



HAL
open science

Simulation multi-agent de l'information des voyageurs dans les transports en commun

Amine Othman

► **To cite this version:**

Amine Othman. Simulation multi-agent de l'information des voyageurs dans les transports en commun. Système multi-agents [cs.MA]. Université Paris-Est, 2016. Français. NNT: . tel-01545376

HAL Id: tel-01545376

<https://hal.science/tel-01545376>

Submitted on 22 Jun 2017

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

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



Pôle de recherche et d'enseignement supérieur

École doctorale MSTIC

**Simulation multi-agent de l'information des voyageurs dans
les transports en commun**

THESE

Présentée pour l'obtention du titre de
DOCTEUR EN INFORMATIQUE

Par
Amine Othman

JURY

- Directeur de thèse : **Gérard Scemama**
Directeur de recherche à IFSTTAR
- Encadrant : **Mahdi Zargayouna**
Chargé de recherche à IFSTTAR
- Rapporteurs : **Fabien Michel**
Maître de conférence HDR à l'Université de Montpellier
- Flavien Balbo**
Professeur à Mines Saint-Etienne
- Examineurs : **Michel Ocello**
Professeur à l'Université Grenoble Alpes - Valence
- Julien Saunier**
Maître de conférence à l'INSA Rouen

Table des matières

Liste des tableaux	ix
Table des figures	xi
Introduction générale	xiii

I État de l'art

Chapitre 1

L'information des voyageurs et l'évaluation de son impact

1.1	Introduction	3
1.2	Les systèmes d'information des voyageurs modernes	5
1.2.1	Rôle et fonctionnalités	5
1.2.2	Les SIV et le transport multimodal	9
1.2.3	Supports et voies de communication	13
1.3	Evaluation de l'impact des SIV	15
1.3.1	L'enquête comme méthode d'évaluation	16
1.3.2	La simulation comme méthode d'évaluation	17
1.3.3	Expérimentations & Résultats	24
1.4	Conclusion	30

Chapitre 2

La simulation multi-agent des déplacements

2.1	Introduction	31
2.2	La notion d'agent	32
2.2.1	Définition	32
2.2.2	Caractéristiques	33
2.2.3	Types d'agents	33
2.3	Les systèmes multi-agents	34
2.3.1	Définition	34
2.3.2	Organisation des SMA	36
2.3.3	Interaction	37
2.3.4	Environnement	37
2.4	Simulateurs multi-agents des déplacements	38
2.4.1	AgentPolis	39
2.4.2	Transims	41
2.4.3	MATSim	43
2.5	Plateformes de simulation multi-Agent	45
2.5.1	Swarm	46
2.5.2	Mason	46
2.5.3	NetLogo	47
2.5.4	Gama	47
2.5.5	Repast	49
2.5.6	Discussion	52
2.6	Conclusion	53

II Contributions

Chapitre 3

Simulateur multi-agent des déplacements

3.1	Introduction	58
-----	------------------------	----

3.2	La plateforme de voyage multimodal	58
3.2.1	Scénario et hypothèses	59
3.2.2	Interface publique	59
3.2.3	Modèle	61
3.3	La plateforme de simulation Repast Symphony	62
3.3.1	Les contextes	63
3.3.2	Les projections	64
3.3.3	L'ordonnancement	64
3.4	Le simulateur	66
3.4.1	Données et paramètres	66
3.4.2	Ordonnanceur parallèle	67
3.4.3	Données et paramètres du simulateur	69
3.5	Déplacement dans la géographie	76
3.5.1	Contextes et projection	76
3.5.2	Planification des itinéraires	76
3.5.3	Le déplacement	77
3.6	Le système multi-agent	78
3.6.1	Les agents	78
3.6.2	Packages	80
3.7	Optimisations	80
3.7.1	Des données erronées ou manquantes	80
3.7.2	Problème de performance	82
3.8	Exécution	83
3.9	Conclusion	83

Chapitre 4

Impact de l'information des voyageurs

4.1	Introduction	87
4.2	Gestion de l'information des voyageurs	88
4.2.1	Flux d'information des voyageurs	88
4.2.2	Flux d'information dans le simulateur	89
4.3	Système multi-agent	90
4.3.1	Agents voyageurs	90
4.3.2	Agents d'information locale	92
4.3.3	Service de planification	93

4.3.4	Agents véhicules	93
4.4	Les perturbations	93
4.4.1	Injection de perturbations	93
4.4.2	Impact des perturbations sur le comportement des agents véhicules	93
4.5	Représentation spatiotemporelle de l'environnement	94
4.5.1	Problématique	94
4.5.2	Proposition	95
4.5.3	Les perturbations dans l'environnement spatiotemporel	97
4.5.4	Usage de l'environnement spatiotemporel par les agents d'informa- tion locale	97
4.5.5	Usage de l'environnement spatiotemporel par le service de planifi- cation	98
4.6	Modèle temporel de la simulation	99
4.7	Conclusion	100

Chapitre 5

Résultats

5.1	Introduction	101
5.2	Configuration	101
5.2.1	Environnement informatique	101
5.2.2	Données en entrée	102
5.2.3	Injection des perturbations	102
5.2.4	Tirage aléatoire selon le type d'agent	102
5.3	Scénarios	104
5.4	Résultats	106
5.5	Temps d'exécution	110
5.6	Discussion	114
5.7	Conclusion	116

Conclusion générale et perspectives	117
--	------------

Bibliographie	121
----------------------	------------

III Annexes

Annexes

Annexe A

Formats de données en entrée du simulateur

- A.1 Formats des données du réseau de transport 135
- A.2 Formats des données d'un tableau de marche 135

Annexe B

Résultats détaillés

- B.1 Temps de parcours 139

Liste des tableaux

1.1	Les caractéristiques des évaluations menées par les travaux étudiés	18
3.1	Exemple de matrice D (avec $N = 3$)	71
3.2	Exemple de matrice S avec $ A = 10$ et $ M = 20$	71
3.3	Exemple de matrice MP correspondant aux proportions dans S	72
3.4	Exemple de table P	72
3.5	P après le choix de (Z_1, Z_2)	72
5.1	Les six scénarios en fonction des niveaux d'informations.	105
5.2	Les capacités des bus dans la simulation	106
5.3	Synthèse des temps de parcours	109
B.1	Temps de parcours (1.000 voyageurs)	139
B.2	Temps de parcours (5.000 voyageurs)	139
B.3	Temps de parcours (10.000 voyageurs)	140
B.4	Temps de parcours (20.000 voyageurs)	140
B.5	Temps de parcours (30.000 voyageurs)	140

Table des figures

1.1	L'architecture cadre FRAME	6
1.2	Chaîne de service de l'information des voyageurs	7
1.3	Les données et leurs sources dans un SIV moderne	8
1.4	Exemple d'un réseau de transport multimodal	12
1.5	L'architecture du système PECITAS	14
1.6	Exemple d'un processus de choix d'itinéraire chez un voyageur	20
1.7	Représentation du modèle de génération d'information	23
1.8	Comparaison entre les différents niveaux d'information en fonction du temps moyen de parcours	25
1.9	Analyse résultant de 9 scénarios	26
1.10	Les 6 combinaisons de test dans [Coppola and Rosati2009]	26
1.11	La distribution des voyageurs entre 3 chemins reliant une paire origine-destination sous les différents scénarios	28
1.12	Coefficient de variation du nombre de voyageurs embarqués à l'arrêt F de la ligne verte	29
1.13	Temps de parcours moyen de différentes paires origines-destination	29
2.1	Illustration de la notion de système multi-agent selon Ferber	35
2.2	Les formes d'interactions entre agents	36
2.3	Le lien entre l'environnement et l'agent	37
2.4	Un cliché d'affichage d'une simulation Transims	43
2.5	Exemple d'un plan journalier d'un agent dans MATSim	44
2.6	Exemples d'intégration de GIS dans NetLogo (a) utilisation de shapefiles (b) gradient (c) le modèle de croisière	48
2.7	Fenêtres de visualisation d'une simulation Gama	50
2.8	Exemple de résultat obtenu après l'ajout d'un MNT	51
2.9	Exemples d'un modèle vectoriel à base d'agent sur RepastS. (a) Des agents se déplaçant sur un réseau (b) Un modèle superposé sur <i>Nasa World Wind</i>	52
3.1	Scénario MMT	60
3.2	Vue extérieure de la plateforme multimodale de voyage MMT	60
3.3	Le modèle de la plateforme MMT	61
3.4	Le module de communication	62
3.5	Le module de planification	63
3.6	Le modèle du système multi-agent	67

Table des figures

3.7	Le <i>workflow</i> de la simulation	69
3.8	Le formatage du temps	74
3.9	Le formatage du temps - Résultat	74
3.10	La définition des paramètres dans l'interface du simulateur	75
3.11	La récupération des paramètres	75
3.12	La transformation du réseau de transport en commun	77
3.13	Le déplacement d'un agent	78
3.14	Le diagramme d'activité du déplacement de l'agent voyageur.	81
3.15	Les packages	82
3.16	Exécution de la simulation	84
4.1	La représentation mentale d'un voyageur non connecté	91
4.2	L'impact des informations locales sur les voyageurs non connectés	91
4.3	Le graphe spatiotemporel	96
4.4	L'implémentation du graphe spatiotemporel dans le simulateur	97
4.5	La souscription de l'agent voyageur connecté sur le graphe spatiotemporel.	98
4.6	Les graphes du simulateur.	99
5.1	Exécution d'une simulation	103
5.2	Génération des agents de chaque type	104
5.3	Améliorations par rapport à un scénario sans information (1.000 voyageurs)	106
5.4	Améliorations par rapport à un scénario sans information (5.000 voyageurs)	107
5.5	Améliorations par rapport à un scénario sans information (10.000 voyageurs)	107
5.6	Améliorations par rapport à un scénario sans information (20.000 voyageurs)	108
5.7	Améliorations par rapport à un scénario sans information (30.000 voyageurs)	108
5.8	Améliorations comparées des temps de parcours (voyageurs connectés vs. voyageurs non connectés) avec 1.000 voyageurs	109
5.9	Améliorations comparées des temps de parcours (voyageurs connectés vs. voyageurs non connectés) avec 5.000 voyageurs	110
5.10	Améliorations comparées des temps de parcours (voyageurs connectés vs. voyageurs non connectés) avec 10.000 voyageurs	111
5.11	Améliorations comparées des temps de parcours (voyageurs connectés vs. voyageurs non connectés) avec 20.000 voyageurs	112
5.12	Améliorations comparées des temps de parcours (voyageurs connectés vs. voyageurs non connectés) avec 30.000 voyageurs	112
5.13	Détérioration des temps d'exécution pour les simulations longues	113
5.14	Détérioration des temps d'exécution pour les simulations longues	113
5.15	Variation des temps d'exécution par rapport au nombre de voyageurs	114
5.16	Usage de la mémoire par rapport au nombre de voyageurs	114
A.1	Schéma XSD d'un réseau de transport (1/2)	136
A.2	Schéma XSD d'un réseau de transport (2/2)	136
A.3	Schéma XSD d'un tableau de marche	137

Introduction générale

Motivations

Selon les Nations Unis¹, la moitié de la population mondiale vit dans les grandes villes métropolitaines et d'ici 2050, plus des deux tiers (6,4 milliards) seront des urbains. Ce mouvement d'urbanisation galopante affecte directement nos modes de vie et sollicite des méthodes et des moyens de gestions efficaces afin de pouvoir suivre les changements résultants de ce mouvement. Parmi les domaines les plus touchés, la mobilité nécessite un intérêt particulier vu son importance et son implication dans tous les domaines de notre vie. D'ailleurs, plusieurs villes ont montré très tôt leur intérêt à cette problématique et ont pris conscience du besoin de fournir à leurs habitants des modes de déplacement fiables et capables de suivre la croissance démographique en marche. C'est bien dans cette optique que les transports en commun (TC) étaient nés. Des villes telles que Londres (dès 1863), Chicago (en 1892), Paris (en 1900) ou Berlin (en 1902) ont depuis plus d'un siècle déjà mis en place leurs systèmes de transport en commun. Aujourd'hui, ces systèmes sont présents dans 70% des villes avec une population autour des 5 millions et 100% de celles qui comptent plus de 10 millions d'habitants [Roth *et al.*2012].

De nos jours, le transport en commun est parmi les modes les plus soutenus des politiques de développement durable et représente le centre des solutions de mobilité urbaine pour des considérations écologiques face à une situation environnementale alarmante. Ces systèmes se développent et s'agrandissent constamment. Nous prenons l'exemple du système MRT² de Singapour qui couvre 149 km de réseau ferré et 99 stations et qui sert 2,3 millions de voyages par jour. Les gestionnaires de ce système travaillent sur son élargissement pour atteindre 278 Km d'ici 2020 afin de pouvoir répondre à une demande estimée à 14,3 millions de voyage par jour [EI2012].

Face à l'évolution de ces systèmes et l'explosion de la demande de transport, les gestionnaires des réseaux se trouvent dans l'obligation d'avoir les bons outils qui leur permettent de gérer l'interaction entre l'offre et la demande qui s'avère une tâche complexe. En effet, les réseaux de transport de nos jours sont complexes et fortement dynamiques. Leurs états changent constamment suite à plusieurs facteurs tels que les pannes des matériels, les accidents, les retards, etc. Sollicitant ainsi une interaction permanente entre les gestionnaires et les usagers afin de fournir une offre de voyage valide. Cette interaction est centrée autour de l'information qui représente le lien entre le gestionnaire et l'utilisateur des réseaux de transport. Le rôle de l'information est de plus en plus critique, particulièrement en cas de dysfonctionnement des réseaux. Dans cette optique, des sous-systèmes de transport appelés systèmes d'information voyageur

1. <http://www.unpopulation.org>.

2. *Mass Rapid Transit*

(SIV) ont vu le jour afin d'assurer cette interaction. Ces systèmes d'information des voyageurs ont pour objectif de fournir aux utilisateurs des réseaux de transport toutes les informations nécessaires pour effectuer leurs déplacements dans les meilleures conditions et d'offrir aux gestionnaires un outil de gestion des flux permettant de contrôler la congestion de leurs réseaux. Parmi les grands défis de ces systèmes est de pouvoir fournir aux voyageurs une information fiable en temps-réel pendant toutes les étapes de leurs déplacements pour faire face à une offre de voyage fortement dynamique. Ces systèmes profitent des avancées technologiques en particulier Internet et les systèmes intelligents afin de suivre l'évolution du transport moderne.

