

La convergence, une manière de concevoir un TAD

Sommaire

4.1	Le principe de convergence	68
4.2	La convergence simple	69
4.2.1	Graphe de convergence issu de la polarisation	70
4.2.2	Les arbres couvrants tentaculaires	72
4.2.3	Un cas particulier de monoconvergence : l’anneau	75
4.2.4	Le principe de monodivergence	76
4.2.5	Discussion	78
4.3	De multiples convergences	79
4.3.1	Méthode d’agrégation	80
4.3.2	Construction du graphe	83
4.3.3	Construction des listes de successeurs	84
4.3.4	Construction de la matrice des temps réduite	86
4.3.5	Méthode sans agrégat	87
4.3.6	Discussion	87
4.4	Conclusion	87

Introduction

La convergence est à la base de nos méthodes d’optimisation, puisqu’elle conditionne la structure des graphes sur lesquels fonctionnent les algorithmes développés. Après une description du principe de convergence, nous nous attachons à deux de ses structures dans le cadre des TAD : la monoconvergence et la multiconvergence.

Ces deux formes de convergence se manifestent soit artificiellement, quand la destination est imposée (monoconvergence), soit naturellement quand l’espace est polarisé

par une ou plusieurs villes (mono- ou multiconvergence). Ces considérations favorisent la construction de graphes de desserte selon des méthodes originales jouant sur la flexibilité du service autorisant des détours.

La monoconvergence donne lieu à une première méthode de construction de graphes, ainsi qu'à une variante, où la convergence se traduit par plusieurs points très proches. En parallèle, la monodivergence peut être traitée de manière quasiment similaire à la monoconvergence.

La multiconvergence, quant à elle, introduit un concept supplémentaire ayant trait à la granularité spatiale, qui nous permet de construire un graphe à partir d'une superstructure établie à différentes échelles.

4.1 Le principe de convergence

Du point de vue mathématique, la convergence existe par exemple dans l'étude des suites réelles. Une suite réelle u est dite convergente lorsque l'ensemble de ses points u_n se rapprochent d'un point limite u^* , on dit dans ce cas que u tend vers u^* :

$$\lim_{n \rightarrow +\infty} u_n = u^* \Leftrightarrow \lim_{n \rightarrow +\infty} (u_n - u^*) = 0$$

En géographie, la convergence caractérise le rapprochement de flux jusqu'à leurs rencontres en un lieu. Elle est visible notamment pour des flux maritimes, nuageux... En géographie humaine, les déplacements de fret ou de personnes sont des flux émis et reçus selon un ensemble de phénomènes comme ceux décrits dans le chapitre 3.

D'une manière générale, le principe de convergence traduit la réunion en un point (lieu) de plusieurs courants. Ainsi les déplacements humains sont animés par les polarités du territoire et se structurent en flux convergeant vers des lieux communs particulièrement attractifs comme les centres-villes pour les commerces et les administrations, les grandes zones commerciales périphériques ou encore les « lieux de vie » (cinémas, salles de concert, stades...).

Les TAD n'échappent pas à cette règle, car ils sont par essence même les produits de la mobilité humaine. Dans la thèse de Castex (2007), figure une typologie exhaustive des différents types de TAD. Parmi les différentes topologies et organisations spatiales des services figurent les formes dites en convergence :

- la monoconvergence qui traduit l'attraction d'un seul point du territoire (C, figure 4.1(a));

- la multiconvergence occasionnée par plusieurs points faisant chacun converger à lui-même plusieurs flux de personnes (C_1, C_2 , fig. 4.1(b)).

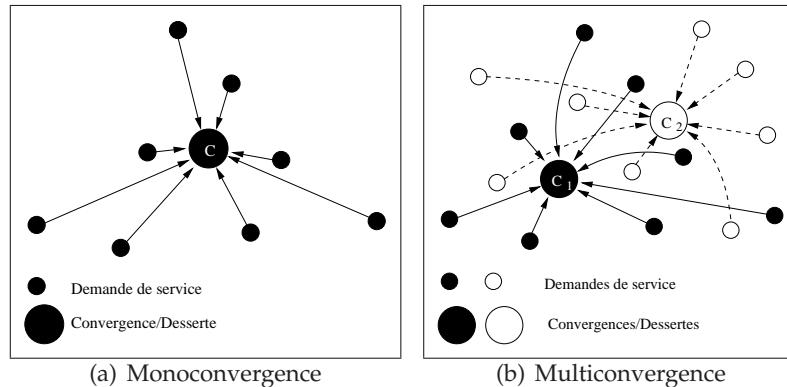


FIG. 4.1: Deux exemples de convergence : l'exemple (a) traduit une seule convergence (C), le (b) indique plusieurs convergences, c'est-à-dire plusieurs dessertes (C_1, C_2).

4.2 La convergence simple

La monoconvergence peut paraître artificielle dans la mesure où il y a nécessairement plusieurs attracteurs sur le territoire. Néanmoins elle traduit une réalité créée par les transporteurs eux-mêmes sur les réseaux de transport, pour répondre à un besoin de mobilité en direction de lieux particuliers comme les gares TGV ou les aéroports. Ces places, souvent excentrées et pourtant fortement attractives, demeurent peu voire mal desservies par les transports publics, notamment en horaires de frange (très tôt ou très tard). Les systèmes de TAD comme « Evolis-Gare » à Besançon répondent à cette demande de mobilité.

Cette polarisation unique permet l'introduction d'un ensemble d'optimisations à travers l'élaboration d'un graphe de convergence représentant le problème de transport à solutionner d'une part, et d'autre part à travers les Arbres Couvrants Tentaculaires (Canalda *et al.*, 2004; Chevrier *et al.*, 2006c) décrivant la solution de transport sous la forme d'un graphe particulier : un arbre.

Par la suite, pour assurer le service, nous disposons d'une flotte de n véhicules à capacités différentes. La méthode que nous proposons est statique dans la mesure où nous connaissons l'ensemble des demandes de service à l'avance.

4.2.1 Graphe de convergence issu de la polarisation

Nous souhaitons dresser un 1-graphe $G(V, E)$ orienté acyclique (*Directed Acyclic Graph*, DAG), avec V l'ensemble des nœuds du graphe (indiquant une prise en charge ou montée et la descente unique du service) et E l'ensemble des arcs (indiquant les chemins possibles). Pour établir ce « graphe de convergence » comme celui de la figure 4.2(a), nous disposons au préalable d'un ensemble de données nécessaires :

- l'horaire h_C de desserte au point C.
- les requêtes de transport r exprimées sous la forme :
 - r^+ pour le point de prise en charge ;
 - h_{r^+} pour les horaires de prise en charge souhaités ;
 - w_r pour le nombre de personnes à transporter de r^+ à C.
- la matrice origines/destinations (OD) en temps (M^T) des points du territoire à desservir, obtenue sur SIG¹. Les temps relevés correspondent aux plus courts chemins ;
- un coefficient de relaxation k appliqué aux temps de parcours $t_{x \rightarrow y}$ des points x vers y : $t_{x \rightarrow y} = k.M_{xy}^T$, pour faciliter les détours dans l'ensemble des plus courts chemins possibles. k est un coefficient à calibrer, qui détend les temps de parcours pour autoriser des détours (dans ce cas $k \geq 1.0$), irréalisables si les temps de parcours sont tendus ($k = 1.0$).

