

## TABLE DES MATIÈRES

	Page
INTRODUCTION .....	1
CHAPITRE 1 ÉTAT DES CONNAISSANCES.....	3
1.1 Mobilité urbaine.....	3
1.1.1 Histoire des transports de l'après-guerre à de nos jours .....	3
1.1.2 Limites et problèmes des transports en commun et individuels .....	7
1.1.3 La situation canadienne et montréalaise .....	9
1.2 Nouvelles approches de la mobilité .....	11
1.2.1 La naissance des transports cybernétiques (ou PRT).....	11
1.2.2 L'héritage des PRT : les projets actuels et futurs .....	13
1.3 Modèle de conception .....	15
1.3.1 Matrice Origine-Destination .....	15
1.3.2 Algorithme de Dijkstra .....	16
1.3.3 Algorithme de colonie de fourmis .....	16
CHAPITRE 2 MÉTHODOLOGIE.....	21
2.1 Modèle dynamique.....	21
2.1.1 Structure du modèle dynamique des déplacements individuels en milieu urbain .....	21
2.1.2 Algorithmes.....	24
2.1.3 Matrice Origine Destination (Matrice OD).....	27
2.2 Paramètres, variables et critères de performance associés.....	29
2.2.1 Structure du réseau.....	30
2.2.2 Paramètres des scénarios de simulation .....	30
2.2.3 Indicateurs de performance.....	31
2.3 Scénarios .....	33
2.3.1 Profil des demandes .....	33
2.3.2 Analyse de la performance des modes opératoires de la flotte de véhicules.....	35
CHAPITRE 3 RÉSULTATS .....	37
3.1 Construction de la matrice des distances .....	37
3.2 Calibrage de l'algorithme de colonie de fourmis.....	39
3.3 Analyse des modes de gestion des véhicules.....	42
3.3.1 Capacité.....	43
3.3.2 Intérêt d'un réseau intelligent .....	45
3.3.3 Comportement du système face à différents types de redistribution .....	47
3.4 Comparaison entre les scénarios .....	51
CHAPITRE 4 DISCUSSION .....	55
4.1 L'influence entre les capsules.....	55
4.2 Les scénarios .....	57

## XII

4.3	Application dans la réalité .....	58
4.3.1	Comparaison entre la capacité d'un PRT et d'une rame de métro .....	58
4.3.2	Possibilité d'application d'un tel système.....	58
4.3.3	Efficacité énergétique .....	61
CONCLUSION.....		63
ANNEXE I	Présentation des algorithmes de utilisés dans le modèle dynamique.....	65
ANNEXE II	Résultats de tous les scénarios .....	81
BIBLIOGRAPHIE.....		87

## LISTE DES TABLEAUX

	Page
Tableau 1.1	Résumé des différents projets de PRT ..... 14
Tableau 1.2	Exemple de matrice OD pour un réseau à 4 nœuds ..... 16
Tableau 2.1	Description des différentes phases du modèle ..... 23
Tableau 2.2	Récapitulatif des différentes variables de l'algorithme de colonie de fourmis ..... 25
Tableau 2.3	Liste des types de redistribution des capsules ..... 26
Tableau 2.4	Résumé des différents algorithmes utilisés et disponibles pour le modèle ..... 27
Tableau 2.5	Liste des paramètres et leurs valeurs ..... 31
Tableau 2.6	Liste des indicateurs de performances ..... 32
Tableau 2.7	Résumé des 6 scénarios étudiés ..... 35
Tableau 2.8	Comparaison de deux algorithmes ..... 36
Tableau 2.9	Résumé des simulations entre les scénarios et les types de redistributions ..... 36
Tableau 3.1	Construction de la matrice des distances ..... 39
Tableau 3.2	Tableau récapitulatif de 10 simulations pour 9 itérations et pour 25 demandes ..... 41
Tableau 3.3	Tableau récapitulatif de 10 simulations pour 10 itérations et pour 25 demandes ..... 41
Tableau 3.4	Tableau récapitulatif de 10 simulations pour 10 itérations et pour 102 demandes ..... 42
Tableau 3.5	Tableau récapitulatif de 10 simulations pour 10 itérations et pour 200 demandes ..... 42
Tableau 3.6	Tableau du temps d'attente moyen en fonction du nombre de demandes (unité : itération) ..... 44

Tableau 3.7	Résumé des résultats de capacité de tous les scénarios pour tous les types de redistributions (en nombre de demandes pour 120 itérations).....51
-------------	--

## LISTE DES FIGURES

	Page
Figure 1.1	Évolution de la croissance démographique et du nombre d'immatriculations de véhicules à moteur au Québec Tirée de Statistiques Canada et SAAQ (2006) .....4
Figure 1.2	Consommation de carburant et densité urbaine Tirée de Newman et Kenworthy (1989) .....6
Figure 1.3	Illustration des calculs de Lowson, vitesse moyenne en fonction de l'espacement des arrêts .....8
Figure 1.4	Moyen de déplacement domicile-travail à Montréal en 2001: Automobile en tant que conducteur Tirée de Statistique Canada (2001).....10
Figure 1.5	Moyen de déplacement domicile-travail à Montréal en 2001: Transport en commun Tirée de Statistique Canada (2001).....10
Figure 1.6	Premier croquis d'un PRT .....12
Figure 1.7	Capsule ULTRa en opération à l'aéroport Heathrow Tirée de ULTRa PRT <a href="http://www.ultraglobalprt.com">www.ultraglobalprt.com</a> .....13
Figure 1.8	Expérience de sélection des branches les plus courtes par une colonie de fourmis: (a) début de l'expérience, (b) fin de l'expérience, Tirée de Dréo et al. (2003, p. 120).....17
Figure 2.1	Schéma bloc du modèle dynamique de gestion des véhicules PRT .....22
Figure 2.2	Schéma simplifié d'une simulation.....22
Figure 2.3	Répartition des origines pour le scénario 4 cas A.....28
Figure 2.4	Répartition des origines pour le scénario 4 cas B .....28
Figure 2.5	Répartition des origines pour le scénario 4 cas C (grand nombre de demandes).....28
Figure 2.6	Schéma du réseau 3x3 régulier .....30
Figure 2.7	Schéma des 6 scénarios étudiés .....34

Figure 3.1	Réseau avec 2 demandes et 5 capsules libres .....	38
Figure 3.2	Moyenne des coûts calculés par l'algorithme en fonction du nombre d'itérations pour 25 demandes .....	40
Figure 3.3	Temps d'attente moyen en fonction du nombre de demandes (illustration de calcul de capacité) .....	44
Figure 3.4	Temps d'attente en fonction du nombre de demandes, comparaison entre 2 types d'attributions (Scénario 1) .....	46
Figure 3.5	Temps d'attente moyen en fonction du nombre de demandes pour deux types de redistribution avec gestion ACO (Scénario 1).....	47
Figure 3.6	Taux d'occupation en fonction du nombre de demandes pour deux types de redistribution avec gestion ACO (Scénario 1) .....	48
Figure 3.7	Temps d'attente moyen en fonction du nombre de demandes pour trois types de redistributions avec gestion ACO (Axes ajustés) (Scénario 2) .....	49
Figure 3.8	Taux d'occupation en fonction du nombre de demandes pour trois types de redistribution avec gestion ACO (Scénario 2) .....	50
Figure 3.9	Représentation du scénario 2 .....	52
Figure 3.11	Temps d'attente moyen en fonction du nombre de demande pour 2 scénarios.....	52
Figure 3.12	Comparaison entre la longueur du trajet moyen et la capacité pour tous les scénarios .....	53
Figure 4.1	Temps d'attente en fonction du nombre de demandes, Scénario 1, redistribution A comparatif avec et sans congestion .....	56
Figure 4.2	Schéma d'un trajet en transport en commun Tirée de Lowson (2004) ) .....	59
Figure 4.3	Calculs de Lowson, vitesse moyenne en fonction de l'espacement des arrêts avec un bus (30km/h).....	60
Figure 4.4	Calculs de Lowson, vitesse moyenne en fonction de l'espacement des arrêts avec un métro (70km/h) .....	61
Figure 4.5	Taux d'occupation en fonction du nombre de demande pour trois types de redistribution avec gestion ACO (Scénario 5).....	62

**Rapport-Gratuit.com**





## INTRODUCTION

Le déplacement au sein des grandes villes est menacé. Que ce soit les problèmes de congestion, de pollution ou de dépendance au pétrole, la mobilité urbaine arrive, telle quelle est connue, à ses limites. La poussée de l'utilisation du transport individuel, perçu premièrement comme une avancée technologique et économique s'est transformée en une aberration sociale et les transports en commun semblent être réservés seulement à une partie des habitants d'une ville.

Les nouvelles technologies ne sont pour l'instant que très peu développées dans le domaine des transports. Alors que la société est de plus en plus servie par les nouvelles technologies, en particulier en termes de communication, ces technologies ne sont que très rarement au service de la communauté et du transport. L'idée d'un réseau intelligent permet de fluidifier les déplacements de tous les citoyens en offrant la possibilité de gérer les véhicules de façon globale.

Un réseau intelligent est un réseau dont ses véhicules peuvent communiquer entre eux. Que se soit les PRT (Personal Rapid Transit) ou des voitures individuelles reliées entre elles (projet Google Car), plusieurs projets de réseaux intelligents voient le jour. Ainsi il est essentiel de comprendre les possibilités, les avantages et les risques d'un tel système. C'est donc en modélisant un tel réseau que ces aspects là d'un nouveau type de transport seront étudiés. L'objectif de ce mémoire est donc d'analyser, par la simulation, les facultés de ces systèmes et ce qu'ils peuvent apporter à la mobilité urbaine.

Le présent document est organisé en quatre principaux chapitres. Le premier chapitre présente la situation de la mobilité, les projets de réseaux intelligents et les algorithmes nécessaires à l'élaboration du modèle dynamique. Le chapitre 2 présente la méthodologie, c'est-à-dire la réflexion de la construction de ce modèle dynamique nécessaire aux simulations, des variables qui le composent et des indicateurs qui permettront l'analyse des

résultats. Le chapitre 3 aborde les résultats obtenus tandis que le chapitre 4 amène une discussion sur ceux-ci avant de présenter les conclusions.

## **CHAPITRE 1**

### **ÉTAT DES CONNAISSANCES**

L'état des connaissances propose d'étudier la question de la mobilité en s'intéressant tout d'abord à l'histoire de notre mobilité de l'après-guerre jusqu'aux projets actuels et de demain.

La première partie s'intéresse donc à l'histoire des transports en termes de mobilité et comment la dépendance à l'automobile s'est installée dans nos sociétés et ses effets, avant de s'attarder à la situation canadienne et montréalaise. La deuxième partie présente les projets et technologies actuels et futurs qui modifient l'approche du transport et favorisent la mobilité des personnes. Enfin la troisième partie introduit l'approche biomimétique et les algorithmes associés qui amènent une dimension écosystémique dans la perspective du développement d'un modèle dynamique de la mobilité individuelle des personnes.

#### **1.1 Mobilité urbaine**

« La motorisation systématique dans de nombreux pays du monde (...) a conduit à des situations apparemment irréversibles. Or aujourd'hui les enjeux énergétiques et le souci croissant d'un développement durable incitent justement à renverser les tendances dans ce domaine. » (Dupuy, 2006)

##### **1.1.1 Histoire des transports de l'après-guerre à de nos jours**

A la fin de la Seconde Guerre mondiale, l'industrie automobile profite de l'essor de l'économie mondiale. Elle se propage aux classes moyennes et devient rapidement le mode de transport privilégié dans tous les pays occidentaux. À titre d'illustration, la Figure 1.1 présente l'évolution des immatriculations de véhicules au Québec comparativement à l'évolution démographique de la province. Alors que la population a quasiment doublé entre 1953 et 2000 (passant de 4 000 000 à 7 500 000 d'habitants), le nombre de véhicules a été

multiplié par 7,2 (passant de 500 000 à 3 600 000). Autrement dit, il y avait un véhicule en circulation pour 8 personnes en 1953 pour atteindre un véhicule pour 2,1 personnes en l'an 2000. Aux États-Unis où le ratio véhicule par nombre d'habitant est le plus élevé au monde, il y a actuellement un véhicule pour 1,3 personne. (Sousanis, 2011).

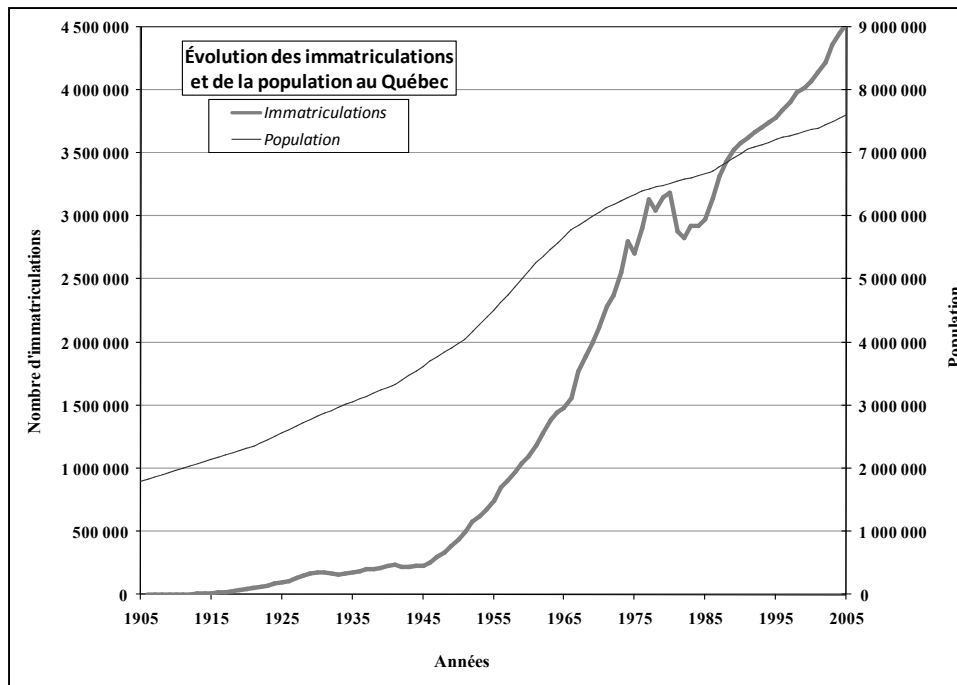


Figure 1.1 Évolution de la croissance démographique et du nombre d'immatriculations de véhicules à moteur au Québec  
Tirée de Statistiques Canada et SAAQ (2006)

La démocratisation de l'automobile et des transports individuels en général a dès lors profondément transformé le paysage urbain ainsi que nos habitudes. À titre d'exemple, les mécanismes mis en place aux États-Unis sont symptomatiques du phénomène d'automobilisation<sup>1</sup> de nos sociétés. En 1962, le Federal-Aid Highway Act alloue 25

---

<sup>1</sup>automobilisation : terme qui se réfère aux activités dépendantes de l'automobile Dupuy, G. 1999. *La dépendance automobile: symptômes, analyses, diagnostic, traitements*. Anthropos.

milliards de dollars à la construction d'autoroutes et va influencer tous les Américains sur leur façon de se déplacer et surtout sur leur façon de penser la planification urbaine (Weiner, 2008). Ces chantiers sont menés au cours d'une période d'abondance du pétrole. La voiture, et plus globalement les transports individuels motorisés, apparaît comme la façon la plus naturelle de se déplacer, entre les villes, mais surtout dans la ville. La planification au sein de la ville va de pair avec cette philosophie du transport. Alors que les villes s'étalent, les problèmes dus à une surconsommation des transports individuels se font ressentir : dislocation entre les zones d'activités et d'habitations, développement urbain, problèmes environnementaux et sociaux (Handy, 1993). La dépendance de nos sociétés au carburant fossile se fait ressentir pendant la crise du pétrole et à partir des années 70 la planification urbaine se concentre sur l'amélioration des infrastructures déjà établies (Weiner, 2008). Cependant la dépendance à l'automobile est déjà réelle et bien établie. Il en est de même dans toutes les villes Européennes, cependant le prix de l'essence et la densité étant plus élevés, les transports en commun sont souvent plus développés que dans les villes américaines. Il en est de même pour les villes asiatiques, traditionnellement plus denses, elles favorisent le transport en commun. La Figure 1.2 présente la consommation annuelle de carburant dans plusieurs villes en fonction de leurs densités. Il apparaît que, globalement, les villes les moins denses sont les plus consommatrices de carburant donc de transports individuels.

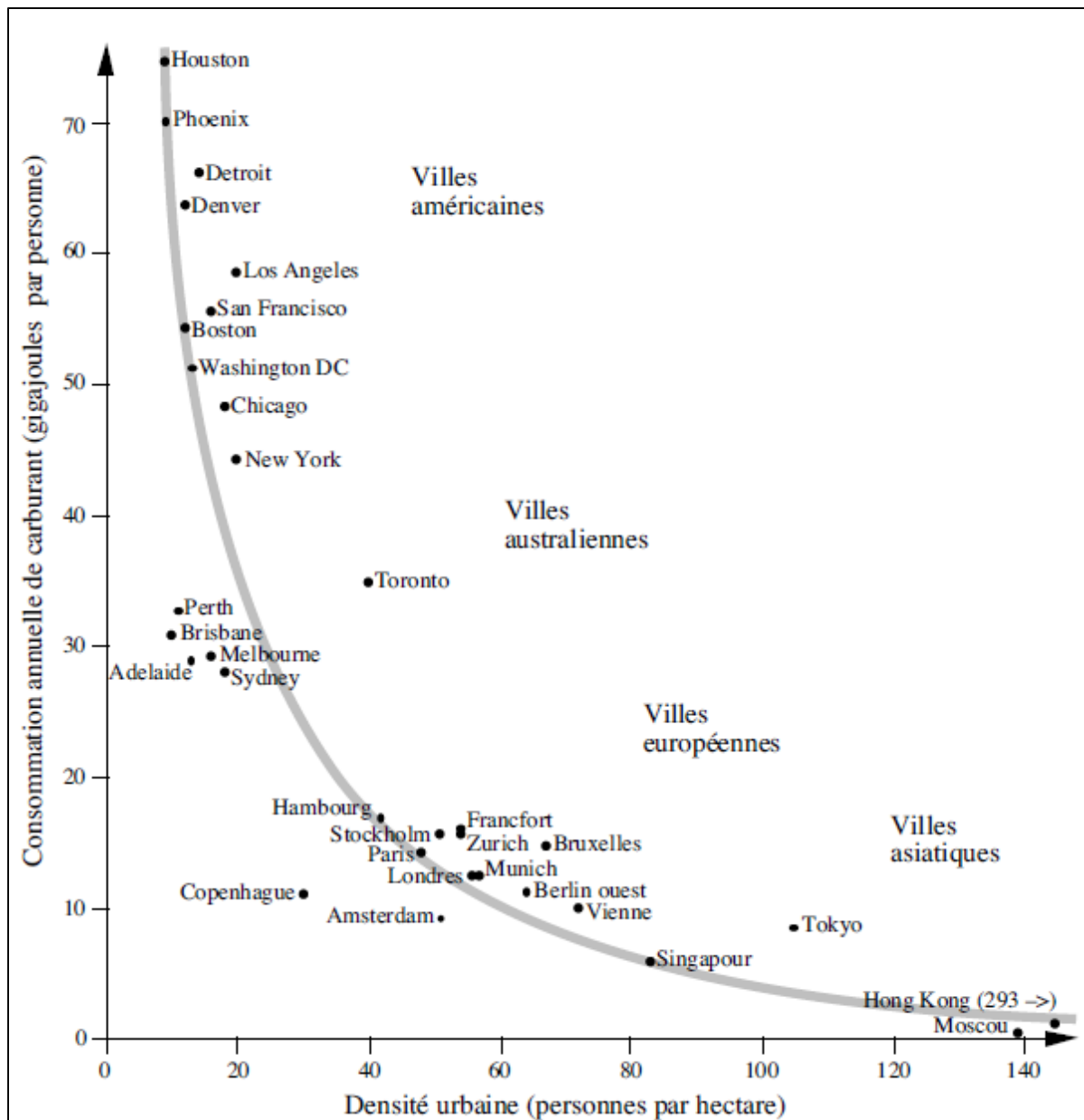


Figure 1.2 Consommation de carburant et densité urbaine  
Tirée de Newman et Kenworthy (1989)

Depuis 40 ans, la planification urbaine des États-Unis n'a guère été modifiée et les seuls efforts en faveur du transport en commun ont été faits sous la pression des crises pétrolières (Weiner, 2008).

Cependant, l'approche du « tous en voiture » engendre des effets néfastes tant sur la qualité de vie des citoyens que sur l'efficacité à déplacer des biens et des personnes. Pire ce qui était perçu comme une avancée technologique et économique (démocratisation de la voiture) s'est transformé en réelle dépendance. L'urbanisation a été telle qu'elle s'est soumise à une planification pour l'automobile et seulement pour l'automobile. Ainsi toutes les villes européennes et américaines disposent de quartiers à faible densité géographiquement éloignés du centre qui ne proposent pas d'alternative à l'automobile. Ces quartiers sont donc clairement dépendants de l'automobile (Turcotte, 2008). Il s'agit de l'effet pervers de l'automobile : la démocratisation de l'automobile a influencé l'urbanisation qui a enfermé les gens dans une dépendance à l'automobile.

### **1.1.2 Limites et problèmes des transports en commun et individuels**

Même si les transports en commun peuvent apparaître comme une solution de substitution les systèmes de transports en commun réguliers présentent cependant des limites. En effet il est démontrable que les transports en commun ont un maximum d'efficacité et donc de capacité (Lowson, 2004). Par ailleurs en décomposant le trajet d'un utilisateur de transport en commun classique en « trois trajets distincts » qui correspondent à un trajet à pied du point de départ à la station, du trajet en transport en commun et enfin du trajet à pied de la station au point d'arrivée il est possible de calculer la vitesse moyenne du trajet global. A la Figure 1.3 la vitesse moyenne du trajet global de 8km est étudiée en fonction de l'espacement des arrêts pour trois vitesses maximales différentes. Il apparaît que peu importe la vitesse maximale du transport considéré le maximum est atteint pour un espacement d'environ 500 mètres. Il apparaît aussi qu'un transport qui a pour vitesse maximale 70 km/h ou 120 km/h a la même vitesse moyenne pour le trajet global. Ainsi il n'est pas utile de viser des transports en commun de plus en plus rapides, car cela est inefficace en termes de vitesses moyennes.

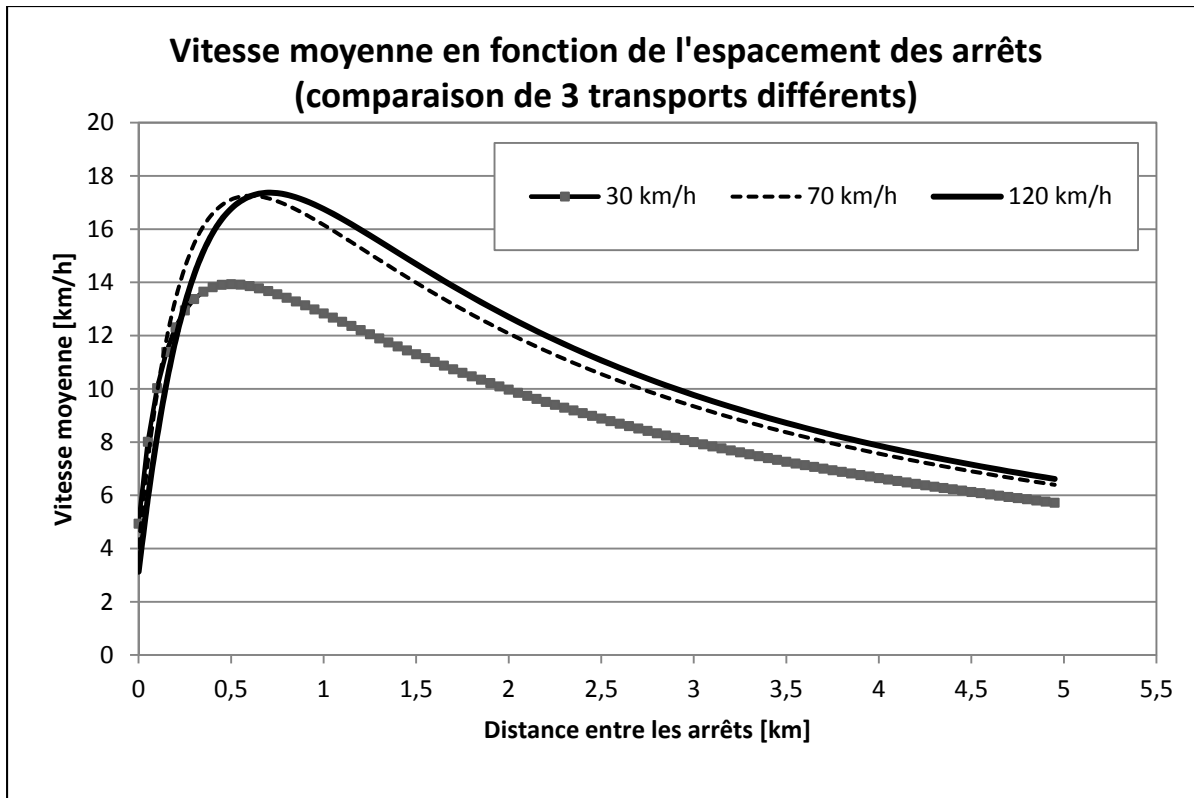


Figure 1.3 Illustration des calculs de Lowson, vitesse moyenne en fonction de l'espacement des arrêts

Ainsi en plus des limites connues des transports en commun il est possible d'énumérer quelques un des principaux problèmes liés à notre système dépendant de la voiture tel que l'augmentation de la congestion, la dépendance au pétrole et les effets néfastes de sa consommation ou les accidents de voiture. (Anderson, 2009)

Tous les transports en commun conventionnels sont des transports de masses (Bus, Métro, Train, Trolleybus) dont la viabilité économique repose sur le remplissage des véhicules. Les transports en commun sont donc soit limités par leurs capacités en zone de haute densité et limités par leur coût en zone de faible densité. Par ailleurs, augmenter la flotte de véhicules d'un transport collectif pour minimiser les temps d'attente et améliorer la qualité du service auprès des utilisateurs paraît difficilement viable économiquement. Dès lors, l'intermodalité,



qui consiste en l'utilisation de plusieurs types de transports pour un trajet, est souvent un choix fait par défaut par les usagers (Litman, 2002). Ainsi il n'y a pour l'instant pas de solution alternative à l'automobile acceptable pour satisfaire le besoin en mobilité en milieu urbain. Il apparaît donc nécessaire de repenser notre mobilité collective. C'est-à-dire que le système tel qu'il est mis en œuvre actuellement (mélange de transport individuel et collectif).