Certes, les avantages de ces systèmes sont nombreux pour les usagers comme pour les gestionnaires. Cependant, sans contrôle, la diffusion massive d'informations, à travers les panneaux à messages variables, les annonces dans les médias ainsi que les équipements de guidage individuel tels que les *smartphones* peuvent avoir des effets pervers et créer de nouvelles congestions. En effet, avec la généralisation de l'information des voyageurs en temps-réel, le comportement des réseaux de transport modernes devient plus difficile à analyser et à prévoir. Il est ainsi important d'observer ces effets afin de choisir les bonnes méthodes pour les traiter. L'information voyageurs peut avoir des effets pervers de différentes natures. Les auteurs dans [Ben-Akiva *et al.*1991] listent les trois phénomènes : saturation, concentration et sur-réaction, qui peuvent être observés après l'utilisation de systèmes d'information voyageurs.

L'évaluation de l'impact de l'information fournie par les SIV est depuis des années le sujet de plusieurs études. Les chercheurs ont eu recours à plusieurs méthodes afin d'analyser l'impact de ces informations. Si les enquêtes ont été utilisées dans plusieurs travaux (e.g. [Ferris *et al.*2009] ou [Watkins *et al.*2011]), la faiblesse de leurs résultats ne fait pas de cette méthode un bon choix d'évaluation. En effet, cette méthode consiste à questionner des voyageurs au niveau des arrêts durant leur attente sans avoir un suivi de leurs déplacements sur le réseau de transport. Nous constatons aussi que le nombre de personnes questionnées est généralement faible pour ce type de d'étude. Ainsi, l'évaluation que proposent ces travaux s'articule autour de l'impact de l'information en temps-réel sur les temps d'attente des voyageurs et ne permet pas d'élargir les critères d'évaluation. Les derniers travaux qui utilisent les enquêtes (e.g [Brakewood *et al.*2014]) commencent à surmonter cette limite en mettant en place un système de suivi de voyageurs témoins via un site web. Cependant, malgré la richesse des critères d'analyse, des résultats médiocres ont été notés à cause d'un grand nombre de désistements chez les participants pendant la durée de l'expérience.

D'autre part, les résultats obtenus par la simulation depuis des années (e.g. [Hickman and Wilson1995], [Coppola and Rosati2009] ou [Cats *et al.*2011]) se montrent encourageants puisqu'ils offrent de nombreux critères d'évaluation à travers différents scénarios de simulation et permettent un suivi microscopique des déplacements des voyageurs simulés ce qui offre une évaluation fine dans la majorité des travaux. Cependant, la nature des informations évaluées jusque là ne suit pas l'évolution que connaissent les SIV modernes où l'information devient de plus en plus personnalisée et accessible pendant toutes les étapes d'un voyage, grâce notamment à un taux d'équipements des voyageurs qui devient de plus en plus conséquent. Aussi, nous estimons que les perturbations sur les réseaux ne sont pas suffisamment considérées dans ces travaux malgré l'importance de leurs effets sur le processus de l'analyse. En effet, nous remarquons un manque d'informations dans ces systèmes concernant l'état dynamique des réseaux, concernant les perturbations par exemple, ou les accidents en cours.

Par ailleurs, les systèmes multi-agent (SMA) est un domaine de recherche qui représente

depuis des années un outil pertinent de résolution de problèmes complexes, intégrant plusieurs entités autonomes et distribuées en interaction évoluant dans un environnement dynamique. Dans un SMA, un agent est une entité autonome ayant une perception de son environnement et capable de réagir aux changements de cet environnement et d'interagir avec les autres agents du système afin de réussir son objectif personnel ou un objectif commun. Cette description correspond à de multiples applications en transport et plus particulièrement à des systèmes de mobilité de voyageurs. Les auteurs dans [Bazzan and Klügl2014]) listent plusieurs raisons pour l'usage privilégié des systèmes multi-agents dans des applications de trafic, telles que la résolution naturelle et intuitive de problèmes, la capacité des agents autonomes pour la modélisation de systèmes hétérogènes, etc. Cette approche est particulièrement pertinente pour la simulation de mobilité de voyageurs puisque l'objectif est de prendre en compte des comportements humains qui interagissent dans un environnement complexe, dynamique et ouvert [Bessghaier *et al.*2012]. En effet, le concept d'agent est bien adapté pour la représentation de voyageurs dans les réseaux de transport. Ce sont des entités autonomes, situées dans un environnement, qui adaptent leurs comportements à la dynamique de l'environnement qu'elles perçoivent et interagissent avec les autres agents pour réaliser des buts spécifiques.

Dans ce contexte, le travail de cette thèse exploite le paradigme multi-agent afin de bâtir un simulateur pour répondre au besoin de l'évaluation de l'impact de l'information voyageur sur les réseaux de transport en commun. Notre simulateur nous permet d'abord de simuler les déplacements des usagers d'un réseau de transport en commun, notamment en situations de perturbations. Le déplacement se fait d'une manière autonome par chaque usager simulé en tenant compte des informations diffusées par les gestionnaires du réseau. Deux types d'informations sont intégrés par le simulateur. Les informations générales diffusées à travers les dispositifs locaux (panneaux d'affichage et sonores) et les informations personnalisées à travers les équipements personnels des voyageurs (*smartphones*). Ensuite, le deuxième objectif du simulateur est de permettre d'analyser l'impact, à un niveau microscopique, des deux types d'informations diffusées sur le réseau. Cette analyse guide les gestionnaires dans le calibrage de leurs stratégies d'informations et de gestion des flux spécialement dans les situations de perturbations.

Contributions

Le travail présenté dans cette thèse est divisé en deux principales contributions. Nous proposons d'une part un simulateur multi-agent de déplacement sur les réseaux de transport en commun. Ce simulateur est originellement développé pour interagir avec la plateforme de voyage multimodal (MMT pour *multimodal travel platform* [Othman *et al.*2014]) conçu lors du projet européen *Instant Mobility*. Cette plateforme guide les voyageurs suivant les meilleurs itinéraires possibles jusqu'à leurs destinations, quelque soient le mode de transport qu'ils empruntent. La plateforme doit être alimentée avec des demandes de déplacement continues. Ainsi, notre simulateur génère des voyageurs et des moyens de transport et interagit avec la plateforme en lui fournissant les demandes et la localisation continue de ces entités. Bien que le simulateur ait été développé originellement pour interagir avec la plateforme, nous en faisons un outil indépendant qui fonctionne en autonomie sur n'importe quelle zone géographique, du moment que nous disposons des données nécessaires à son exécution. L'application simule les déplacements de voyageurs sur les réseaux de transport tout en prenant en compte les changements

d'état des réseaux et de leurs temps de parcours. Puisqu'il simule des positions individuelles des voyageurs, le simulateur peut notamment simuler et évaluer l'impact d'un large éventail d'applications communautaires de transport, telles que l'échange communautaire d'information de temps de parcours, le guidage communautaire de voyageurs, etc.

D'autre part, notre deuxième contribution intervient au niveau de l'intégration de l'information voyageurs dans le simulateur. Nous intégrons une couche informationnelle au sein du simulateur, en intégrant des agents d'information de voyageurs. Ces agents interagissent avec les agents voyageurs connectés et non connectés. Dans ce cadre, nous reprenons un modèle de représentation de l'environnement physique fondé sur les réseaux espace-temps. Cette représentation a été utilisée par le passé dans différentes applications : le transport à la demande, le problème de tournées de véhicules, l'évacuation de populations en situation de crise, etc. Dans notre application, l'environnement spatio-temporel multi-agent représente l'état du réseau de transport en commun dans le temps. Cet environnement est l'interlocuteur privilégié des agents voyageurs et est actif, dans le sens où il sauvegarde des informations et réagit en fonction de leur dynamique.

Organisation du rapport

Cette thèse est divisée en trois parties. La première partie est composée de deux chapitres, la deuxième partie est composée de trois chapitres et la troisième est composée de deux annexes. Les deux premiers chapitres représentent l'état de l'art des deux domaines sur lesquelles nous travaillons : la simulation et les méthodes d'évaluation des systèmes d'information des voyageurs. Les trois chapitres suivants décrivent nos contributions et les résultats de nos expérimentations. Enfin, les deux dernières annexes fournissent les détails sur les formats de données manipulées et fournit les résultats détaillés de nos expérimentations.

Première partie. État de l'art

Chapitre 1. les systèmes d'information des voyageurs et les méthodes d'évaluation de leur impact

Le chapitre 1 présente dans sa première partie les systèmes d'information des voyageurs modernes. Ensuite, une étude des méthodes d'évaluations de leur impact dans différents travaux de la littérature. Ce chapitre conclut que l'évaluation de l'impact de l'information des voyageurs par simulation est plus prometteuse. Elle ouvre le champ d'étude sur la simulation de déplacement faisant l'objet du chapitre 2.

Chapitre 2. La simulation multi-agent et la simulation de trafic

Le chapitre 2 présente la simulation des déplacements, et en particulier la simulation multi-agent des déplacements. Ce type de simulations est le plus adapté lorsque nous désirons simuler des comportements individuels face aux informations des opérateurs. Nous présentons les concepts de base du paradigme multi-agent, puis une étude des architectures des simulateurs de déplacement à base d'agents existants. Concluant qu'aucune de ces plateformes ne répond

à nos problématiques, nous présentons une étude détaillée des plateformes multi-agents généralistes. Cette étude nous permet de choisir la plateforme Repast Symphony comme base de développement de notre plateforme de simulation multi-agent des déplacements.

Deuxième partie. Contributions

Chapitre 3. Un simulateur multi-agent de déplacement sur les réseaux de transport en commun

Conformément au choix de plateforme opéré au chapitre 2, le chapitre 3 décrit d’abord le contexte originel du développement du simulateur (la définition d’une plateforme de guidage dans le cadre du projet Européen FP7 *Instant Mobility*) avant de détailler les briques de base de la plateforme Repast Symphony. Ensuite, il détaille la conception du simulateur : ses paramètres et données, la planification d’itinéraires et la représentation du réseau. Puis, il décrit les agents du système et leurs comportements et détaille l’ensemble des optimisations que nous avons effectuées pour améliorer les performances du système.

Chapitre 4. L’impact de l’information des voyageurs sur les réseaux de transport en commun

Dans le chapitre 4, nous décrivons l’approche utilisée pour l’intégration de l’information dans la plateforme de simulation présentée dans le chapitre 3. Nous intégrons de nouveaux types d’agents et utilisons une représentation spatio-temporelle de l’environnement pour représenter l’état du réseau dans le temps. Cet environnement est l’interlocuteur privilégié des agents voyageurs, ainsi, il est responsable de la diffusion des informations sur le réseau. Nous décrivons également notre méthode d’injection des perturbations pour valider l’impact de l’information temps-réel sur les parcours des voyageurs.

Chapitre 5. Résultats

Dans le chapitre 5, nous décrivons la configuration de nos expérimentations. Nous utilisons des données réelles du réseau de transport public de la ville de Toulouse. Des perturbations aléatoires sont créées sur le réseau afin de valider l’impact des informations sur les voyageurs. Nous exécutons six scénarios différents avec cinq ensembles de voyageurs pour chaque scénario (1.000, 5.000, 10.000, 20.000 et 30.000 voyageurs). Le premier scénario représente le cas où aucune information n’est diffusée sur le réseau et fait office de configuration de référence. Ensuite, le niveau d’information des voyageurs est introduit progressivement dans les autres scénarios. Pour chacune de ces 30 configurations, nous reportons le temps de parcours moyen des passagers. Le résultat le plus important montre que les informations temps-réel ont un impact positif jusqu’à un certain seuil, avant de devenir relativement négatif.

Troisième partie. Annexes

Annexe A. Formats de données en entrée du simulateur

Cette annexe donne les schémas XSD des données décrivant les réseaux de transport (transport en commun et réseau piéton) et les tableaux de marche des véhicules.

Annexe B. Résultats détaillés

Cette annexe fournit les résultats détaillés de nos simulations, qui sont présentés d'une manière synthétique dans le chapitre 5.

Première partie

État de l'art

Chapitre 1

Les systèmes d'information des voyageurs et les méthodes d'évaluation de leur impact

Sommaire

1.1	Introduction	3
1.2	Les systèmes d'information des voyageurs modernes	5
1.2.1	Rôle et fonctionnalités	5
1.2.2	Les SIV et le transport multimodal	9
1.2.3	Supports et voies de communication	13
1.3	Evaluation de l'impact des SIV	15
1.3.1	L'enquête comme méthode d'évaluation	16
1.3.2	La simulation comme méthode d'évaluation	17
1.3.3	Expérimentations & Résultats	24
1.4	Conclusion	30

1.1 Introduction

Ces dernières années, les réseaux de transport deviennent de plus en plus complexes et interconnectés. Cela rend la tâche d'un usager de ces réseaux complexe et nécessitant certaines connaissances pour effectuer son déplacement. Ainsi, des systèmes d'informations des voyageurs (SIV) ont été développés afin d'assister les utilisateurs des réseaux de transport en leur fournissant les informations nécessaires pour pouvoir planifier leurs déplacements. Des informations telles que les tableaux de marche des bus/métros/trains, les modes de transport disponibles, les plus-courts chemins, l'état du trafic, les routes/lignes fermées en cas de travaux ou d'accidents et pleins d'autres. Ces informations permettent aux voyageurs de planifier un trajet entre un point de départ et un point d'arrivée dans les meilleures conditions et l'assistent pendant son voyage. Aussi, cela leur assure un gain de temps précieux, notamment pour les voyageurs journaliers, et augmente leur satisfaction envers les services des systèmes de transport.

Avec l'évolution que connaissent les systèmes de transport aujourd'hui et les technologies avancées qu'ils utilisent, on assiste à une grande transformation dans les systèmes d'information des voyageurs qui deviennent à leur tour de plus en plus avancées. En effet, les gestionnaires de transport ont recours à des systèmes intelligents tels que les systèmes de localisation automatiques des véhicules (e.g. [Barbeau *et al.*2010, Yan2012]) et les nouvelles technologies de communication afin de mieux gérer leurs systèmes d'une manière sûre et efficace. Ces nouvelles technologies rendent les données des systèmes de transport plus dynamiques et volumineuses d'où le grand besoin des systèmes d'information d'abord, de pouvoir traiter ces informations et ensuite, de trouver les canaux capables de les partager avec un public de plus en plus exigeant et connecté. Effectivement, les informations de préparation de voyage ne sont plus suffisantes aux voyageurs face à des réseaux de transport dynamiques où l'environnement de voyage peut changer rapidement plusieurs fois pendant une seule journée. D'où le besoin d'être informé en temps-réel afin d'indiquer aux usagers de ces réseaux les changements d'états et les indications à suivre pendant leurs déplacements. Ces informations sont de grande utilité, particulièrement en cas d'accidents et de dysfonctionnement et permettent de contrôler la congestion sur les réseaux et assistent les voyageurs afin de bien mener leurs déplacements. Ainsi, on distingue deux types d'informations que les systèmes d'information des voyageurs modernes doivent fournir : informations de pré-voyage et informations en-route en temps réel.