La construction du graphe se fait de la manière suivante. Préalablement les nœuds r^+ de prise en charge sont ordonnés selon leurs temps de parcours nécessaires pour gagner la convergence, c'est l'ordre total des nœuds (Chevrier *et al.*, 2006b), tel que

$$\forall x, y \in V, t_{x \rightarrow C} > t_{y \rightarrow C} \Leftrightarrow x \prec y$$

La séquence Θ des sommets V ordonnés est une construction par insertion. Comme les horaires sont déterminés par rapport à l'horaire de convergence, les points de départ les plus éloignés à la convergence se voient attribués un horaire de passage plus tôt. L'ordre de la séquence Θ correspond à l'ordre des horaires de départ, du plus tôt au plus tardif.

¹Système d'Information Géographique

La séquence Θ est construite par insertion (x, y sont deux points de ramassage) :

1. $\Theta(V) \leftarrow \emptyset$
2. **Pour chaque** $x \in V$ **faire**
 - (a) inséré \leftarrow FAUX
 - (b) **Pour chaque** $y \in \Theta(V)$ **faire**
 - Si** $t_{x \rightarrow C} > t_{y \rightarrow C}$ **alors**
 - i. insérer x avant y dans $\Theta(V)$ # x doit être desservi avant y
 - ii. inséré \leftarrow VRAI
 - Fin si**
 - Fait**
 - (c) **Si** inséré = FAUX **alors**
 - ajouter x à $\Theta(V)$ # x est en fin de liste
 - Fin si**
- Fait**

Les retours arrière ne sont donc pas autorisés. Quant aux connexions entre chacun des nœuds, celles-ci doivent répondre à la condition temporelle selon laquelle un point x peut gagner un point y si et seulement si $h_x + t_{x \rightarrow y} \leq h_y$, et nous notons $\langle xy \rangle$ l'arc correspondant. Les arcs notés en pointillés indiquent quant à eux les transitivités :

$$\forall x, y, z \in V, h_x + t_{x \rightarrow y} + t_{y \rightarrow z} \leq h_z \Rightarrow \exists \langle xz \rangle \in E$$

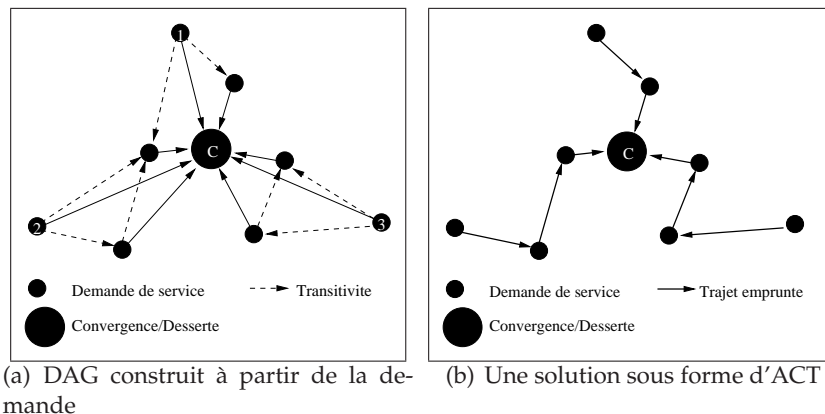


FIG. 4.2: DAG (a) associé au problème de TAD de la figure 4.1(a) et un arbre couvrant tentaculaire optimal (b)

Les liaisons ainsi faites, l'ensemble des nœuds a à la fois des prédécesseurs (des arcs incidents) et des successeurs (arcs sortants). Quant aux autres nœuds, certains n'ont pas de prédécesseurs : ce sont les nœuds minimaux du graphe (fig. 4.2(a) sommets 1, 2 et 3) et un seul n'a pas de successeur : c'est le nœud terminal (l'*antiracine* de l'arbre) qui correspond au point de desserte, la convergence C.

Connaître et distinguer ces sommets nous permet maintenant d'utiliser les arbres couvrants tentaculaires pour couvrir le graphe ainsi créé et ensuite d'apporter des optimisations de résolution dans l'algorithme génétique, qui se révèle particulièrement efficace dans ce type d'application (Julstrom et Raidl, 2002; Raidl et Julstrom, 2003).

4.2.2 Les arbres couvrants tentaculaires : une solution à la monoconvergence

Les arbres couvrants tentaculaires (ACT) sont issus de travaux consacrés au transport à la demande en convergence simple (Chevrier *et al.*, 2006a,c).

Définitions

Nous posons préalablement un ensemble de définitions nécessaires à l'explicitation des ACT :

1. un arbre $A(V, E)$ est un graphe connexe sans cycle (ayant une arête de moins que de sommets) ;
2. un arbre couvrant $A(V, E)$ est un sous-graphe de $G(V, E)$ contenant tous les sommets de G ;
3. l'*antiracine* de l'arbre couvrant $A(V, E)$ est le sommet n'ayant que des arcs incidents (i.e. pas d'arcs sortants) ;
4. un chemin terminant est une branche de l'arbre couvrant $A(V, E)$, se terminant à l'*antiracine* de l'arbre ;
5. un arbre couvrant tentaculaire est un arbre couvrant dont tous les chemins terminants sont arcs-disjoints deux à deux et sommets-disjoints deux à deux (hormis à l'*antiracine*).

L'unique jonction des branches de l'arbre couvrant tentaculaire correspond à l'*antiracine*. En effet, ce sommet ne peut être une racine en raison de l'orientation des arcs, ceux-ci indiquant une destination et non une source.

Les figures 4.3(b,c,d,e,f) sont des exemples d'ACT du graphe de la figure 4.3(a). Un ACT est dit minimal lorsque le nombre de branches le composant est minimal.