### **1.1.3 La situation canadienne et montréalaise**

Depuis les années 90 une légère augmentation de la fréquentation des transports en commun a été constatée, ce qui ne signifie pas que la proportion d'automobilistes a baissé. Effectivement pendant la même période de temps la distance moyenne des Canadiens pour se rendre au travail a augmenté de 7,0 à 7,2 kilomètres, soit 2,9% alors que les temps moyens sont passés de 54 à 63 minutes, soit une augmentation de 16,6%. Donc une augmentation du trajet moyen de 0.2 kilomètre engendre une augmentation de neuf minutes de trajet en plus (Turcotte, 2005). Cette évolution est globale c'est-à-dire qu'elle concerne autant les automobilistes que les usagers des transports en commun. L'étude montre même que ce sont les transports en commun qui sont le plus touchés par cette augmentation, ainsi les trajets en transport en commun sont plus longs que ceux en voiture, malgré les problèmes de congestion. Cela met bien en évidence le déclin de l'efficacité du système global des transports. Ainsi il est difficile aux transports collectifs d'être attractifs pour les passagers. Il a été vu que plus les villes sont denses plus elles favorisent l'utilisation du transport en commun, mais cela se vérifie aussi à l'échelle de la ville elle-même. Le cas de Montréal est présenté Figure 1.4 et Figure 1.5. Il en est une bonne représentation. Les zones les plus denses auront plus tendance à utiliser les transports en commun alors qu'au contraire les zones éloignées du centre, moins denses, auront tendance à utiliser les transports individuels.

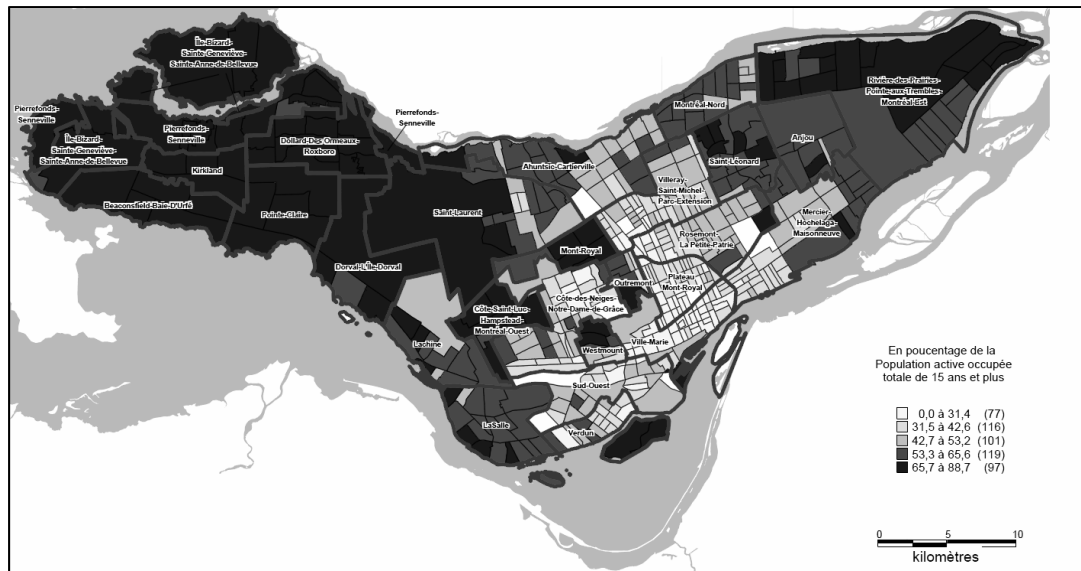


Figure 1.4 Moyen de déplacement domicile-travail à Montréal en 2001:  
Automobile en tant que conducteur  
Tirée de Statistique Canada (2001)

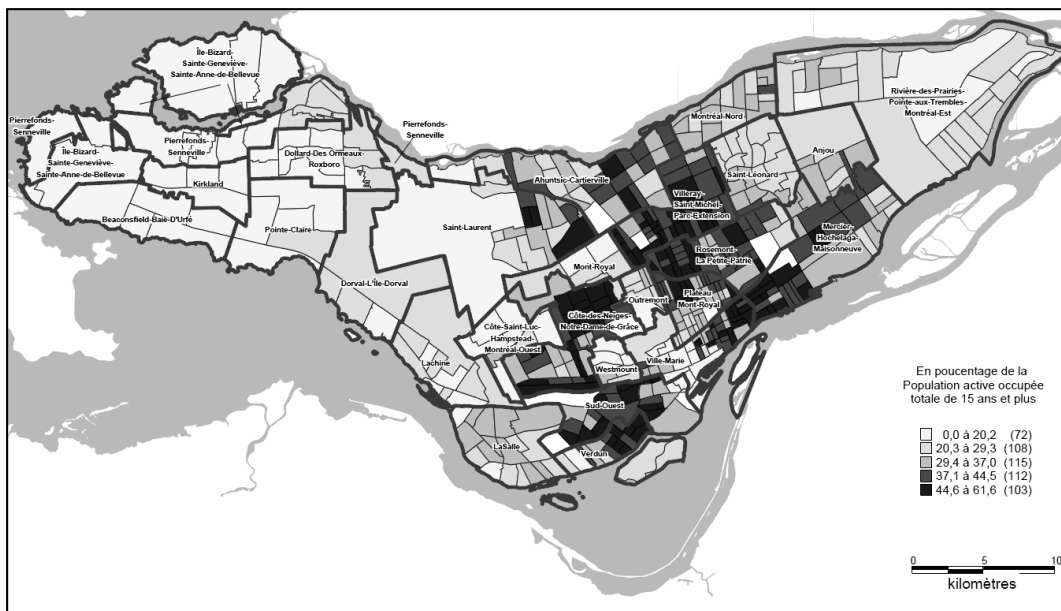


Figure 1.5 Moyen de déplacement domicile-travail à Montréal en 2001:  
Transport en commun  
Tirée de Statistique Canada (2001)

Il y a bien une corrélation entre la densité de la population, les services proposés des transports en commun et l'utilisation de la voiture. Plus la densité de la population est élevée, plus les services proposés en termes de transport collectif sont denses et moins la voiture est utilisée. Alors que, logiquement, un manque de service entraîne une dépendance à la voiture.

## **1.2 Nouvelles approches de la mobilité**

Plusieurs projets se proposent comme des alternatives ou comme complément à notre mode de transportation actuel. En effet le concept des systèmes intégrés permet de redéfinir la mobilité dans le sens large du terme. En particulier les réseaux intelligents qui seront présentés de cette partie

### **1.2.1 La naissance des transports cybernétiques (ou PRT)**

C'est Donn Fichter alors ingénieur en planification des transports pour l'état de New York qui imagine un nouveau type de transport en 1953. Fichter veut développer des transports pour les zones résidentielles à densité faible souvent très mal desservies par les transports en commun conventionnels. Il publie un livre où il propose un système de transport en commun entièrement automatisé (Fichter, 1964). La recherche pour ce genre de système connaît un réel engouement dans les années 60 et 70 avec entre autres les travaux de plusieurs ingénieurs et chercheurs avec notamment l'introduction du terme PRT (Anderson, 1974). Le premier croquis publié est présenté Figure 1.6.

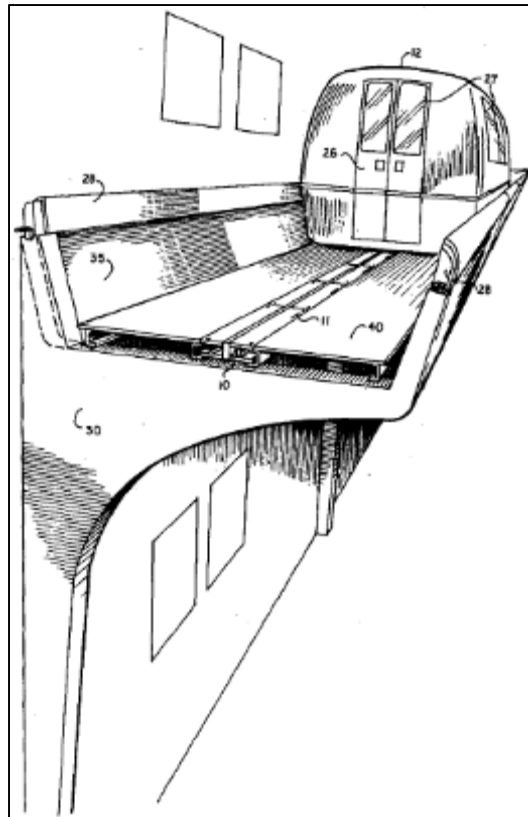


Figure 1.6 Premier croquis d'un PRT  
Tirée de Sawyer (1977)

En Europe les premières recherches commencent dans les années 70 à Paris avec le projet ARAMIS qui se soldera malheureusement par un échec (Latour, 1992). De même aux États-Unis et en Allemagne où tous les chantiers entrepris dans les années 70 et 80 n'ont pu aboutir (Anderson, 2006, p. 25). Pour Anderson (2006) les fautes de ces succès sont :

- une méconnaissance de l'électronique et de l'informatique trop peu développée à l'époque;
- de nombreux problèmes légaux;
- le manque de conscience politique et publique des problèmes environnementaux, qui n'ont pas favorisé ce genre de projet.

### 1.2.2 L'héritage des PRT : les projets actuels et futurs

Cependant, plusieurs projets ont pu voir le jour notamment ULTRa à Londres et 2getthere à Masdar. Le système ULTRa PRT a été installé à l'aéroport Heathrow de Londres et est en opération depuis 2011 (Figure 1.7).



Figure 1.7 Capsule ULTRa en opération à l'aéroport Heathrow  
Tirée de ULTRa PRT [www.ultraglobalprt.com](http://www.ultraglobalprt.com))

En moyenne, ce moyen de transport consomme 0,55MJ par passager par km comparativement à 1,2 à 2,4 pour des transports en commun classiques. Le système s'est montré fiable et 67% des passagers ont jugés la globalité de l'expérience ULTRa comme « excellente » (Lowson, 2011). Cela dit, il existe d'autres projets de PRT à travers le monde (Tableau 1.1).

Tableau 1.1 Résumé des différents projets de PRT

Nom du système	Ville	Nombre de passagers par véhicule	Caractéristiques
ULTRa	Londres, Heathrow, Royaume-Unis	4 à 6	3,8km de rail pour 21 véhicules et 3 stations. Vitesse : 40kmh
2getthere	Masdar, Abu Dhabi	2 à 6	1,5km de rail pour 13 véhicules et 5 stations. Vitesse : 40kmh
Vectus	Suncheon, Corée du Sud	3 à 6	9,8km de rail pour 40 véhicules et 2 stations. Vitesse : 40kmh
ULTRa	Amritsar, Inde	4 à 6	7,0km de rail pour 200 véhicules et 7 stations. Vitesse : 40kmh
Serpentine	Lausanne, Suisse	4	2,6km de rail pour 6 stations

Mais en plus des PRT, l'idée d'un réseau de transport intelligent sur route devient concrète avec les recherches actuelles de Google d'une voiture qui se conduit toute seule. En effet sous la tutelle de Sebastian Thrun le projet d'une voiture sans chauffeur a été développé par l'équipe de recherche de Google. Cette voiture ne nécessite pas de conducteur, la voiture est capable de s'adapter à tous les types de routes grâce à notamment un GPS (Global

Positioning System) intégré (Source : [Google : Official Blog, 2012](#)). Cela apparaît donc aussi comme un pas en avant vers les PRT ou plutôt la possibilité d'un réseau entièrement intelligent où tous ses éléments communiquent entre eux. Les systèmes intégrés permettent donc la gestion globale des véhicules. Ainsi on peut oublier le comportement égoïste des transports individuels ainsi que système rigide des transports en commun actuels.

### **1.3 Modèle de conception**

La conception d'un modèle dynamique capable de simuler des déplacements individuels pose quelques problèmes de programmation. Le premier est l'élaboration des matrices origine-destination qui retranscrivent l'intention des utilisateurs. Deuxièmement ces matrices origine-destination demandent aussi l'identification des itinéraires, ainsi l'algorithme de Dijkstra sera utilisé pour pallier à ce problème. Enfin, la gestion des déplacements individuels d'un réseau intelligent demande de lourds calculs c'est pourquoi l'algorithme de colonie de fourmis est présenté en troisième partie.

#### **1.3.1 Matrice Origine-Destination**

La matrice origine-destination (O-D) détermine les habitudes en déplacements de la part des résidents associés à un territoire (Spiess, 1987). Elle est généralement adaptée mathématiquement à partir d'études d'origine-destination (O-D). Ces études sont menées régulièrement pour analyser les besoins en mobilité des habitants d'une ville. La dernière en date pour Montréal date de 2008 (Source : Agence métropolitaine des transports, 2008). Cette étude consiste à tenir les statistiques des habitudes de mobilités par quartier, par âge, par groupe socioprofessionnel, par période de la journée et bien sûr les modes de transport utilisés. Le Tableau 1.2 présente un exemple d'une matrice OD utilisée dans le modèle.

Tableau 1.2 Exemple de matrice OD pour un réseau à 4 nœuds

Origine	Destination	Nœud A	Nœud B	Nœud C	Nœud D	Total
Nœud A		0	5	1	9	15
Nœud B		2	0	10	4	16
Nœud C		7	2	0	2	11
Nœud D		6	6	1	0	13
Total		15	13	12	15	

A gauche les nœuds origines dans cet exemple il y a donc à partir du nœud A cinq demandes pour aller au nœud B. Il y a donc 15 demandes à partir du nœud A, c'est-à-dire 15 usagers en partance du nœud A vers les autres nœuds du réseau.

### 1.3.2 Algorithme de Dijkstra

L'algorithme de Dijkstra qui porte le nom de son inventeur Edsger Dijkstra est un algorithme qui sert à résoudre le problème du chemin le plus court entre une origine et une destination parmi plusieurs trajets possibles. Cet algorithme est notamment utilisé par les GPS pour, entre autres, calculer le trajet le plus court, le trajet le plus rapide ou le plus économique (Elhaffar et Lehtonen, 2005). En effet l'algorithme peut aussi bien prendre en compte les distances, que le temps ou la consommation de carburant.

### 1.3.3 Algorithme de colonie de fourmis

Le biomimétisme est une démarche scientifique qui consiste à s'inspirer de phénomènes naturels afin de servir la science ou la technologie. Ainsi les algorithmes de colonies de fourmis forment une classe des métaheuristiques (soit un algorithme d'optimisation) récemment proposés pour des problèmes d'optimisation difficile. Ces algorithmes s'inspirent des comportements collectifs de dépôt et de suivi de pistes observés dans les colonies de fourmis. (Dréo et al., 2003, p. 115). En effet les fourmis bien que ne possédant pas d'intelligence individuelle sont capables de trouver le chemin le plus court lorsqu'elles sont



en groupe. Ces insectes dits sociaux peuvent ainsi résoudre des problèmes complexes. C'est grâce aux phéromones que les fourmis peuvent communiquer entre elles et ainsi s'échanger des informations. Les fourmis laissent après elles une trace de phéromones et plus la concentration de phéromones est importante, plus les autres fourmis seront incitées à prendre ce même chemin. Ainsi, après un certain temps, toutes les fourmis, ou presque, auront tendance à converger vers le même tracé qui sera aussi le plus court comme illustré à la Figure 1.8. Les fourmis qui choisissent le chemin le plus court déposeront plus de phéromones sur leur trajet et influenceront ainsi les futures fourmis. Ceci dit, certains individus choisiront toutefois un trajet différent malgré les phéromones et ce choix différent peut aboutir à une nouvelle amélioration du trajet.

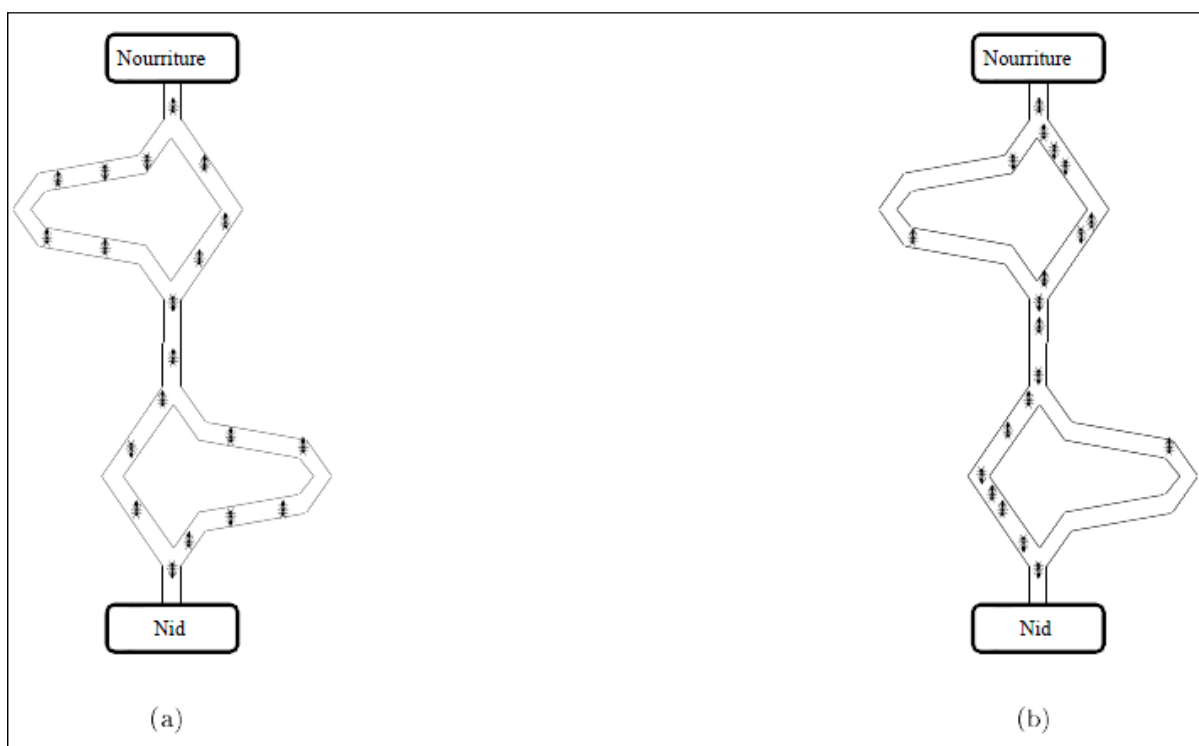


Figure 1.8 Expérience de sélection des branches les plus courtes par une colonie de fourmis: (a) début de l'expérience, (b) fin de l'expérience, Tirée de Dréo et al. (2003, p. 120)

L'algorithme original a été utilisé pour la recherche de chemins optimaux notamment dans le problème du voyageur de commerce (Coloni, Dorigo et Maniezzo, 1991) puis il s'est diversifié pour résoudre toutes sortes de problèmes d'optimisation (Dréo et al., 2003).

Le problème du voyageur de commerce consiste à rejoindre un nombre  $N$  de villes par le chemin le plus court. La résolution exacte de ce problème est de complexité  $N!$ , il est donc obligatoire pour résoudre ce genre de problème d'utiliser un algorithme d'optimisation. Le but n'étant pas de trouver la solution exacte mais de s'en approcher le plus possible en un temps minimal. Voici donc le fonctionnement de l'algorithme

Soit :

$N$ : nombres de villes

$m$ : nombres de fourmis

A chaque itération, chaque fourmi va faire un tour de toutes les villes. La première ville est choisie au hasard. Puis d'une ville  $i$  à une ville  $j$  la fourmi sera influencée par :

- la liste des villes déjà visitées, sachant qu'une fourmi ne retourne pas sur ses pas;
- la distance entre les villes ou plus exactement la *visibilité*  $\eta$  qui correspond à l'inverse de la distance :

$$\eta_{ij} = \frac{1}{d_{ij}} \quad (1.1)$$

- la quantité de phéromones déposée, autrement dit l'intensité de la piste  $\tau$  qui évoluera à chaque itération.

Ainsi toutes les villes qui n'ont pas été visitées peuvent être candidates. Afin que la fourmi « fasse un choix », à toutes les villes candidates est attribué un coefficient selon la formule (1.2)

Soit, à l'itération  $t$  la fourmi  $k$  allant des villes  $i$  à  $j$  et soit  $J_i^k$  l'ensemble des villes « candidates » de la fourmi  $k$  :

$$p_{ij}^k(t) = \frac{(\tau_{ij}(t))^\alpha \cdot (\eta_{ij})^\beta}{\sum_{l \in J_i^k} (\tau_{il}(t))^\alpha \cdot (\eta_{il})^\beta} \quad (1.2)$$

Une fois ce coefficient attribué à toutes les villes, la prochaine ville est choisie aléatoirement de façon pondérée. L'introduction de deux nouveaux coefficients  $\alpha$  et  $\beta$  dans la formule permet de favoriser respectivement soit l'importance de la visibilité, soit l'intensité. Les valeurs de  $\alpha$  et  $\beta$  sont donc importantes. Pour  $\alpha = 0$ , la visibilité n'est pas prise en considération tandis que pour  $\beta = 0$ , seules les pistes de phéromones seront importantes dans le modèle.

Toutes les fourmis ayant fini leur tour déposent une quantité de phéromones sur leurs parcours suivant la formule (1.3):

$$f = L - \epsilon L \cdot \min(L) \quad (1.3)$$

Avec :

$f$  : le coût ajusté de chaque trajet

$L(t)$  : l'ensemble des longueurs  $L_k$  à l'itération  $t$  (c'est donc un ensemble de  $m$  éléments) où  $L_k(t)$  représente la longueur parcourue par une fourmi à l'itération  $t$

$\epsilon$  : coefficient d'ajustement

puis :

$$\Delta\tau_k = \frac{1}{f_k} \quad (1.4)$$

La formule suivante permet d'ajuster la quantité de phéromones entre chaque ville :

$$\tau_{ij}(t+1) = (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau_k \quad (1.5)$$

Il y a apparition de la constante  $\rho$  qui est le coefficient d'évaporation, elle permet d'« évaporer » les phéromones et ainsi d'oublier les mauvaises solutions.

Il est donc nécessaire de paramétrer ces variables :

$\alpha$  : coefficient d'intensité de phéromone

$\beta$  : coefficient de visibilité

$m$  : nombre de fourmis

$\alpha$  : coefficient d'ajustement

$\rho$  : coefficient d'évaporation

Toutes les valeurs suivantes sont tirées directement de la littérature. La littérature conseille un nombre de fourmis égale au nombre de ville ( $N = m$ ) (Dréo et al., 2003, p. 126),  $\alpha = 1$ ,  $\beta = 8$ ,  $\rho = 0,01$ ,  $\alpha = 0,96$  (Khichane, Albert et Solnon, 2009, p. 6).

Ainsi l'algorithme de colonie de fourmis permet, en un temps limité, de trouver une solution optimale non pas pour un usager mais pour l'ensemble des usagers du système. Cela permet de voir les demandes comme une matrice OD auquel le système se doit de répondre..

## CHAPITRE 2

### MÉTHODOLOGIE

Ce deuxième chapitre aborde les approches méthodologiques à la base du développement du modèle mathématique permettant la simulation d'un réseau intelligent dédiée à l'opération des véhicules PRT. La première partie présente la structure du modèle qui permet la simulation de déplacements de véhicules dans un réseau. Une fois le modèle présenté la deuxième partie présente les paramètres qui le composent ainsi que les indicateurs de performances à relever qui serviront aux analyses. Enfin le modèle sera étudié au travers de plusieurs scénarios qui seront exposés en troisième partie.

#### 2.1 Modèle dynamique

Le modèle doit répondre à certaines spécifications de base qu'il est important d'explicitier ci-après. Le raisonnement ainsi que la méthodologie permettant au modèle de réaliser les attentes du cahier des charges sera ainsi développé.

##### 2.1.1 Structure du modèle dynamique des déplacements individuels en milieu urbain

Afin de s'intéresser aux déplacements et aux comportements de capsules au sein d'un réseau il faut établir un modèle. Ce modèle doit répondre à certaines exigences qui seront établies ci-après. Le modèle doit être capable de simuler les déplacements et la gestion de véhicules PRT dans un réseau défini tout en répondant au besoin de mobilité exprimé sous la forme d'une matrice Origine Destination (Matrice OD) tel qu'exprimé à la Figure 2.1. La structure du modèle permet ainsi de simuler par itérations l'état du système modélisé. La simulation se définit par une phase d'initiation suivie d'une boucle de  $n$  itérations tel qu'illustré à la Figure 2.2. Le modèle est développé sous Matlab (R2010b).