En plus des technologies utilisées par les systèmes de transport qui les rendent intelligentes, ces derniers deviennent multimodaux. Dans plusieurs grandes villes, un voyageur peut emprunter plusieurs modes de transport (bus, métro, voiture, vélo) afin de se déplacer entre deux points du réseau ainsi qu'un nouveau mode (*Park & Ride*) est introduit afin de gérer les problèmes de congestion dans ces villes (e.g. [Zhang *et al.*2012, Li *et al.*2012]). Par conséquent, les systèmes d'information des voyageurs doivent pouvoir suivre l'évolution des modes de transport d'aujourd'hui et fournir aux utilisateurs des informations adaptées à ces changements. Dans ce contexte, plusieurs études sont menées ces dernières années afin de permettre aux systèmes d'information des voyageurs de s'adapter aux réseaux multimodaux [Zhang *et al.*2012] et d'intégrer les nouveaux modes de déplacement multimodaux (e.g. [Li *et al.*2012], [Pajor2009], [Zhang *et al.*2011], [Zografos and Androusoyopoulos2008]).

Dans ce chapitre, nous étudions les SIV et leur évolution par rapport à celle des systèmes de transport moderne et par le changement de leur environnement, ainsi que les travaux évaluant leur impact. La section 1.2 présente les SIV. Elle est divisée en trois parties : la section 1.2.1 examine le rôle et les fonctionnalités d'un SIV, dans la section 1.2.2, nous étudions l'évolution des SIV pour répondre aux besoins du transport moderne et multimodal, et la section 1.2.3 met le point sur les nouveaux outils de communication utilisés par les SIV modernes. La deuxième partie du chapitre (section 1.3) est consacrée aux approches utilisées dans la littérature pour évaluer l'impact des SIV sur les réseaux de transport en commun. La section 1.3.1 présente l'évaluation par enquête tandis que la section 1.3.2 présente l'évaluation par simulation, qui inspire notre proposition. La section 1.4 est notre conclusion.

1.2 Les systèmes d'information des voyageurs modernes

1.2.1 Rôle et fonctionnalités

L'information des voyageurs est de nos jours parmi les domaines les plus importants des systèmes de transport intelligents (ITS³) et touche plusieurs domaines fonctionnels à la fois. Selon ACTIF, le dispositif d'aide à la conception de transports interopérables en France⁴ mis en place par le MEDDE (Ministère de l'Écologie, du Développement durable et l'Énergie), les secteurs d'activité des ITS sont regroupés dans les domaines fonctionnels suivants :

1. Fournir des moyens de paiement électroniques.
2. Gérer les services d'urgences et de sécurité.
3. Gérer les infrastructures de transport et leur trafic.
4. Exploiter les transport publics.
5. Fournir des systèmes d'assistance aux conducteurs.
6. Coordonner l'offre de transport et informer sur les déplacements.
7. Faire appliquer la réglementation.
8. Exploiter les marchandises et les flottes.
9. Gérer les données partagés.

Le dispositif ACTIF est inspiré de l'architecture cadre européenne des systèmes de transport intelligents FRAME⁵. Cette architecture a été créée afin de fournir une structure stable pour le déploiement d'ITS interopérables au niveau de l'union européen. La figure 1.1 illustre cette architecture et donne une vision d'ensemble des systèmes de transport intelligents.

Ainsi, l'information voyageur et traitée au travers du domaine fonctionnel 6 et également 3 et 9 d'où son importance dans le transport moderne. En effet, les systèmes d'information des voyageurs ne cessent pas de se développer et de devenir parmi les principaux ITS. Ces systèmes permettent de gérer simultanément l'offre et la demande de voyages et constituent un outil pertinent pour les gestionnaires comme pour les usagers des réseaux de transport. Les SIV jouent un rôle important en offrant aux gestionnaires des systèmes de transport un outil pour améliorer l'efficacité et la sûreté des réseaux et diminuer les impacts environnementaux. À travers l'information voyageurs, les gestionnaires peuvent orienter les déplacements des flux de voyageurs et appliquer des stratégies d'affectation afin de :

- Contrôler les congestion sur les réseaux,
- Gérer les perturbations de fonctionnement (e.g. retards, pannes),
- Gérer des situations de crise nécessitant des mesures de sécurité particulières (ex. évacuation).

Ensuite, ils permettent de fournir aux voyageurs les informations nécessaires à leurs déplacements. Des informations telles que les modes de transport disponibles, les plus courts chemins, les tableaux de marches des bus/métros/trains, l'état du trafic, les routes/lignes fermées en cas de travaux ou d'accidents et les consignes de sécurité et de régulation pendant les situations particulières. Ces informations sont diffusées pendant deux phases essentielles du voyage :

3. *Intelligent Transport Systems*

4. <http://www.its-actif.org/>

5. <http://frame-online.eu/>

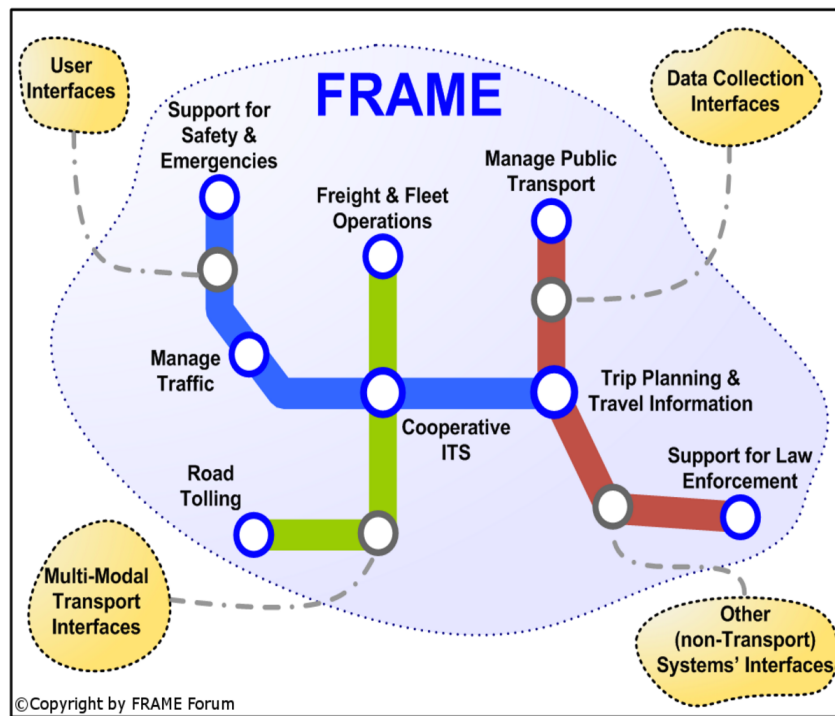


FIGURE 1.1 – L'architecture cadre FRAME

- **pré-voyage** : avant le début du voyage, permettant aux voyageurs de planifier un trajet entre un point de départ O et un point d'arrivée D dans les meilleures conditions. Les informations sont connues en avance et de nature statique ou semi-statique.
- **en-route** : pendant le déplacement, permettant d'assister les voyageurs et leur transmettre les derniers états des réseaux. Les informations sont de nature temps-réel et changent fréquemment en fonction de la dynamique des réseaux de transport et des événements imprévus.

Généralement, les informations sont transmises aux voyageurs à travers plusieurs équipements et technologies. Cela dépend essentiellement de la nature de l'information et de la phase de voyage pendant laquelle elle intervient : pré-voyage (internet, assistance téléphonique); en-route (affichage, messages sonores, radio, internet-mobile, équipements de navigation). Ces informations contiennent des données en provenance de différentes sources ayant subi des traitements particuliers avant d'être diffusées. La figure 1.2 issue de [Ambrosino *et al.*2010] illustre une représentation de la chaîne de services sous-jacente à la provision et traitement de l'information des voyageurs. Elle implique des intervenants publics et privés qui opèrent selon différents rôles et responsabilités dans le processus de production.

Les SIV jouent ainsi un rôle important parmi les systèmes de transport intelligents, ce qui explique l'important intérêt que portent les grandes villes aujourd'hui à ces systèmes. En effet, plusieurs villes dans le monde commencent à s'intéresser particulièrement à ces systèmes et essaient de suivre leur évolution. Divers travaux et études ont été menés ces dernières années par les autorités à une échelle régionale au niveau des grandes villes comme en Inde [Kumar *et al.*2005, Jain *et al.*2014] et aux USA [Cambridge Systematics2009] afin de

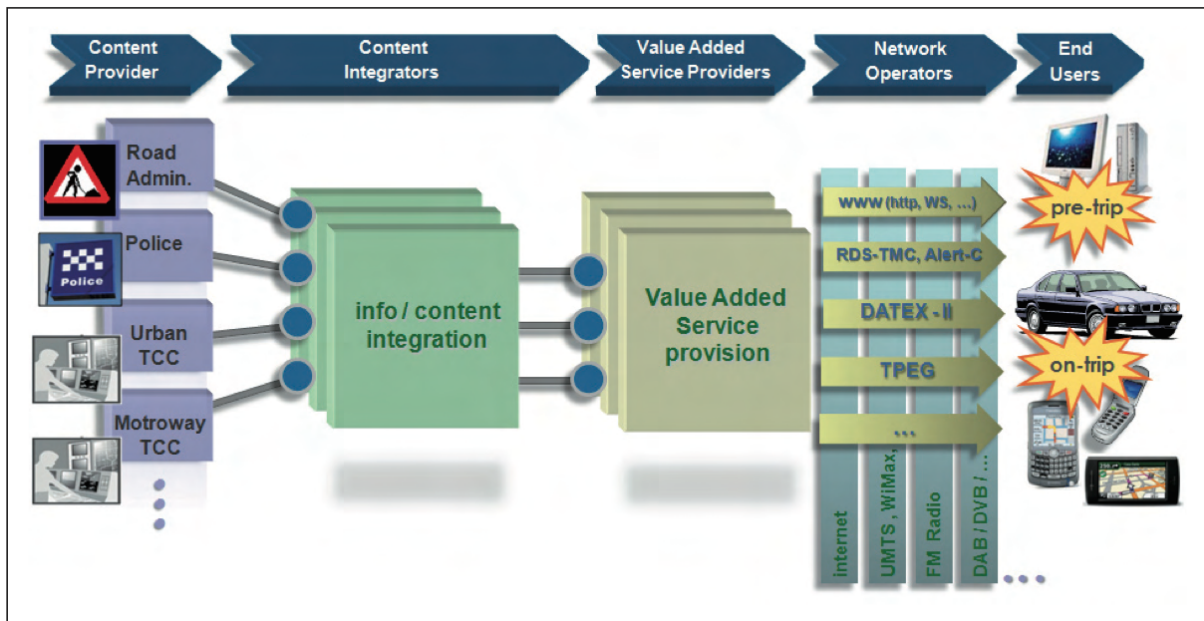


FIGURE 1.2 – Chaîne de service de l'information des voyageurs

concevoir des SIV convenables à leurs villes et qui répondent aux besoins de leurs populations d'un côté et aux différents acteurs en relation avec les réseaux de transport de l'autre côté (gestionnaire réseaux de transports en commun (TC) et routier, gestionnaire de crises, sécurité civile, universités, police, médias, aéroports, hôpitaux etc.).

A l'échelle d'un pays comme d'une grande ville, une bonne infrastructure de transport est parmi les bases de développement de ces derniers et offre des avantages directs à la société. Le transport est fortement lié aux principaux domaines vitaux pour une société, en particulier les domaines économiques et sociaux. Une grande ville qui développe des pôles industriels doit pouvoir assurer aux travailleurs le moyen d'atteindre leur travail dans les meilleures conditions. Les systèmes de transport en général et les SIV en particulier sont indispensables au bon développement des grandes villes sur tous les secteurs et jouent un rôle important dans l'amélioration de la vie quotidienne de ses habitants.

Selon une étude du cabinet de conseil « Cambridge Systematics Inc » pour le compte de la commission d'aménagement régionale de Mid-Ohio aux USA en 2009 [Cambridge Systematics 2009], un système d'information moderne doit couvrir 12 fonctionnalités majeures. La figure 1.3 illustre ces 12 fonctionnalités qui sont intégrées dans un système en relation avec les acteurs directement concernés ainsi que les sources des informations. Cette étude a été menée dans le but de concevoir un SIV moderne qui fournit toutes les informations nécessaires aux utilisateurs des réseaux de transport et couvre tous les modes dans la région. L'étude a pris en considération les besoins et les préférences de tous les acteurs émetteurs et récepteurs des informations puis a étudié les systèmes existants dans la région afin de concevoir un outil qui répond à tous les besoins. Les 12 fonctionnalités sont les suivantes : Congestion & Temps de voyage - Transport en Commun TC - Plans - Covoiturage - Vélo&Piétons - Parking - Tourisme - Travaux - Alertes - Evénements - Météo - Aéroports.