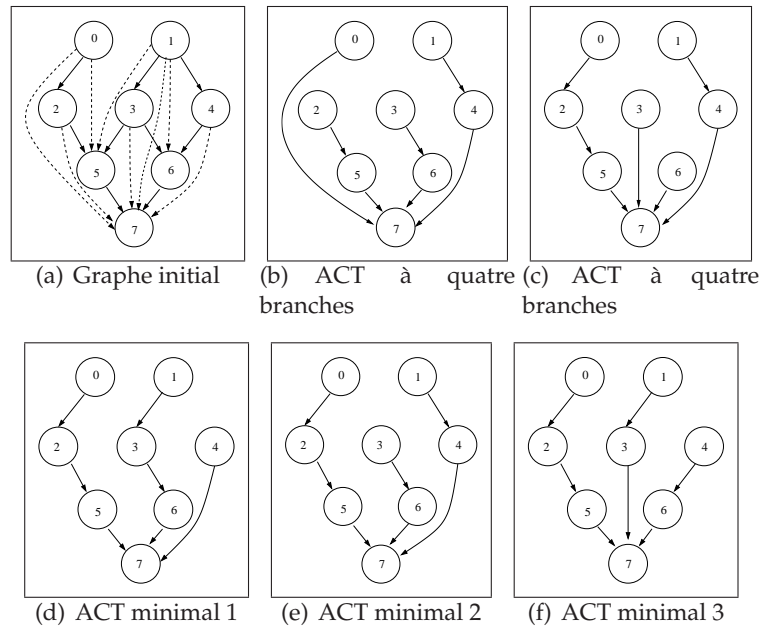


FIG. 4.3: Un DAG à 8 nœuds (a) et 5 ACT : les ACT (b,c) sont non-minimaux (4 branches) tandis que les ACT (d,e,f) sont minimaux (3 branches) et équivalents (au sens de la minimisation du nombre de branches). D'autres ACT sont possibles.

Une branche de l'ACT est une séquence, dont l'ordre des nœuds indique l'itinéraire d'un véhicule, la feuille de l'arbre étant le point de départ du véhicule tandis que l'antiracine de l'ACT correspond au point de convergence. Cinq exemples d'ACT sont donnés sur la figure 4.3, les arcs pointillés représentent les transitivités. Sur la figure 4.3(d), la suite de sommets $S_1 = (1, 3, 6, 7)$ indique au véhicule l'ordre de passage à ces arrêts. De plus, dans ce même exemple on distingue trois branches, dont S_1 ainsi que $S_2 = (0, 2, 5, 7)$ et $S_3 = (4, 7)$. La présence de ces trois branches nous indique que, dans cette configuration, il faut trois véhicules pour assurer le service. Les ACT des figures 4.3(b) et 4.3(c) représentent des dessertes nécessitant un véhicule supplémentaire car il y a quatre branches.

L'ACT de la figure 4.2(b) est l'ACT optimal représentant le meilleur service au problème de transport de la figure 4.2(a).

Application de la relaxation

Dans le bref exemple suivant, nous allons justifier l'utilité de la relaxation dans la cadre de notre application. Nous travaillons en secondes. Considérons le problème simple de TAD monoconvergent suivant :

- le rendez-vous au point C de convergence est donné à 20H. Pour avoir une marge de manœuvre en cas de retard inopiné, le rendez-vous en C est fixé 10 minutes en avance soit 19H50 : $h_C = 19 \times 3600 + 50 \times 60 = 71400$;
- deux demandes de transport vers C sont issues des points A et B ;
- les temps de parcours *tendus* (c'est-à-dire ceux inscrits dans la matrice OD initiale, sans relaxation temporelle) sont :
 - $t_{A \rightarrow C} = 900$ (15 minutes) ;
 - $t_{B \rightarrow C} = 480$ (8 minutes) ;
 - $t_{A \rightarrow B} = 480$ (8 minutes).

Nous pouvons en déduire les horaires théoriques de prises en charge aux points A et B :

$$h_A = h_C - t_{A \rightarrow C} = 70500$$

$$h_B = h_C - t_{B \rightarrow C} = 70920$$

Or, si un véhicule part de A, il risque d'arriver en retard au point B :

$$h_A + t_{A \rightarrow B} = 70500 + 480 = 70980$$

$$h_A > h_B$$

Le détour par B en venant de A n'est donc pas possible. Appliquons alors un coefficient de relaxation $k_r = 1.5$ qui va augmenter de 50% les temps de parcours et modifier les horaires de prise en charge :

$$h'_A = h_C - t'_{A \rightarrow C} \text{ avec } t'_{A \rightarrow C} = k_r \cdot t_{A \rightarrow C} = 1350$$

$$h'_A = 71400 - 1350 = 70050 \text{ et}$$

$$h'_B = h_C - t'_{B \rightarrow C} \text{ avec } t'_{B \rightarrow C} = k_r \cdot t_{B \rightarrow C} = 720$$

$$h'_B = 71400 - 720 = 70680$$

Maintenant avec ces nouveaux horaires h' , le détour vers le point B est possible et le parcours (A, B, C) ne nécessite finalement qu'un seul véhicule, au lieu de deux initialement, sous contrainte d'un changement horaire peu préjudiciable pour les usagers. Cette technique repose bien sur « l'élasticité » des temps de parcours sous couvert de la tolérance des usagers à accepter des détours.

4.2.3 Un cas particulier de monoconvergence : l'anneau

Lorsque la convergence n'est pas limitée à un seul point, plusieurs points de desserte suffisamment proches peuvent constituer une zone de convergence que nous pouvons associer à un point pour chercher une résolution *via* les ACT (Chevrier *et al.*, 2008).

Le cas que nous étudions ici, traduit par exemple la desserte d'un centre ville, autour duquel la circulation se fait sur une route à sens unique (c'est le cas de nombreuses villes moyennes comme Montbéliard, pour laquelle nous avons déployé une solution étudiée en section 7.4.1).

Dans l'exemple de la figure 4.4(a), six points de desserte se situent sur une zone restreinte (400m de diamètre par exemple) sur le pourtour d'une rocade à sens unique, que l'on peut formellement représenter par un anneau. Pour revenir à une résolution à l'aide des ACT, nous transformons cet anneau en un point de convergence C . L'idée est d'agrégier les points en un seul, en jouant sur une modification raisonnée des horaires des points sur l'anneau. Notons que l'anneau n'est qu'un cas exemplaire d'une approche plus générale d'agrégation de points.