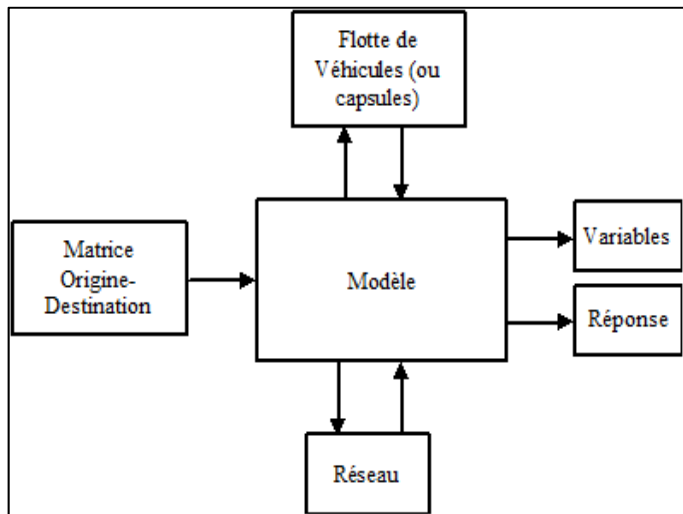


Figure 2.1 Schéma bloc du modèle dynamique de gestion des véhicules PRT

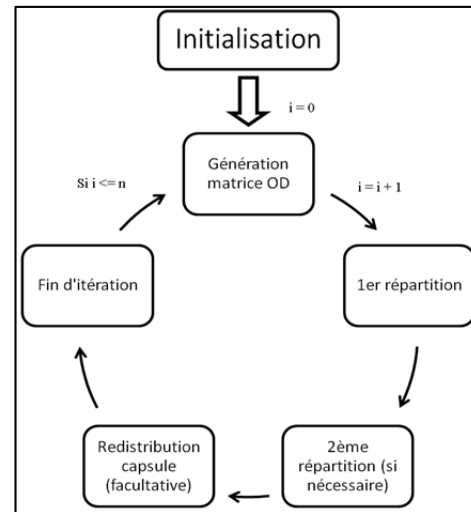


Figure 2.2 Schéma simplifié d'une simulation

La Figure 2.1 correspond à un schéma bloc ou schéma fonctionnel, il permet de visualiser les entrées et les sorties ainsi que les paramètres qui peuvent influencer la réponse. La Figure 2.2 quant à elle décrit concrètement le déroulement d'une simulation. A chaque phase de ce schéma correspond une fonction mathématique, ce schéma reprend donc la logique du code du modèle. Le Tableau 2.1 présente la description de toutes ces phases.

Tableau 2.1 Description des différentes phases du modèle

Étapes	Description / Conditions
Initialisation	Les paramètres de base sont définis tels que le nombre d'itérations, le nombre de capsules ou le réseau.
Génération matrice OD	A chaque itération une nouvelle liste d'OD est créée. La nouvelle liste OD se greffe aux anciennes demandes qui n'ont pas encore été traitées.
1 <sup>ère</sup> répartition des véhicules	Si une capsule est présente là ou un utilisateur en a besoin alors l'utilisateur devient prioritaire vis-à-vis de cette capsule. La demande est donc instantanément remplie.
2 <sup>ème</sup> répartition des véhicules	Les demandes non complétées dans la 1 <sup>ère</sup> phase de répartition font donc appel à des capsules associées à des nœuds voisins. Dans cette phase de répartition les capsules voisines seront donc amenées aux nœuds demandeurs. <b>C'est dans cette phase que plusieurs algorithmes de gestion seront testés.</b>
Redistribution des véhicules	Phase facultative (suivant la simulation) Une fois toutes les capsules attribuées aux utilisateurs, s'il reste des capsules libres et qu'il apparait un déséquilibre dans la répartition des capsules libres dans le réseau, il est alors possible de redistribuer les capsules dans le réseau afin de répondre de façon anticipée aux futures demandes. La moyenne des véhicules libres est calculée, si à un nœud il y a plus ou moins de véhicules que cette moyenne alors il y a déséquilibre. Un nœud pourra alors envoyer ses véhicules en surplus vers d'autres nœuds.
Fin d'itération	Phase représentant le mouvement des capsules dans le réseau. Si une capsule arrive à destination, elle devient libre.

### 2.1.2 Algorithmes

La 1<sup>ère</sup> phase de répartition correspond à la répartition instantanée, c'est-à-dire que si un utilisateur demande un véhicule et qu'il y en a un de libre à cette même station ce véhicule lui sera attribué automatiquement et instantanément. Cependant la 2<sup>ème</sup> phase de répartition correspond au cas où une capsule doit être amenée d'un autre nœud pour répondre à la demande de l'utilisateur. Deux façons de répartir les véhicules seront étudiées correspondants à deux algorithmes différents, l'une dite « répartition évidente » et l'autre basée sur l'algorithme de colonie de fourmis.

La répartition « évidente » envoie les véhicules les plus proches, les demandes ne sont pas considérées comme un ensemble mais individuellement. La 1<sup>ère</sup> demande recevra la capsule la plus proche, puis la 2<sup>ème</sup> etc. L'algorithme qui sera utilisé dans le modèle est une légère modification de l'algorithme de base de colonie de fourmis, cependant la seule différence provient dans la construction de la matrice des distances. L'algorithme est donc identique mais la matrice des distances, au lieu d'être la distance entre toutes les villes comme dans le problème du voyageur de commerce, il s'agit de la distance entre les capsules et les nœuds.

L'algorithme de colonie de fourmis permet quant à lui de considérer toutes les demandes en tant qu'ensemble. L'algorithme présenté dans la partie 1.3.3 du mémoire est l'algorithme de base appliqué au problème du voyageur de commerce. Ici le problème est légèrement différent. En effet, lors du problème du voyageur de commerce la matrice des distances exprime celles qui séparent les villes les unes des autres, tandis que dans le cas qui nous occupe, cette matrice exprime la distance entre les capsules libres et les demandes individuelle, telle que :

Soit OD : l'ensemble des demandes

A : l'ensemble des capsules disponibles

D : la matrice distance ayant pour élément  $d_{i,j}$

$$\forall i \in OD, \forall j \in A \quad d_{i,j} = dist(OD_i, A_j) \quad (2.1)$$



C'est cette matrice D qui est utilisée dans les simulations. De plus l'algorithme de colonie de fourmis est dépendant de plusieurs paramètres qui ont été définis dans l'état des connaissances (Tableau 2.2).

Tableau 2.2 Récapitulatif des différentes variables de l'algorithme de colonie de fourmis

Variable	Sigle	Valeur
Coefficient d'intensité de phéromone	$\alpha$	1
Coefficient de visibilité	$\beta$	8
Nombre de fourmis	m	Égale au minimum du nombre de demandes et du nombre de capsules libres
Coefficient d'ajustement	el	0,96
Coefficient d'évaporation	$\rho$	0,01
Nombre d'itérations	n	À déterminer par calibration

Le nombre d'itérations dépend de la complexité du problème et doit être soumis à un calibrage. Cet algorithme étant un algorithme d'optimisation qui ne renvoie pas forcément la solution optimale mais une réponse qui s'en approche. Comme pour tout algorithme d'optimisation il est nécessaire de calibrer afin d'avoir un bon compromis entre la pertinence des résultats et le temps de calcul.

Pour une matrice de distance donnée, le minimum absolu est recherché. Pour cela l'algorithme est calibré plusieurs fois et avec de nombreuses itérations avant d'arriver à un consensus de résultats. Une fois le minimum trouvé, on garde la même matrice des distances et l'algorithme est calibré pour donner des résultats en moyenne à 10% de ce résultat minimum sans jamais donner un résultat aberrant c'est-à-dire un résultat dépassant les 20% du résultat minimum. Si pour un certain nombre d'itérations l'algorithme renvoi en moyenne des résultats en dessous de 10% mais que une des simulations renvoi un résultat aberrant dépassant les 20% alors l'algorithme ne sera pas considéré comme viable. Ces critères ont été

définis de façon arbitraire. Enfin à chaque fois qu'un véhicule se déplace d'un nœud à l'autre l'algorithme de Dijkstra voir Figure-A I-18 en annexe est utilisé pour calculer le meilleur trajet.

Il existe plusieurs façons de redistribuer les capsules libres et à chaque type de redistribution, un algorithme qui exécute cette redistribution. La liste des différents types de redistribution possible ainsi que leur description est présentée au Tableau 2.3.

Tableau 2.3 Liste des types de redistribution des capsules

Nom	Description
La non-redistribution	Le système est laissé tel quel. Si une capsule se déplace et arrive à sa destination elle reste à ce nœud jusqu'à répondre à une future demande la plus proche. Il n'y a pas d'anticipation de la demande ni de recherche d'équilibre.
La redistribution équilibrée	A chaque fin d'itération, le système se rééquilibre c'est-à-dire que d'un nœud à l'autre il y aura un nombre équivalent de capsules libres prêtes à répondre à des demandes futures.
La redistribution anticipée	<i>(Seulement pour les scénarios 2 à 6)</i> Connaissant le profil anticipé des demandes, les capsules libres sont envoyées aux nœuds susceptibles d'être des demandes. La redistribution est ainsi optimisée.

Plusieurs algorithmes ont été utilisés afin de créer le modèle dynamique de gestion et de déplacement des véhicules afin de répondre aux besoins individuels. Le Tableau 2.4 présente une synthèse de ces algorithmes ainsi que de la fonction à laquelle ils sont attribués.

Tableau 2.4 Résumé des différents algorithmes utilisés et disponibles pour le modèle

Action ou phase	Algorithme disponible utilisé
Calcul d'un trajet d'un véhicule	Algorithme de Dijkstra (voir 1.3.2)
2 <sup>ème</sup> Répartition	Algorithme dit « évident » (voir 2.1.2)
	Algorithme de colonie de fourmis (voir 1.3.3 et 2.1.2)
Redistribution des véhicules	Algorithme de non-redistribution (voir Tableau 2.3)
	Algorithme redistribution équilibrée (voir Tableau 2.3)
	Algorithme redistribution anticipée (voir Tableau 2.3)

### 2.1.3 Matrice Origine Destination (Matrice OD)

La matrice OD exprime une demande qui est caractérisée par un nœud « Origine » et un nœud « Destination ». Ces matrices OD sont générées aléatoirement suivant un scénario de profils des demandes. Le scénario détermine si un nœud peut être « Origine » et/ou « Destination » mais il n'y a pas de pondérations quant à la probabilité pour un nœud d'être origine ou destination. Chaque nœud, s'il est défini comme « Origine » possible, a autant de chance qu'un autre nœud d'être « Origine ». Un nœud aura un nombre différent de demandes à chaque simulation due au caractère aléatoire de la simulation. Évidemment, si un seul nœud est défini comme « Origine » possible ou « Destination » possible ce nœud le sera pour toutes les demandes. Une demande ne peut avoir la même origine que sa destination. Le modèle introduit volontairement une dimension aléatoire. C'est-à-dire qu'entre deux simulations qui ont comme même variable, même scénario et mêmes conditions initiales, la matrice OD sera différente et donc la réaction du modèle aussi. Par exemple, dans les Figure 2.3 et Figure 2.4, la répartition des nœuds origines entre deux simulations pour le scénario 4 est différente.

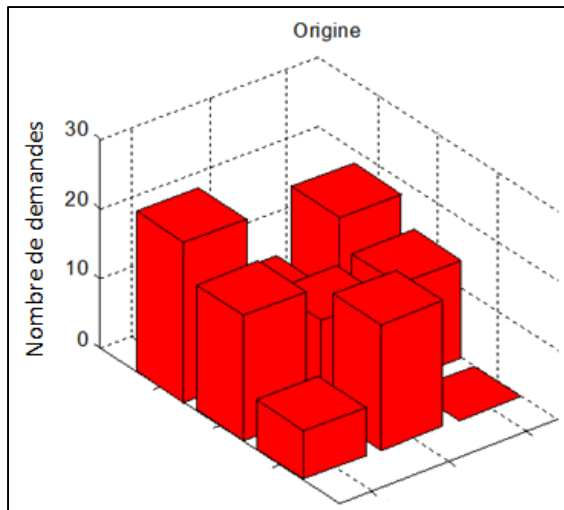


Figure 2.3 Répartition des origines pour le scénario 4 cas A

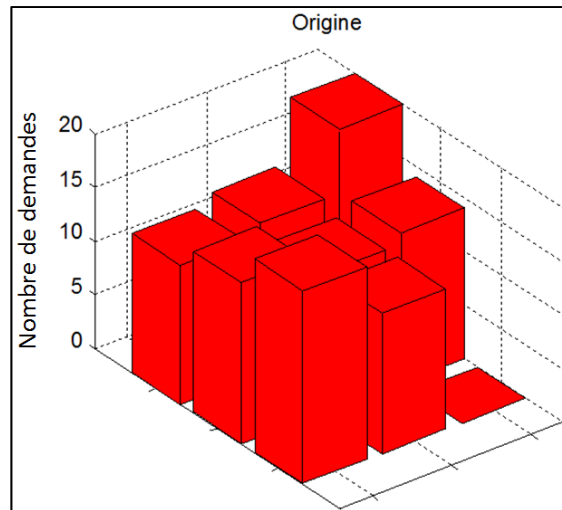


Figure 2.4 Répartition des origines pour le scénario 4 cas B

En effet ces deux figures représentent la répartition des nœuds origines pour deux simulations aux conditions identiques. Le réseau est représenté et chaque « bâton » représente le nombre de demandes à un nœud du réseau. Les résultats sont différents dus aux conditions d'équiprobabilité de la génération de la matrice OD. Ces disparités tendront à disparaître pour un grand nombre de demandes comme dans la Figure 2.5.

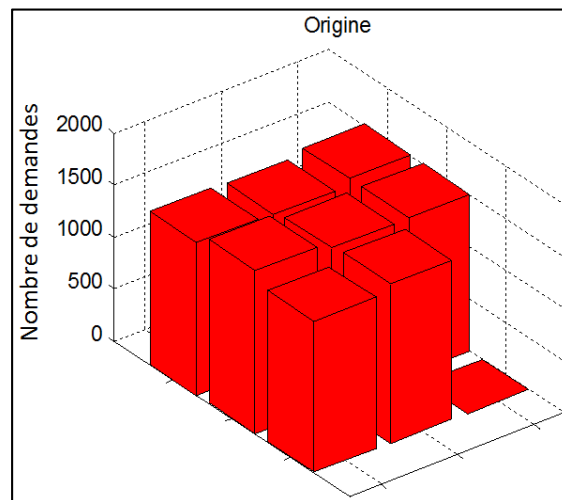


Figure 2.5 Répartition des origines pour le scénario 4 cas C (grand nombre de demandes)

## 2.2 Paramètres, variables et critères de performance associés

Que cela soit externe (Matrice OD, Flotte de véhicules ou Réseau) ou interne au modèle (Type de répartition ou de redistribution), il existe plusieurs paramètres à déterminer. En effet le modèle peut influencer sur l'état du réseau, cela est utile pour simuler les problèmes de congestion. La flotte des véhicules est, elle aussi indépendante, c'est-à-dire que le modèle et/ou l'utilisateur peuvent la modifier. Il est donc possible de modifier le nombre de véhicules pendant une simulation pour simuler des pannes ou un ajout de véhicules. Un des intérêts majeurs des PRT est leur gestion « intelligente ». Il doit donc être possible d'utiliser plusieurs types de gestions. Enfin toutes les variables devront être enregistrées à chaque simulation afin de les utiliser pour les analyser. Les paramètres fixes et variables sont distingués. En effet les paramètres fixes seront les mêmes pour toutes les simulations alors que pour chaque variation d'un paramètre variable il y a une nouvelle simulation. Ainsi cette partie s'attardera à présenter la structure du réseau, les variables et les critères de performances.

### 2.2.1 Structure du réseau

La structure du réseau à l'étude est caractérisée par neuf nœuds (3x3) reliés par des tronçons bidirectionnels d'égale longueur qui forment quatre mailles identiques. La Figure 2.6 présente la configuration de ce réseau régulier.

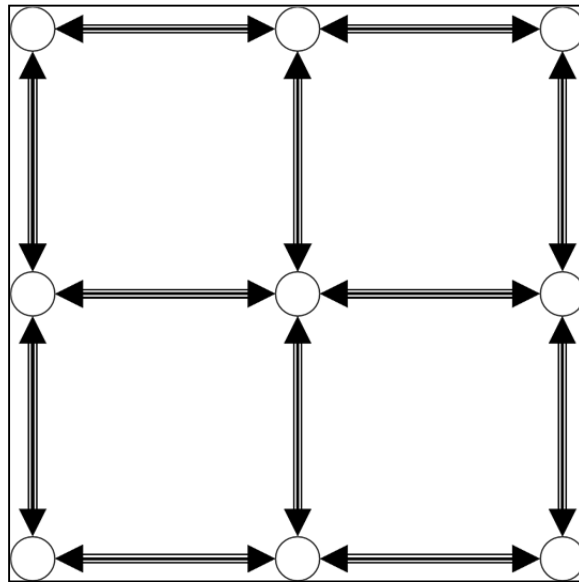


Figure 2.6 Schéma du réseau 3x3 régulier

### 2.2.2 Paramètres des scénarios de simulation

Dans le contexte de la présente étude, il est considéré l'équivalence suivante : une itération dans la simulation correspond à une minute de temps réel. Par ailleurs, considérant que les véhicules des projets ULTRa et 2getthere se déplaçant à 40 km/h (section 1.2.2, p.13), la vitesse fixée pour les véhicules est de 30 km/h (prise en compte les temps d'accélération, de décélération et des temps de transferts des gens, soit le temps de sortir ou de rentrer dans le véhicule). Le réseau maillé (3x3 nœuds) est de huit itérations de longueur et de largeur (quatre itérations par tronçon), ce qui correspond à un réseau de 4km de côté soit une aire de 16 km<sup>2</sup>. Le Tableau 2.5 présente les paramètres qui ont été utilisés pour les simulations ainsi que leurs caractéristiques.

Tableau 2.5 Liste des paramètres et leurs valeurs

Nom du paramètre	Valeur
<b>Paramètres fixes</b>	
Type de réseau	3x3 régulier
Nombre de capsules	200
Vitesse des capsules	30 km/h
Nombre d'itérations par simulation	120 (soit deux heures en « temps réel »)
Durée d'une itération	une minute
<b>Paramètres variables</b>	
n (nombre de demandes par itérations)	2 à 100 par itération (soit de 240 à 1200 demandes par simulation)
Type de demande	Six types de « scénarios » possibles (voir 2.3.1)
Type de redistribution des capsules	Deux à trois types de redistribution suivant les scénarios (voir Tableau 2.3)

### 2.2.3 Indicateurs de performance

L'évaluation de la performance des scénarios d'opération des véhicules en fonction de la demande en mobilité des utilisateurs relativement à des profils de répartition de ces demandes, repose sur des indicateurs. Il est donc nécessaire d'établir des paramètres mesurables afin de juger quantitativement l'efficacité du modèle face à un certain type de scénario.

Ces paramètres sont les suivants : Pour un véhicule  $k$  donné, le système enregistre tous ses déplacements ainsi que leur nature, soit des déplacements d'utilisateurs (déplacements utiles  $d_k$ ), soit des déplacements à vide, nécessaires pour équilibrer le réseau dans le but de faciliter la réponse à des demandes à venir;

- pour chaque demande  $j$ , le temps d'attente  $t_j$  qui correspond au temps qu'il faut au véhicule avant de répondre à la demande ;
- $E_d$  ensemble des demandes et  $E_{dc}$  ensemble des demandes complétées.

De ces paramètres, il est possible de calculer les indicateurs de performances présentés au Tableau 2.6.

Tableau 2.6 Liste des indicateurs de performances

Nom de l'indicateur et/ou sigle attribué	Description
Taux d'occupation : $\theta$	$\theta = \frac{\Sigma \text{Trajet utile}}{\Sigma \text{déplacements}} \quad (2.2)$ <p>Il s'agit du rapport entre la somme des déplacements d'utilisateurs et la somme de l'ensemble des déplacements dans le réseau qui comprend les déplacements à vide des véhicules.</p>
Temps d'attente moyen : $T_{moy}$	$T_{moy} = \frac{1}{card(E_d)} \sum_{j \in E_d} t_j \quad (2.3)$
Temps de trajet moyen : $D_{moy}$	$D_{moy} = \frac{1}{card(E_{dc})} \sum_{k \in E_{dc}} d_k \quad (2.4)$
Critère d'efficacité : $Ce$	<p>Il s'agit d'un critère qui sert à déterminer la capacité du réseau (voir ci-dessous). Il correspond à 25% (critère arbitraire) du temps de trajet moyen.</p> $Ce = 0,25 * D_{moy} \quad (2.5)$
Capacité du réseau : $Nd_{capacité}$	<p>Soit <math>Nd</math> le nombre de demandes et <math>T_{moy}</math> le temps d'attente moyen en fonction du nombre de demandes, la capacité du réseau correspond à la valeur du nombre de demandes <math>Nd_{capacité}</math> tel que :</p> $T_{moy}(Nd_{capacité}) = Ce \quad (2.6)$



## **2.3 Scénarios**

La structure du réseau ainsi que les profils des matrices origine-destination sont indépendants du modèle. Ainsi plusieurs scénarios seront étudiés afin d'étudier l'influence de la répartition des demandes dans un réseau dynamique intelligent.

### **2.3.1 Profil des demandes**

La nature des demandes de la part des usagers peut être équirépartie entre les différents nœuds du réseau ou à l'inverse peu ou pas distribuée. Cette variation des profils a priori un effet sur la gestion de la flotte de véhicules et sur l'efficacité du service de l'offre vis-à-vis de la demande. La Figure 2.7 présente le profil des demandes pour les six scénarios étudiés.

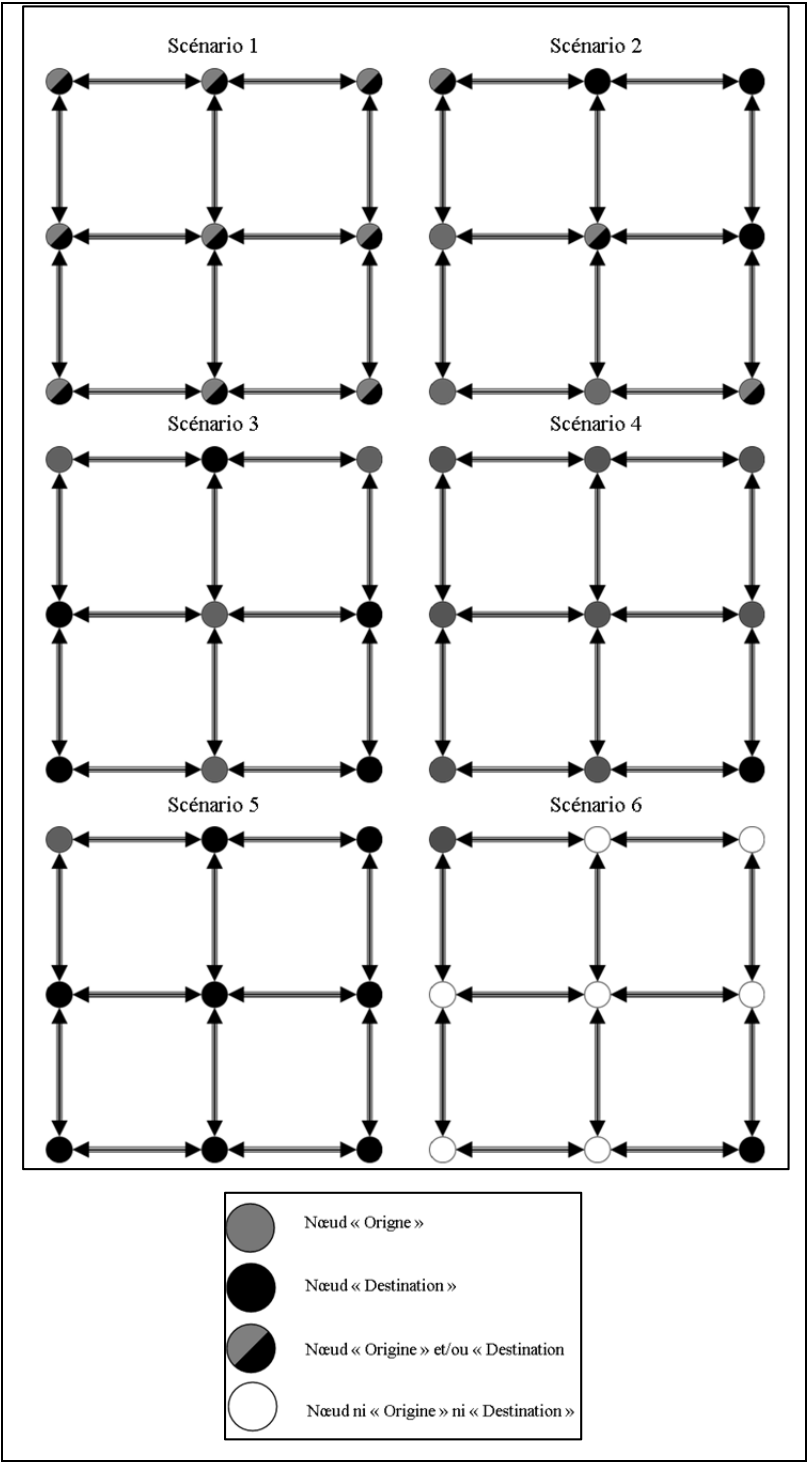


Figure 2.7 Schéma des 6 scénarios étudiés

Tableau 2.7 Résumé des 6 scénarios étudiés

Nom du scénario	Description
Scénario 1	Tous les nœuds sont origines et destinations, il n'y a donc pas de nœuds préférentiels
Scénario 2	Seulement la moitié du réseau est origine alors que l'autre est destination, il y a donc une symétrie dans le réseau
Scénario 3	Les nœuds origines et destinations sont répartis dans le réseau avec la particularité de ne pas être l'un <b>et</b> l'autre. Une capsule qui arrive donc à destination devra donc être obligatoirement déplacer pour répondre à une nouvelle demande
Scénario 4	Tous les nœuds sont origines sauf un. Par exemple dans un quartier tous les utilisateurs se dirigent vers une station de métro.
Scénario 5	Tous les nœuds sont destination sauf un. Par exemple dans un quartier où tous les utilisateurs sortent d'une station de métro jusqu'à leurs destinations.
Scénario 6	Un seul nœud est origine et un seul nœud est destination et ces deux nœuds sont les plus éloignés possibles. Cela correspond au pire scénario envisageable pour se type de transport en commun

### 2.3.2 Analyse de la performance des modes opératoires de la flotte de véhicules

Les deux paramètres qui permettent de comparer toutes les simulations sont donc le **taux d'occupation** et la **capacité du système**. Le premier rend compte de la bonne utilisation des véhicules, afin de juger de l'efficacité du système. Le deuxième permet de savoir combien de personnes peuvent être déplacées en un temps donné.

