet (pondération des flèches) estimés de l'origine (cercle surmonté de personnages) vers la destination. L'heure d'arrivée souhaitée est indiquée à côté de la destination. Les cercles accôtés à un véhicule représentent les dépôts des taxis.

La figure 3 représente des tournées de véhicules répondant à la demande. L'itinéraire de chaque taxi est représenté en pointillés. L'heure de passage du taxi est indiquée pour chaque lieu (ramassage ou livraison).

3.3. Positionnement bibliographique

Le TAD du Doubs Central est donc du type (**m-DARPTW**) (*c.f.* section 2.2), correspondant à un problème « Dial-a-Ride », réalisé par une flotte de plusieurs véhicules, utilisant des données statiques, sous contrainte de fenêtres de temps. Les caractéristiques principales de ce DARPTW sont les suivantes :

- 1) plusieurs véhicules de capacité limitée, basés dans des dépôts,
- 2) plusieurs requêtes de transport origine-destinations émanant de plusieurs personnes, soumises à des contraintes de qualité de service (respect de fenêtres de temps et des délais),

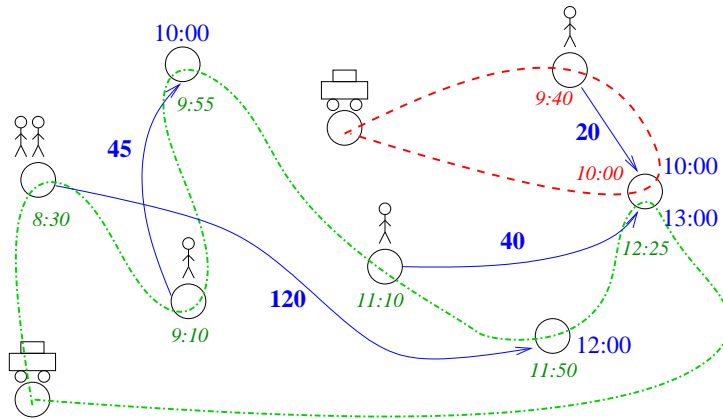


Figure 3. Représentation du Transport à la Demande dans le Doubs Central - Solutions

- 3) matrices des temps, distances et/ou coûts, correspondant au réseau de transport,
- 4) objectif de minimisation des coûts,
- 5) toutes ces données sont connues à l'avance.

Le DARP statique est un problème ayant fait l'objet de nombreux travaux. Ces travaux sont relatifs à des applications réellement mises en place. Ainsi, les problèmes de DARP dynamique restent marginalement étudiés faute d'applications. Les systèmes faisant appel à des procédures d'optimisation et au fonctionnement proche de celui du Doubs Central, concernent généralement le transport de catégories spécifiques de population (handicapés, écoliers...) en milieu urbain. Les volumes de requêtes traités sont assez importants (jusqu'à 2000 requêtes par jour). La littérature propose ainsi davantage de méthodes heuristiques à la résolution de problèmes de DARP.

Un tel problème a une décomposition naturelle en trois sous-problèmes :

- 1) *un problème d'affectation*, qui consiste à déterminer par quel véhicule est visité chaque lieu.
- 2) *un problème de séquençement*, qui consiste à déterminer dans quel ordre chaque véhicule visite les lieux qui lui sont attribués.
- 3) *un problème d'horodatage*, qui consiste à déterminer l'heure de passage des véhicules en chaque lieu.

Ces trois sous-problèmes peuvent être traités globalement ou indépendamment.

En ce qui concerne les approches globales, on trouve des méthodes d'insertion qui consistent à insérer au fur et à mesure les requêtes dans une tournée, c'est-à-dire positionner le point de ramassage et celui de livraison dans la séquence composant déjà la tournée. Par exemple, Toth et Vigo [TOT 96] construisent ainsi plusieurs tournées en parallèle en autorisant des échanges intra et inter tournées ; ils améliorent ensuite leurs

résultats par une méthode d'optimisation locale de type recherche tabou ([GLO 89], [GLO 90]).