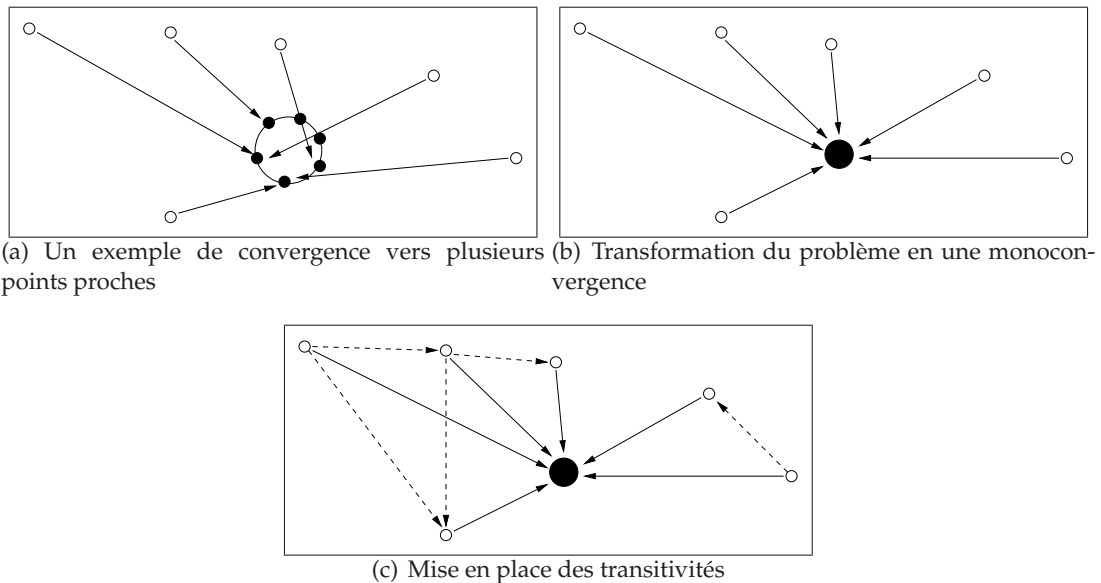


FIG. 4.4: Un cas particulier de monoconvergence : problème initial (a), transformation de l'anneau en un point (b) et mise en place des transitivités (c).

Néanmoins, cette technique nécessite quelques modifications des horaires. Pour chaque requête de transport r , nous cherchons le point de destination r^- situé sur l'anneau le plus proche (en temps) du point r^+ . La distance à r^- est utilisée pour évaluer la distance de r^+ à la convergence. Autrement dit, nous transposons la distance à la

convergence à celle au point d'entrée sur l'anneau le plus proche du point de montée.

Ainsi pour l'ACT formulé (fig. 4.5(a)), il faut définir l'ordre de passage aux points de desserte avec une post-optimisation cherchant les meilleurs chemins pour chaque véhicule (fig. 4.5(b)) : entrée dans l'anneau par le point de desserte le plus proche, puis dessertes des clients au fur et à mesure.

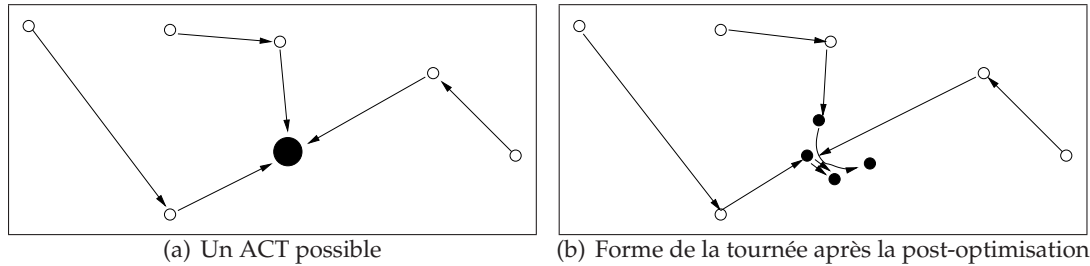


FIG. 4.5: Une solution au cas particulier de monoconvergence : un ACT (a) et la solution de desserte après la post-optimisation (b).

4.2.4 Le principe de monodivergence

La monoconvergence représente l'ensemble des trajets allers vers un point commun C. La mono-divergence indique quant à elle les trajets retours, partant de ce même point C. Dans sa représentation, le problème peut lui aussi être assimilé à un DAG, dont la solution prendra également la forme d'un ou plusieurs ACT. Fondamentalement, par rapport à la monoconvergence seuls les horaires et donc l'orientation des arcs diffèrent.

La résolution de la monodivergence consiste simplement à renverser le DAG $G(V, E)$ (fig. 4.6(a)), c'est-à-dire à inverser l'orientation des arcs pour revenir à un problème simple de monoconvergence pour lequel nous avons déjà une méthode de résolution.

À cette fin, nous constituons le DAG $G'(V, E')$ qui est le *graphe inverse* de G (fig. 4.6(b)), avec E' l'ensemble des arcs inversés de E . Pour réaliser cette inversion, nous modifions les horaires h_{r^-} de chaque requête r du problème initial en horaires h'_{r^-} au moyen d'une fonction $f : h_{r^-} \rightarrow h'_{r^-}$:

$$h'_{r^-} = 2 \times h_C - h_{r^-} - t_{C \rightarrow r^-}$$

Le développement ci-après prouve l'inversion d'ordre des horaires h_i devenant h'_i , telle que $(h_{r_1^-} > h_{r_2^-}) \xrightarrow{f} (h'_{r_1^-} < h'_{r_2^-})$:

$$h'_{r^-} = 2 \times h_C - h_{r^-} - t_{C \rightarrow r^-} \quad (4.1)$$

$$h'_{r^-} = h_C - 2 \times t_{C \rightarrow r^-} \text{ car } h_{r^-} = h_C - t_{C \rightarrow r^-}$$

$$\text{si } h_{r_1^-} > h_{r_2^-} \quad (4.2)$$

$$\text{avec } h_{r_1^-} = h_C - t_{C \rightarrow r_1^-}$$

$$h_{r_2^-} = h_C - t_{C \rightarrow r_2^-}$$

$$\text{alors } t_{C \rightarrow r_1^-} > t_{C \rightarrow r_2^-} \quad (4.3)$$

$$\text{et } t_{C \rightarrow r_1^-} = t_{C \rightarrow r_2^-} + \omega$$

$$\text{par conséquent } h'_{r_1^-} = h_C - 2 \times t_{C \rightarrow r_1^-}$$

$$h'_{r_1^-} = h_C - 2 \times t_{C \rightarrow r_2^-} - 2\omega$$

$$\text{et } h'_{r_2^-} = h_C - 2 \times t_{C \rightarrow r_2^-}$$

$$\text{enfin } h'_{r_1^-} < h'_{r_2^-} \quad (4.4)$$

De cette manière, les horaires h_{r^-} se retrouvent inversés par rapport à l'horaire de départ h_C :

$$h_{r^-} \xrightarrow{f} h'_{r^-} | h_{r^-} > h_C, h'_{r^-} < h_C$$

Une fois cette inversion réalisée, il reste à mettre à place les transitivités (fig. 4.6(c)) selon la méthode indiquée en section 4.2.1.