Selon cette étude, un système idéal est multimodal, intégrant les modes de transport public

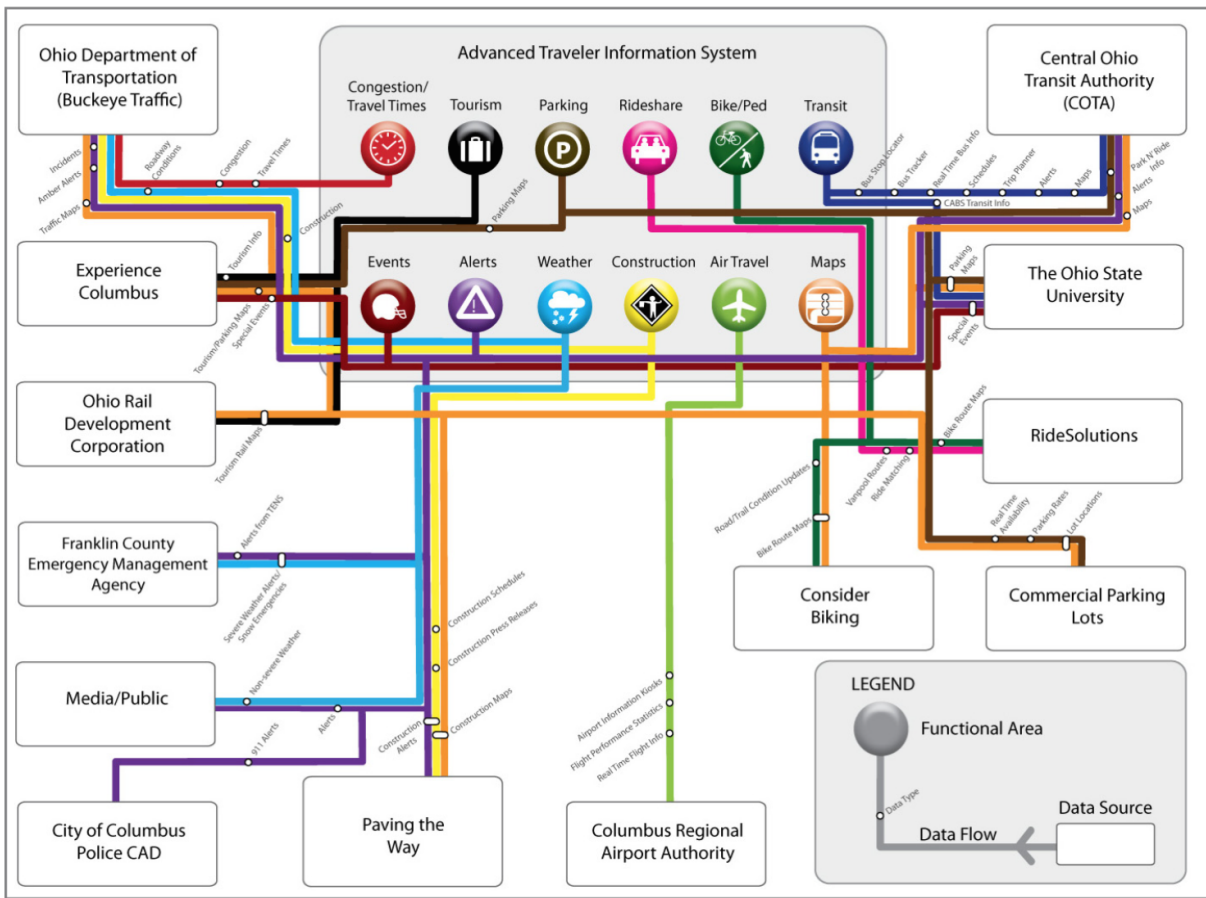


FIGURE 1.3 – Les données et leurs sources dans un SIV moderne

et privé et pouvant inclure des informations provenant de différentes sources et les partager en temps-réel. Comme indiqué dans la figure 1.3, chaque fonctionnalité est liée à un acteur qui représente sa source d'informations. L'étude a aussi conclu que le système doit être fondé sur le Web qui a prouvé qu'il est un moyen efficace et une méthode polyvalente de partage d'informations. Un site Web est un outil relativement simple et peu coûteux pour sa gestion et mise à jour. Ensuite, l'avantage de pouvoir intégrer le Web dans les innovations technologiques, tels que les véhicules connectés et les équipements mobiles, offre à un SIV une variété de méthodes d'accès aux utilisateurs afin de fournir une information en temps-réel.

Le rôle essentiel d'un SIV est de fournir aux utilisateurs d'un réseau de transport les informations nécessaires pour planifier leurs déplacements et s'organiser avant leurs départs. Cependant, l'évolution technologique et le développement des modes et réseaux de transport ont entraîné un grand changement dans le rôle qu'un SIV doit jouer. En plus de l'information des voyageurs, le SIV est un outil important qui aide les gestionnaires à gérer leurs réseaux surtout à résoudre les problèmes de congestion. Un SIV aujourd'hui doit fournir aux usagers des informations en temps-réel avant et pendant leurs déplacements : deux types d'informations sont alors intégrés, informations de pré-voyage et informations en-route. Côté communication, Internet offre un canal de transmission qui permet d'atteindre un voyageur même dans les véhicules qu'il utilise pour se déplacer. Ainsi, les voyageurs peuvent être informés de tous les changements sur des réseaux de plus en plus complexes et dynamiques et profitent des consignes de sécurité et de régulation. Un autre changement affecte directement la nature des SIV et oblige les responsables à repenser leurs systèmes existants. En effet, le transport d'aujourd'hui est devenu multimodal ce qui demande des SIV multimodaux pouvant combiner les modes de transport public et privé et intégrer les nouveaux modes émergents tels que le *Park & Ride* et le covoiturage. Cependant, cette combinaison s'avère délicate, voire devient un obstacle devant les concepteurs car elle nécessite un modèle de réseau de transport multimodal comme base à leurs SIV. Et selon [Zhang *et al.*2012], il n'existe pas un modèle général de réseau de transport multimodal. Néanmoins de plus en plus de standards sont définis ou en cours de définition concernant la représentation des réseaux multimodaux.

1.2.2 Les SIV et le transport multimodal

Face à un transport devenu multimodal dans la majorité des grandes villes du monde, les gestionnaires des réseaux de transport sont dans l'obligation d'adapter leurs SIV à cette évolution. Cette adaptation s'avère délicate pour deux raisons majeures. La première est que le SIV doit intégrer différents types d'information provenant de sources différentes et l'adapter à une seule forme de données dans un contexte dynamique et temps-réel. La deuxième qui est la plus importante, est l'absence d'un modèle général de réseau de transport multimodal sur lequel se base le SIV. Cet obstacle a poussé plusieurs chercheurs à traiter ce problème en particulier. Dans [Pajor2009], l'auteur a concentré sa recherche sur l'extension d'un réseau mono-modal (routier) vers un réseau multimodal. Les auteurs dans [Li *et al.*2012, Jariyasunant *et al.*2010] se sont focalisés sur la conception et la réalisation d'applications de planification de voyages multimodaux. [Kumar *et al.*2005] a développé un SIV multimodal pour la ville d'Hyderabad en Inde. Du côté industriel, plusieurs compagnies s'intéressent à leurs tours aux SIV multimodaux. La compagnie Mentz a développé un système de planification de voyage multimodal et l'a appliqué sur une échelle régionale (Service 511 à San Francisco et le service londonien *Transport For*

London). Trapeze, Logica et Google ont aussi développé leurs applications de planification de voyage multimodaux. Selon [Zhang *et al.*2012], malgré le grand nombre de travaux effectués, il n'existe toujours pas une représentation générale pour les réseaux de transport multimodaux qui permet de prendre en considération plusieurs attributs tels que l'aspect temporel des services de différents modes en particulier le transport en commun qui dépend des tableaux de marche de ses véhicules. La majorité des travaux ne décrivent pas suffisamment la connexion entre les différents modes de transport surtout entre le transport public et privé. Ainsi, les auteurs se focalisent sur la conception d'un modèle général de réseau de transport multimodal destiné aux SIV modernes. Ce modèle est basé sur la notion du super-réseau où chaque mode de transport est représenté sous la forme d'un sous-réseau du super-réseau qui les unie.

Le super-réseau : le concept

Un super-réseau consiste à combiner plusieurs réseaux en un seul offrant une représentation générale des fonctionnalités et attributs de chacun des sous-réseaux.

La première étape consiste à choisir le type de cette représentation générale puisqu'un réseau de transport peut être observé selon plusieurs points de vues. D'un point de vue physique, on peut le classer en route, chemin de fer, voie maritime et voie aérienne. D'un point de vue fonctionnel, on peut le classer en deux modes : mode privé (voiture, vélo, piéton) et mode public (transport en commun). L'avantage de la représentation fonctionnelle est qu'elle permet d'intégrer l'offre des services destinés aux voyageurs. Un réseau de transport privé permet de représenter un service continu à travers l'association des nœuds et arcs physiques. Cependant, un réseau de transport public offre un service discret lié à des tableaux de marche ainsi les nœuds physiques (arrêts) sont visibles et les arcs physiques sont invisibles.

Ainsi, le point de vue fonctionnel est convenable à la modélisation d'un réseau de transport multimodal, mais nécessite un traitement particulier pour intégrer l'offre du transport en commun [Zhang *et al.*2012]. Deux solutions sont cités dans la littérature : L'approche *time-dependent* et l'approche *time-expanded*. La première approche consiste à représenter les événements des tableaux de marche sous forme de propriétés des arcs. La deuxième approche extrait ces événements et les affiche sous forme de nœuds d'événements [Pajor2009].

Cette représentation générale permet de représenter de la même manière tous les sous-réseaux associés aux différents modes de transport puis d'assurer la liaison entre eux à travers les arcs de correspondance.

L'étape suivante consiste à élaborer un modèle de mesure de coûts généralisés. Cette mesure de coûts doit intégrer le temps, le coût monétaire et le confort. Concernant le temps, le traitement diffère entre le transport privé et le transport public. Les arcs du transport privé peuvent être classés en trois types :

- Indépendant du temps où le coût est statique.
- Dépendant du temps où le coût varie en fonction du temps d'une manière connue d'avance.
- Stochastique où des données temps-réel de l'état du réseau peuvent être considérées.

Au niveau du transport public, tous les arcs sont dépendant du temps et parfois stochastiques. Une solution générale est d'associer un horodatage à chaque nœud du réseau et un temps de parcours dépendant du temps à chaque arc.

Concernant le coût monétaire, la méthode ordinaire où le coût d'un voyage est accumulé suite à chaque fragment du réseau traversé peut être utilisée. Le mode de l'arc définit le coût de

confort et les temps d'attente et de transfert doivent être considérés.

La dernière étape est la sélection d'un algorithme approprié de calcul de plus court chemin (e.g. Dijkstra, A*, etc.). La figure 1.4 représente un exemple d'un réseau de transport multimodal proposé par [Zhang *et al.*2012].

Limites et perspectives

L'évolution des SIV fait encore face à plusieurs obstacles qui ne leur permettent pas de suivre l'avancée du transport moderne malgré les efforts menés dans ce domaine. Par exemple, le mode *Park & Ride* n'est toujours pas visible sur les SIV des grandes villes malgré les travaux focalisés dessus [Li *et al.*2012]. Cela est dû principalement au problème de partage d'informations entre les différentes sources dans un contexte dynamique et temps-réel. Selon un rapport fait par le centre national de recherche en transport en commun (NCTR) à l'université South Florida [Hillsman and Barbeau2011], le problème du partage d'informations entre les différents acteurs du transport constitue avec le coût élevé du développement des planificateurs de voyages multimodaux les obstacles majeurs devant les autorités locales pour offrir à leurs utilisateurs des SIV modernes intégrant les différents modes de déplacement. Cette étude, met le point sur une solution qui s'avère prometteuse et qui attire de plus en plus les gestionnaires des réseaux de transport et qui consiste à utiliser des plateformes open-source qui contiennent des bases de données ouvertes permettant le partage d'informations. En effet, deux grands projets ont été lancés selon cette approche et qui attirent en permanent des opérateurs de transport dans le monde. Le premier est *Open Street Map* (OSM)⁶, maintenue par la fondation caritative Open Street Map. OSM est un dépôt international ouvert et gratuit de données géographiques. Les agences de transport public peuvent intégrer des données géographiques dans le système et peuvent facilement être partagées avec les autres utilisateurs via une page Web. OSM offre aussi OpenTriPlanner, un système de planification de voyage open source géré par une communauté de développeurs.

Le deuxième projet General Transit Feed Specification (GTFS) est développé par Google et devient un standard de description des systèmes de transport en commun et une plateforme pour plusieurs sites Web et applications mobiles. Google offre aussi un outil de planification de déplacement en transport en commun basé sur GTFS. Plus de 800 villes dans le monde utilisent cet outil et ont introduit leurs données sous le format GTFS.

La limite de l'outil de planification de déplacement de google est qu'il n'intègre pas encore le nouveau mode *Park & Ride*. Cependant, l'utilisation des données GTFS est possible par d'autres plateformes. La communauté OSM travaille sur un outil nommé GTFS-OSM_SYNC (GO_SYNC) permettant la synchronisation des données entre les deux plateformes. Cette démarche est utilisée pour intégrer les données des réseaux Vélo et Piéton à partir de OSM et les données des réseaux de transport en commun à partir de GTFS dans OpenTripPlanner pour la ville de Tampa en Floride-USA [Hillsman and Barbeau2011]. Les solutions pour introduire les nouveaux modes de déplacement sont désormais le sujet de plusieurs travaux que ce soit dans le monde académique qu'industriel. Cet intérêt porté à cette problématique ne peut qu'améliorer la qualité des SIV modernes et leur permettre de suivre l'évolution des systèmes de transport et satisfaire les besoins des gestionnaires comme des voyageurs. Cette évolution touche aussi

6. www.openstreetmap.org

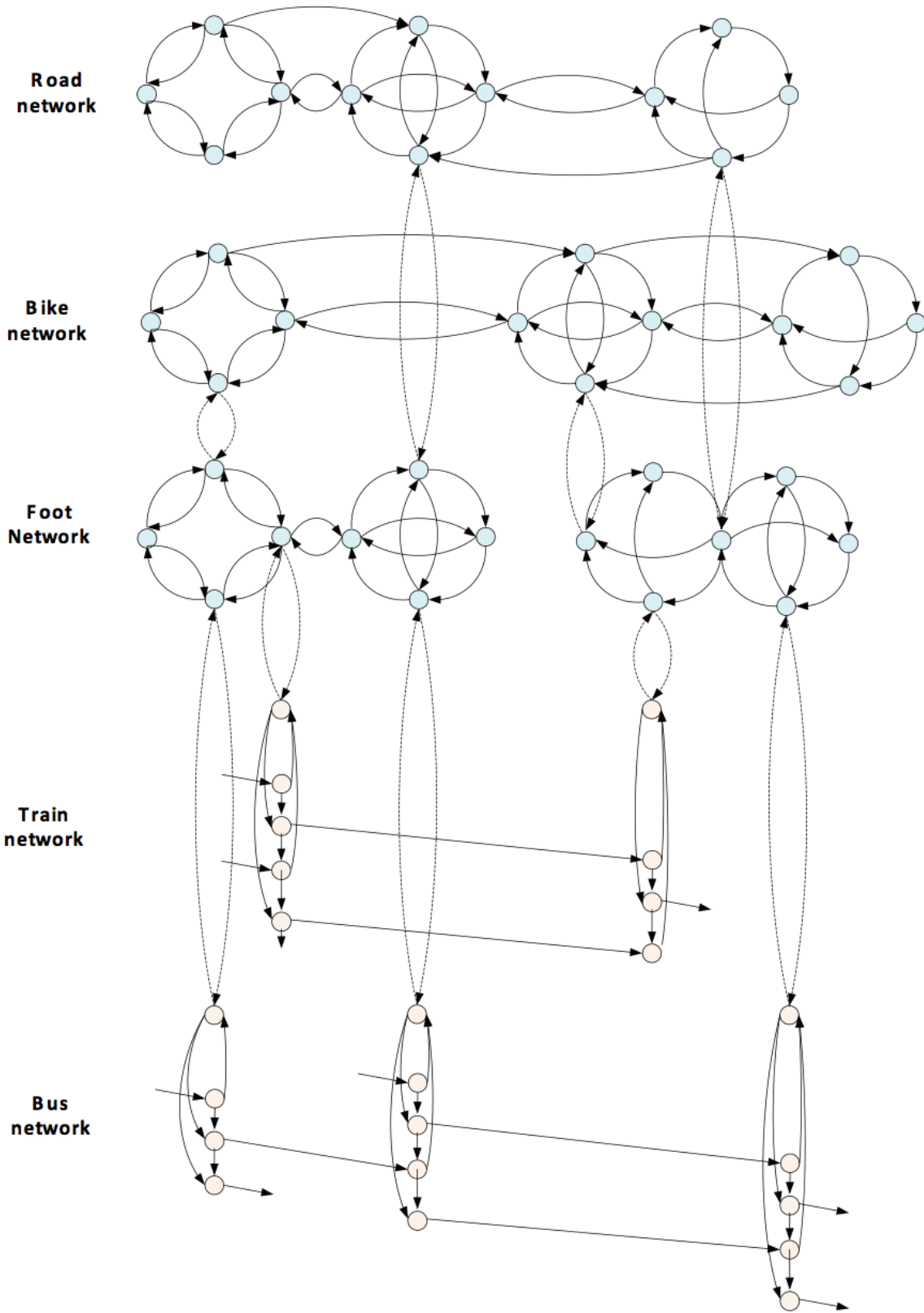


FIGURE 1.4 – Exemple d'un réseau de transport multimodal

les supports utilisés par les SIV afin d'interagir avec les voyageurs. En effet, les SIV modernes se basent essentiellement sur le Web qui à son tour ne cesse de se développer et d'offrir des nouvelles technologies de communication et d'échange. Dans section suivante, nous nous intéressons à cet aspect et nous étudions les nouvelles voies de communication offertes par le Web et comment les SIV s'y adaptent.