Une autre approche courante, appelée *clustering first - routing second*, revient à traiter dans une première phase le problème d'affectation avant celui de routage (séquencement et horodatage). Dumas, Desrosiers et Soumis [DUM 89] regroupent les demandes par véhicule suivant des critères de proximité puis construisent les tournées réalisables par un algorithme de résolution du *Single-Vehicle* DARP. Ceci se fait par une méthode de génération de colonnes utilisant un algorithme exact de programmation dynamique (Desrosiers et al. [DES 86]) pour le sous-problème de routage (*1-PDPTW*). Un récent état de l'art à ce sujet est proposé par Cordeau, Laporte [COR 03].

L'objectif principal de ces algorithmes est de minimiser le coût d'exploitation en minimisant les distances parcourues. La limitation des détours autorisés est la principale contrainte de qualité de service imposée. Les désagréments infligés aux passagers, comme les temps d'attente, sont pris en compte de façon secondaire.

Une des originalités du problème dans le Doubs Central est de limiter les détours autorisés en termes de délai et non de distance. De plus, le pays du Doubs Central est une zone rurale (distances longues) où le potentiel de demande est peu élevé. La nature des instances proposées est donc sensiblement différente de celles rencontrées dans la littérature.

4. Formalisation d'un problème théorique optimal pour la collectivité avec fenêtres de temps – *m-DARPTW*

La collectivité est le client de l'application, son objectif est que toutes les demandes des clients soient satisfaites à un moindre coût pour eux.

4.1. Description des données

Le service doit répondre à n requêtes de transport d'un point de ramassage (élément de l'ensemble \mathcal{P}) vers un point de livraison (élément de l'ensemble \mathcal{D}). Servir une requête consiste à ramasser un nombre l_i de personnes en i et de toutes les déposer en $i + n$; on a $-l_{i+n} = l_i$. Ainsi, des personnes partant du même endroit pour des destinations différentes sont modélisées par des requêtes différentes. On dispose d'une flotte K de m véhicules, de dépôts, de capacités, de coûts fixes d'utilisation journalière et de coûts kilométriques respectifs o_k , C_k , pc_k et c_k . Le ramassage en i doit démarrer dans la fenêtre de temps $[a_i, b_i]$ et la livraison doit démarrer dans la fenêtre de temps $[a_{i+n}, b_{i+n}]$. Ces fenêtres de temps sont déduites de la requête client (*i.e.* date limite d'arrivée) et du délai maximal autorisé (pourcentage du temps de parcours du chemin le plus court). On se ramène ainsi à un problème de type *m-PDPTW* standard. Les véhicules partent à vide, $l_{o_k} = 0$. Les temps de parcours t_{ij} entre deux points (ramassage, livraison, dépôt) sont connus. On définit ainsi un graphe complet ($A = V^2$) orienté $G = (V, A)$ doublement valué, où V est l'ensemble des points de

ramassage, de livraison et les dépôts ($V = \mathcal{P} \cup \mathcal{D} \cup o_k | k \in K$). Les arcs (i, j) sont pondérés par les valeurs d_{ij} et t_{ij} . Pour chaque arc i, j et véhicule k on associe un coût c_{ijk} fonction de la distance, $c_{ijk} = c_k \times d_{ij}$.

Pour déterminer si la tournée du véhicule k passe par l'arc $(i, j) \in A$, on utilise la variable de décision x_{ijk} dont la valeur est fixée à 1 si l'arc est emprunté et 0 sinon. La variable T_{ik} représente l'instant où le véhicule k débute le service en $i \in V$ et L_{ik} représente la charge (nombre de passagers présents dans le véhicule k) à la fin du service en $i \in V$.