Avec un tel réseau il est intéressant de mettre en évidence deux phénomènes typiques des problèmes de transport. Le premier est que la distribution et la nature de la matrice OD ont un effet sur la fluidité du trafic. C'est-à-dire que pour un même nombre de demandes et un même nombre de capsules le profil des demandes influe sur l'efficacité du système et cela

indépendamment de la longueur moyenne des trajets. Le deuxième est l'importance de la gestion des capsules, et le type de répartition des capsules; il est possible de montrer que le système peut être plus ou moins efficace en fonction du mouvement à vide des capsules.

**C'est donc principalement avec ces deux critères (taux d'occupation et capacité) que seront étudiés : l'importance de la redistribution des capsules, l'intérêt d'un réseau intelligent et l'influence de la matrice OD sur l'efficacité du réseau.**



















L'intérêt d'un réseau intelligent sera étudié avec le scénario 1, qui correspond au scénario le plus favorable à la circulation des capsules. Ce scénario sera soumis à deux algorithmes de répartition des capsules (Tableau 2.8).

Tableau 2.8 Comparaison de deux algorithmes

	Algorithme Évident	Algorithme ACO
Scénario 1		

Le Tableau 2.9 est quant à lui un résumé des redistributions étudiées par rapport aux scénarios.

Tableau 2.9 Résumé des simulations entre les scénarios et les types de redistributions

	Non redistribution	Redistribution équilibrée	Redistribution anticipée
Scénario 1			
Scénario 2			
Scénario 3			
Scénario 4			
Scénario 5			
Scénario 6			

## **CHAPITRE 3**

### **RÉSULTATS**

Après s'être assuré que le modèle est bien calibré pour donner des résultats cohérents en un temps minimal (Chapitre 3.2), les résultats des différents scénarios seront discutés. Enfin une comparaison entre les scénarios, soit la répartition des nœuds attractifs, sera présentée.

#### **3.1 Construction de la matrice des distances**

Le but de cette étape est de déterminer le nombre optimal d'itérations à faire pour obtenir des résultats viables par l'algorithme de colonie de fourmis en minimisant les temps de calculs. Dans un premier temps la matrice des distances est établie. La matrice des distances correspond à toutes les possibilités entre capsules et demandes. S'il y a  $A$  demandes et  $B$  capsules libres la matrice des distances est une matrice de taille  $A \times B$ . En pratique,  $A$  est majoré, c'est-à-dire que s'il y a trop de demandes le modèle en considère seulement les  $A_{\max}$  premières qui correspondent aux nombres de capsules libres (si  $A \geq B$  alors  $A = A_{\max} = B$ ).

Voici un exemple de comment est construit cette matrice dans le cas où il y a 2 demandes pour 5 capsules. Dans le réseau  $3 \times 3$  chaque nœud est identifié par un numéro voir Figure 3.1.

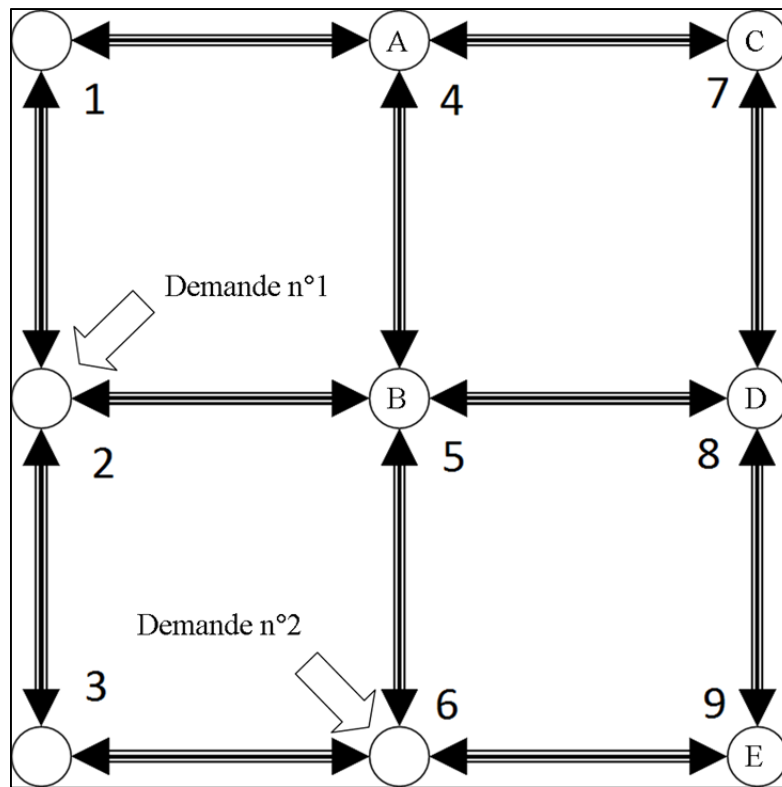


Figure 3.1 Réseau avec 2 demandes et 5 capsules libres

Il y a donc 2 demandes, une au **nœud 2** et l'autre au **nœud 6** et 5 capsules de libres aux **nœuds 4, 5, 7, 8 et 9**, les capsules sont respectivement appelées A, B, C, D et E. Sachant qu'il faut 2 itérations pour rejoindre deux nœuds adjacents on en déduit la matrice des distances.

Tableau 3.1 Construction de la matrice des distances

	<b>Capsule A (nœud 4)</b>	<b>Capsule B (nœud 5)</b>	<b>Capsule C (nœud 7)</b>	<b>Capsule D (nœud 8)</b>	<b>Capsule E (nœud 9)</b>
Demande n°1 (nœud 2)	4	2	6	4	6
Demande n°2 (nœud 6)	4	2	6	4	2

C'est cette structure de matrice qui sera utilisée dans l'algorithme de colonie de fourmis afin de déterminer quelles sont les meilleures répartitions possibles.

### 3.2 Calibrage de l'algorithme de colonie de fourmis

L'algorithme de colonie de fourmis est calibré afin de déterminer le nombre d'itérations minimal pour permettre des calculs fiables et rapides. Ce résultat sera appelé résultat étalon. Les paramètres tels que le nombre de fourmis (égal au nombre de véhicules, ici 200), alpha (=1), beta (=8) ou le coefficient d'évaporation (=0,01) ont été choisis dans la partie précédente. Ce type d'algorithme d'optimisation, pour de faibles itérations, peut être légèrement aléatoire. C'est-à-dire que pour deux simulations le résultat obtenu pourra être différent. Cet effet tend à disparaître pour un grand nombre d'itérations et si l'algorithme est bien calibré. Dans le cas de cette étude, l'algorithme sera calibré pour donner des résultats en moyenne à 10% du résultat minimum sans jamais donner un résultat aberrant c'est-à-dire un résultat dépassant les 20% du résultat minimum. Les algorithmes ne renvoyant pas forcément les solutions « parfaite » mais des résultats approchés il est important de maîtriser la fourchette d'incertitude dans laquelle se situe les résultats. Ces seuils de 10% et 20% ont été choisis arbitrairement. Le résultat « minimum » est le résultat qui serait obtenu avec un très grand nombre d'itérations, il s'agit donc d'un résultat fiable et très probablement du meilleur résultat dans l'absolu. Tous les paramètres sont donnés dans la littérature (Dréo et al., 2003) (de même que le nombre d'itérations), il s'agit donc de déterminer le plus petit nombre d'itérations nécessaires pour obtenir des résultats fiables.

Pour calibrer l'algorithme de colonie de fourmis on construit d'abord 3 matrices de distances qui seront fixes pour la suite du calibrage. Une matrice qui représente 25 demandes pour 200 capsules de libres, une pour 102 demandes et 200 capsules de libres et une pour 200 demandes et 200 capsules libres. Comme le nombre de demandes considéré variera entre 0 et 200 lors des « vraies » simulations, ces trois matrices de calibrage représentent un échantillon de ces futures simulations. En effet les demandes oscillent entre 0 et 200, 200 étant la valeur plus contraignante en termes de calculs.

Le « coût » est la somme de toutes les distances une fois la solution décidée. Le « coût minimum » est donc la somme de toutes les distances de la meilleure solution. Pour la première matrice, calculé plusieurs fois avec l'algorithme de colonie de fourmis, est de 50. L'algorithme sera donc calibré pour donner en moyenne des résultats entre 50 et 55 (soit 10% de 50) sans jamais donner un résultat aberrant (c'est-à-dire supérieur à 20% de 50). Les « coûts » en fonction du nombre d'itérations sont présentés Figure 3.2

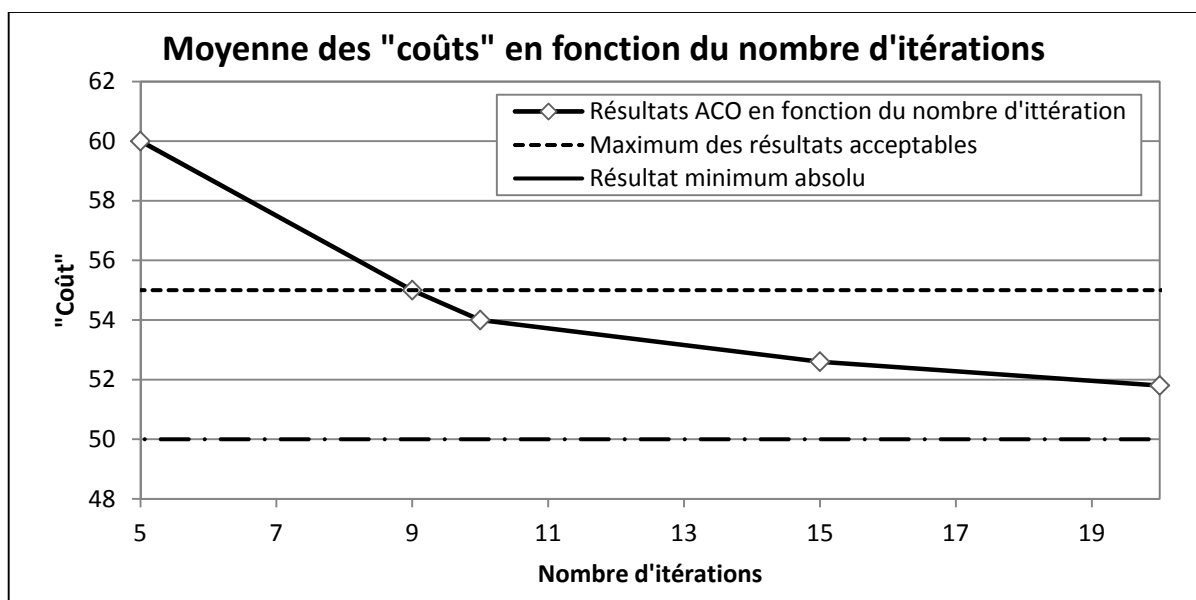


Figure 3.2 Moyenne des coûts calculés par l'algorithme en fonction du nombre d'itérations pour 25 demandes



La Figure 3.2 montre que moins l'algorithme effectue d'itérations moins le système est précis. Il apparaît que le système semble viable à partir de neuf itérations, il reste cependant à vérifier que ce modèle ne fournit pas de résultats aberrants d'un écart supérieur à 20% du résultat étalon, soit supérieur à un coût de 60. Le Tableau 3.2 présente les résultats de 10 itérations ainsi que la valeur moyenne de ces 10 itérations. Il faut donc que la valeur moyenne soit inférieure ou égale à 55 et que toutes les valeurs des itérations soit inférieurs à 60.

Tableau 3.2 Tableau récapitulatif de 10 simulations pour 9 itérations et pour 25 demandes

Numéro de la simulation	1	2	3	4	5	6	7	8	9	10	Moy
Coût	52	56	52	54	54	54	54	54	60	60	55

Tel que le montrent les valeurs du coût des simulations 9 et 10 du Tableau 3.2 pour les conditions associées à 9 itérations, il y a une forte probabilité (20%) de renvoyer des résultats aberrant c'est-à-dire trop éloignés du résultat étalon. Les résultats spécifiques obtenus dans le cas des simulations pour 10 itérations sont présentés au Tableau 3.3.

Tableau 3.3 Tableau récapitulatif de 10 simulations pour 10 itérations et pour 25 demandes

Numéro de la simulation	1	2	3	4	5	6	7	8	9	10	Moy
Coût	56	56	54	54	54	52	52	54	52	56	54

Tel que le montre les valeurs du Tableau 3.3, les résultats obtenus par l'algorithme sont acceptables avec 10 itérations lorsqu'il y a 25 demandes. Il reste à vérifier que ce calibrage est valable peu importe le nombre de demandes. Le maximum de demandes considérées est égal aux nombres de capsules. C'est-à-dire lors de la construction de la matrice des distances, s'il y a 200 capsules dans le réseau alors seules les 200 premières demandes seront intégrées dans cette matrice. Cela sera le cas pour la majorité des cas traités dans l'étude. Les vérifications sont faites pour 102 demandes (Tableau 3.4) et 200 demandes (Tableau 3.5).

Tableau 3.4 Tableau récapitulatif de 10 simulations pour 10 itérations et pour 102 demandes

Numéro de la simulation	1	2	3	4	5	6	7	8	9	10	Moy
Coût	344	358	350	350	354	356	354	366	348	344	352

Le coût minimum pour la matrice des distances utilisé pour la vérification avec 102 demandes est de 322 (calculé avec l'algorithme de colonie de fourmis). Le critère de moyenne est donc 354 (10% supérieur à 322) et le « critère d'aberration » est de 386 (20% supérieur à 322). La moyenne étant de 353,4 et aucunes des simulations ne présentant d'aberrations l'algorithme est calibré pour 102 demandes.

Tableau 3.5 Tableau récapitulatif de 10 simulations pour 10 itérations et pour 200 demandes

Numéro de la simulation	1	2	3	4	5	6	7	8	9	10	Moy
Coût	1062	1070	1058	1070	1058	1070	1058	1058	1074	1054	1063

Le coût minimum pour la matrice des distances utilisé pour la vérification avec 200 demandes est de 1026 (calculé avec l'algorithme de colonie de fourmis). Le critère de moyenne est donc 1128 (10% supérieur à 1026) et le « critère d'aberration » est de 1231 (20% supérieur à 1026). La moyenne étant de 1063,2 et aucune des simulations ne présentant d'aberrations l'algorithme est calibré pour 200 demandes.

L'algorithme calibré à 10 itérations permet d'obtenir des résultats qui satisfont les conditions fixées d'acceptabilité et cette calibration sera utilisée pour la suite des simulations.

### 3.3 Analyse des modes de gestion des véhicules

L'un des principaux avantages des PRT est la faculté à communiquer entre tous les véhicules et ainsi de gérer collectivement la flotte de véhicule. L'analyse du bénéfice de la gestion collective des véhicules sera effectuée comparativement à un système qui n'utiliserait pas cette approche. Par ailleurs, un avantage d'un réseau où les véhicules sont mobiles et régulés par ordinateur est qu'ils peuvent permettre d'anticiper les demandes et de gérer au mieux ces

flux. Cette capacité à anticiper la demande par une réallocation des véhicules dans le réseau sera évaluée selon plusieurs approches en regard des indicateurs de performance retenus.

### **3.3.1 Capacité**

La capacité a été défini mathématiquement au 2.2.3, concrètement lorsque la courbe du temps d'attente moyen est étudiée la « capacité du réseau » correspond à la valeur du nombre de demandes pour laquelle le fonctionnement du système n'est plus fluide. Les résultats suivants ont été obtenus pour 200 véhicules, l'évolution du temps d'attente moyen est étudiée sur une période de 120 itérations (soit 2 heures en temps réels) en fonction du nombre de demandes. Pour ce modèle une itération représente donc une minute. Plus le nombre de demandes augmente plus le temps d'attente du passager est long. La capacité correspond à la valeur pour laquelle ce temps d'attente est trop important pour l'utilisateur.

Avant de commencer l'analyse voici un exemple de comment la capacité du réseau est calculée :

Ci-dessous la réponse du modèle face à des demandes du scénario 1 ayant un type de non-redistribution des capsules. La Figure 3.3 présente donc le temps d'attente moyen en fonction du nombre de demandes pour 2 heures de simulation

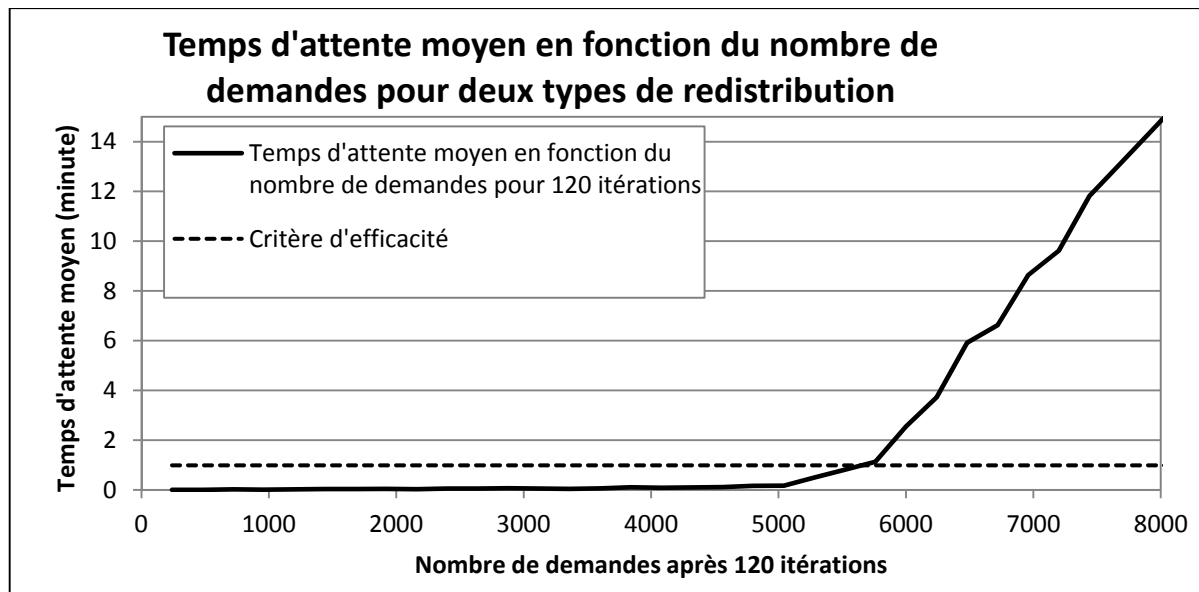


Figure 3.3 Temps d'attente moyen en fonction du nombre de demandes (illustration de calcul de capacité)

Si le temps d'attente moyen est inférieur au critère d'efficacité alors le système est fluide c'est-à-dire qu'il répond aux demandes en un temps correct. Une fois qu'il passe au-dessus le système n'est plus fluide. L'intersection des deux renseigne sur la **capacité du réseau** soit le nombre de demandes que le réseau est capable de gérer en deux heures. Voici Tableau 3.6 les résultats pour quatre nombres de demandes différentes lorsque le système passe d'un état non saturé à un état saturé.

Tableau 3.6 Tableau du temps d'attente moyen en fonction du nombre de demandes (unité : minute)

Scénario 1, Type: Non-redistribution				
Moyenne globale des trajets	3,93827			
Critère d'efficacité	0,98456			
Nombre de demande	5280	5520	5760	6000
Temps d'attente moyen	0,4905303	0,80923	1,12690	2,54766

La moyenne des trajets étant égale à 3,93 it (itérations) le critère de capacité est égale à  $0,25 \times 3,93 = 0,98 \text{it}$ . Pour 5520 demandes le système n'est pas saturé alors que pour 5760 il l'est. La capacité du réseau est donc entre 5520 et 5760. Pour des raisons de simplicité on considère donc que la capacité est la valeur inférieure donc ici de 5520 demandes. La capacité de notre système est donc, dans les conditions fixées de : 5520 demandes / 2 heures.

### **3.3.2 Intérêt d'un réseau intelligent**

L'intérêt principal d'un réseau PRT est la communication. En effet les capsules communiquent toutes entre elles, ou plutôt à un « cerveau » central qui peut ainsi régir leurs comportements. Il a été énoncé que l'algorithme de colonie de fourmis a été utilisé dans l'élaboration du modèle, il s'agit donc d'en démontrer les atouts. Lorsque le système est soumis à plusieurs demandes d'utilisateurs en même temps, il peut analyser ces demandes afin de répartir au mieux les capsules vers les utilisateurs. Or cette répartition n'est pas forcément évidente. Les résultats entre une comparaison qui serait évidente, c'est-à-dire que chaque demande est traitée individuellement en attribuant le véhicule le plus proche et l'attribution par ACO (algorithme de colonie de fourmis) dans le cas du scénario 1 sont présentés à la Figure 3.4.

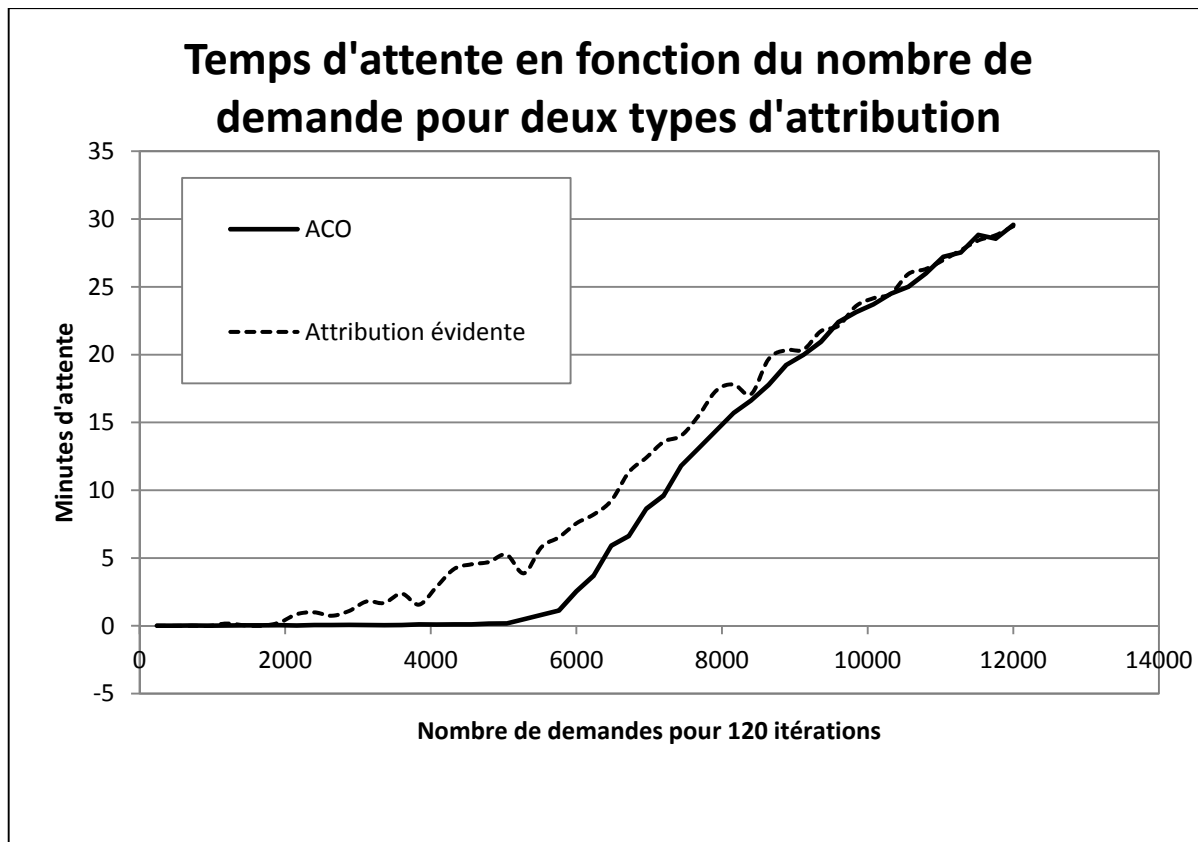


Figure 3.4 Temps d'attente en fonction du nombre de demandes, comparaison entre 2 types d'attributions (Scénario 1)

Les résultats sont identiques pour un nombre très faible (inférieur à 2000) ou très élevé de demandes (supérieur à 10 000). Cependant il y a de grande disparité pour un nombre de demandes entre ces deux seuils. En effet, l'attribution évidente arrive à capacité pour 2640 demandes contre 5520 demandes pour une attribution de type ACO. Ainsi le système peut, pour un nombre identique de capsules et de demandes, être deux fois plus efficace par la méthode ACO comparativement à la méthode évidente.

### 3.3.3 Comportement du système face à différents types de redistribution

Tous les scénarios sont étudiés au travers de trois redistributions des véhicules vides excepté le scénario 1 qui par construction confond les redistributions anticipées et équilibrées. Les résultats associés au cas particulier du scénario 1 sont illustrés à la Figure 3.5.

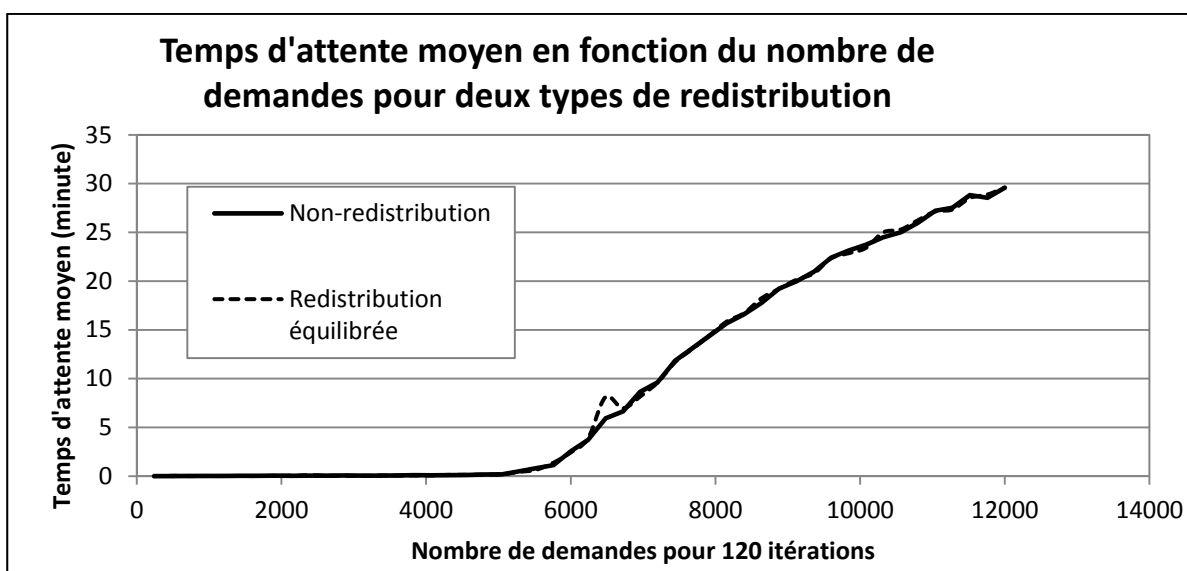


Figure 3.5 Temps d'attente moyen en fonction du nombre de demandes pour deux types de redistribution avec gestion ACO (Scénario 1)

Peu importe le type de redistribution celles-ci elles semblent être adaptées au scénario 1 qui se caractérise par un profil équiprobable des demandes. En effet la saturation apparait pour le même nombre de demande (soit 5520 demandes pour 2heures de simulation). La capacité de ce genre de scénario n'est donc pas dépendante de la redistribution à condition évidemment que cette redistribution soit « relativement adaptée ». La Figure 3.6 s'intéresse à l'évolution du taux d'occupation en fonction du nombre de demandes.