1.2.3 Supports et voies de communication

Le Web : un nouveau support d'information

Depuis plusieurs années, les gestionnaires des réseaux de transport ont choisi le Web comme premier support à leurs SIV. [Infopolis1998] liste plus de 300 sites-Web de services de transport en commun en Europe. [Jain *et al.*2014] nomme plusieurs villes indiennes ayant créé leurs SIV sur le Web ou utilisent le service *Google Map Transit* qui est déjà utilisé par plus de 500 villes dans le monde dont 125 aux USA [Hillsman and Barbeau2011]. L'étude menée dans [Cambridge Systematics2009] indique qu'un SIV qui répond aux besoins du transport moderne doit impérativement être basé sur le Web. En effet, le Web est un outil pertinent qui offre aux gestionnaires la possibilité de présenter des données spatiales et non spatiales en intégrant les systèmes d'informations géographiques et bénéficier de leurs avantages. Face à des réseaux de transport de plus en plus complexes et larges, le Web permet la manipulation de grandes quantités de données et leur traitement dans un contexte dynamique grâce à l'évolution continue des équipements informatiques. Une fois traitées, le Web permet ensuite de partager en temps-réel ces données, un atout qui répond à l'un des besoins majeurs des SIV.

Cette fonctionnalité ne cesse de se développer grâce à l'évolution du Web-mobile qui a rendu les *smartphones* un outil d'assistance de voyage en plus de ses fonctionnalités de base et a su profiter de leurs services de géo-localisation. L'information peut désormais être envoyée au voyageur au cours de son déplacement et devient même personnalisée.

Le Web mobile : vers une information personnalisée

Dans le domaine du transport, plusieurs chercheurs focalisent leurs travaux autour de l'information voyageur via l'internet mobile (*smartphones*).

Les auteurs dans [Jariyasunant *et al.*2010] exploitent l'union des *smartphones* avec les GPS et propose un système de planification de voyage en temps-réel destiné aux *smartphones* qui intègre des données provenant de plusieurs sources. Ce système calcule le plus court chemin entre origine et destination à travers un système de prévision utilisant les données GPS des véhicules (bus) en circulation. De leur côté, [Tumas and Ricci2009] se focalisent sur l'information personnalisée et utilisent le concept du profil voyageur afin de pouvoir personnaliser l'information et fournir à l'utilisateur les bonnes informations qui correspondent à ses préférences. Cette personnalisation se fait à l'aide de la technologie de recommandation à base de connaissances. La figure 1.5 illustre l'architecture du système nommé PECITAS qu'ils proposent. Tandis que dans [Lathia *et al.*2012], les auteurs utilisent l'exploitation des données *datamining* et le traitement automatique du langage naturel afin d'avoir une représentation explicite des préférences de l'utilisateur. Dans [Zenker and Ludwig2010], l'information personnalisée est exploitée dans un contexte différent. La personnalisation se base simultanément sur les préférences et la po-

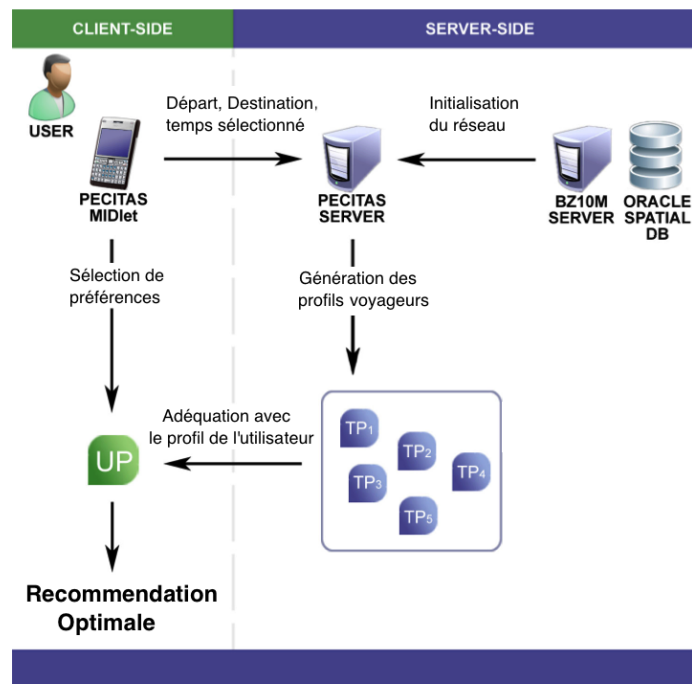


FIGURE 1.5 – L'architecture du système PECITAS

sition géographique afin de guider un utilisateur vers des destinations touristiques ou événementielles. L'information voyageur se voit ainsi de nouveaux angles ce qui élargit son domaine d'exploitation.

L'information personnalisée devient ainsi le centre d'intérêt de la nouvelle génération des SIV qui veut mieux exploiter cette nouvelle forme afin de pouvoir offrir une meilleure assistance aux utilisateurs des réseaux de transport de plus en plus complexes. Par exemple, le service maRATP proposé sur le site de la ratp, offre aux utilisateurs de leur réseau la possibilité de créer un espace personnel qui leur permettra de recevoir des informations personnalisées concernant leurs déplacements selon leurs préférences. Aussi cette nouvelle forme de données offre aux SIV la possibilité de s'ouvrir vers des nouvelles fonctionnalités et ensuite sur de nouveaux domaines d'activité (Sécurité, Tourisme, etc) qui peuvent accélérer leur développement.

Les réseaux sociaux : une nouvelle manière d'interagir

Selon l'institut Gartner⁷, les ventes de *smartphones* dans le monde en 2014 ont atteint les 1,2 milliards unités. La majorité des acheteurs se trouvent dans les grands villes et représentent pour la plupart les utilisateurs de leurs réseaux de transport. Ainsi, le taux d'équipements des voyageurs dans les grandes villes est important en considérant que les *smartphones* sont des outils de voyageurs très performants devenant l'un des importants canaux de diffusion des SIV. Ce canal d'information ne cesse de changer le comportement des voyageurs de plus en plus connectés. Et sachant que 2 milliards d'internautes sont actifs sur les réseaux sociaux aujourd'hui

7. <http://www.gartner.com/technology/home.jsp>

d'hui, cette nouvelle forme de support de diffusion attire les gestionnaires des SIV et leur offre des nouvelles possibilités qui peuvent être très bénéfiques dans leur mission d'approcher au maximum le voyageur en lui offrant une information personnalisée tout en optimisant au maximum le partage. En effet, les opérateurs de transport de villes telles que Paris et New York ont déjà adapté leurs SIV à ce nouveau mode d'interaction. Leurs services sont présents sur les deux plus importants réseaux sociaux (Facebook et Twitter) à travers des pages communicantes et des flux à suivre qui permettent de segmenter leurs flux d'informations d'une manière claire et familière aux utilisateurs. Plusieurs autres services et opérateurs se lancent dans cette démarche et développent leurs stratégies d'information en suivant les évolutions technologiques.

Dans la partie suivante de ce chapitre, nous interrogeons la littérature afin d'étudier les travaux d'évaluation de l'impact des systèmes d'information des voyageurs sur les réseaux de transport en commun qui sont adéquates avec notre problématique. Cette étude est dans le but d'analyser les approches utilisées dans ces travaux et plus particulièrement pour voir si ces travaux suivent l'avancée des SIV modernes devenus à leur tour complexes et dynamiques.

1.3 L'évaluation de l'impact de l'information des voyageurs sur les réseaux de transport en commun

Entre des voyageurs de plus en plus connectés, des techniques de partage bien adaptées et des informations temps-réel et personnalisées, les SIV sont aujourd'hui capables de fournir aux voyageurs les informations nécessaires dont ils ont besoin, que ce soit pour préparer leurs déplacements ou pour se guider pendant le voyage en fonction des changements qui peuvent survenir à tout moment. En effet, nous pouvons maintenant fournir des itinéraires optimaux pour les voyageurs mais nous pouvons également les mettre à jour en temps réel en nous fondant sur les nouveaux états des réseaux (congestions, accidents, pannes, etc.). Fournir de l'information trafic aux usagers des réseaux est généralement positif et permet de fluidifier les flux de trafic. Néanmoins, sans contrôle, la diffusion massive d'informations, à travers les panneaux à messages variables, les annonces dans les médias ainsi que les dispositifs de guidage individuel peut avoir des effets pervers et créer de nouvelles congestions. En effet, avec la généralisation des informations temps-réel des voyageurs, l'environnement des réseaux de transport modernes devient difficile à analyser et à prédire. Ainsi, il est important d'observer les effets des SIV afin de choisir les bonnes méthodes pour les traiter. [M.Ben-Akiva *et al.*1991] liste trois phénomènes qui peuvent être observés après l'utilisation des SIV :

- Sur-saturation : quand les voyageurs sont confrontés à une grande masse d'informations, ils vont avoir tendance à ignorer ces informations et essaient de se débrouiller seuls.
- Concentration : quand des voyageurs ayant les mêmes préférences reçoivent les mêmes chemins, une congestion se crée sur ces chemins.
- Sur-réaction : si plusieurs voyageurs répondent à une alternative proposée, la congestion peut se déplacer vers le point alternatif.

Cette problématique a poussé plusieurs chercheurs à se focaliser dessus et à mener des recherches dans le but d'évaluer l'impact des SIV sur les réseaux de transport. Selon [Hu2009], les méthodes d'évaluation des SIV sont généralement classifiés en 3 catégories :

- Les observations de terrains qui comparent la différence des temps de parcours d'un

véhicule ou d'un voyageur avec ou sans informations provenant des SIV.

- Les enquêtes, qui visent le retour d'expériences des utilisateurs afin de mieux juger l'impact des SIV.
- Les simulations qui permettent de mesurer l'impact des différentes possibilités d'utilisation des SIV sur un réseau de transport.

Dans la suite de cette section, nous présentons les deux dernières méthodes : les enquêtes et la simulation, que nous adoptons dans le cadre de ce travail de thèse.

1.3.1 L'enquête comme méthode d'évaluation

Parmi les travaux qui se sont intéressés à évaluer l'impact des SIV dans les réseaux de transport en commun, plusieurs recherches utilisent les enquêtes afin de mesurer cet impact sur les expériences de voyage des usagers. Nous en donnons un aperçu dans ce qui suit.

Deux travaux récents évaluent le même SIV *OneBusAway* (OBA) mais dans deux villes différentes en utilisant deux méthodes d'enquêtes différentes. OBA [Ferris *et al.*2009] est un SIV destiné à fournir aux usagers des réseaux de bus un compte à rebours temps réel des prochaines arrivées des bus. Les informations sont fournies via les équipements mobiles (*smartphones*) à travers le Web-mobile et les messages cellulaires (sms). OBA est développé en tant que système open-source afin d'être utilisé par d'autres opérateurs de transport.

La première étude [Watkins *et al.*2011] effectue une enquête de terrain dans la ville américaine de Seattle-Washington. Cette enquête mesure les temps d'attente des usagers du réseau de bus en leur posant une série de questions pendant leur attente à l'arrêt. Huit chercheurs travaillent en binôme dans 14 arrêts. Le premier se tient face à l'arrêt et enregistre les temps d'arrivée des bus et les informations concernant les usagers en attente. Tandis que le deuxième se met avec les voyageurs et choisit aléatoirement quelque uns pour leur poser une série de questions à propos de l'attente.

À la fin de l'enquête, 655 témoignages valides ont été récoltés. Parmi ces 655, 88 utilisent déjà OBA, 23 utilisent un autre SIV et 544 n'ont pas accès à des informations temps-réel. L'enquête a été centrée seulement sur le temps d'attente. Les résultats affichent un impact minime sur le temps d'attente des voyageurs. Les auteurs estiment leur besoin d'élargir le champs de test et de concevoir une enquête d'avant et après utilisation du SIV, ce qui implique le suivi des témoins, afin de pouvoir comparer les résultats.

Cette perspective a mené à la deuxième étude de l'impact de OBA dans la ville de Tampa-Floride dans [Brakewood *et al.*2014]. Le choix du réseau de la ville de Tampa est fait parce qu'il garantit un environnement contrôlé, puisque OBA n'était pas encore fonctionnel, ainsi les participants à l'enquête ont été les seuls utilisateurs du système. Les participants ont été contactés via le site Web de l'opérateur de transport et toute l'enquête s'est faite en ligne. L'enquête s'est faite sur deux phases, la première avant l'utilisation de OBA et la deuxième après. Ainsi, les données de voyage utilisant OBA sont comparés avec celles sans informations temps réel. Dans cette étude, les critères d'analyses sont élargis pour contenir les changements de comportements des usagers (fréquences de correspondance et temps d'attente), leurs sentiments pendant l'attente (ennui, productivité, anxiété, détente, frustration, sécurité) et leurs satisfactions envers le service du réseau (fréquence de bus, temps d'attente, ponctualité des bus, nombre de correspondances, arrivée à destination à temps et le service général). Malgré la richesse des critères d'analyse, des résultats médiocres ont été notés. En effet, après avoir commencé avec

534 participants, 268 seulement ont terminé l'enquête qui a duré 3 mois. Ce faible nombre de témoignages fiables ainsi que l'absence d'alternatives de voyages sont responsables du résultat.

Des résultats meilleurs sont notés dans [Dziekhan and Kottenhoff2007], une enquête par questionnaire effectuée sur une ligne de Tramway. Cette enquête évalue l'utilisation d'information en temps-réel dans les arrêts de la ligne et estime que cela réduit le temps d'attente des voyageurs en moyenne de 20%. Cette réduction est aussi observée par [Schweiger2003] qui utilise des enquêtes sur le terrain. Ces résultats sont présentés sur une échelle réduite vu la difficulté et le coût élevé d'effectuer de telles études sur un grand nombre d'usagers. Ainsi, cette méthode ne se voit pas bien efficace pour évaluer les SIV modernes qui s'étalent sur des réseaux de plus en plus larges et multimodaux.