4.2. Modèle mathématique

$$\min f(x_{ijk}) \quad [1]$$

sujet à

$$\sum_{k \in K} \sum_{j \in V} x_{ijk} = 1 \quad \forall i \in \mathcal{P}, \quad [2]$$

$$\sum_{j \in \mathcal{P} \cup \mathcal{D}} x_{ijk} - \sum_{j \in \mathcal{P} \cup \mathcal{D}} x_{j,n+i,k} = 0 \quad \forall k \in K, \forall i \in \mathcal{P}, \quad [3]$$

$$\sum_{i \in V} x_{ijk} - \sum_{i \in V} x_{jik} = 0 \quad \forall k \in K, \forall j \in \mathcal{P} \cup \mathcal{D}, \quad [4]$$

$$x_{ijk} (T_{ik} + t_{ij} - T_{jk}) \leq 0 \quad \forall k \in K, (i, j) \in A, \quad [5]$$

$$a_i \leq T_{ik} \leq b_i \quad \forall k \in K, i \in V, \quad [6]$$

$$T_{ik} + t_{i,n+i} \leq T_{n+i,k} \quad \forall k \in K, i \in \mathcal{P}, \quad [7]$$

$$x_{ijk} (L_{ik} + l_j - L_{jk}) = 0 \quad \forall k \in K, (i, j) \in A, \quad [8]$$

$$l_i \leq L_{ik} \leq C_k \quad \forall k \in K, i \in \mathcal{P}, \quad [9]$$

$$0 \leq L_{n+i,k} \leq C_k - l_i \quad \forall k \in K, n+i \in \mathcal{D}, \quad [10]$$

$$L_{o_k,k} = 0 \quad \forall k \in K, \quad [11]$$

$$x_{ijk} \in \{0, 1\} \quad \forall k \in K, (i, j) \in A. \quad [12]$$

$$f(x_{ijk}) = \sum_{k \in K} \sum_{(i,j) \in A} c_{ijk} x_{ijk} + \sum_{k \in K} \left(p c_k \sum_{i \in \mathcal{P}} x_{o_k i k} \right) \quad [13]$$

4.3. Explication des contraintes

[1],[13] La fonction objectif minimisant le coût total des trajets.

[2] et [3] Chaque requête est servie exactement une fois et ceci par un seul véhicule. [2] Exactement un véhicule k quitte le noeud de ramassage i et ce une seule fois. [3] Si le véhicule k quitte le noeud de ramassage i alors k arrive au noeud de livraison correspondant $n + i$.

[4] Le véhicule k quitte et arrive autant de fois sur un noeud j , que ce soit un noeud de ramassage, de livraison ou de dépôt.

[5] Si le véhicule k emprunte l'arête de i vers j , il doit débiter le service en j après avoir pu finir le service en i puis voyagé par (i, j) . Cette contrainte assure l'élimination des sous-tours⁴ Desrosiers et al. [DES 84].

[6] Le véhicule k respecte la fenêtre de temps pour le début du service au noeud i .

[7] Le ramassage doit se faire avant la livraison.⁵

[8] Les véhicules ont l'obligation d'effectuer la requête complète (on prend ou on dépose toutes les personnes) au passage en j . On admet qu'on ne dépose personne en route.

[9] Le nombre de personnes dans le véhicule k après un ramassage en i est compris entre le nombre de personnes à prendre en i et la capacité de k .

[10] Après une livraison en $n + i$, il doit y avoir au moins autant de places libres dans le véhicule que ce qui doit être chargé en i .

[11] Les véhicules démarrent à vide.

[12] Le véhicule k prend ou ne prend pas la route de i vers j .

Ce modèle mathématique peut être transcrit dans un solveur de programmation linéaire, théoriquement capable de fournir une solution optimale au problème. Les résultats ainsi obtenus sont comparés à ceux de notre algorithme dans la section 6.

5. Résolution par un algorithme d'insertion

5.1. Idée générale

Nous proposons un algorithme d'optimisation qui fonctionne en deux phases :

- une première phase pendant laquelle on construit des tournées faisables par une procédure d'insertion gloutonne (algorithme 1),
- une seconde phase où, par le biais d'insertions et de suppressions de requêtes dans les tournées, on cherche à améliorer la qualité de la solution.

4. un sous-tour est un circuit n'ayant pas de dépôt pour origine.