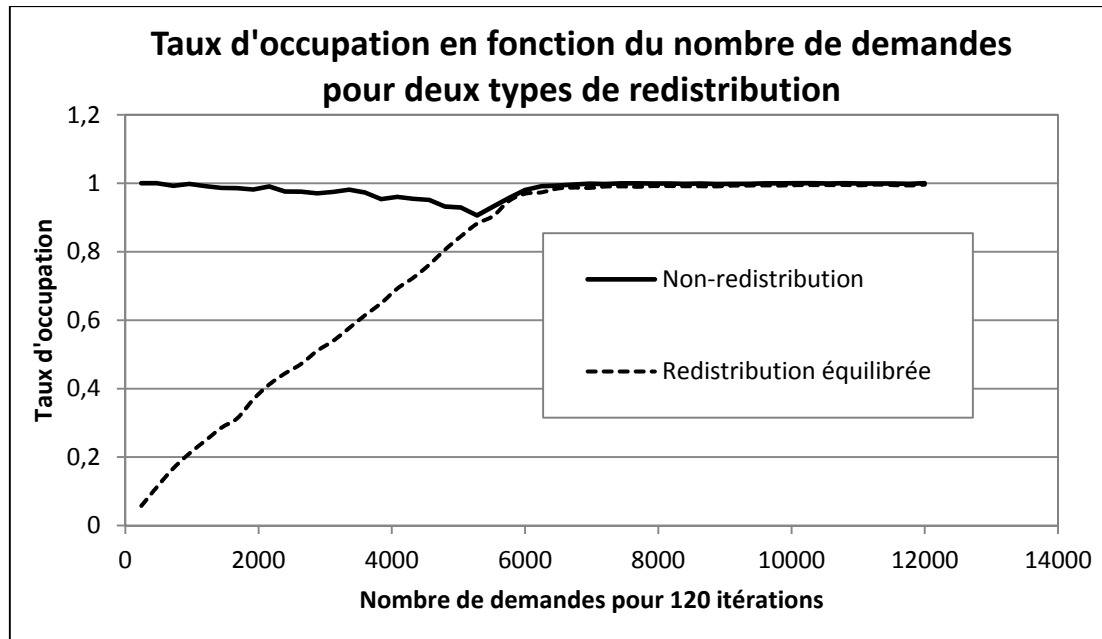


Figure 3.6 Taux d'occupation en fonction du nombre de demandes pour deux types de redistribution avec gestion ACO (Scénario 1)

Plus le taux d'occupation est élevé (proche de un), plus le système est performant énergétiquement (minimise le déplacement de véhicules vides). Or, contrairement au temps d'attente, il y a une grande disparité en fonction des deux types de redistributions des véhicules vides. Pour une même performance du point de vue du temps d'attente pour l'utilisateur, le type de redistribution équilibrée n'est pas efficace en termes énergétique lorsque le nombre de demande est inférieur à 5520. La non-redistribution a une efficacité quasi-parfaite quelque soit le nombre de demande car elle ne cherche pas à réattribuer les véhicules « inutilement ». Après saturation (nombre de demandes supérieur à 5520), les deux redistributions sont équivalentes en tous points autant au niveau énergétique que satisfaction du client.

En ce qui concerne les prochains scénarios, malgré leurs différences de profils les résultats sont, qualitativement, les mêmes. Afin d'illustrer cette tendance la Figure 3.7 présente les résultats du scénario 2.



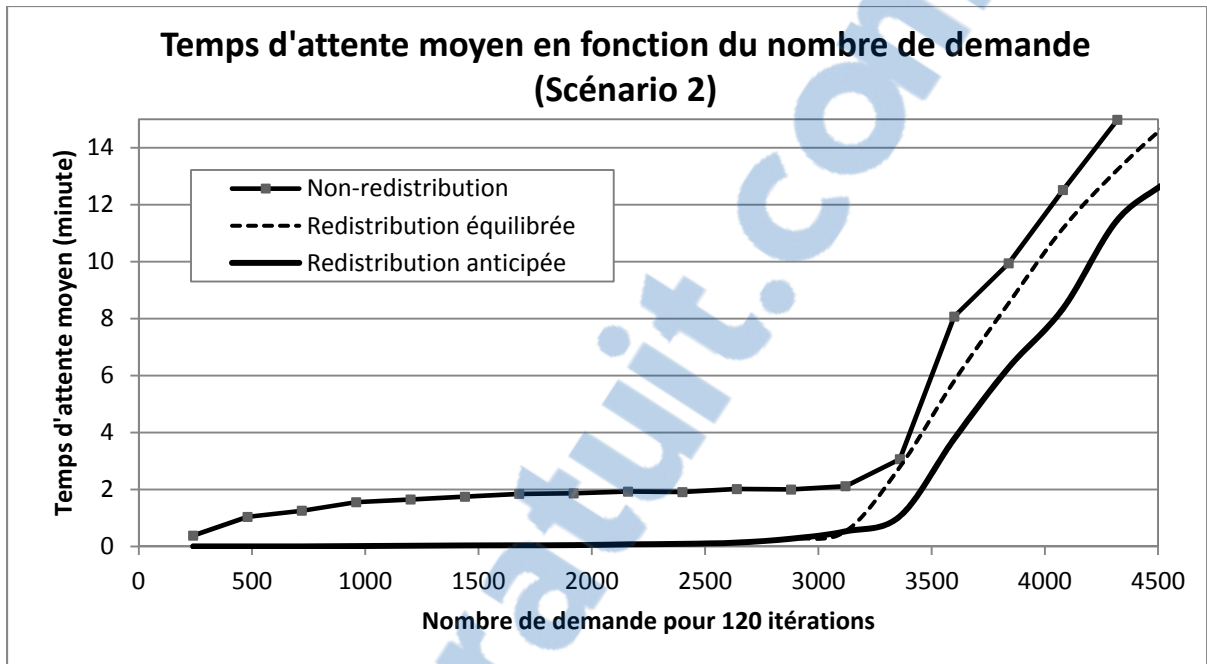


Figure 3.7 Temps d'attente moyen en fonction du nombre de demandes pour trois types de redistributions avec gestion ACO (Axes ajustés) (Scénario 2)

La non-redistribution qui correspond à laisser faire le système sans essayer de le gérer entraîne des temps d'attente non négligeable même pour un nombre faible de demandes. Selon le critère d'efficacité qui a été établi il y a saturation à partir de 240 demandes pour deux heures de simulation, soit 360 demandes/heure alors que 200 capsules sont présentes dans le réseau. L'importance de la gestion des capsules est donc primordiale. Ainsi la redistribution équilibrée n'est pas la redistribution la plus adaptée à ce scénario mais permet de retarder l'effet de saturation jusqu'à 3120 demandes, enfin la capacité pour la redistribution anticipée sera de 3360. La Figure 3.8 présente les résultats du taux d'occupation pour le scénario 2.

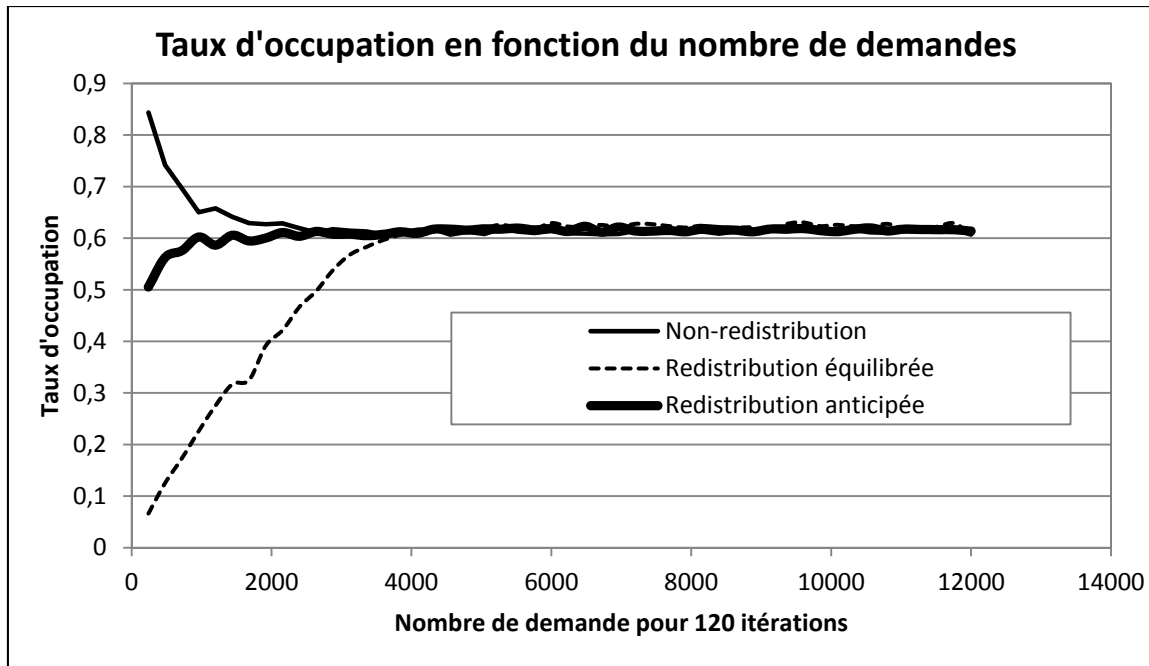



Figure 3.8 Taux d'occupation en fonction du nombre de demandes pour trois types de redistribution avec gestion ACO (Scénario 2)

À très faible demande, la non-redistribution fournit les meilleures performances énergétiques car elle répond instantanément aux demandes sans déplacements inutiles. La redistribution équilibrée propose les moins bonnes performances énergétiques à cause du nombre de déplacements élevés nécessaires pour rééquilibrer systématiquement les nœuds origines. En effet, pour un faible taux de demandes, le déplacement d'une capsule va entraîner un déséquilibre dans le système et une autre capsule devra être mobilisée pour rééquilibrer le système. Enfin à partir de 1680 demandes par deux heures la redistribution anticipée offre les mêmes performances que la non-redistribution. Il est important de noter l'existence d'un état stationnaire après saturation. Le but d'une bonne redistribution anticipée est donc de « faire rentrer » le système à l'équilibre le plus vite possible. L'importance de la gestion d'un tel système est donc primordiale, un réseau cybernétique mal configuré peut donner de mauvais résultats. Tout l'intérêt d'un tel système peut donc être perdu s'il n'est pas bien pensé en amont.

La totalité des capacités en fonction des scénarios et des redistributions est résumée dans le Tableau 3.7. Il peut être constaté que la redistribution anticipée est effectivement la plus adaptée à ce système alors que la non-redistribution offre bien des performances mauvaises excepté pour le scénario 1.

Tableau 3.7 Résumé des résultats de capacité de tous les scénarios pour tous les types de redistributions (en nombre de demandes pour 120 itérations)

	<b>Non-redistribution</b>	<b>Redistribution équilibrée</b>	<b>Redistribution anticipée</b>
Scénario 1	5520	5520	
Scénario 2	240	3120	3360
Scénario 3	720	3600	3840
Scénario 4	240	1680	1680
Scénario 5	240	2400	2640
Scénario 6	240	960	1440

### 3.4 Comparaison entre les scénarios

Il est aussi intéressant de comprendre en quoi la répartition des nœuds influe sur l'efficacité du réseau. La longueur trajet moyen expérimental est assimilée avec la capacité. Si le trajet moyen est plus long il est logique de penser que la capacité en sera diminuée (les capsules passent plus de temps à transporter les gens et donc satisferont moins les demandes des utilisateurs). Cependant à trajet moyen égal, le nombre de capsules étant fixe on s'attend à avoir la même capacité pour tous les scénarios. Or cela n'est pas le cas comme il sera démontré ci-dessous. Il est intéressant de comparer le scénario 2 et 3 (Figure 3.9).

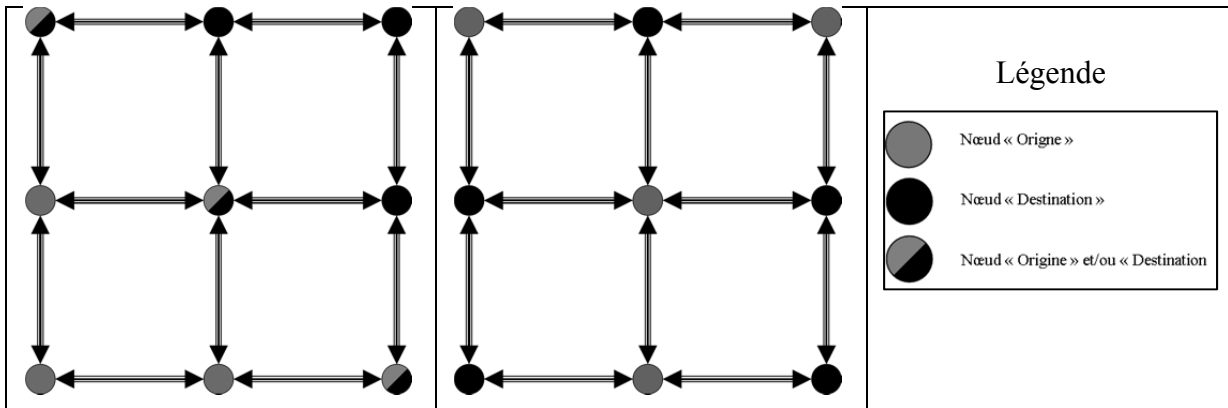


Figure 3.9 Représentation du scénario 2 (à gauche) et du scénario 3 (à droite)

Dans le scénario 2 il y a plus de nœuds « Origines » que dans le 3 (6 nœuds « Origines » dans le scénario 2 contre 4 dans le scénario 3). Ainsi les temps d'attente moyen sont comparés pour les deux scénarios voir Figure 3.10.

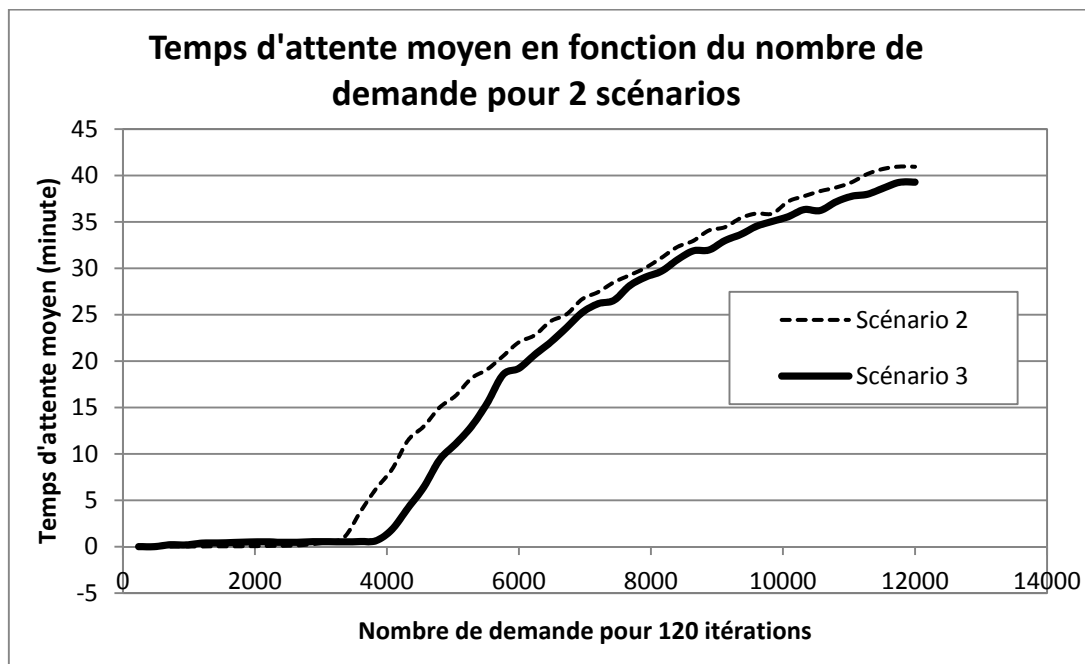


Figure 3.10 Temps d'attente moyen en fonction du nombre de demande pour 2 scénarios

Le scénario 3 offre une capacité supérieure (3840 demandes contre 3360 demandes pour le scénario 2). Il est donc plus important de répartir les pôles attractifs plutôt que leur nombre. Les nœuds « Destinations » ont aussi leurs importances en effet en comparant les scénarios 5 et 6, qui offrent sensiblement les mêmes performances, il apparaît que répartir les nœuds « Origine » ne sert à rien si on n'équilibre pas les nœuds « Destinations ». Pour maximiser le flux de transport il est important d'utiliser le plus de tronçons possibles. Si un pôle attractif le permet (zone marchande, commerces etc.) il est préférable de multiplier les arrêts/stations à ces endroits. Ainsi multiplier et décentraliser les points attractifs sur un territoire et synonyme d'amélioration de la capacité à répondre aux besoins individuels en déplacement. Ainsi la Figure 3.11 présente cette comparaison.

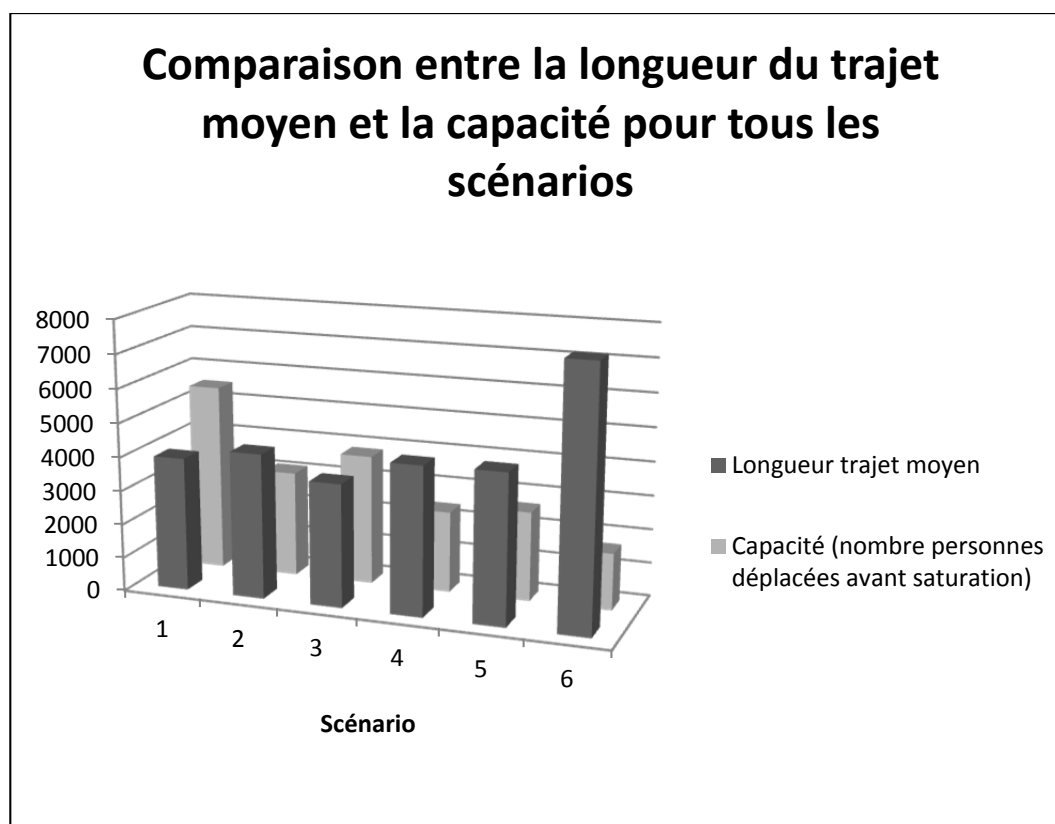


Figure 3.11 Comparaison entre la longueur du trajet moyen et la capacité pour tous les scénarios

Plus les nœuds « Origines » sont nombreux et répartis partout sur le réseau plus le système sera efficace. Par exemple le scénario 1 et 3 ont sensiblement les mêmes longueurs de trajet moyen (respectivement 3,9 itérations et 3,6 itérations) cependant le scénario 1 est celui qui va offrir les meilleures performances (Capacité de 5520 demandes pour le scénario 1 contre 3840 demandes pour le scénario 3). Ainsi plus les nœuds sont équitablement réparties plus le système est fluide. Cela va à l'encontre des plans d'urbanisation des grandes villes. Généralement les pôles d'activités sont regroupés et forme un point unique attractif. Cette méthode est contre efficace quant à sa faculté de mobilité. Le scénario le plus efficace en terme de mobilité est le scénario 1, celui où tous les nœuds sont « Origines » et « Destinations ».

Les performances d'un tel système sont donc grandement variables, ainsi la technologie en elle-même ne présente pas forcément une solution efficace. Un transport cybernétique doit être grandement étudié avant d'être installé afin d'en maximiser sa capacité surtout si il est dépendant de stations fixes. Enfin la partie intelligente n'est elle aussi pas à négliger et il est important de gérer les véhicules même après la fin de leur trajet. Si le flux est anticipable alors il faut utiliser cette connaissance, s'il ne l'est pas alors il est important de répartir le plus justement les véhicules libres.

## **CHAPITRE 4**

### **DISCUSSION**

Les résultats ont été obtenus pour des conditions précises. Il s'agit dans ce chapitre discussion d'aborder la pertinence de ces conditions ainsi que les améliorations qui peuvent être apportées au modèle.

#### **4.1 L'influence entre les capsules**

L'influence entre les capsules n'a pas été abordée car elle dépend principalement de la technologie utilisée, il peut être quasiment nul par exemple pour un réseau suspendu sans intersection ou conséquent si l'on considère les voitures intelligentes. Il n'existe donc pas de réelle étude quant à l'influence et la congestion d'un système PRT cependant il est possible d'en faire une étude qualitative. En considérant le scénario 1 avec une redistribution de type : non-redistribution (les capsules sont laissées à elle-même). Le schéma de congestion suivant est considéré : chaque véhicule présent sur un tronçon augmente d'une itération le trajet sur ce tronçon. Par exemple si un véhicule se déplace entre les nœuds 5 et 6, il augmente, le temps de sa présence sur le tronçon, le temps de trajet d'un nouveau véhicule d'une itération. Le premier véhicule parcourra le tronçon en deux itérations, le deuxième parcourra le tronçon en trois itérations. Ceci n'a pas pour but d'être quantifié en termes de temps de trajet. Cependant il est possible d'étudier le comportement du modèle et donc du système face aux problèmes de congestions. Les résultats du scénario 1 type : non-redistribution avec ou sans la congestion sont présentés Figure 4.1 :

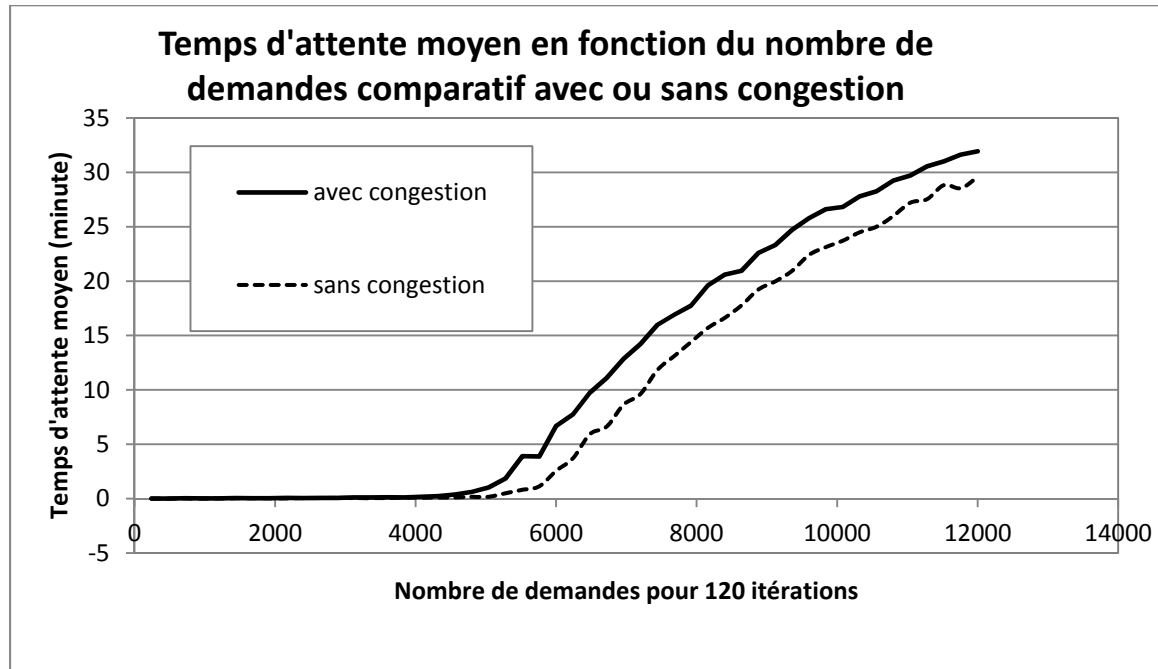


Figure 4.1 Temps d'attente en fonction du nombre de demandes, Scénario 1, redistribution A comparatif avec et sans congestion

La courbe qui représente les résultats avec congestion reste au dessus de la courbe représentant la non-congestion. Cependant l'écart reste assez faible par rapport à ce que l'on pourrait attendre. Cela est du à la gestion du système, en particulier, l'algorithme de Dijkstra, qui permet de calculer à tout instant le trajet le plus court. Si un véhicule suit un trajet donné mais qu'un tronçon va être congestionné inopinément, sa trajectoire sera corrigée. Ainsi le système peut « contourner », dans la mesure du possible, le problème. Le problème de congestion dans les PRT mériterait un approfondissement, cependant il reste très complexe à étudier car il n'existe pas encore assez de réelles données sur le sujet. Il est de plus très dépendant de la technologie étudiée. Et cette étude doit être faite dans un cas bien précis.