1.3.2 La simulation comme méthode d'évaluation

Dans cette section, nous nous intéressons à la simulation comme méthode d'évaluation. Il ressort de notre étude bibliographique que cinq principaux travaux répondent aux besoins de notre problématique et offrent un panorama complet de critères d'analyse. Le tableau 1.1 montre les caractéristiques des évaluations menées par ces travaux.

Les travaux à base de simulation intègrent souvent différentes conditions de fonctionnement des réseaux de transport telles que l'irrégularité du service et la congestion des lignes. Ils intègrent également plusieurs types d'informations que le SIV diffuse et considèrent plusieurs niveaux d'information, ce qui permet une évaluation plus fine et plus riche. Aussi, une variété de processus de prise de décision chez les utilisateurs des réseaux de TC est observée dans ces travaux ce qui garantit une diversité dans les comportements des voyageurs.

Toute cette richesse offerte par la simulation informatique aide à affiner les expériences menées dans ces travaux en permettant de concevoir plusieurs scénarios possibles. La diversité de ces scénarios impacte directement la qualité des analyses menées par ces expériences en offrant plus de critères d'analyse et ainsi plus de précision et de richesse.

Ainsi, on peut distinguer 3 modules essentiels qui forment la base d'un modèle de simulation destiné à mesurer l'impact des informations temps-réel dans un réseau de TC :

- Le module **offre de Transport** simule la performance du réseau de transport ainsi que ses fonctionnalités de base.
- Le module **comportement voyageur** simule le processus de prise de décision chez le voyageur à la réception des informations de voyage qui lui sont destinées.
- Le module **information voyageur** simule le processus de l'acquisition d'informations (réception et/ou estimation) jusqu'à sa diffusion dans le réseau.

Module : offre de transport

La première fonctionnalité de ce module est d'assurer la simulation des déplacements des véhicules en fixant une approche à suivre. Dans la majorité des travaux étudiés il s'agit d'approches à base de calendrier où des tableaux de marches sont utilisés pour organiser le mouvement des véhicules sur les réseaux. Dans [Hickman and Wilson1995], une approche stochastique dépendante du temps est utilisée. Cette approche a été choisie vu que les temps de parcours des arcs sont estimés en utilisant une approche probabiliste. La deuxième fonctionnalité de ce

-	Hickman	Coppola	Cats	Fonzone	Estrada
Gestion de l'offre	Stochastique dépendante du temps	Approche à base de calendrier	Approche à base de calendrier	Approche à base de calendrier	Approche à base de calendrier
Différentes conditions de fonctionnement	NON	Congestion des lignes et Régularité du service	Retards et Réduction de fréquence	NON	Modification des Tables de marche et Changement des fréquences
Nature des Informations fournies par le SIV	Temps du prochain départ estimé + Temps de parcours estimé + Niveau de précision	Temps d'attente + Capacité du bus	Temps d'attente estimé + Retard estimé	-	Temps d'arrivée du prochain bus
Différents niveaux d'information	7 niveaux	3 niveaux	3 niveaux	3 niveaux	6 niveaux
Localisations des informations	Arrêts et à bord des véhicules	Arrêts et à bord des véhicules	Arrêts	Arrêts	Arrêts
Processus de prise de décision (voyageurs)	Choix d'itinéraire dynamique	Behavioral discrete choice model (Random Utility Theory)	Random utility discrete choice model	3 catégories : à base de calendrier / Arriver le plutôt possible / à base de fréquence	À base de ligne (pas de changements)
Plateforme de simulation	SIMAN (Pegden et al. 1990)	-	BusMezzo	Matlab 2013	EOSimulation
Taille du réseau	O-D / 3 itinéraires	9 Lignes /188 voyages	3 lignes /100 stations /210 quais	4 stations / 4 lignes	13 lignes / 522 noeuds /1528 arcs
Nombre de Scénarios	7	24	9	3	6
Critères d'analyse	Temps de parcours moyen + écart des temps de parcours + probabilité qu'un voyageur arrive à destination à $t > X_{min}$ + nbre de voyageurs choisissant chaque itinéraire	Temps d'attente moyen + Temps de parcours moyen + Utilité systématique moyenne	Temps moyen de parcours (marche/à bord)+ Nb Changes + Distribution des choix d'itinéraires	Temps moyen de parcours + Charges des lignes	Temps moyen de parcours et ses composants (temps à bord- temps de marche- temps d'attente)

TABLE 1.1 – Les caractéristiques des évaluations menées par les travaux étudiés

module permet de manipuler le fonctionnement du réseau dans le but de pouvoir injecter des perturbations sur les réseaux afin de mieux évaluer l'impact des informations temps-réel. Ces informations sont connues pour être plus efficace en cas de dysfonctionnement sur les réseaux. Dans [Coppola and Rosati2009], une fonction de réinitialisation des temps de parcours des arcs est introduite afin de créer une irrégularité du service. Aussi, la congestion sur les lignes du même réseau est modulable. Ainsi, l'impact des SIV est évalué sous différentes conditions de fonctionnement du réseau. Dans [Cats *et al.*2011], deux cas de perturbation sont injectés dans le réseau séparément. Un retard de 15 minutes sur la ligne bleue du réseau entre les deux stations. Puis, une réduction de la fréquence de la ligne verte de 18 véhicules/heure à 6 véhicules/heure. Une perturbation du même type est aussi intégrée dans [Estrada *et al.*2015]. L'évaluation de l'impact des SIV en situation d'irrégularité des services est très importante car c'est sous ces conditions que les informations temps-réel sont plus utiles. En effet, un voyageur en attente d'embarquement ne se voit pas dans le besoin d'avoir des informations en temps-réel si aucun retard ou dysfonctionnement n'est observé. Tandis que, si le bus ou métro n'arrive pas à temps, le voyageur commence à douter et aura besoin d'être informé afin de savoir s'il doit changer d'itinéraires ou de mode afin d'atteindre sa destination et éviter un temps d'attente important. Ainsi, l'intégration de différentes conditions de fonctionnement est un élément indispensable dans l'évaluation des SIV spécialement afin d'enrichir les expérimentations par des scénarios permettant une analyse riche et précise de l'impact des informations temps-réel sur les déplacements des voyageurs et sur le fonctionnement des réseaux de transport et la qualité des services qu'ils offrent à ses usagers.

Module : comportement des voyageurs

Le deuxième module du modèle général de simulation simule le processus de prise de décision chez le voyageur à la réception des informations de voyage qui lui sont destinées. Ce processus est responsable du calcul et du choix de l'itinéraire à suivre par le voyageur, puis il se poursuit pendant le déplacement en réponse aux informations temps-réel selon l'approche de déplacement choisie. Dans la littérature, nous estimons que les approches proposées dans [Cats *et al.*2011], [Coppola and Rosati2009] et [Hickman and Wilson1995] semblent les plus réalistes et proches du comportement d'un voyageur moderne. Cependant, la variété d'approches intégrée dans [Fonzone and Schmöcker2014] est la plus adéquate dans un contexte d'évaluation d'impact des SIV dans un réseau de transport. Dans [Cats *et al.*2011], le modèle de prise de décision chez le voyageur est divisé en deux sous-modèles : le générateur de l'ensemble des choix et le processus de choix d'itinéraire. Le premier définit l'ensemble des itinéraires possibles entre l'origine et la destination du voyageur. Les itinéraires sont choisis selon des paramètres de préférences prédéfinis comme le nombre de correspondances. Ensuite, le processus de choix d'itinéraire détermine la manière dont le voyageur avance dans le réseau. En effet, ce processus est différent d'un choix statique d'itinéraire. Il ne choisit pas un itinéraire fixe parmi l'ensemble pré-choisi, mais il choisit une séquence de décisions en fonction de son avancement sur le réseau. Chaque décision est définie par le besoin de choisir le prochain élément de l'itinéraire (arrêt, ligne, correspondance) parmi l'ensemble des alternatives associées à cet élément. La figure 1.6 présente ce processus [Cats *et al.*2011].

La prise de décision implique une comparaison de l'utilité de chaque alternative disponible. $U(i)$ représente l'utilité d'une alternative i et elle est calculée par la fonction suivante basée sur

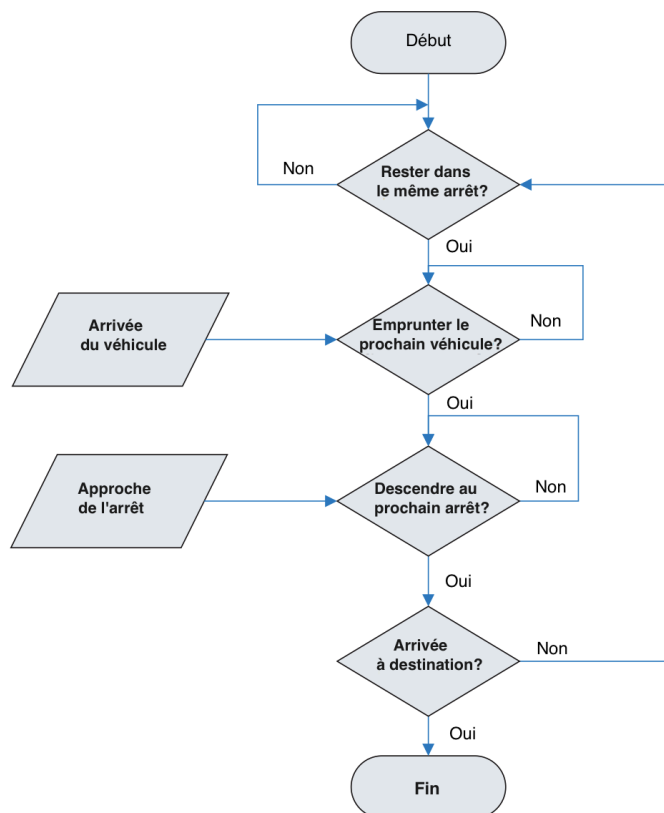


FIGURE 1.6 – Exemple d'un processus de choix d'itinéraire chez un voyageur

une enquête de préférences :

$$U_i = \beta_{wait}T_wait_i + \beta_{ivt}T_ivt_i + \beta_{walk}T_walk_i + \beta_{trans}trans_i + \epsilon_i$$

Avec :

- T_wait_i , T_ivt_i , T_walk_i et $trans_i$ respectivement, le temps d'attente, le temps à bord d'un véhicule, le temps de marche et le nombre de correspondances, associés à l'alternative i ;
- β le coefficient correspondant ;
- ϵ_i variable d'erreur.

La marche et l'attente ont le même coefficient qui est le double du temps à bord du véhicule, tandis que la correspondance est pénalisée par $4,77 \times$ le temps à bord.

Une autre fonction d'utilité est proposée dans [Coppola and Rosati2009] afin de guider le choix d'embarquement des voyageurs entre deux alternatives : emprunter le prochain véhicule disponible ou emprunter le suivant dans la liste des choix. Cette fonction est aussi basée sur une enquête auprès des usagers du réseau de transport public de la ville de Naples en Italie. La fonction est la suivante :

$$U_r = \beta_{TB}TB_r + \beta_{TE}TE_r + \beta_{CFW}CFW_r + \epsilon_r$$

$$U_{r'} = \beta_{TW}TW_{r'} + \beta_{TB}TB_{r'} + \beta_{TE}TE_{r'} + \beta_{CFW}CFW_{r'} + \epsilon_{r'}$$

avec

- r est le prochain véhicule disponible à l'arrêt.
- r' est le véhicule suivant selon l'ensemble des choix initiaux du voyageur.
- TW est le temps d'attente sur l'itinéraire minimal entre l'arrêt s et la destination.
- CFW est un indice de confort ; égal à 1 si le taux d'occupation du véhicule est supérieur à 50% et égal à 0 sinon.
- TB est le temps à bord jusqu'à la destination de l'itinéraire minimal à partir de l'arrêt s .
- TE est le temps de sortie entre l'arrêt d'arrivée et le centroïde de destination.

Le paramètre β est initié en fonction de la catégorie à laquelle appartient le voyageur.

Dans la même direction, [Hickman and Wilson1995] utilisent dans leur plateforme analytique un modèle de choix d'itinéraire dynamique pour gérer le processus de prise de décision chez les voyageurs. Ce modèle suggère que le voyageur attend dans son arrêt de départ jusqu'à l'arrivée d'un véhicule pour étudier son choix d'embarquement en évaluant le temps de parcours estimé. Selon les auteurs, cette stratégie permet aux voyageurs de profiter des informations temps-réel perçues pendant l'attente. Contrairement à ces travaux, 3 approches différentes de prise de décision sont utilisées dans [Fonzone and Schmöcker2014]. Les deux premières sont à base de calendrier prenant en compte les informations temps-réel et la troisième est à base de fréquence et ne considère pas les informations temps-réel. La différence entre les deux premières approches repose dans les critères de choix d'itinéraires : la première choisit un itinéraire parmi 4 propositions en fonction du temps de parcours estimé de chacune, tandis que la deuxième choisit l'itinéraire qui atteint la destination au plus tôt même si son temps de voyage est plus lent. Cela signifie que le voyageur préfère prendre le prochain véhicule et arriver plus tôt qu'attendre le suivant, qui a un temps de parcours plus court mais arrive après le premier.

La troisième approche ne considère pas les informations temps-réel et tend à choisir son itinéraire sous une approche attractive. Ce qui est remarquable dans cette étude est la diversité dans les processus de décision des voyageurs, ce qui offre une richesse aux expériences menées permettant une analyse proche d'une situation réelle où les utilisateurs d'un réseau de transport en commun forment une population variée. Généralement les populations (utilisateurs de réseaux de transport) peuvent être classées en 3 ou 4 catégories principales ce qui correspond au choix fait dans cette étude. Nous estimons qu'il est nécessaire d'avoir une variété dans les processus de prise de décision chez les usagers d'un réseau de transport en commun afin que la simulation permet de transposer au mieux la situation réelle afin que l'analyse soit réaliste. Ensuite, l'approche dynamique semble la plus adéquate pour simuler le processus de prise de décision chez le voyageur moderne sous l'influence des informations temps-réel essentiellement celles perçues pendant le déplacement (en-route). Ce module est fortement lié au module de l'information voyageur. La nature des informations et leurs niveaux d'accessibilité dans le modèle de simulation définissent l'approche de prise de décision des voyageurs que ce modèle peut intégrer. Dans la section suivante, on étudie ces deux aspects dans la littérature.