5. à remplacer par $T_{ik} \leq T_{n+i,k} \forall k \in K, i \in \mathcal{P}$ si l'inégalité triangulaire n'est pas respectée par les t_{ij} .

L'idée est d'insérer les requêtes dans la tournée où cette insertion est la moins coûteuse. L'insertion d'une requête dans une tournée ne doit pas modifier l'ordre déjà établi entre les points composant cette tournée à une itération précédente.

5.2. Description de l'algorithme

On considère une tournée par véhicule disponible, et, à chaque itération, une nouvelle requête que l'on va insérer dans la tournée offrant la possibilité d'insertion au meilleur coût. Insérer une requête dans la tournée d'un véhicule, c'est insérer un point de ramassage et un point de livraison. Pour trouver la meilleure configuration d'insertion possible dans une même tournée, l'algorithme teste toutes les configurations réalisables. Dans notre implémentation, on réduit l'espace de recherche principalement en utilisant les contraintes de fenêtres de temps.

Le coût d'insertion $coutInsertion(i, k, u, v)$ d'un point i dans la tournée du véhicule k ($tour_k$), entre les points u et v (u précédant v) s'exprime comme suit : $coutInsertion(i, k, u, v) = c_{uik} + c_{ivk} - c_{uvk}$.

Algorithm 1 Algorithme d'insertion

```

1: procedure INSERTION( $\mathcal{P} \cup \mathcal{D}, K$ )
2:   Trier les requêtes
3:   for all requête  $(i, i + n) \in \mathcal{P} \times \mathcal{D}$  do
4:     for all véhicule  $k \in K$  do
5:       for all position d'insertion de  $i$  dans  $tour_k$  do
6:         for all position d'insertion  $(u, v)$  de  $i + n$  dans  $tour_k$  postérieure
           à  $i$  do Calculer  $coutInsertion(i, k, u, v)$ 
7:         if  $coutInsertion(i, k, u, v) < meilleurCoûtInsertion$  then
8:            $meilleurCoûtInsertion = coutInsertion(i, k, u, v)$ 
9:         end if
10:      end for
11:    end for
12:  end for
13:  Insérer la requête  $(i, i + n)$  dans la tournée à la position ayant le meilleur
    coût d'insertion.
14: end for
15: end procedure

```

L'algorithme est « rapide » $O(n^3)$, et de plus aisément parallélisable (les calculs de moindre coût d'insertion d'une requête dans chaque tournée étant indépendants). Cet algorithme est une heuristique et ne donne, a priori, aucune garantie de « proximité » par rapport à une solution optimale.

5.3. Détails d'implémentation et traitements annexes

L'ordre dans lequel sont traitées les requêtes a une influence sur la qualité de la solution. L'algorithme s'exécutant rapidement, on pourrait le lancer sur différents ordres aléatoires. Les paragraphes 5.3.1 et 5.3.2 présentent les deux traitements permettant d'obtenir des ordres « non naïfs » de traitement des requêtes. Dans le paragraphe 5.3.3, nous présentons un algorithme de post-optimisation.

5.3.1. Pré-tri basé sur les contraintes temporelles

Il paraît raisonnable de supposer que des bonnes tournées (peu coûteuses) vont effectuer des regroupements de requêtes intéressants. De tels regroupements doivent répondre à des contraintes de proximité spatiale et temporelle. Il est clair que le regroupement de deux requêtes « proches » spatialement et temporellement, a des chances d'améliorer le coût de la solution. D'après le calcul du coût d'une insertion et de la formule déterminant les c_{ijk} (c.f. section 4.1), l'algorithme 1 utilisant une insertion au moindre coût assure une certaine proximité spatiale sans tenir compte aucunement des distances temporelles entre les deux requêtes. Ceci diminue, a priori, l'intérêt de tels regroupements. En ordonnant préalablement les requêtes en fonction d'une certaine proximité temporelle, on favorise ensuite dans l'algorithme 1 les regroupements de requêtes proches temporellement et spatialement. Cet ordonnancement des requêtes peut par exemple se faire à partir des différentes bornes des fenêtres de temps imposées en chaque point.