## 4.2 Les scénarios

L'enjeu d'un système intelligent est qu'il doit être réactif, il a été vérifié que si un tel système n'est pas correctement configuré il renvoie alors des performances moyennes voire mauvaises. Les performances d'un tel système ne sont pas garanties si sa configuration n'est pas adaptée. Il a surtout été question ici de scénarios de mobilité types déployés sur un réseau relativement simple. Cela peut être différent pour un système à échelle réelle avec un réseau plus étendu (liaison d'un plus grand nombre de centres d'attraction) mettant en jeu un nombre plus important de véhicules. Si un système est considéré à la taille d'une ville il serait évidemment impossible de prédire toutes les demandes. Dans ce cas, pour appréhender la réponse à des besoins en mobilité il serait alors possible d'anticiper les flux en fonctions des pôles attractifs (Zone d'habitation ou industrielle etc.) mais aussi en fonction de la météo. Par exemple le comportement des utilisateurs pourrait être modifié s'il neige ou s'il pleut, le système peut faire alors preuve de mémoire et donc réagir en fonction de paramètres extérieures (conditions météorologique, grèves etc.). Tous ces paramètres peuvent ainsi rentrer en jeu pour ainsi créer un vrai réseau intelligent au service des utilisateurs.

Dans cette étude l'existence de stations ou plutôt de nœuds à été considérée même si la technologie la plus intéressante est celle qui ne propose pas de tels arrêts. L'importance de la prévision à été démontrée, or de nos jours les flux de transports sont dictés par les réseaux rigides des transports en commun. Ainsi la prévision d'un tel flux paraît difficile car elle ne peut pas vraiment être étudiée avec les conditions actuelles. Comme les trajets des utilisateurs sont définis par des « chemins » préétablis, les trajets des utilisateurs s'ils ont à disposition un système libre restent encore à définir.

### **4.3 Application dans la réalité**

Les possibilités d'application ainsi que leurs conséquences d'un système de transport intelligent sont nombreuses et certaines d'entre elles sont présentées dans cette section.

#### **4.3.1 Comparaison entre la capacité d'un PRT et d'une rame de métro**

Il est intéressant de s'interroger sur la capacité d'un tel système, c'est-à-dire le nombre de passagers qu'il peut transporter. Pour le scénario 1 un tel système a une capacité de 5520 demandes pour un temps de deux heures. En moyenne, les projets actuels proposent des véhicules de 4 places, ce qui fait donc une capacité de 22 080 personnes. La STM (Société de Transport de Montréal), dans un communiqué de presse annonce que les métros montréalais ont une capacité de 100 à 110 passagers par voiture soit de 900 à 990 passagers par rame (Source : Société de Transport de Montréal). Il est à noter que la capacité énoncée pour un système PRT est pour une attente nulle ou quasi nulle alors que les temps d'attente du métro sont indépendants de l'utilisateur. Évidemment ces deux types de transports sont difficiles à comparer et le but n'est pas de déterminer si l'un est meilleur que l'autre mais de montrer que malgré ce que l'on pourrait penser le fait d'avoir des petits véhicules ne diminue pas la capacité, au contraire, ce qui est perdu en capacité pour un véhicule est gagné en fréquence d'usage. Une voiture transporte moins de passagers mais elle en transporte plus souvent.

#### **4.3.2 Possibilité d'application d'un tel système**

Comme le montre Lawson (2004), un trajet d'un point A à un B en transport en commun se dissèque en 4 étapes :

- Marcher de A à la station C
- Attente à la station C
- Trajet en transport en commun (bus, métro etc.) de C à D, avec tous les temps d'attentes aux stations intermédiaire
- Marcher de D à B

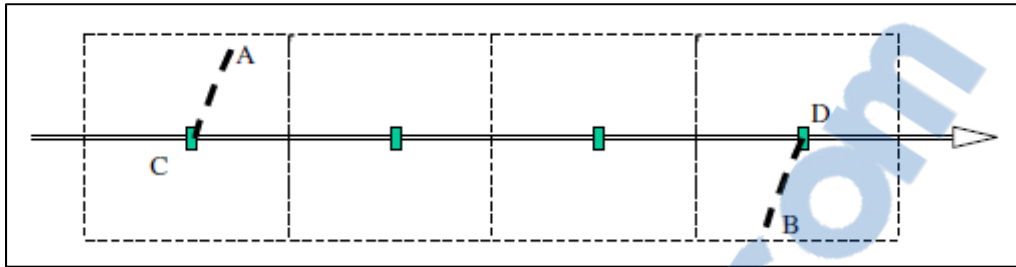


Figure 4.2 Schéma d'un trajet en transport en commun  
Tirée de Lawson (2004)

Ainsi il est possible d'imaginer plusieurs utilisations au réseau PRT. Soit l'utilisateur utilise le transport cybernétique du point C au point D, il remplace donc le bus ou le métro par une capsule privée et ainsi il supprime les temps d'attente aux stations intermédiaire. L'intérêt majeur de ce genre de transport est évidemment de faire un trajet sans interruption. Soit l'utilisateur l'utilise du point A au point C ou du point D à B, autrement dit il le remplace par le temps de marche. Ainsi dans l'équation de la multimodalité, le transport PRT rajoute une variable intéressante. Il n'est pas nécessaire de voir un réseau PRT comme un remplacement ou une amélioration des transports connus mais plutôt comme un complément. Dans l'équation de Lawson les transports en commun sont majorés par ce temps de marche car il oblige d'avoir les stations rapprochées (d'environ 500 mètres pour un maximum d'efficacité), diminuer ce temps de marche permet donc d'éloigner les stations. La vitesse de marche (4,8 km/h) est remplacée par la vitesse « pessimiste » d'un PRT (20 km/h, mais les valeurs réels sont plus proche de 40 km/h), le trajet entre C et D est fait en bus (30 km/h), les résultats sont présentés Figure 4.3.

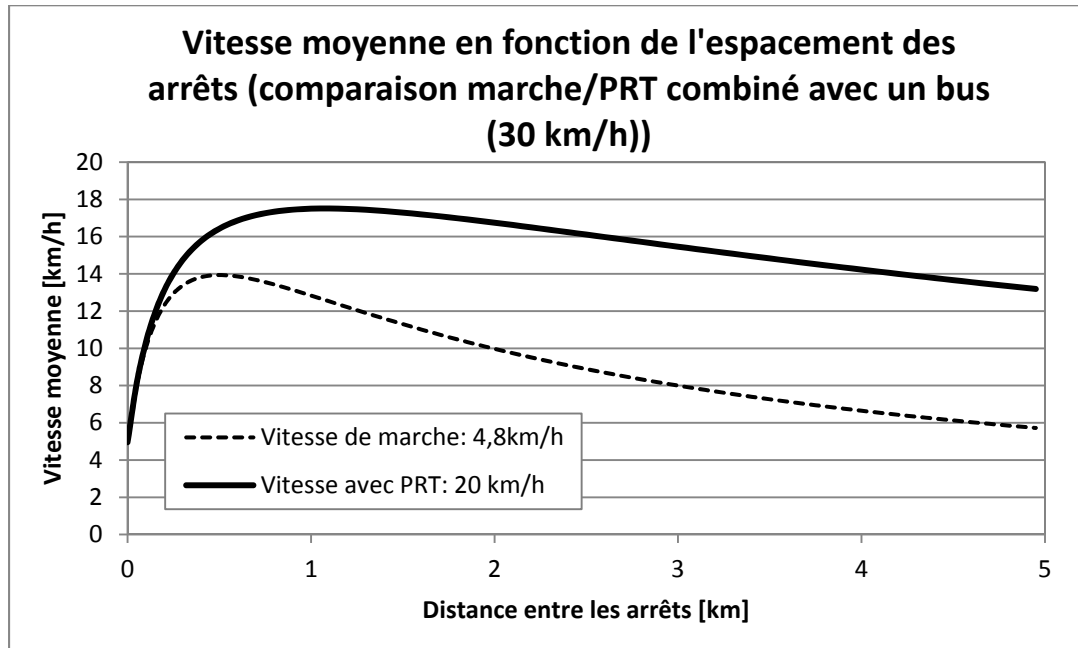


Figure 4.3 Calculs de Lowson, vitesse moyenne en fonction de l'espacement des arrêts avec un bus (30km/h)

Ainsi la vitesse moyenne maximale passe de 14 km/h à 18 km/h, cela signifie qu'il est possible de diminuer drastiquement les temps d'attentes ainsi que les phases d'accélération. Il est possible d'améliorer encore cette idée si on combine un PRT avec un métro ou un train. La Figure 4.4 présente les résultats de vitesse moyenne si un train (70km/h) est combiné avec la marche (4,8 km/h) et un PRT (40 km/h).

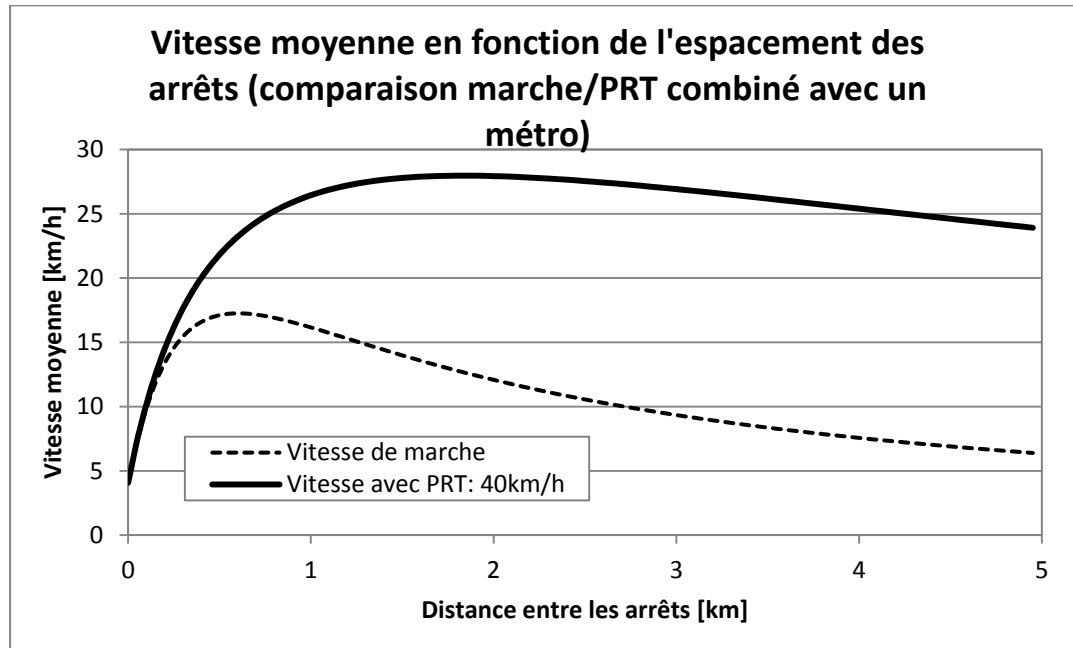


Figure 4.4 Calculs de Lowson, vitesse moyenne en fonction de l'espacement des arrêts avec un métro (70km/h)

Ici la vitesse moyenne maximale passe de 17 km/h pour la marche à 28 km/h pour un PRT. Ainsi la combinaison d'un transport rapide plus d'un PRT permet d'augmenter considérablement l'efficacité de ce même système.

### 4.3.3 Efficacité énergétique

Le taux d'occupation détermine si les véhicules roulent souvent « à vide » c'est-à-dire sans passager. Il détermine donc aussi si le réseau est efficace énergétiquement. Ce coefficient permet donc de choisir une solution à une autre d'un point de vue énergétique. Pour un temps d'attente moyen équivalent, ce qui arrive souvent lors de petites demandes, le taux d'occupation est plus grand pour une non-redistribution. La redistribution équilibrée est celle qui fournit les pires résultats énergétiques dus à trop de mouvement inutiles des capsules. Cependant il s'agit de la redistribution la plus apte à être utilisée car la redistribution anticipée n'est pas forcément réalisable. Par exemple, la Figure 4.5 présente le taux d'occupation pour le scénario 5.

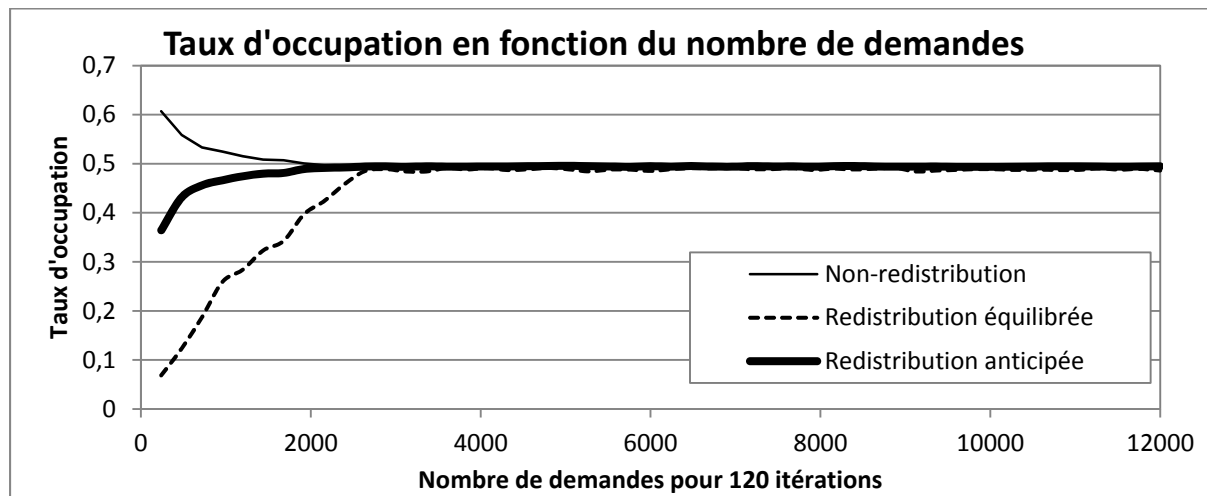


Figure 4.5 Taux d'occupation en fonction du nombre de demande pour trois types de redistribution avec gestion ACO (Scénario 5)

Une redistribution semi-équilibrée peut être développée, qui consisterait à répartir au moins un véhicule à chaque station. Lorsque les demandes sont faibles, pendant la nuit par exemple, cette redistribution s'assurerait qu'au moins un véhicule soit présent à chaque station sans pour autant s'assurer de l'équité de cette redistribution.

A l'ère où notre consommation devient « intelligente » et « verte » (téléphone, logement etc.) il paraît rationnel de penser que la prochaine étape sera à faire dans notre façon de penser la mobilité. Beaucoup d'efforts sont fait technologiquement pour améliorer notre consommation individuelle mais peu d'efforts sont faits pour améliorer les transports en commun. Les progrès proposés semblent être anecdotiques face aux technologies existantes. Par exemple il est facilement imaginable qu'un téléphone intelligent couplé avec des transports PRT et classiques pourront, en calculant en temps réel, le trajet optimal en temps réel et réservé une capsule individuelle précisément quand un utilisateur en aura besoin. Ces technologies peuvent être complémentaires et ainsi profondément changer notre mobilité. Même si le temps de trajet ne sera pas forcément grandement atteint par ces avancées, il est important de modifier notre mobilité de façon responsable.

## CONCLUSION

La mobilité urbaine arrive inexorablement à sa limite, la croissance de la population, l'étalement urbain et le manque d'alternative à l'automobile condamne la mobilité à une saturation certaine. Ce phénomène de saturation est déjà courant dans toutes les métropoles du monde pendant les heures de pointe où la congestion des automobiles et l'engorgement des transports en commun provoquent pollution, ralentissement économique, stress etc. Les PRT (Personal Rapid Transit) sont la combinaison parfaite du transport en commun et individuel. La maîtrise de l'électronique, de la mécanique et de l'informatique ont permis de repousser les limites technologiques des premiers projets PRT. De tels systèmes sont désormais fiables, technologiquement et économiquement viables. Ce type de transport ne se propose pas comme un remplacement des transports en commun mais plutôt comme un complément. Il permet de redéfinir la mobilité afin d'améliorer tous les transports déjà existants.

Cependant la capacité d'un tel système est très soumise aux variables qui le composent. Une fois la technologie arrêtée et définie il est important de comprendre les enjeux d'un réseau intelligent où les capsules ne sont pas faites pour transporter beaucoup d'usagers. Ce qu'un réseau PRT perd en capacité par capsule il le gagne en efficacité. Il est donc nécessaire de maximiser cette efficacité.

Cette étude a permis de montrer que les performances sont dépendantes de la répartition des points attractifs et donc de la distribution des flux de passagers et de la gestion de la flotte de véhicule. La compétitivité d'un tel système repose donc grandement sur la préparation des infrastructures ainsi que de la programmation de sa partie commande. Effectivement l'emplacement des stations est déterminant. La partie commande se scinde en deux critères : la façon de répartir en temps réel les capsules afin de fluidifier le trafic et la façon d'anticiper les futures demandes. Alors que l'anticipation des futures demandes peut être faite grâce à une fonction mémoire du réseau, la répartition doit être faite en temps réel par de puissants ordinateurs afin de pouvoir à tout moment calculer le meilleur trajet possible non pas pour un

véhicule mais pour l'ensemble de tous les véhicules. Ainsi naît l'idée d'un réseau intelligent où tous les véhicules communiquent entre eux. L'apparition de voitures qui se conduisent toutes seules relance aussi de leur côté l'idée un réseau intelligent. Les voitures reliées à un GPS pourraient toutes communiquer leurs positions afin de gérer les flux automobiles d'une façon altruiste. Ainsi, la mobilité urbaine actuelle n'est pas à l'image des possibilités technologiques et il existe un réel potentiel pour l'améliorer.



## ANNEXE I

### Présentation des algorithmes de utilisés dans le modèle dynamique

Premièrement la structure du modèle est rappelée sous forme de schéma à la Figure-A I-1 puis chaque fonction est présentée sous sa forme telle qu'elle à été utilisé sous Matlab.

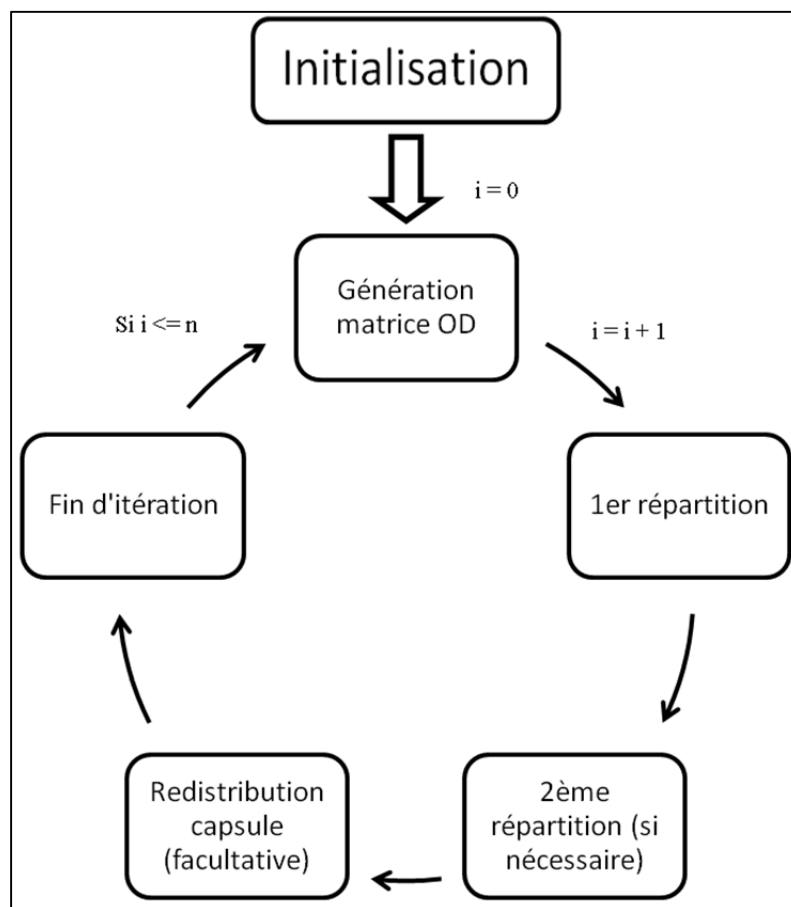


Figure-A I-1 Schéma simplifié du modèle dynamique

Ce schéma simplifié correspond à la base du code présentée

Figure-A I-2. Pour des soucis de lisibilité les variables sont remplacées par des points. Le fichier main.m correspond donc au code de base qui fera appel à d'autres fonctions pendant les itérations.

```
[..]=informations; % Phase d'initialisation
[..] = initialisation_capsules(..); % Phase d'initialisation des capsules
for itt=1:dt_max %Pour le nombre d'itération n, usuellement 120 itérations
    [OD,OD_memoire,OD_x]=generation_matrice_od(..); %Génération de la matrice OD
    [..] = repartition_caps_pers(..); %1er répartition
    if size(OD,1)~=0
        [..]=identification_capsules_proche(..);%Création de la matrice des distances
        [..] = ACO_original(..);%Utilisation de l'algorithme de colonie de fourmis
        [..] = repartition_caps_pers2(..);
        %2ème répartition
    end
    [..] = redistribution_capsules(..);
    %Redistribution des capsules
    [..] = fin_tour(..);
    %Fin d'itération
end
```

Figure-A I-2 Fichier main.m simplifié

```
function [..]=informations
n_veh=200; %nombre de véhicules/capsules
dt_max=120; %nombre d'itérations
OD=[];
OD_memoire=[];
capsules=zeros(n_veh,10);
OD_x=1;
compteur_capsules=zeros(1,n_veh);
compteur_kilometrage=zeros(1,n_veh);

[matrice_reseau]=reseau;
taille_reseau=size(matrice_reseau,1);
```

Figure-A I-3 Initialisation-1

```

function [ matrice_reseau ] = reseau()
%etablit la matrice du reseau, pour l'instant le reseau est un reseau à 4
%mailles et 9 noeuds.
matrice_reseau=zeros(9);
t=zeros(9);
t(1,2)=2;
t(1,4)=2;
t(2,1)=2;
t(2,5)=2;
t(2,3)=2;
t(3,2)=2;
t(3,6)=2;
t(4,1)=2;
t(4,5)=2;
t(4,7)=2;
t(5,2)=2;
t(5,4)=2;
t(5,6)=2;
t(5,8)=2;
t(6,3)=2;
t(6,5)=2;
t(6,9)=2;
t(7,4)=2;
t(7,8)=2;
t(8,5)=2;
t(8,7)=2;
t(8,9)=2;
t(9,6)=2;
t(9,8)=2;

for i=1:9
    for j=1:9
        if t(i,j)==0
            matrice_reseau(i,j)=inf;
        else
            matrice_reseau(i,j)=t(i,j);
        end
    end
end
end
end

```

Figure-A I-4 Initialisation-2

```

function [ capsules ] = initialisation_capsules( capsules,matrice_reseau )
%INITIALISATION_CAPSULES répartie à t=0 les capsules dans le réseau. Ici on
%prend une répartition égales à tous les noeuds.

n=size(matrice_reseau,1);
a=size(capsules,1);
b=fix(a/n);

for i=0:n-1
    for j=1:b
        capsules(j+b*i,2)=i+1;
        capsules(j+b*i,3)=i+1;
        capsules(j+b*i,5)=i+1;
    end
end

for i=b*n:a
    a=fix(rand*n+1);
    capsules(i,2)=a;
    capsules(i,3)=a;
    capsules(i,5)=a;
end

end

```

Figure-A I-5 Initialisation-3

```

function [..]=generation_matrice_od(..)
a=fix(n_veh*alpha);
D=[1 2 3 5 6 9];
E=[1 4 5 7 8 9];
for i=1:a
    E_temp=E;
    d=fix(rand*size(D,2))+1;
    B=E==D(d);
    E_temp(B)=[];
    e=fix(rand*size(E_temp,2))+1;
    OD=[OD; D(d) E_temp(e) OD_x 0];
    OD_memoire=[OD_memoire; D(d), E_temp(e), OD_x, 0, 0, itt, 0, 0];
    OD_x=OD_x+1;
    clear E_temp;
end

```

Figure-A I-6 Génération Matrice O-D (Pour scénario 2)

```

function [...] = repartition_caps_pers(...)
%Fonction de répartition des capsules pour répondre aux demandes des
%usagers

%Code capsules:
%0: véhicule libre et immobile
%1: véhicule libre et mobile (pour équilibrer la répartition aux noeuds)
%2: véhicule occupé avec un passager
%3: véhicule attribué (mobile, sans passager)

%Les véhicules libres et immobiles (marqués 0) sont attribués directement
%aux usagers présents aux même noeuds.
nbr_demandes=size(OD,1);
nbr_capsules=size(capsules,1);
OD_temp(:,1:2)=OD(:,1:2);

OD_unique=unique(OD_temp,'rows');
matrice_reseau_critere=zeros(size(matrice_reseau,1));
OD_unique_prochain_noeud=zeros(size(OD_unique,1),3);

for i=1:size(OD_unique,1)
    [trajet,temps_trajet]=dijkstra(matrice_reseau,OD_unique(i,1),OD_unique(i,2));
    OD_unique_prochain_noeud(i,:)=[OD_unique(i,:),trajet(2)];
    matrice_reseau_critere(OD_unique(i,2),OD_unique(i,1))=temps_trajet;
end

for i=1:nbr_capsules
    if capsules(i,1)==0
        for j=1:nbr_demandes
            if OD(j,1)==capsules(i,2)
                a=OD_unique_prochain_noeud(:,1)==OD(j,1);
                b=OD_unique_prochain_noeud(:,2)==OD(j,2);
                c=a.*b;
                d=find(OD_unique_prochain_noeud(:,3).*c);
                capsules(i,1)=2;
                capsules(i,3)=OD_unique_prochain_noeud(d,3);
                capsules(i,4)=matrice_reseau(OD(j,1),OD_unique_prochain_noeud(d,3));
                capsules(i,5)=OD(j,2);
                capsules(i,6)=matrice_reseau_critere(OD(j,2),OD(j,1));
                capsules(i,9)=OD(j,3);
                compteur_capsules(1,i)=compteur_capsules(1,i)+1;
                x=OD_memoire(:,3)==OD(j,3);
            end
        end
    end
end