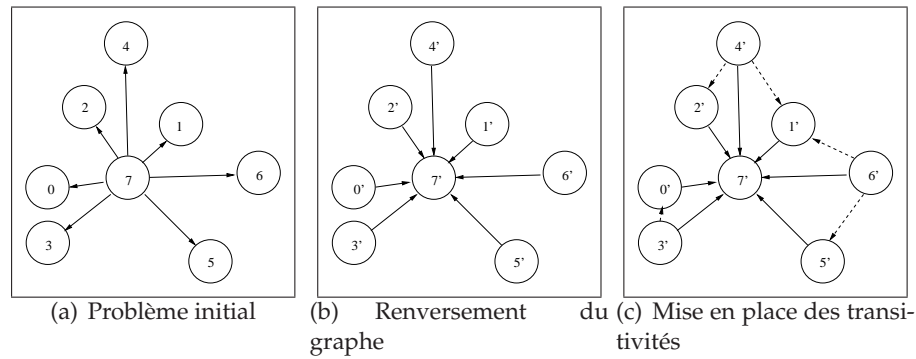


FIG. 4.6: Un exemple de monodivergence : problème initial (a), inversion du problème (b) et mise en place des transitivités (c).

Une fois que nous disposons d'un graphe en monoconvergence, nous pouvons chercher des ACT répondant au problème correspondant. Les solutions à la monodivergence s'obtiennent ensuite par inversion de la séquence. Ainsi l'ACT de la figure 4.7(a)

dispose de quatre branches B_1, B_2, B_3, B_4 qui, inversées, donnent les séquences des chemins des véhicules (fig. 4.7(b)) :

- $B_1 = (4', 2', 7')$ correspond au chemin $S_1 = (7, 2, 4)$;
- $B_2 = (3', 0', 7')$ correspond au chemin $S_2 = (7, 0, 3)$;
- $B_3 = (1', 7')$ correspond au chemin $S_3 = (7, 1)$;
- $B_4 = (6', 5', 7')$ correspond au chemin $S_4 = (7, 5, 6)$.

À l'instar de la convergence en anneau, notons que cette méthode fonctionne également avec un service en divergence dont le point de départ est un anneau.

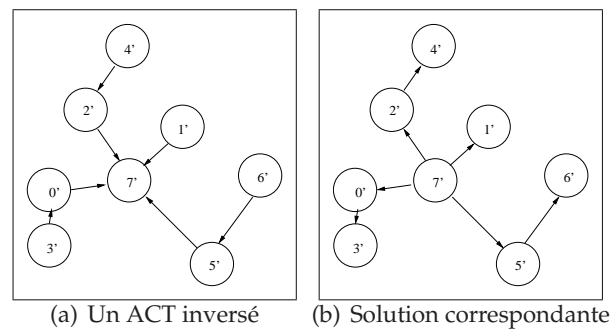


FIG. 4.7: Une solution à la monodivergence : un ACT inversé (a) et la solution correspondante (b).

4.2.5 Discussion

La méthode exposée dans cette section est née du principe même de polarité et de centralité (contrainte ou non), dont l'une des manifestations réside dans la desserte d'un lieu particulier (par exemple desservir la gare de Besançon en horaires de frange (Canalda *et al.*, 2004) dans le cas du TAD Evolis-Gare). Elle constitue de ce fait un modèle simple et réduit du fonctionnement d'un TAD optimisé en monoconvergence.

Le champs d'application de la méthode est cependant trop restreint et se limite à une desserte unique. La desserte de deux lieux distincts nécessiterait en effet deux services, ce qui serait évidemment absurde, d'où la nécessité de généraliser le concept de convergence à un ensemble de lieux à desservir pour mutualiser les services en un seul.

4.3 De multiples convergences

Nous étudions maintenant le cas généralisé, où plusieurs destinations sont possibles et ne sont pas imposées. Pour construire un graphe, le processus est différent de celui employé dans le cas de la monoconvergence. Même s'il demeure orienté, le graphe généré n'est plus acyclique, impliquant en cela l'impossibilité d'utiliser les ACT.

Comme nous souhaitons faire face à des problèmes de taille croissante, il faut considérer le problème à différents niveaux qui sont autant d'échelles distinctes. L'étape préliminaire à l'optimisation du DARP est une construction de graphe selon une approche multiscalaire. Celle-ci utilise les unités spatiales (US), point ou zone (ou *aire* : agrégation de points), décrites en section 3.2.2.

À travers l'IRIS ou la commune, utiliser ces US nous permet d'agréger les demandes de transport selon leurs US d'origines et de destinations, et d'obtenir ainsi un premier niveau agrégé de représentation. Les US ainsi connectées forment un ensemble de sous-graphes dont la forme de chacun rappelle celle d'un graphe de convergence. Le nœud terminal d'un tel graphe correspond à l'US de destination commune aux demandes de transport dudit graphe.

À ce niveau, nous pouvons dissocier deux cas de multiconvergence. Dans le premier cas, nous assurons la desserte de plusieurs points de convergence connus et fixés à l'avance. C'est le cas de l'application « Mod'Gen » (section 7.4.2) qui est une application de multiconvergence *explicite*. Une méthodologie simpliste consisterait à voir dans ce problème plusieurs sous-problèmes de monoconvergence, que l'on pourrait traiter respectivement avec des ACT. Certes, cette méthode serait fonctionnelle, mais ne serait en aucune manière efficace, ne serait-ce qu'en terme de regroupement, puisque les combinaisons des ACT de chaque graphe ne seraient pas évaluées. En effet, si on résoud chaque problème indépendamment, on construit pour chacun un ACT dont l'antiracine est une destination. Or une branche d'un ACT pour une desserte C_1 pourrait éventuellement être mutualisée avec une branche d'un ACT pour une desserte C_2 , en supposant que les horaires soient compatibles, de telle sorte que le transport des passagers vers C_1 et des autres vers C_2 ne nécessite qu'un véhicule.

Dans le second cas, la multiconvergence est *implicite* et nous amène à identifier les pôles attracteurs du territoire, et ainsi à dégager une structure de flux caractérisant la polarité formalisée dans le modèle gravitaire, sur lequel nous nous appuyons pour générer des instances de simulations (chap. 7). Notre hypothèse est en effet qu'à différents niveaux d'échelles peuvent apparaître des structures de flux (corridors) utiles pour disposer de graphes pertinents pour l'optimisation.

Une fois les différents pôles mis en évidence, nous cherchons les détours possibles, sous relaxation, pour interconnecter les sous-graphes les uns avec les autres, et établir de cette manière la super-structure de graphe que nous appelons « *métagraphe* ».