5.3.2. L'algorithme comme son propre prétraitement

Il s'agit de déterminer l'ordre de traitement des requêtes à partir d'une solution obtenue par l'algorithme à partir d'un ordre quelconque. Le nouvel ordre dans lequel les requêtes seront considérées est celui qui est décroissant par rapport au gain qu'induirait leur suppression. En les insérant en premier, on peut ainsi espérer diminuer l'impact négatif de leur insertion en fin de traitement. Le gain de suppression d'une requête $(i, i + n)$ (i entre u et v et $i + n$ entre w et x) d'une tournée du véhicule k , est défini par la formule : $gainSuppression(i, k, u, v, w, x) = coutInsertion(i, k, u, v) + coutInsertion(i + n, k, w, x) / d_{i, i+n}$. On divise le coût par la distance entre le point de ramassage et celui de livraison afin de ne pas pénaliser les requêtes les plus longues.

5.3.3. Optimisation locale par suppression puis insertion

L'idée est de retirer certaines requêtes de la solution, puis de les réinsérer au moindre coût (avec la même procédure d'insertion que pour l'algorithme 1). La nouvelle solution est ainsi soit identique à l'ancienne, soit plus avantageuse. On peut espérer modifier la solution courante par cette opération puisque, par rapport à l'insertion lors dans l'algorithme 1, toutes les requêtes sont desservies. Ce traitement est appliqué aux requêtes dans leur ordre initial d'insertion. Cette opération peut être répétée jusqu'à ce qu'on ne constate plus d'amélioration de la solution.

6. Résultats et analyse

Les tests sont effectués sur des données réelles du pays du Doubs Central : réseau de transport, flotte de véhicules, localisation des arrêts. A partir de ces informations, les demandes de transport sont estimées aléatoirement sous contrainte de probabilité à partir des statistiques sur les flux de population, le potentiel de pénétration du service sur le marché auprès des différentes catégories de population (Fig. 4).

nombre de requêtes	délai(%) maximum	heuristique			Cplex		
		solution	cpu (s)	unité	solution	cpu (s)	unité
10	150	464	0	46,4	452	3	45,2
10	200	443	0	44,3	430	36	43,0
25	150	1047	0	41,9	-	3600	-
25	200	964	0	38,5	-	3600	-
50	150	1940	0	38,8	-	3600	-
50	200	1757	0	35,1	-	3600	-

Tableau 1. résultats comparatifs entre notre heuristique et une méthode exacte

Dans le tableau 1, nous présentons des résultats expérimentaux sur 3 jeux de 40 instances (à 10, 25 et 50 requêtes par jour, respectivement). Pour les instances à 10 (*resp.* 25 et 50) requêtes, on dispose de 5 (*10 et 15*) véhicules. Pour chacune des 3 séries d'instances, nous avons testé deux modes de calcul des délais maximaux autorisés de détours : 200% et 150% du temps de parcours du chemin le plus rapide du point de montée dans le véhicule à celui de descente. Pour évaluer les coûts, nous avons retenu les tarifs préfecture de prise en charge de 1,9 Euros et de 1,26 Euros au km taxi, ainsi qu'une portée des tournées de 15 à 20 km. Ne sont pas retranchées à ce coût les recettes venant des clients : il s'agit d'un coût brut.

Les deux premières colonnes du tableau caractérisent donc la nature des instances. Dans la catégorie *heuristique*, est indiquée le coût moyen des solutions (coût total divisé par le nombre de requêtes) (colonne *solution*) obtenues avec notre algorithme d'insertion enrichi par les améliorations évoquées dans la section 5.3, ainsi que leur temps de calcul moyen (*cpu*) et le coût unitaire moyen (en Euros) de prise en charge d'une requête (*unité*). La colonne *Cplex*, présente sous le même modèle les résultats obtenus avec le solveur de programmation linéaire commercial *Cplex 9.0* (paramétré avec les valeurs par défaut).