Module : information des voyageurs

Ce module gère les aspects des informations destinées aux voyageurs dans le modèle de simulation du réseau de transport en commun. Il assure la fonctionnalité d'un SIV dans le monde réel. La nature et les niveaux d'information sont les deux principaux aspects gérés par ce module. La nature de l'information est fortement liée au module de prise de décision chez le voyageur, tandis que le niveau d'information est un facteur essentiel des scénarios de simulation qui définissent le type des analyses. Concernant la nature des informations, tous les travaux étudiés intègrent des informations générales d'avant-voyage et/ou en-route (tableaux de marche, fréquences de ligne, horaires d'arrivée de véhicules, temps d'attente).

Des modules d'estimation sont généralement utilisés pour prédire les informations telles que les temps de parcours des arcs et les temps d'attente dans les arrêts. Ces modules nécessitent les données des positions des véhicules sur le réseau pour estimer les temps de parcours ou les nombres de voyageurs embarqués pour déduire les capacités d'embarquement des véhicules. Ces données peuvent être collectées en utilisant des technologies de géolocalisation telles que les GPS et les AVM (*Automated Vehicles Monitoring*) et des détecteurs infrarouges de mouvement. La figure 1.7 illustre le modèle de génération d'information utilisé dans [Coppola and Rosati2009]. Ce modèle génère les estimations des temps d'attentes dans les arrêts et les taux d'occupation des bus en temps-réel. Dans ce cas, les temps d'attente sont déduits à travers l'estimation des temps de parcours des arcs come suit : à un instant t , le modèle prévoit le temps de parcours de chaque arc du réseau pour n'importe quel instant du futur $\tau' > \tau$, en utilisant un algorithme basé sur le filtre de Kalman et en se basant sur les données de localisation des bus provenant du système de surveillance.

$$KALMANFILTRE \rightarrow t^{for}(\tau') \forall \tau' > \tau; \tau' \in \Theta$$

$t^{for}(\tau)$ est le temps de parcours estimé au temps τ .

Une fois le temps de parcours de l'arc $t^{for}(\tau')$ est estimé, le temps d'attente à un arrêt s

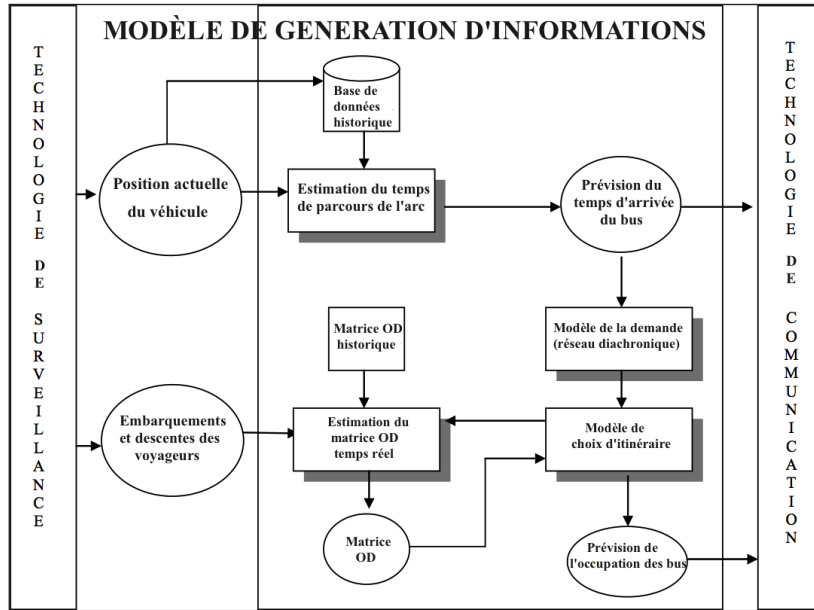


FIGURE 1.7 – Représentation du modèle de génération d'information

pour un véhicule r à l'instant futur τ' est donné ainsi :

$$wt_{r,s}^{for}(\tau) = at_{r,s}^{sched} - \tau + \sum_{\tau' < \tau} \sum_{ij} \delta_{ij,r}(\tau') \cdot [t_{ij}^{act}(\tau') - t_{ij}^{sched}(\tau')] + \\ + \sum_{\tau' > \tau} \sum_{ij} \delta_{ij,r}(\tau') \cdot [t_{ij}^{for}(\tau') - t_{ij}^{sched}(\tau')]$$

Avec :

- $wt_{r,s}^{for}(\tau)$ est le temps d'attente estimé à l'arrêt s pour le véhicule r au temps τ .
- $at_{r,s}^{sched}$ est le temps d'arrivée programmé du passage r à l'arrêt s .
- $\delta_{ij,r}(\tau)$ est l'élément générique de la matrice d'incidence dynamique "Arc, Passage" au temps τ , égal à 1 si le véhicule r a quitté l'arc ij et égal à 0 sinon.
- $t_{ij}^{act}(\tau), t_{ij}^{for}(\tau), t_{ij}^{sched}(\tau)$ sont respectivement les temps de parcours actuel, estimé et programmé de l'arc ij à $t=\tau$.

Ces données estimées sont ensuite communiquées aux voyageurs à travers les différents modes de diffusion afin de guider leurs déplacements.

La localisation de l'information (à l'arrêt ou à bord du véhicule) indique si l'information est d'avant-voyage ou en-route. Quand l'information est à l'arrêt, elle peut avoir les deux types d'information, alors que lorsqu'elle est à-bord du véhicule, il s'agit d'une information en-route seulement. Les informations en-route alimentent les processus de prise de décision dynamique des voyageurs et la localisation exacte peut définir quel type de décision est impacté par l'information. Par exemple, si l'information survient au niveau d'une station où des connexions sont possibles, le choix de correspondance et d'embarquement sont influencés. On remarque que les informations personnelles résultant de l'utilisation d'équipements comme les *smartphones* sont non considérées séparément. Une simple hypothèse, concernant les voyageurs équipés, est faite

au niveau d'accès aux informations temps-réel. On trouve cette hypothèse dans [Cats *et al.*2011] et [Estrada *et al.*2015] où simplement un accès permanent aux informations temps-réel représente ce type de voyageurs. Nous jugeons que ce type d'accès à l'information temps-réel nécessite plus de considération car il représente aujourd'hui l'un des modes d'informations les plus sollicités dans les réseaux de transport en commun. Ce mode nécessite plus d'attention par les modèles de simulation pour que leurs analyses soient adéquates avec le transport moderne. Le deuxième aspect de l'information est son niveau d'accès par les usagers des réseaux de transport. Dans la simulation, cet aspect sert à calibrer les expérimentations pour bien analyser l'impact des SIV. Dans la littérature, différents niveaux d'information sont toujours utilisés dans les expérimentations. Parmi les travaux étudiés, on choisit l'exemple de [Estrada *et al.*2015] qui est le plus détaillé. 6 niveaux d'accès aux informations temps-réel sont étudiés :

1. Niveau 1 : (RTI-allways) représente le cas où tous les voyageurs ont accès aux informations temps-réel sur tout le réseau. Ça représente aussi les utilisateurs équipés d'un équipement mobile (*smartphone*).
2. Niveau 2 : (RTI@origin) Les voyageurs n'ont accès aux informations temps-réel qu'à l'origine. Cela représente les voyageurs qui accèdent aux informations via un internet avant leur départ (maison/bureau).
3. Niveau 3 : (RTI-1Line) Les voyageurs ont accès aux informations spécifiques à une ligne particulière à tout moment de leur déplacement.
4. Niveau 4 : (STT) Pas d'accès aux informations temps-réel. Les voyageurs utilisent les tableaux de marche statiques et ne changent plus leur itinéraire.
5. Niveau 5 : (RTI@stops) Les voyageurs accèdent seulement aux informations temps-réel affichées dans les arrêts.
6. Niveau 6 : (FBA) Les tableaux de marche ne sont pas disponibles. Les voyageurs prennent leurs décisions en suivant une approche à base de fréquence.

La figure 1.8 montre comment ces niveaux d'information sont utilisés pour évaluer l'impact des informations temps-réel. Une analyse en fonction du temps moyen de parcours et ses composantes (temps de marche, temps d'attente, temps d'embarquement) nous montre la différence entre les différents niveaux.

Dans la section suivante, nous étudions les méthodologies d'expérimentation suivies dans la littérature pour analyser et évaluer l'impact des SIV à travers les modèles de simulation et nous examinons leurs résultats.

1.3.3 Expérimentations & Résultats

Scénarios

Si la majorité des recherches actuelles choisissent la simulation comme outil, c'est pour sa grande capacité à étendre les champs d'expérimentation à travers la possibilité de tester plusieurs scénarios à la fois. Entre les différents niveaux d'information et les différentes conditions de fonctionnement des systèmes de transport, les scénarios utilisés dans les travaux étudiés ont permis d'analyser les SIV sous plusieurs angles offrant une évaluation riche et multi-critères. En prenant comme exemple la méthodologie suivie par [Cats *et al.*2011], on remarque que l'analyse fine de l'impact des informations temps-réel résultent de la combinaison entre 3 niveaux

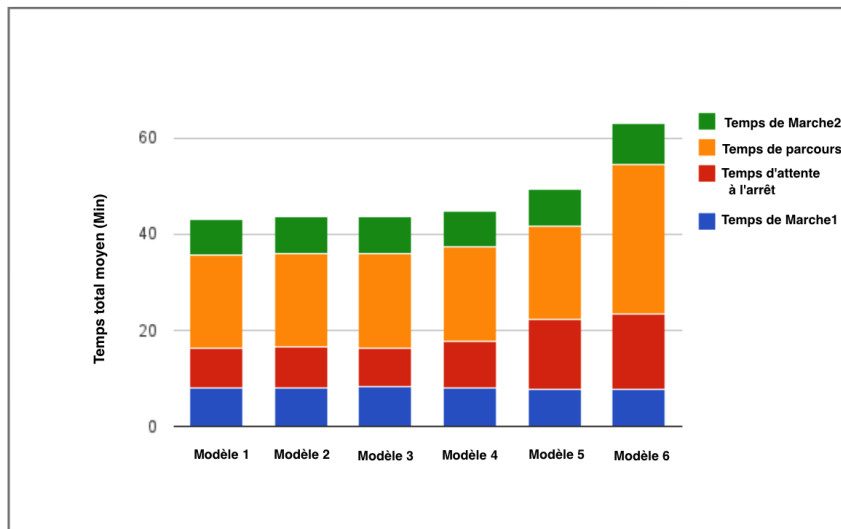


FIGURE 1.8 – Comparaison entre les différents niveaux d'information en fonction du temps moyen de parcours

d'information temps-réel et 3 conditions de fonctionnement du réseau de métro de Stockholm dans les 9 scénarios testés par le modèle de simulation. Les 3 niveaux d'information sont les suivants :

1. *Platform* : info-temps-réel pour tous les trains partant d'un quai spécifique.
2. *Station* : info-temps-réel pour tous les trains partant d'une station spécifique.
3. *Network* : info-temps-réel pour tous les trains du réseau.

Les 3 conditions de fonctionnement sont :

1. R : service normal.
2. DR : un retard de 15 min sur la ligne Bleue entre deux stations Fridhemsplan et T-Centralen.
3. DF : une réduction de la fréquence de la ligne Green de 18 véhicules heure à 6.

Ainsi, les 9 scénarios résultent de la combinaison des 3 niveaux d'information 1, 2 et 3 avec les 3 conditions R, DR et DF. La figure 1.9 montre la richesse de l'analyse extraite des 9 scénarios [Cats *et al.*2011].

Un deuxième exemple de résultats pertinents à cette problématique est donné dans [Coppola and Rosati2009], où 24 scénarios sont testés. 4 niveaux de demande de voyage (congestion de lignes de bus) sont analysés en fonction de 3 niveaux d'information et de 2 niveaux de régularité de service. Ces scénarios permettent d'évaluer l'impact des SIV sous différentes conditions de fonctionnement du réseau. Les scénarios ont été conçus comme suit : 4 niveaux de congestion de lignes (0/ Faible-25% / Moyenne-50% / Forte-80%) sont simulés chacun sous 6 combinaisons en fonction de la régularité du service (Oui / Non), de la disponibilité de l'information de temps d'attente (Oui / Non) et la disponibilité de l'information des capacités des bus (Oui / Non). La figure 1.10 décrit les 6 combinaisons.

Scénarios	Temps de parcours total (s)	Changement du temps de parcours total (%)	Temps à bord d'un véhicule(s)	Temps de marche et d'attente(s)
R1	1,081		554	527
R2	1,046	-3.2	557	489
R3	1,035	-4.3	538	497
DF1	1,418		553	865
DF2	1,293	-8.8	545	748
DF3	1,260	-11.1	523	737
DR1	1,771		1,116	655
DR2	1,733	-2.2	1,115	617
DR3	1,603	-9.5	1,054	549

FIGURE 1.9 – Analyse résultant de 9 scénarios

Indice	Irrégularité du service	Informations de temps d'attente	Informations de capacité de bus
1		non	non
2	non	oui	non
3		oui	oui
4		non	non
5	oui	oui	non
6		oui	oui

FIGURE 1.10 – Les 6 combinaisons de test dans [Coppola and Rosati2009]

Une méthodologie plus basique mais pertinente a été utilisée par [Hickman and Wilson1995] dans leur étude de l'impact des informations temps-réel sur le temps de voyage et le processus de choix d'itinéraire chez les voyageurs. 7 scénarios ont été conçus en fonction du processus de décision chez les voyageurs, leurs degrés d'information et du niveau d'exactitude des informations diffusées.

Nous constatons que les méthodologies d'analyse et de test ont suivi l'évolution que connaissent les SIV et sont devenus plus complexes. Les niveaux d'information, les degrés d'informations des voyageurs, les différentes conditions de fonctionnement du réseau sont des éléments imbriqués dans le processus d'évaluation des SIV. Leur combinaison devient essentielle afin d'avoir une évaluation complète qui intègre tous les aspects d'un réseau de transport en commun.

Résultats

Dans cette partie, nous exposons quelques résultats issus des travaux étudiés dans la littérature. Afin d'être cohérents, nous choisissons les résultats liés aux scénarios décrits dans la section précédente.

Dans [Hickman and Wilson1995], le cas étudié relie deux points origine et destination où 3 choix d'itinéraires sont possibles utilisant 3 lignes différentes. La simulation fournit aux voyageurs une estimation du temps de départ, du temps de parcours et le degré de précision des informations afin de les utiliser dans leur processus de choix d'itinéraire dynamique. Comme précisé ultérieurement, 7 scénarios sont testés. Malgré ce nombre de scénarios, les résultats obtenus dans cette étude indiquent en général un faible impact des informations temps-réel sur le temps de parcours et sur le processus de choix d'itinéraires des voyageurs. Selon les auteurs, ces résultats sont dus en partie à la taille réduite de l'axe étudié. En effet, le modèle de choix d'itinéraire modélisé dans ce travail nécessite un axe d'application plus large et riche en alternatives afin que les informations temps-réel puissent avoir un impact. Aussi, l'absence de situations de perturbations favorise ce type de résultats car c'est en de telles situations que les informations temps-réel se montrent utiles. D'autre part, l'étude a montré que le degré de certitude des informations diffusées est important. Les informations temps-réel doivent être exactes afin d'être utiles.