Nous suivons maintenant en détails et pas à pas la construction d'un métagraphe. Préalablement, nous disposons des données suivantes :

- les requêtes de transport r ($r \in R$, R est l'ensemble des requêtes) sont définies comme suit :
 - un point de départ x^+ et un point d'arrivée x^- ;
 - les horaires théoriques de montée et de descente h_{r^+}, h_{r^-} ;
 - le nombre de clients w_r à transporter de x^+ à x^- .
- la matrice origines/destinations (OD) des temps de trajet entre les points du territoire desservi (M^T) ;
- la matrice OD des temps de trajet des US du territoire desservi (M^{TU}) ;
- un coefficient de relaxation k_r appliqué aux durées de parcours $t_{x^* \rightarrow y^*}$ des points x^* vers les points y^* ($x^* = \{x^+ | x^-\}$) :

$$t_{x^* \rightarrow y^*} = k_r \cdot M_{x^* y^*}^T$$

Le même coefficient k_r s'applique également aux durées de parcours des US α^* aux US β^* :

$$t_{\alpha^* \rightarrow \beta^*} = k_r \cdot M_{\alpha^* \beta^*}^{TU}$$

4.3.1 Méthode d'agrégation

La première phase de la méthode consiste à dresser les graphes d'US et à trouver les détours possibles en exploitant la matrice M^{TU} et le coefficient de relaxation k_r appliqué aux temps de parcours. En jouant ainsi sur l'élasticité des temps, les détours sont possibles pour atteindre et desservir une zone supplémentaire. Les étapes de cette phase sont les suivantes :

1. à partir de l'ensemble des requêtes, trouver celles ayant une US commune de destination et extraire les n sous-graphes de convergence $\Gamma_i(V_i, E_i)$, où V_i est l'ensemble des sommets de Γ_i , E_i est l'ensemble des arcs, $i \leq n$;
2. chercher les flux pouvant être détournés sous réserve de la relaxation ;
3. pour chaque sous-graphe $\Gamma_i(V_i, E_i)$, construire un graphe de convergence $G_i(V_i, E_i)$.

Identification des structures de flux

Cette étape consiste à identifier les requêtes x ayant une même aire de destination A :

- le point de descente x^- est situé dans l'aire A : $x^- \in A$;
- le point de montée x^+ se trouve dans une aire B_j : $x^+ \in B_j$;

Finalement, nous avons un graphe de convergence orienté acyclique $G_i(V_i, E_i)$, dont le nœud central correspond à l'aire A et les autres sommets sont associés aux différentes aires B_j :

$$\forall j \leq |V_i|, \{B_j, A\} \subset V_i$$

La figure 4.8(a) décrit un exemple de problème de transport. Chacune des flèches représente une requête (de son origine à sa destination). En regroupant par zones d'émission et de destination, nous obtenons une super-structure de flux qui est le prémiss du méta-graphe (cf. figure 4.8(b)). Sont identifiés deux sous-graphes Γ_1, Γ_2 , dont les nœuds centraux sont respectivement C_1, C_2 et les nœuds connectés sont, respectivement, 1,2,4 et 0,3,4. Le sommet 4 a un double statut, car il appartient à la fois au sous-graphe Γ_1 et au sous-graphe Γ_2 .

Recherche des détours

Nous examinons les flux du sous-graphe Γ_i pouvant être détournés (fig. 4.9) sous contrainte de relaxation k_r . Un flux dirigé d'une zone α vers une zone β peut être détourné vers le centre γ du sous-graphe Γ_j si et seulement si

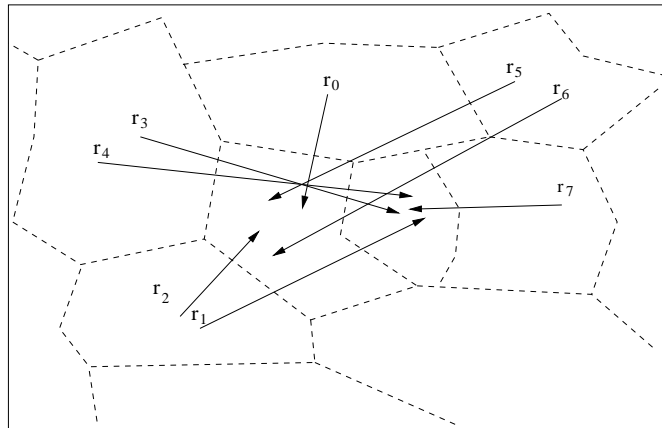
$$t_{\alpha \rightarrow \gamma} + t_{\gamma \rightarrow \beta} \leq k_r \cdot t_{\alpha \rightarrow \beta}$$

Dans ce cas, nous connectons γ au sous-graphe Γ_i :

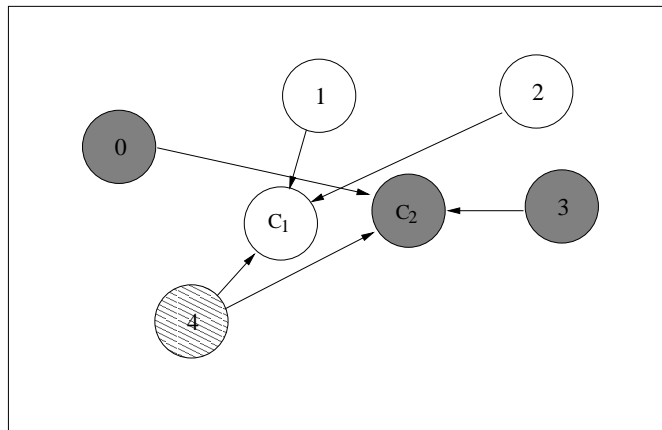
$$V_i = V_i \cup \gamma$$

Le nœud de détour appartenant au sous-graphe Γ_j est ajouté au sous-graphe Γ_i . De cette manière et ainsi de suite, tous les sous-graphes sont, potentiellement, interconnectés les uns aux autres.

Donc, les flux démarrant du nœud 0 peuvent faire un détour par C_1 et ceux démarrant en 2 peuvent passer par C_2 (cf. fig. 4.9, en pointillés longs). Les sous-graphes Γ_1, Γ_2 sont ainsi chacun étendus d'un nouveau sommet. La dernière opération à effectuer pour compléter le méta-graphe consiste à connecter les aires sous réserve des



(a) Problème initial



(b) Regroupement

FIG. 4.8: Un problème de transport multiconvergent : (a) chaque flèche indique une demande de service (origine vers destination). Le découpage territorial apparaît en pointillés. (b) Regroupement et agrégation des requêtes : les requêtes sont groupées par zones d'émission et de destination, et forment ainsi le squelette du métagraphe.

contraintes temporelles (cf. fig. 4.9 en pointillés petits). Dans un graphe $\Gamma_i(V_i, E_i)$ de centre γ , les US α et β sont connectées si et seulement si :

$$t_{\alpha \rightarrow \beta} + t_{\beta \rightarrow \gamma} \leq k_r \cdot t_{\alpha \rightarrow \gamma} \Rightarrow \exists \langle \alpha \beta \rangle \in E_i$$