Pour utiliser ce solveur, on utilise le modèle mathématique⁶ présenté dans la section 4.2. Le solveur exécute un *Branch & Bound* (méthode exacte de résolution arborescente) où il résout à chaque noeud une relaxation linéaire par l'algorithme du *simplexe* [HIL 86]. Il permet donc de calculer une solution optimale au problème.

6. après l'avoir linéarisé.

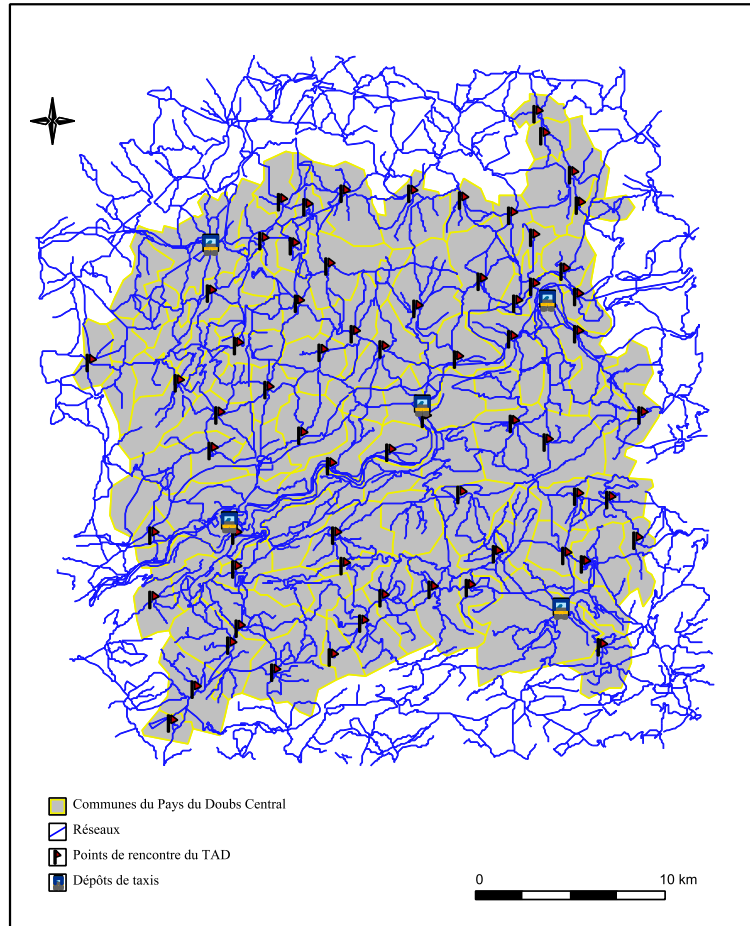


Figure 4. Localisation des points de rencontre et des dépôts de taxis pressentis du TAD du Pays du Doubs Central

A partir d'une quinzaine de requêtes, la méthode exacte ne parvient pas à résoudre optimalement le problème (même au bout d'1 heure). On parvient toujours à obtenir une borne inférieure⁷ mais rarement une borne supérieure⁸ (d'une valeur largement supérieure à celle obtenue par l'heuristique).

Notre algorithme, qui est de complexité polynomiale, est beaucoup plus rapide à s'exécuter que le *Branch & Bound* de *Cplex*. Nous obtenons des solutions près de 3% (pour les instances à 10 requêtes) plus coûteuses que les solutions optimales. Nous

7. coût que l'on sait inférieur à celui d'une solution optimale.

8. solution réalisable mais pas forcément optimale.

remarquons que plus le délai maximal autorisé est long, plus la résolution est coûteuse en temps de calcul. Ceci est dû à une augmentation du nombre de solutions possibles.

L'observation des résultats de l'heuristique indique que le coût moyen unitaire d'une requête diminue avec l'augmentation du délai maximal autorisé (de 150% à 200%). On constate par exemple un gain moyen de 4,5% pour 10 requêtes et de 9,5% pour 50 requêtes. Ce coût unitaire diminue aussi fortement avec l'augmentation du nombre de requêtes à servir. Pour un délai maximal de 200%, on a un gain de 21% entre les instances à 10 requêtes et celles à 50 (13% par rapport à celles à 25 requêtes).