```

Figure-A I-71er répartition

```

        compteur_capsules(1,i)=compteur_capsules(1,i)+1;
        x=OD_memoire(:,3)==OD(j,3);
        y=find(x);
        OD(j,:)=[];
        OD_memoire(y,5)=1;
        OD_memoire(y,7)=i;
        nbr_demandes=size(OD,1);
        break
    end
end
end
end

%Mise à jour de la matrice OD_unique sans refaire les calculs par Djisktra
clear OD_temp;
OD_temp(:,1:2)=OD(:,1:2);
OD_unique=unique(OD_temp,'rows');
OD_unique_prochain_noeud_temp=OD_unique_prochain_noeud;
OD_unique_prochain_noeud=zeros(size(OD_unique,1),3);
for i=1:size(OD_unique,1)
    a=OD_unique_prochain_noeud_temp(:,1)==OD_unique(i,1);
    b=OD_unique_prochain_noeud_temp(:,2)==OD_unique(i,2);
    c=a.*b;
    d=find(OD_unique_prochain_noeud_temp(:,3).*c);
    OD_unique_prochain_noeud(i,:)=[OD_unique(i,:), OD_unique_prochain_noeud_temp(d,3)];
end
OD_unique_prochain_noeud_temp=[];

```

Figure-A I-8 1er répartition (suite)

```

function [..]=identification_capsules_proche(..)
% Création de la matrice h utile pour l'algorithme ACO

A=capsules(:,1)==0;
B=capsules(:,1)~=0;
identification_capsules=find(A);
capsules_temp=capsules;
capsules_temp(find(B),:)=[];
clear A

if size(OD,1)<=size(capsules_temp,1)
    OD_temp=OD;
else
    OD_temp=OD(1:size(capsules_temp,1),:);
end

d=zeros(size(OD_temp,1),size(capsules_temp,1));
h=d;

for i=1:size(matrice_reseau,1)
    nbr_demandes=OD_temp(:,1)==i;
    for j=1:size(matrice_reseau,1)
        [~,dist(j,i)]=dijkstra(matrice_reseau,j,i);
        A=capsules_temp(:,5)==j;
        A=+A;
        delta_t=capsules_temp(:,6).*A;
        A=A';
        A=A*dist(j,i);
        nbr_demandes=+nbr_demandes;
        B=nbr_demandes*A+nbr_demandes*delta_t';
        d=d+B;
    end
end

for i=1:size(d,1)%generating sight matrix.
    for j=1:size(d,2)
        if d(i,j)==0
            h(i,j)=1;
        else
            h(i,j)=1/d(i,j);
        end
    end
end
end

```

Figure-A I-9 Identification Capsules Proches

```

function [besttour,inv] = ACO_original(h,d,OD,capsules)
    %ACO Algorithmme de colonie de fourmis
    besttour=[];

    a=size(h,1);%nombre de demandes
    b=size(capsules,1);%nombre de véhicules
    c=min(a,b);
    end
    [t,h,iter,alpha,beta,e,m,el]=ants_information(h,OD);%Initialisation de tous les paramètres
    mincost=sum(sum(h));
    for i=1:iter
        [app]=ants_primaryplacing(m,c);%Placement des fourmis, la "ville" est choisit aléatoirement
        [app,at,pers]=ants_cycle(app,m,a,b,c,h,t,alpha,beta);%Les fourmis font leurs trajet
        if size(at)~= [0,0]
            at=horzcat(at,at(:,1));
        end
        [cost,f]=ants_cost(m,c,d,pers,el);%Calcul du "coût" de chaque trajet
        [t]=ants_traceupdating(m,c,t,at,pers,f,e);%Mise à jour des pistes de phéromones
        costoa(i)=mean(cost);
        [mincosttemp,bestanttemp]=min(cost);
        if mincost>mincosttemp %Determination du trajet le plus court
            mincost=mincosttemp;
            besttour=pers(bestanttemp,:);
        end
        iteration(i)=i;
    end
end

```

Figure-A I-10 « Pseudo code » de l'algorithme de colonie de fourmis utilisé dans le modèle



```

function [...] = repartition_caps_pers2(...)
%REPARTITION_CAPS_PERS2 Summary of this function goes here
% Detailed explanation goes here
id_capsules=identification_capsules;
OD_temp=OD;
if inv==0
    liste=[];
    for i=1:size(besttour,2)
        A=besttour(i);
        [trajet_temp,f] = dijkstra(matrice_reseau, capsules(id_capsules(A),2), OD_temp(i,1));
        if capsules(id_capsules(A),1)==0 && numel(trajet_temp)~=1
            capsules(id_capsules(A),1)=3;
            capsules(id_capsules(A),3)=trajet_temp(2);
            capsules(id_capsules(A),4)=matrice_reseau(capsules(id_capsules(A),2),capsules(id_capsules(A),3));
            capsules(id_capsules(A),5)=OD_temp(i,1);
            capsules(id_capsules(A),6)=f;
            capsules(id_capsules(A),7)=OD_temp(i,2);
            capsules(id_capsules(A),10)=OD_temp(i,3);
            compteur_capsules(1,id_capsules(A))=compteur_capsules(1,id_capsules(A))+1;
            x=OD_memoire(:,3)==OD_temp(i,3);
            y=find(x);
            OD_memoire(y,5)=2;
            OD_memoire(y,7)=id_capsules(A);
            liste=[i,liste];
        elseif capsules(id_capsules(A),1)==0 && numel(trajet_temp)==1
            [trajet_temp,f]=dijkstra(matrice_reseau, capsules(id_capsules(A),2), OD_temp(i,2));
            capsules(id_capsules(A),1)=2;
            capsules(id_capsules(A),3)=trajet_temp(2);
            capsules(id_capsules(A),4)=matrice_reseau(capsules(id_capsules(A),2),capsules(id_capsules(A),3));
            capsules(id_capsules(A),6)=f;
            capsules(id_capsules(A),5)=OD_temp(i,2);
            capsules(id_capsules(A),9)=OD_temp(i,3);
            compteur_capsules(1,id_capsules(A))=compteur_capsules(1,id_capsules(A))+1;
            x=OD_memoire(:,3)==OD_temp(i,3);
            y=find(x);
            OD_memoire(y,5)=1;
            OD_memoire(y,7)=id_capsules(A);
            liste=[i,liste];
        end
    end
end

```

Figure-A I-11 2<sup>ème</sup> répartition (1)

```

else
    if capsules(id_capsules(A),5)==OD_temp(i,1) && capsules(id_capsules(A),7)==0
        capsules(id_capsules(A),7)=OD_temp(i,2);
        capsules(id_capsules(A),10)=OD_temp(i,3);
        compteur_capsules(1,id_capsules(A))=compteur_capsules(1,id_capsules(A))+1;
        x=OD_memoire(:,3)==OD_temp(i,3);
        y=find(x);
        OD_memoire(y,5)=2;
        OD_memoire(y,7)=id_capsules(A);
        liste=[i,liste];
    elseif capsules(id_capsules(A),5)~=OD_temp(i,1) && capsules(id_capsules(A),7)==0
        capsules(id_capsules(A),7)=OD_temp(i,1);
        capsules(id_capsules(A),8)=OD_temp(i,2);
        capsules(id_capsules(A),10)=OD_temp(i,3);
        compteur_capsules(1,id_capsules(A))=compteur_capsules(1,id_capsules(A))+1;
        x=OD_memoire(:,3)==OD_temp(i,3);
        y=find(x);
        OD_memoire(y,5)=2;
        OD_memoire(y,7)=id_capsules(A);
        liste=[i,liste];
    end
end
end
OD(liste,:)=[];
else
    liste=[];
    for i=1:size(besttour,2)
        A=besttour(i);
        if capsules(i,7)==0
            if capsules(i,5)==OD_temp(A,1);
                capsules(i,7)=OD_temp(A,2);
                capsules(i,10)=OD_temp(A,3);
                x=OD_memoire(:,3)==OD_temp(A,3);
                y=find(x);
                OD_memoire(y,5)=2;
                OD_memoire(y,7)=i;
            else
                capsules(i,7)=OD_temp(A,1);
                capsules(i,8)=OD_temp(A,2);
                capsules(i,10)=OD_temp(A,3);
                x=OD_memoire(:,3)==OD_temp(A,3);
                y=find(x);
                OD_memoire(y,5)=2;
                OD_memoire(y,7)=i;
            end
        end
    end
end

```

Figure-A I-12 2ème répartition (2)

```

        end
        liste=[A,liste];
    end
end
    OD(liste,:)=[];
end

%Mise à jour de la matrice OD_unique sans refaire les calculs par Djisktra
clear OD_temp;
OD_temp(:,1:2)=OD(:,1:2);
OD_unique=unique(OD_temp,'rows');
OD_unique_prochain_noeud_temp=OD_unique_prochain_noeud;
OD_unique_prochain_noeud=zeros(size(OD_unique,1),3);
for i=1:size(OD_unique,1)
    a=OD_unique_prochain_noeud_temp(:,1)==OD_unique(i,1);
    b=OD_unique_prochain_noeud_temp(:,2)==OD_unique(i,2);
    c=a.*b;
    d=find(OD_unique_prochain_noeud_temp(:,3).*c);
    OD_unique_prochain_noeud(i,:)=[OD_unique(i,:), OD_unique_prochain_noeud_temp(d,3)];
end
OD_unique_prochain_noeud_temp=[];

```

Figure-A I-13 2ème répartition (3)

```

function [...] = redistribution_capsules(...)
%REDISTRIBUTION CAPSULES A la fin d'un cycle on répartit les capsules pour
%qu'elles soient equitalement presentent sur le reseau.
demande_origine=[1 2 3 4 5 6 7 8 9];
nbr_demandes=size(demande_origine,2);
A=capsules(:,1)==0;
B=fix(sum(A)/nbr_demandes);
temp=zeros(nbr_demandes,1);
limite=size(nbr_demandes,1);
for i=1:nbr_demandes
    C=capsules(:,2)==demande_origine(i);
    D=A.*C;
    temp(i,1)=B-sum(D); %Vecteur de demandes le but étant de ramener les valeurs positives à 0
end
clear A;
clear B;
clear C;
clear D;
noeuds_visites=[];
noeuds_temp=[];
for i=1:nbr_demandes
    A=temp(i,1);
    if A>0
        n_stop=A-1; %nombre de capsules que l'on veut considerer dans notre calcul
        a=0;
        noeuds=demande_origine(i);
        stop=0;
        while a<=n_stop
            stop=stop+1;
            if stop==limite
                a=n_stop+1;
                noeuds_visites=1:size(matrice_reseau,1);
            end
            for j=1:size(noeuds,2)
                C=capsules(:,5)==noeuds(j);
                D=capsules(:,1)==0;
                E=C.*D;
                b=sum(E);%denombre le nombre de capsules presente au noeud i
                a=a+b;
                noeuds_visites=unique([noeuds, noeuds_visites]);
                noeuds_temp=unique([find(matrice_reseau(noeuds(j),:)<Inf), noeuds_temp]);
                if intersect(noeuds_visites,noeuds_temp)~=0
                    d=intersect(noeuds_visites,noeuds_temp);
                    for l=1:size(d,2)
                        k= noeuds_temp==d(l);
                        noeuds_temp(k)=[];
                    end
                end
            end
        end
    end
end

```

Figure-A I-14 Redistribution capsules

```

        end
    end
    end
    noeuds=noeuds_temp;
    noeuds_temp=[];
end
for j=1:size(noeuds_visites,2)
    [trajet,temps_trajet]=dijkstra(matrice_reseau,noeuds_visites(j),demande_origine(i));
    matrice_reseau_critere(j,demande_origine(i))=temps_trajet;
    if noeuds_visites(j)~=demande_origine(i)
        temp1=capsules(:,1)==0;
        temp2=capsules(:,2)==noeuds_visites(j);
        temp3=temp1.*temp2;
        x=find(temp3);
        if numel(x)~=0
            for k=1:size(x,2)
                capsules(x(k),1)=1;
                capsules(x(k),3)=trajet(2);
                capsules(x(k),4)=matrice_reseau(capsules(x(k),2),capsules(x(k),3));
                capsules(x(k),5)=demande_origine(i);
                capsules(x(k),6)=matrice_reseau_critere(j,demande_origine(i));
            end
        end
    end
end
noeuds_visites=[];
end
end
OD_temp(:,1:2)=OD(:,1:2);
OD_unique=unique(OD_temp,'rows');
OD_unique_prochain_noeud_temp=OD_unique_prochain_noeud;
OD_unique_prochain_noeud=zeros(size(OD_unique,1),3);
if size(OD,1)==0
    OD_unique_prochain_noeud=[];
else
    for i=1:size(OD_unique,1)
        a=OD_unique_prochain_noeud_temp(:,1)==OD_unique(i,1);
        b=OD_unique_prochain_noeud_temp(:,2)==OD_unique(i,2);
        c=a.*b;
        d=find(OD_unique_prochain_noeud_temp(:,3).*c);
        OD_unique_prochain_noeud(i,:)=[OD_unique(i,:), OD_unique_prochain_noeud_temp(d,3)];
    end
    OD_unique_prochain_noeud_temp=[];
end
end

```

Figure-A I-15 Redistribution Capsules (suite)

```

function [...] = fin_tour(...)
%FIN_TOUR On mets à jour la matrice capsules
for i=1:size(capsules,1)
    if capsules(i,1)==2;
        A=capsules(i,9);
        OD_memoire(A,8)=OD_memoire(A,8)+1;
    end
end

for i=1:size(capsules,1)
    if capsules(i,4)~=0
        capsules(i,4)=capsules(i,4)-1;
        capsules(i,6)=capsules(i,6)-1;
        compteur_kilometrage(1,i)=compteur_kilometrage(1,i)+1;
    end
    if capsules(i,4)==0%Si capsules(i,4)=0 alors la capsule est arrivé à un noeud
        capsules(i,2)=capsules(i,3);
        if capsules(i,3)==capsules(i,5)
            if capsules(i,8)~=0 && capsules(i,3)==capsules(i,5) && capsules(i,4)==0
%C'est le cas où la capsule récupère un passager et où les colonnes 7 et 8 sont non nuls
                capsules(i,1)=3;
                [trajet_temp,temps_trajet_temp] = dijkstra(matrice_reseau, capsules(i,5), capsules(i,7));
                if numel(trajet_temp)==1
                    i
                    capsules(i,:)
                end

                capsules(i,3)=trajet_temp(2);
                capsules(i,4)=matrice_reseau(capsules(i,2),capsules(i,3));
                capsules(i,5)=capsules(i,7);
                capsules(i,6)=temps_trajet_temp;
                capsules(i,7)=capsules(i,8);
                capsules(i,8)=0;
                capsules(i,9)=0;
                clear trajet_temp
                clear temps_trajet_temp

            elseif capsules(i,7)~=0 && capsules(i,8)==0
                capsules(i,1)=2;
                [trajet_temp,temps_trajet_temp] = dijkstra(matrice_reseau, capsules(i,5), capsules(i,7));
                capsules(i,3)=trajet_temp(2);
                capsules(i,4)=matrice_reseau(capsules(i,2),capsules(i,3));
                capsules(i,5)=capsules(i,7);
                capsules(i,6)=temps_trajet_temp;
                capsules(i,7)=0;
                capsules(i,9)=capsules(i,10);
            end
        end
    end
end

```

Figure-A I-16 Fin d'itération

```

capsules(i,10)=0;
clear trajet_temp
clear temps_trajet_temp
clear a
a=capsules(i,9)==OD_memoire(:,3);
b=find(a);
OD_memoire(b,5)=1;
else%C'est le cas où la capsule devient libre
capsules(i,1)=0;
capsules(i,6)=0;
capsules(i,9)=0;
end
else
%La capsule est donc à un noeud mais pas encore à destination,
%on cherche par dijkstra le prochain noeud
[trajet_temp,temps_trajet_temp] = dijkstra(matrice_reseau, capsules(i,2), capsules(i,5));
capsules(i,3)=trajet_temp(2);
capsules(i,4)=matrice_reseau(capsules(i,2),capsules(i,3));
capsules(i,6)=temps_trajet_temp;
clear trajet_temp
clear temps_trajet_temp
end

end
if capsules(i,5)==capsules(i,7) && capsules(i,8)~=0
capsules(i,7)=capsules(i,8);
capsules(i,8)=0;
end

end
for i=1:size(OD,1)
OD(i,4)=OD(i,4)+1;
end
clear a;
a=OD_memoire(:,5)~=1;
OD_memoire(a,4)=OD_memoire(a,4)+1;
end

```

Figure-A I-17 Fin d'itération (suite)

Enfin beaucoup de fonctions font appel à une autre fonction Dijkstra dont le code est présenté Figure-A I-18.

```

%Initialisation
n=size(matrix_costo,1);
S(1:n) = 0;      %liste des noeuds visités
dist(1:n) = inf; %Distance (ou "poids") des noeuds, il est infini pour tous les noeuds
prev(1:n) = n+1; % Noeud précédent, cela informe du meilleur chemin à prendre
dist(s) = 0;

%Fonction principale
while sum(S)~=n
    candidate=[];
    for i=1:n
        if S(i)==0
            candidate=[candidate dist(i)]; %On dresse la liste des candidats avec leurs "poids"
        else
            candidate=[candidate inf]; %Les noeuds non joignable ont un "poids" infini
        end
    end
    [~, u]=min(candidate); %Le noeud "u" est donc le noeud avec le moins de "poids"
    S(u)=1; %On se place donc au noeud "u"
    for i=1:n
        if(dist(u)+matrix_costo(u,i)<dist(i))
            dist(i)=dist(u)+matrix_costo(u,i); %Mise à jour du "poids" de tous les noeuds
            prev(i)=u; %Le paramètre "prev" sert à se souvenir du chemin parcouru
        end
    end
end

%Une fois tous les noeuds étudiés on "récupère" le chemin le plus court
trajet = [d]; %On part du noeud d'arrivé

while trajet(1) ~= s
    if prev(trajet(1))<=n
        trajet=[prev(trajet(1)) trajet];%Puis on remonte noeud à noeud
    else
        error;
    end
end;

temps_trajet = dist(d); %Le trajet le plus court est donc le "poids" total

```

Figure-A I-18 Algorithme de Dijkstra utilisé dans le modèle



## ANNEXE II

### Résultats de tous les scénarios

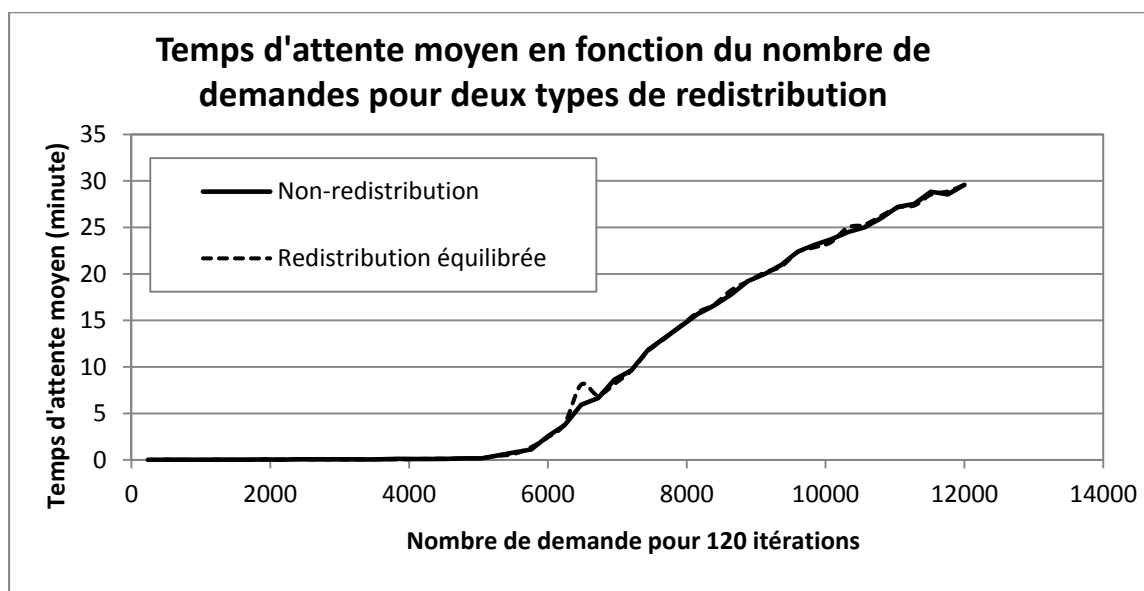


Figure-A II-1 Temps d'attente moyen en fonction du nombre de demandes (Scénario 1)

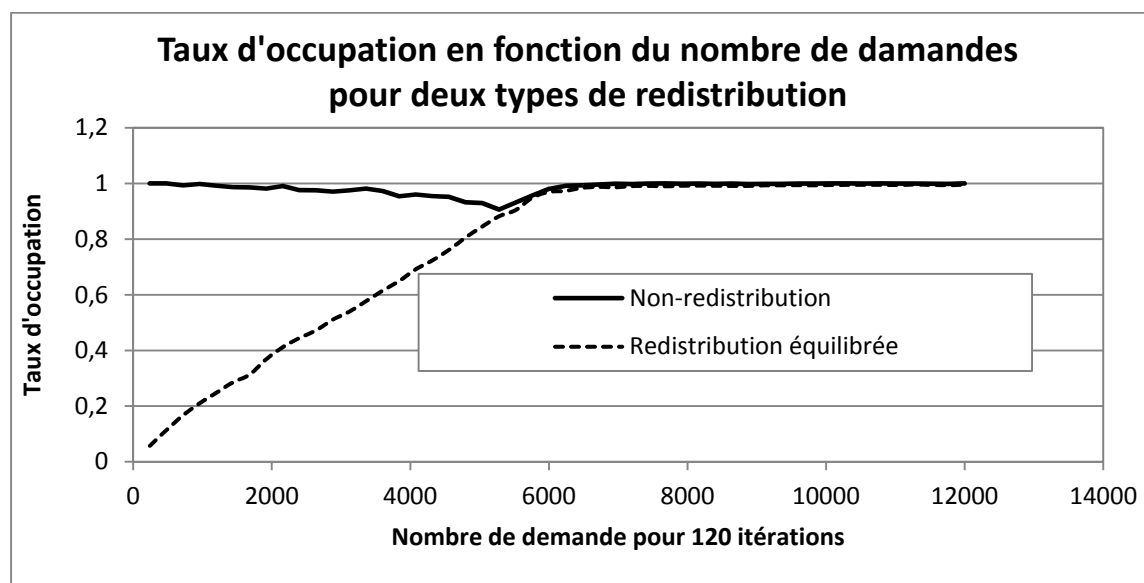


Figure-A II-2 Taux d'occupation en fonction du nombre de demandes (Scénario 1)



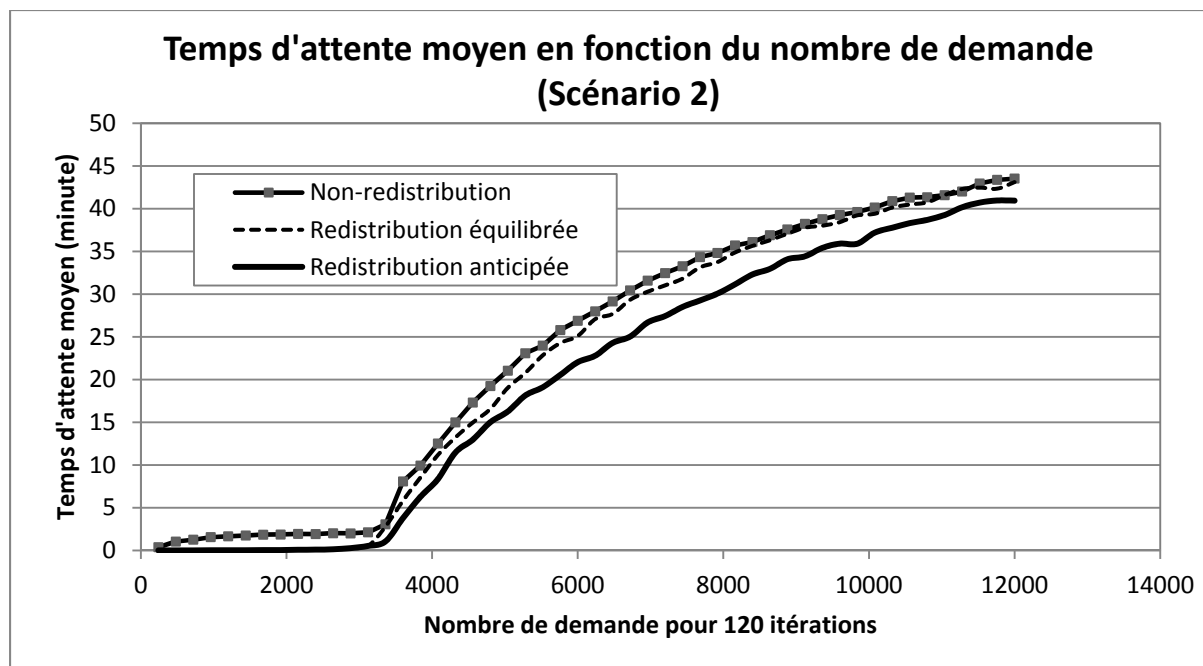


Figure-A II-3Temps d'attente moyen en fonction du nombre de demandes (Scénario 2)