Tri des sommets

Pour chaque sous-graphe Γ_i , toutes les origines et destinations des requêtes sont ordonnées selon leurs horaires de passage. Les connexions entre chacun de ces points

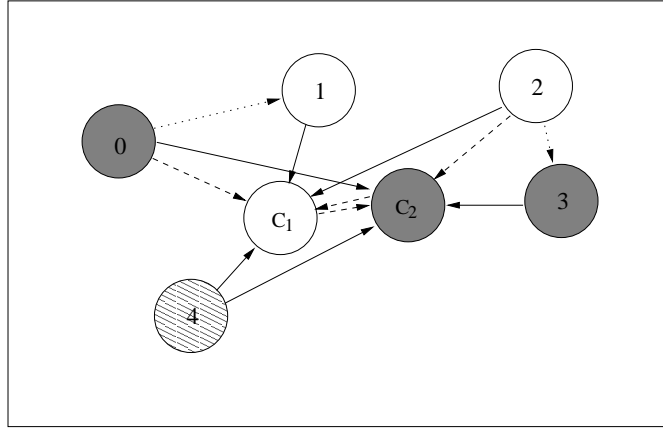


FIG. 4.9: Deuxième étape : réalisation possible des détours

sont effectuées seulement ensuite, et sous contraintes temporelles. Ainsi, le point le plus éloigné (départ le plus tôt) est également un nœud minimal du graphe de convergence et le dernier point correspond au point de convergence, qui est un nœud terminal du graphe de convergence.

Tout comme il y a des nœuds minimaux dans les graphes de convergence, nous pouvons parler de requêtes minimales, dont les points de départ n'ont pas de prédécesseurs.

Ainsi, avec deux requêtes $r_1, r_2 \in R$, respectivement définies avec leurs horaires de départ $h_{r_1^+}, h_{r_2^+}$, nous notons $r_1 \leq r_2$ si et seulement si la requête r_1 démarre avant la requête r_2 :

$$\forall r_1, r_2 \in R, h_{r_1^+} \leq h_{r_2^+} \iff r_1 \leq r_2$$

4.3.2 Construction du graphe

Une fois les sommets triés et liés dans chaque sous-graphe Γ_i , nous cherchons les connexions possibles entre les sommets pour établir de nouveaux arcs dans le graphe $G_i(V, E)$. Les sommets sont reliés selon des conditions temporelles très proches de celles précédemment énoncées, mais avec cette fois en plus les fenêtres de temps, qui sont proportionnelles aux durées des trajets.

Nous définissons le temps de parcours du point x^* vers le point y^* , sujet à la relaxation k_r :

$$\forall x^*, y^* \in V, t'_{x^* \rightarrow y^*} = k_r \cdot t_{x^* \rightarrow y^*}, t_{x^* \rightarrow y^*} = M_{x^* y^*}^T$$

La fenêtre de temps (*time window* dans la littérature) tw en un point correspond à une marge temporelle permettant des regroupements, ou à une marge de sécurité. Nous définissons sa durée comme relative au temps de parcours nécessaire pour aller du point x au point y , donc la fenêtre de temps tw_{xy} après avoir parcouru la distance $x \rightarrow y$ vaut :

$$tw_{xy} = k_w \cdot t'_{x \rightarrow y}$$

k_w est un coefficient à calibrer. Si $k_w = 0.2$, alors la fenêtre de temps créée vaut 20% du temps de parcours.

Les sommets x^*, y^* (avec les horaires respectifs h_{x^*} et h_{y^*}) sont connectés si et seulement si la condition suivante est respectée :

$$\forall x^*, y^* \in V, h_{x^*} + t_{x^* \rightarrow y^*} \leq h_{y^*} + tw_{x^* y^*} \Rightarrow \exists \langle x^* y^* \rangle \in E$$

Pour terminer la construction, il faut lier les points du problème initial selon cette dernière condition et nous obtenons le graphe de la figure 4.10.

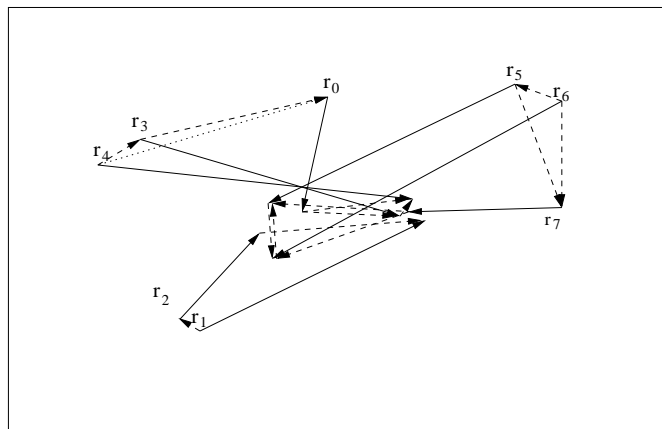


FIG. 4.10: Liaison des points sous contraintes temporelles

4.3.3 Construction des listes de successeurs

À partir du graphe de la figure 4.10 issu du problème initial (fig. 4.8(a)), nous trions les requêtes r_i ($r_i \in R, i \in \mathbb{N}$) selon leurs horaires de départ h_{r_i} et nous obtenons les relations suivantes :

Rang	Requête	Liste des successeurs
0	4	$r'_{0+} : r'_{3+} \rightarrow r'_{4+} \rightarrow r'_{4-} \rightarrow r'_{3-} \rightarrow r'_{0-} \rightarrow /$ $r'_{0-} : /$
1	1	$r'_{1+} : r'_{7+} \rightarrow r'_{7-} \rightarrow r'_{1-} \rightarrow /$ $r'_{1-} : /$
2	6	$r'_{2+} : r'_{6+} \rightarrow r'_{5+} \rightarrow r'_{6-} \rightarrow r'_{5-} \rightarrow r'_{2-} \rightarrow /$ $r'_{2-} : /$
3	3	$r'_{3+} : r'_{4+} \rightarrow r'_{4-} \rightarrow r'_{3-} \rightarrow r'_{0-} \rightarrow /$ $r'_{3-} : r'_{0-} \rightarrow /$
4	0	$r'_{4+} : r'_{4-} \rightarrow r'_{3-} \rightarrow r'_{0-} \rightarrow /$ $r'_{4-} : r'_{3-} \rightarrow r'_{0-} \rightarrow /$
5	5	$r'_{5+} : r'_{6+} \rightarrow r'_{6-} \rightarrow r'_{5-} \rightarrow r'_{2-} \rightarrow /$ $r'_{5-} : r'_{2-} \rightarrow /$
6	7	$r'_{6+} : r'_{6-} \rightarrow r'_{5-} \rightarrow r'_{2-} \rightarrow /$ $r'_{6-} : r'_{5-} \rightarrow r'_{2-} \rightarrow /$
7	2	$r'_{7+} : r'_{7-} \rightarrow r'_{1-} \rightarrow /$ $r'_{7-} : r'_{1-} \rightarrow /$