7. Conclusion

Dans cet article, nous avons proposé un algorithme adhoc dans le cadre de l'implantation d'un transport à la demande points à points dans une zone rurale (le pays du Doubs Central). Notre algorithme propose des solutions de bonne qualité dans des temps de calcul extrêmement courts. Les résultats ont montré par ailleurs que la montée en charge du service et l'incitation au regroupement grâce aux détours pouvaient avoir un effet non négligeable sur les coûts.

Toutefois, des améliorations peuvent être apportées :

- d'une part, sur les solutions calculées, plus précisément la capacité à trouver des solutions réalisables et sur d'autres critères de qualité non exprimés dans la fonction objectif (répartition des charges, temps d'attente, etc.),
- d'autre part, sur la plausibilité des tournées (induisant un coût extrêmement élevé), qui ne tiennent pas compte actuellement des effets de distance, tendant naturellement à favoriser les requêtes de courte portée, par des effets de polarisation locale.

Remerciements

Nous tenons à remercier chaleureusement les responsables et professionnels du Pays du Doubs Central pour leur participation financière et méthodologique active dans ce projet de recherche-action.

8. Bibliographie

- [AMB 04] AMBROSINO G., NELSON J., *Demand Responsive Transport Services : Toward the Flexible Mobility Agency*, ENEA, Italy, 2004.
- [COR 03] CORDEAU J.-F., LAPORTE G., « The dial-a-ride problem (DARP) : Variants, modeling issues and algorithms. », *4OR*, vol. 1, 2003, p. 89-101.
- [DES 84] DESROSIERS J., SOUMIS F., DESROCHERS M., « Routing with time windows by column generation. », *Networks*, vol. 14, 1984, p. 545-565.

- [DES 86] DESROSIERS J., DUMAS Y., SOUMIS F., « A dynamic programming solution of the large-scale single-vehicle dial-a-ride problem with time windows », *American Journal of Mathematical Sciences*, vol. 6, 1986, p. 301-325.
- [DUM 89] DUMAS Y., DESROSIERS J., SOUMIS F., « Large scale multi-vehicle dial-a-ride systems », *Gerad, Ecole des Hautes Etudes de Commerciales, Montréal*, vol. G-89-30, 1989.
- [DUP 99] DUPUY G., *La dépendance automobile, symptômes, analyses, diagnostic, traitements*, Anthropos, Collection Villes, Paris, 1999.
- [FIN 02] FINKE G., Ed., *Recherche opérationnelle et réseaux – Méthodes d'analyse spatiale*, Hermès Sciences Publications, Paris, 2002.
- [GLO 89] GLOVER F., « Tabu search part I. », *ORSA journal on computing*, vol. 1, 1989, p. 190-206.
- [GLO 90] GLOVER F., « Tabu search part II. », *ORSA journal on computing*, vol. 2, 1990, p. 4-32.
- [HIL 86] HILLIER F., LIEBERMAN G., *Introduction to operations research*, Holden-Day, Oakland, Californie, 1986.
- [HOU 04] HOUZET S., JOSSELIN D., « Rapport d'étude de la faisabilité de l'extension du service Evolis-gare sur le périmètre de la Communauté d'Agglomération du Grand Besançon », Survey, Mai 2004, CAGB-ADEME.
- [TOT 96] TOTH P., VIGO D., « Fast local search algorithms for the handicapped persons transportation problem. », OSMAN I. I., KELLY J., Eds., *Meta-heuristics : Theory and applications.*, p. 677-690, Kluwer, Boston, 1996.
- [TOT 01] TOTH P., VIGO D., *The vehicle routing problem*, Society for industrial and Applied Mathematics, Philadelphia, PA, USA, 2001.
- [WIE 02] WIEL M., *Ville et automobile*, Descartes Cie, Paris, 2002.