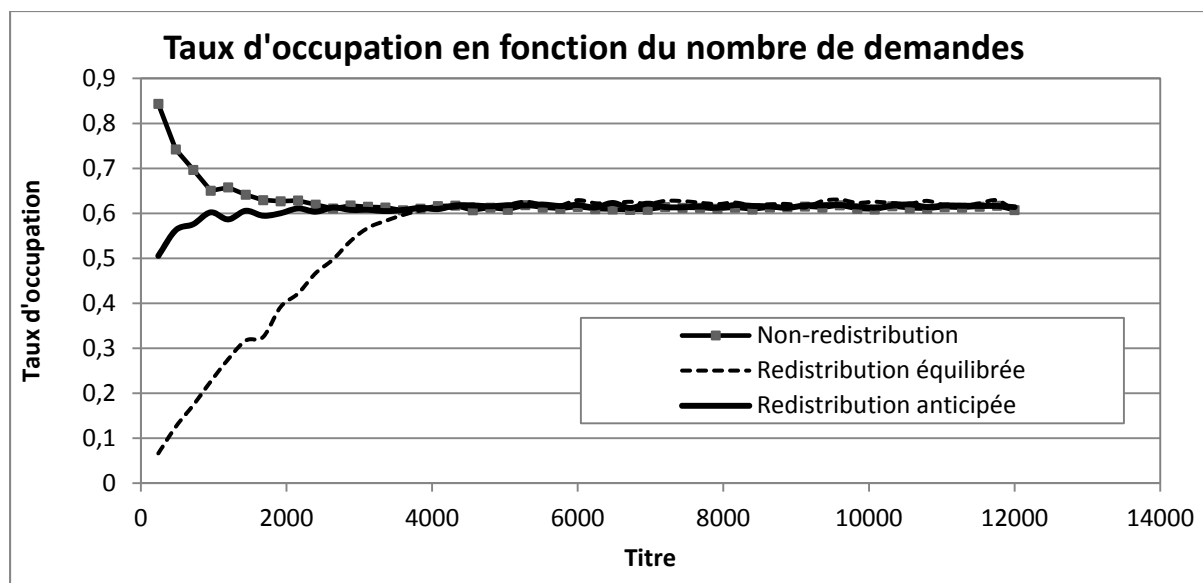


Figure-A II-4Taux d'occupation en fonction du nombre de demandes (Scénario 2)

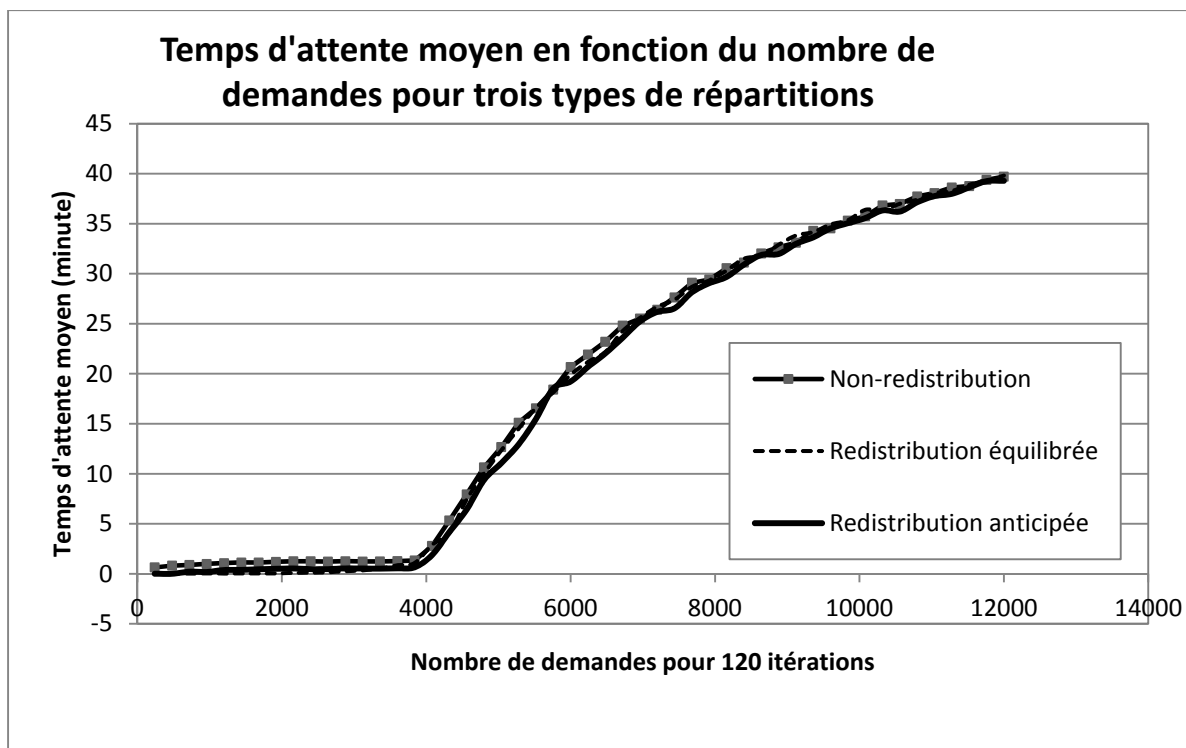


Figure-A II-5 Temps d'attente moyen en fonction du nombre de demandes (Scénario 3)

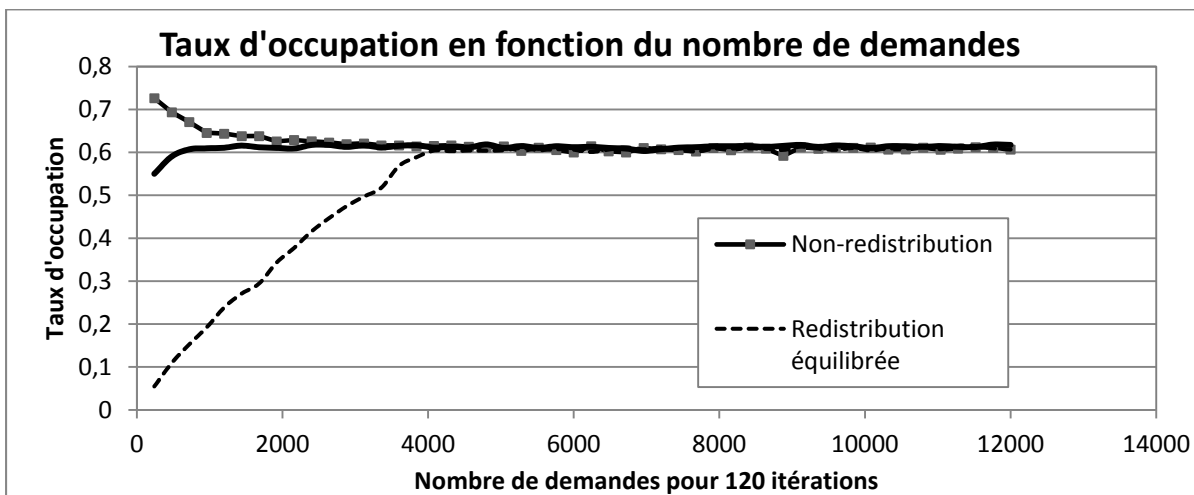


Figure-A II-6 Taux d'occupation en fonction du nombre de demandes (Scénario 3)

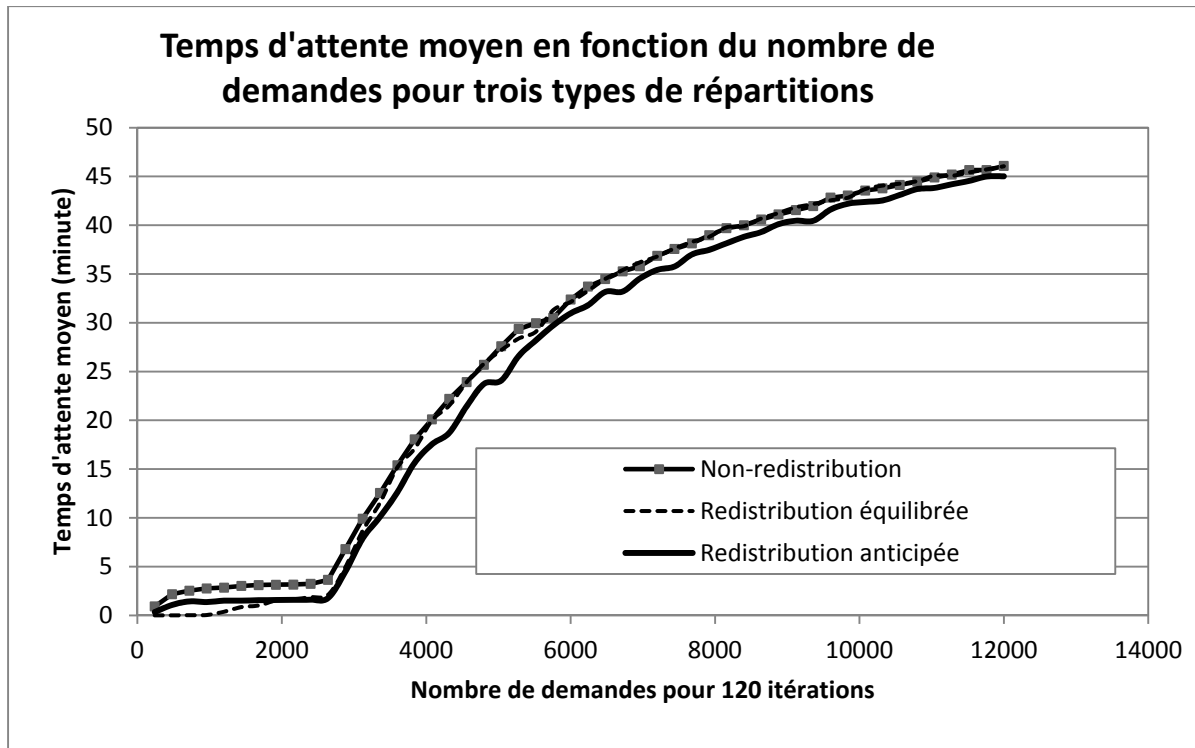


Figure-A II-7Temps d'attente moyen en fonction du nombre de demandes (Scénario 4)

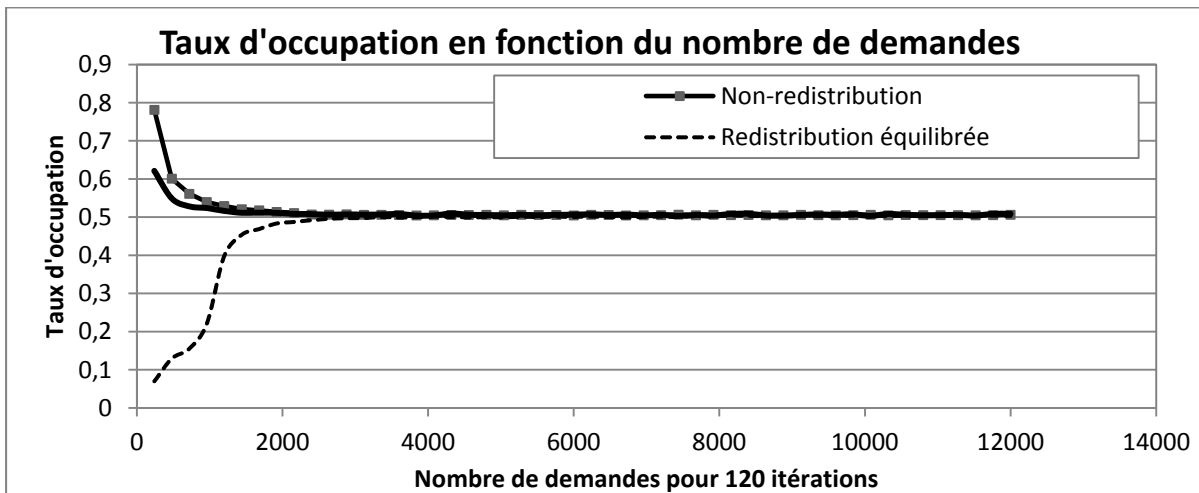


Figure-A II-8Taux d'occupation en fonction du nombre de demandes (Scénario 4)

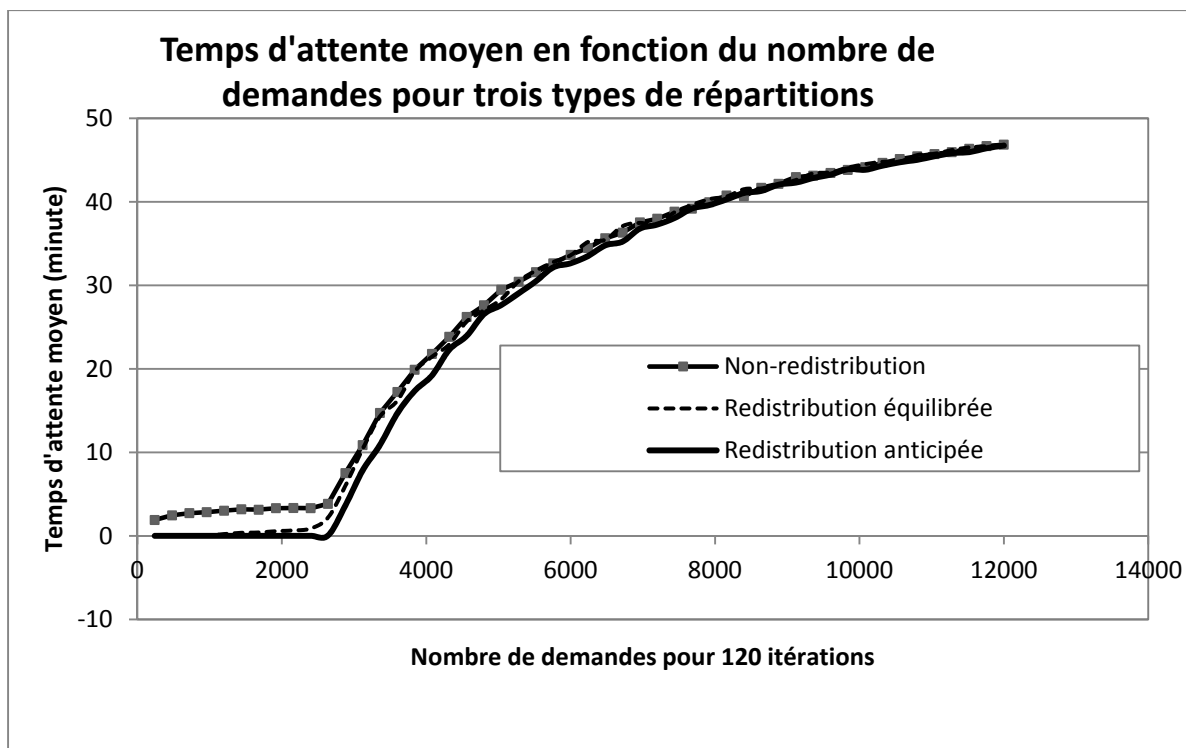


Figure-A II-9 Temps d'attente moyen en fonction du nombre de demandes (Scénario 5)

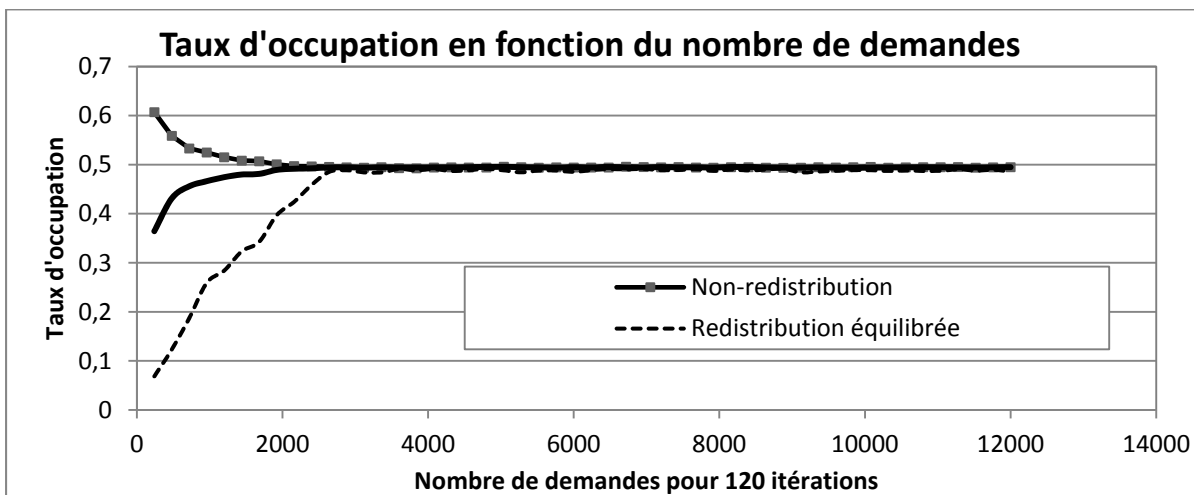


Figure-A II-10 Taux d'occupation en fonction du nombre de demandes (Scénario 5)

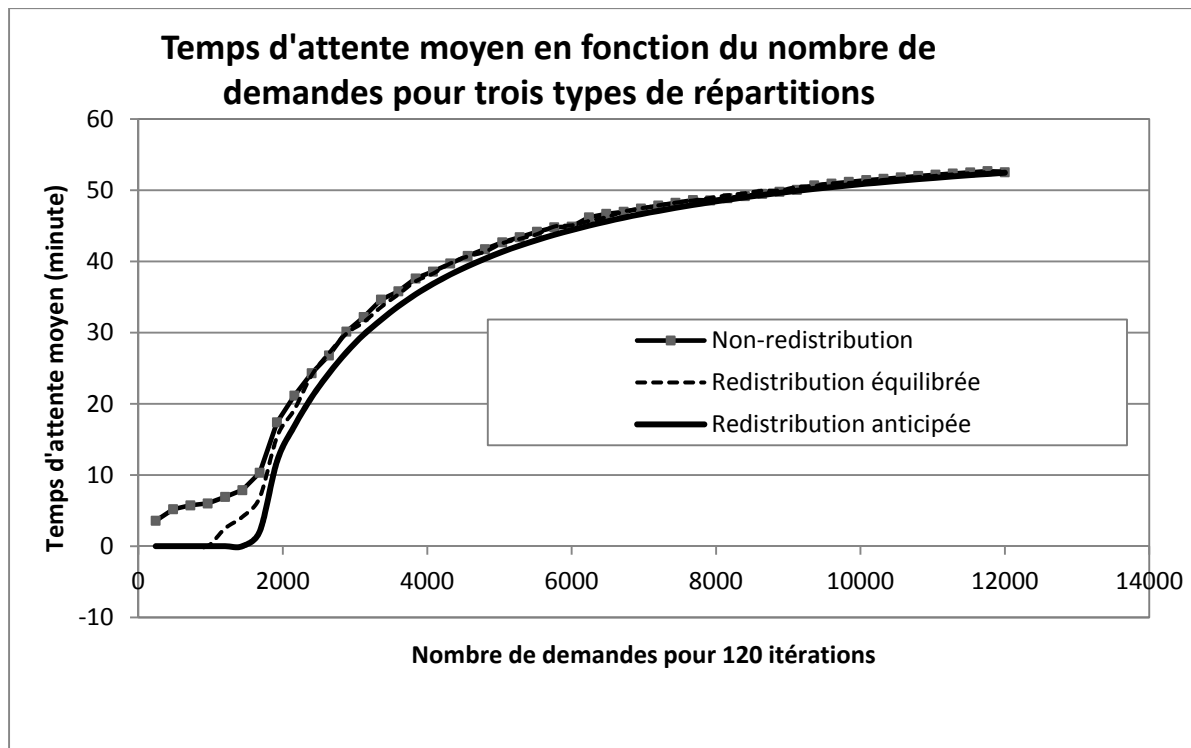


Figure-A II-11 Temps d'attente moyen en fonction du nombre de demandes (Scénario 6)

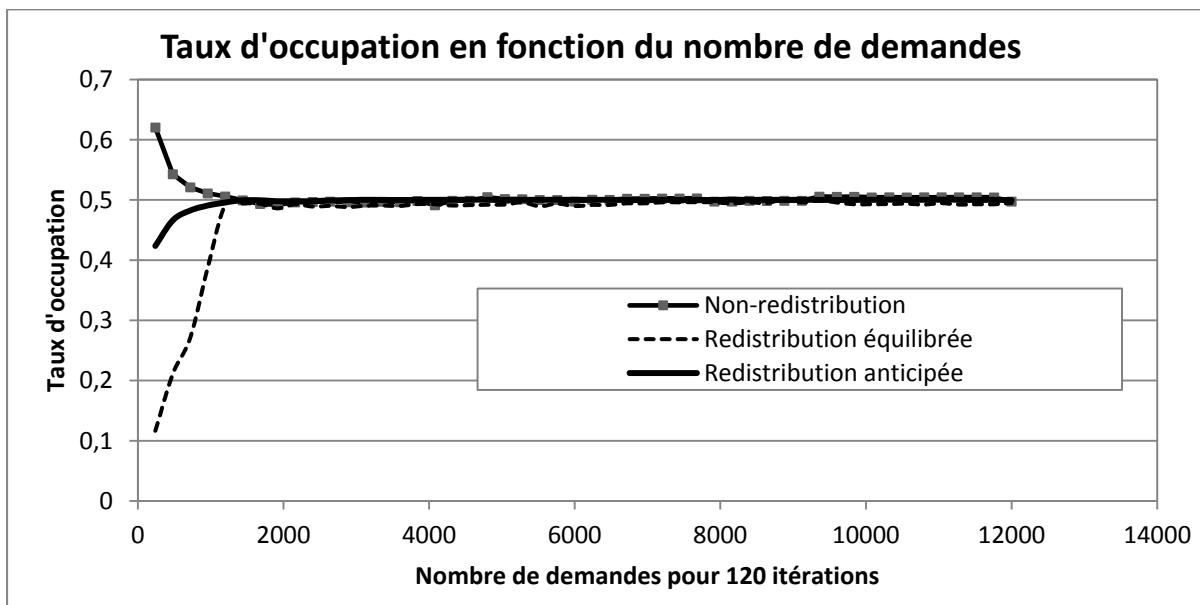


Figure-A II-12 Taux d'occupation en fonction du nombre de demandes (Scénario 6)

## BIBLIOGRAPHIE

- Anderson, J.E. 1974. « Personal Rapid Transit ». *Environment: Science and Policy for Sustainable Development*, vol. 16, n° 3, p. 6-11.
- Anderson, J.E. 2009. « How to design a PRT guideway ». In. Vol. 343, p. 436-449. American Society of Civil Engineers, 1801 Alexander Bell Drive Reston VA 20191-4400 USA.
- Dréo, J., A. Pétrowski, É.D. Taillard et P. Siarry. 2003. « Métaheuristiques pour l'optimisation difficile ».
- Dupuy, G. 2006. « La dépendance à l'égard de l'automobile ». *Transports, recherche, innovation le point sur*.
- Fichter, D. 1964. *Individualized automatic transit and the city*.
- Khichane, M., P. Albert et C. Solnon. 2009. « Un modèle réactif pour l'optimisation par colonies de fourmis: application à la satisfaction de contraintes ».
- Lowson, M. 2004. « Idealised models for public transport systems ». *International Journal of Transport Management*, vol. 2, n° 3-4, p. 135-147.
- Lowson, M.V. 2011. « Sustainable personal transportation ». In., p. 251-257. IEEE.
- Veuillez sélectionner un type de document autre que « Generic » afin de faire afficher la référence bibliographique.
- Turcotte, M. 2005. « Le temps pour se rendre au travail et en revenir ». *Enquête sociale sur l'emploi du temps: cycle*, vol. 19, p. 89-622.
- Anderson, J.E. 2006. « The future of high-capacity personal rapid transit ». *PRT International, LLC. Minneapolis, MN. Retrieved January*.
- Anderson, J.E. 2009. « How to design a PRT guideway ». In. Vol. 343, p. 436-449. American Society of Civil Engineers, 1801 Alexander Bell Drive Reston VA 20191-4400 USA.
- Coloni, A., M. Dorigo et V. Maniezzo. 1991. « Distributed optimization by ant colonies ». In. Vol. 142, p. 134-142.
- Dréo, J., A. Pétrowski, É.D. Taillard et P. Siarry. 2003. « Métaheuristiques pour l'optimisation difficile ».

- Dupuy, G. 1999. *La dépendance automobile: symptômes, analyses, diagnostic, traitements*. Anthropos.
- Elhaffar, A., et M. Lehtonen. 2005. « An improved GPS current traveling-wave fault locator in EHV transmission networks using few recordings ». In., p. 5 pp.-5. IEEE.
- Handy, S. 1993. « A cycle of dependence: automobiles, accessibility, and the evolution of the transportation and retail hierarchies ». *Berkeley Planning Journal*, vol. 8, p. 21-43.
- Latour, B. 1992. « Aramis, ou, L'amour des techniques (Textes à l'appui) ».
- Litman, T. 2002. « Transportation Cost and Benefit Analysis ». *Techniques, Estimates and Implications*, Victoria Transport Policy Institute ([www.vtpi.org/tca](http://www.vtpi.org/tca)).
- Lowson, M. 2004. « Idealised models for public transport systems ». *International Journal of Transport Management*, vol. 2, n° 3-4, p. 135-147.
- Newman, P.G., et J.R. Kenworthy. 1989. *Cities and automobile dependence: an international sourcebook*.
- Sousanis, J. 2011. « World Vehicle Population Tops 1 Billion Units ». *WardsAuto.com*, Aug.
- Turcotte, M. 2008. « Dépendance à l'automobile dans les quartiers urbains ». *Tendances sociales canadiennes*, vol. 85, p. 21-32.
- Weiner, E. 2008. *Urban transportation planning in the United States: history, policy, and practice*. Springer Verlag.
- Dréo, J., A. Pétrowski, É.D. Taillard et P. Siarry. 2003. « Métaheuristiques pour l'optimisation difficile ».
- Spiess, H. 1987. « A maximum likelihood model for estimating origin-destination matrices ». *Transportation Research Part B: Methodological*, vol. 21, n° 5, p. 395-412.