TAB. 4.1: Table de correspondance des nœuds du graphe ordonnés et listes des successeurs de chaque point de montée et de desserte indexé par rang.

$$\forall i, 0 \leq i \leq 7, r_i \in R \quad (4.5)$$

$$h_{4+} < h_{1+} < h_{3+} < h_{6+} < h_{0+} < h_{5+} < h_{7+} < h_{2+} \quad (4.6)$$

$$\iff r_4 < r_1 < r_3 < r_6 < r_0 < r_5 < r_7 < r_2 \quad (4.7)$$

Cependant, même si $h_{3+} < h_{6+}$, il n'y a pas de détour possible vers r_{6+} , ce qui rend ce point minimal, i.e. r_6 est une requête minimale. L'ordre final des requêtes est donc le suivant :

$$r_4 < r_1 < r_6 < r_3 < r_0 < r_5 < r_7 < r_2$$

De plus, afin de prendre en compte l'ordre des nœuds et les minimalités, nous utilisons de nouveaux nœuds correspondant aux points de passage des requêtes ordonnés dans le tableau 4.1. La requête r_4 , première à être traitée, devient r'_0 (requête de rang 0) et ainsi de suite les requêtes sont renumérotées selon leur rang. Ainsi la dernière requête à être traitée, r_2 , devient r'_7 . Puis, pour chaque r'_{i^*} nous cherchons ses successeurs r'_{j^*} .

4.3.4 Construction de la matrice des temps réduite

La matrice des temps de parcours réduite aux points considérés (${}^r M^T$) est construite en considérant tous les successeurs possibles de chaque point. Les temps indiqués dans cette matrice correspondent à ceux de la matrice initiale modifiés par le coefficient de relaxation k_r .

Posons préalablement ces variables :

- r' correspond à la requête r une fois le tri effectué. Par exemple, la requête r_4 se situe au rang 0 et est notée *in fine* r'_0 (cf. tab 4.1) ;
- a, b sont les indices respectifs dans la matrice M^T des points notés i^*, j^* ;
- u, v sont les indices dans la matrice ${}^r M^T$ des points correspondants aux indices a, b de la matrice initiale M^T .

Les règles de construction de la matrice ${}^r M^T$ sont les suivantes :

$${}^r M^T_{uv} = \begin{cases} k_r \cdot M^T_{ab} & \text{si } \exists \langle r'_i, r'_j \rangle \\ -1 & \text{sinon.} \end{cases}$$

Dès que la matrice réduite est complétée, nous pouvons procéder à la seconde grande étape de la méthode d'optimisation consistant à utiliser un algorithme génétique pour produire des solutions efficaces telles que celle de la figure 4.11.

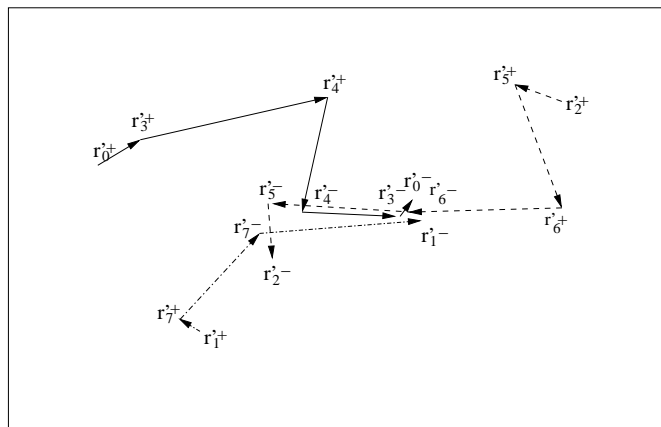


FIG. 4.11: Une solution au problème de la figure 4.8(a) : trois véhicules sont nécessaires pour prendre en charge tous les clients.

4.3.5 Méthode sans agrégat

Nous pouvons évidemment construire un graphe sans utiliser d'étape préalable d'agrégation dans des sous-graphes. Ainsi, il suffit de connecter les points du graphe selon les conditions exposées dans la sous-section 4.3.2. De cette manière, le graphe construit est plus connectif (cf. déf. annexe A.2.2) que ceux construits sur la base d'agrégats.

La mise au point des listes de successeurs et de la matrice des temps réduite demeure inchangée.

4.3.6 Discussion

La bonne marche de la méthode de création de graphe en multiconvergence est tributaire du bon positionnement des centres des US utilisées pour agréger les flux. En effet, selon les positions géographiques données aux centres des US, ce sont tous les détours US à US qui peuvent être remis en question, et de fait modifier la connectivité du graphe et donc les solutions potentielles construites sur celui-ci.

Ainsi la méthode dépend du découpage zonal et donc de la taille et de la forme des US. Son intérêt majeur est, d'une part, une simplification de la structure des graphes (agrégation des requêtes par zone) qui induit une plus grande rapidité de calcul, et d'autre part, une prise en compte à un niveau d'échelle géographique plus petit, de la structure territoriale (flux, zones polarisantes, densités de population). Nous verrons dans le dernier chapitre que, selon les cas, la prise en compte des arrêts, des IRIS ou des communes permet d'obtenir la solution la plus optimale.

4.4 Conclusion

L'introduction de la convergence facilite l'élaboration des graphes des dessertes, qui constitue l'étape préalable de l'optimisation. Avec une destination unique et un horaire fixe d'arrivée, il devient aisé de construire un DAG et de définir rétroactivement les horaires de prises en charge de chacun des clients. De plus, cette construction permet de définir quelques uns des points de départ obligatoires des différentes tournées.

Quant à la construction du graphe dans le cadre de la multiconvergence, celle-ci introduit le concept de granularité spatiale, pour lequel nous utilisons les niveaux d'échelle suivants : arrêts, IRIS et communes. Ces deux derniers sont des agrégats, à

partir desquels nous pouvons élaborer le métagraphe, tandis que les arrêts constituent la plus grande échelle de représentation du TAD.

Le territoire est ainsi représenté de deux façons : par la structure du graphe décrivant les flux sous-jacents, et par la granularité spatiale qui intègre des découpages administratifs et niveaux d'échelles différenciés.

Ces mêmes arrêts peuvent être connectés directement sans étape préalable d'agrégation pour constituer un graphe plus connectif, alors que le métagraphe permet de simplifier le graphe en prenant en compte la structure territoriale.