Dans [Coppola and Rosati2009], les auteurs tendent à évaluer l'impact des informations temps-réel dans un réseau de transport en commun sous différentes conditions du réseau (congestion de ligne et irrégularité de service). Les résultats montrent que la provision de l'information du temps d'attente estimé implique la hausse du temps moyen d'attente contre la baisse du temps moyen de voyage. Cela s'explique par le fait que grâce à cette information, les voyageurs adoptent un comportement adaptatif intelligent de choix du véhicule à emprunter. C'est à dire qu'ils n'empruntent pas le premier véhicule disponible (comportement adaptatif indifférent). Concernant la capacité de bus, l'impact de cette information est nul dans le cas d'un service régulier et négligeable en cas de service irrégulier. Cependant, son impact est très significatif en termes d'utilité de voyageurs : une hausse de 6-9% est observée lors de la simulation. La hausse la plus importante correspond au niveau élevé de congestion sur le réseau.

Dans [Cats *et al.*2011], l'effet des informations temps-réel sur le processus de choix d'itinéraire des voyageurs dans un réseau de transport en commun est étudié. Le modèle de simulation est testé sur des données réelles du métro de la ville de Stockholm en Suède. Les 9 scénarios de test sont présentés dans la section précédente. La figure 1.9 montre l'impact des niveaux

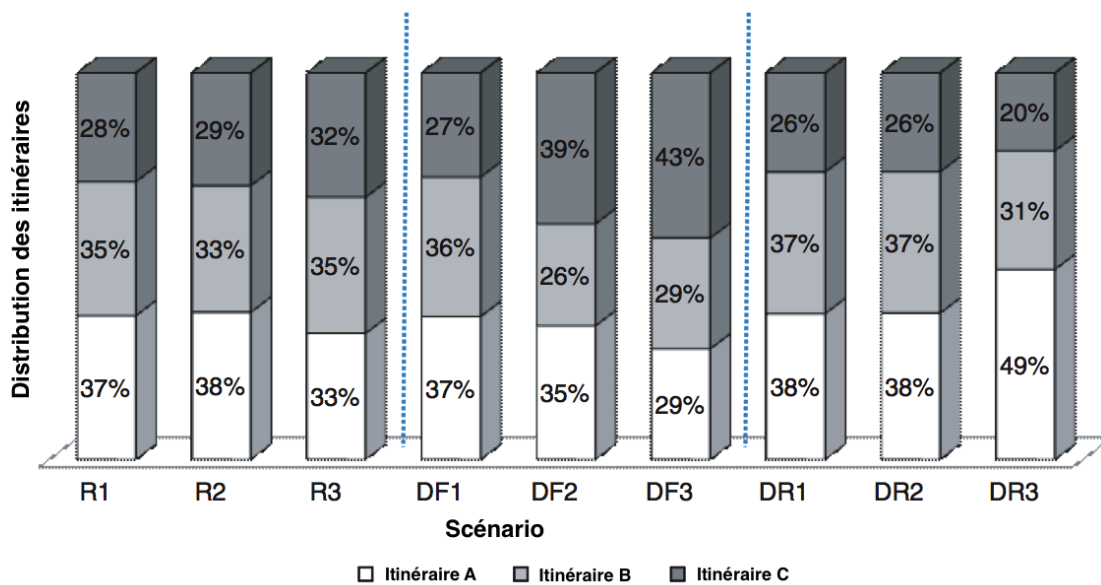


FIGURE 1.11 – La distribution des voyageurs entre 3 chemins reliant une paire origine-destination sous les différents scénarios

d'information et des scénarios de perturbations sur les conditions de voyage des usagers du réseau (temps d'attente et de marche, temps de parcours) [Cats *et al.*2011]. Concernant le choix d'itinéraires, les résultats indiquent que les niveaux d'information affectent ce processus. La figure 1.11 nous montre la distribution des voyageurs entre 3 chemins reliant une paire origine-destination sous les différents scénarios. On constate que cette distribution varie remarquablement en fonction des niveaux d'information et des conditions de fonctionnement du réseau.

Une mesure à l'échelle d'arrêt est aussi faite dans cette étude. La figure 1.12 nous montre le résultat de l'analyse du coefficient de variation du nombre de voyageurs embarqués à l'arrêt F de la ligne verte. Cette analyse évalue l'impact des différents scénarios (niveaux d'information et conditions de fonctionnement) sur le choix de l'arrêt F pour commencer un voyage ou faire une correspondance.

On trouve d'autres manières d'analyse dans [Estrada *et al.*2015] où différents niveaux d'information présentés dans la section 1.3 sont évalués en comparant leurs temps de parcours sur une série de paires origines-destinations (Figure 1.13). Aussi, une comparaison de leurs temps de parcours moyen en fonction de forte condition de congestion et d'irrégularité de service est présentée.

Discussion

Les modèles de simulations présentés dans cette section présentent un outil pertinent d'évaluation de l'impact des SIV dans un réseau de transport en commun. Les 3 modules assurent une richesse au niveau des critères d'analyse, et offrent une évaluation fine qu'on peut constater en observant les différents types de résultats affichés dans la littérature. Cette richesse se caractérise par les possibilités de combinaisons des 3 modules : l'offre de voyage, le processus de prise de décision du voyageur et l'information voyageur. Ces combinaisons assurent un

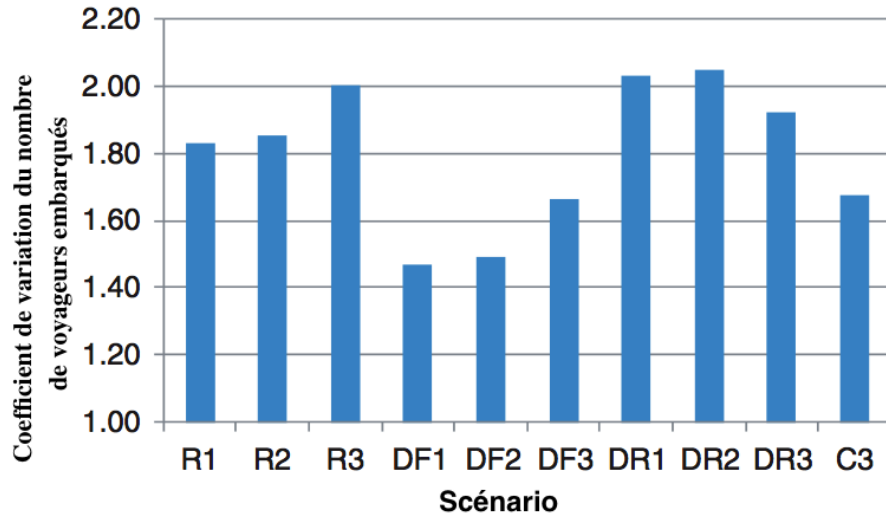


FIGURE 1.12 – Coefficient de variation du nombre de voyageurs embarqués à l'arrêt F de la ligne verte

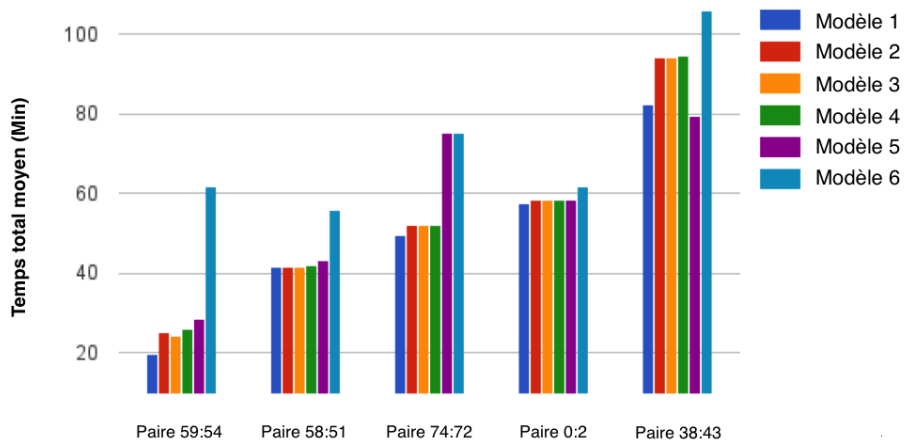


FIGURE 1.13 – Temps de parcours moyen de différentes paires origine-destination

champ expérimental large et ouvert permettant de multiplier les scénarios étudiés. Cependant, nous notons que l'information temps-réel personnalisée n'est jamais considérée en tant que telle dans la littérature. Or, ce type d'information représente un élément important dans les SIV modernes et joue un rôle considérable dans le transport d'aujourd'hui. En effet, l'information temps-réel évaluée dans les travaux étudiés est de nature générale qu'un voyageur reçoit par les panneaux d'affichage et sonores dans les stations et aussi par le Web et qui est accessible à tous. Ce type d'information diffère de l'information personnalisée qui est accessible via des équipements mobiles, généralement les *smartphones*, et qui guide les voyageurs essentiellement pendant leur déplacement en leur fournissant des informations spécifiques à leurs itinéraires et qui ne sont pas fournies aux autres voyageurs non équipés. Le taux d'équipement des voyageurs des réseaux de transport est de plus en plus important et nécessite d'être intégré en tant que paramètre d'analyse dans l'évaluation des SIV modernes.

Dans les travaux cités, les tailles des réseaux simulés et le nombre de voyageurs dans les expérimentations effectuées sont assez peu importants. Nous pensons que, pour avoir une évaluation réaliste, il faudrait que les cas simulés soient les plus proches possibles des situations réelles. Or, dans le cas du transport en commun, le nombre d'usagers est généralement élevé ainsi que les tailles des réseaux qui s'étendent de plus en plus en suivant l'explosion démographique des villes. Ainsi, nous estimons que pour évaluer un SIV sur un réseau de transport en commun moderne, le cas étudié doit respecter ces deux caractéristiques.

1.4 Conclusion

Dans ce chapitre, nous avons présenté les SIV et leur réponse aux exigences posées par l'évolution des systèmes de transport et par le changement de leurs environnement, ainsi que les travaux évaluant leur impact. En conclusion de ce chapitre, nous croyons que la simulation est une méthode pertinente et capable de mieux évaluer l'impact des informations temps-réels sur les réseaux de transport modernes.

Cependant, les travaux de la littérature sur cette question souffrent d'un inconvénient majeur : la non-considération explicite des voyageurs connectés, et par conséquent des informations personnalisées. Nous remarquons aussi que la taille des échantillons simulés est généralement non mentionnée dans ces travaux. Dans le chapitre suivant, nous élargissons notre étude des simulateurs dans le but de pallier cette limitation. Notre étude se concentre plus particulièrement sur la simulation multi-agent des déplacements, pour les raisons que nous exposons dans le chapitre suivant.

Chapitre 2

La simulation multi-agent des déplacements

Sommaire

2.1	Introduction	31
2.2	La notion d'agent	32
2.2.1	Définition	32
2.2.2	Caractéristiques	33
2.2.3	Types d'agents	33
2.3	Les systèmes multi-agents	34
2.3.1	Définition	34
2.3.2	Organisation des SMA	36
2.3.3	Interaction	37
2.3.4	Environnement	37
2.4	Simulateurs multi-agents des déplacements	38
2.4.1	AgentPolis	39
2.4.2	Transims	41
2.4.3	MATSim	43
2.5	Plateformes de simulation multi-Agent	45
2.5.1	Swarm	46
2.5.2	Mason	46
2.5.3	NetLogo	47
2.5.4	Gama	47
2.5.5	Repast	49
2.5.6	Discussion	52
2.6	Conclusion	53

2.1 Introduction

Dans le chapitre précédent, nous avons exprimé notre préférence pour la simulation comme moyen d'évaluation de l'impact de l'information des voyageurs. Dans ce cadre, il existe plu-

sieurs simulateurs de mobilité des voyageurs dans la littérature. Cependant, dans le cadre d'une évaluation d'impact de l'information voyageurs par simulation, il faut que les comportements des voyageurs puissent intégrer les informations dont on désire évaluer l'impact. Ainsi, un processus de prise de décision individuel doit accompagner chaque voyageur simulé. Ce processus définit par la suite les réactions des voyageurs suite aux informations perçues. Ces réactions représentent l'objet de l'analyse nécessaire afin d'évaluer l'impact de ces informations. Par conséquent, les voyageurs simulés doivent être munis d'un degré d'autonomie et de perception de leur environnement pendant leurs déplacements. Pour cette raison, ce chapitre traitera exclusivement des approches pour la simulation des déplacements, qui représentent les voyageurs d'une manière individuelle sous la forme d'agents : les simulations multi-agents de déplacements.

Les systèmes multi-agents (SMA) mettent en réseau un ensemble dynamiques d'agents, et offrent en effet un outil pertinent dans la gestion de systèmes complexes distribués en général. Les SMA sont efficaces pour la modélisation et la simulation de systèmes nécessitant l'interaction de plusieurs entités. En effet, la simulation de certaines situations (évacuations, attentats, etc.) est délicate, coûteuse et difficile à réaliser. Ainsi, la simulation multi-agent est une solution pertinente pour ce genre de problématiques. Un SMA permet d'intégrer un nombre important d'individus dans une simulation tout en assurant une gestion et un suivi individuel de chacun d'eux. Grâce aux interactions, un SMA permet de modéliser différents modèles sociaux où plusieurs formes d'échanges et d'interactions sont simulées. Ces capacités permettent de simuler, par exemple, une population, des piétons, des voyageurs dans un réseau de transport, des spectateurs dans une salle de spectacle ou un stade, des visiteurs d'un salon ou d'un centre commercial, etc. La simulation des déplacements permet aux architectes de bien modéliser les plans de déplacements avec l'avantage de pouvoir tester et calibrer plusieurs scénarios. Elle offre aux gestionnaires de ses réseaux plusieurs avantages et possibilités en leur permettant d'étudier et d'analyser d'une manière microscopique leurs méthodes d'affectation ou leurs systèmes d'information voyageurs essentiellement en situation de dysfonctionnement.

Ce chapitre est structuré comme suit. La section 2.2 définit la notion d'agent et la section 2.3 définit les systèmes multi-agents. La section 2.4 présente trois simulateurs multi-agents de déplacement de la littérature. Concluant que ces simulateurs ne sont pas pertinents pour nos objectifs, nous présentons dans la section 2.5 les plateformes de simulation multi-agents généralistes. Nous en choisissons une qui sera développée dans le chapitre suivant. La section 2.6 est notre conclusion.

2.2 La notion d'agent

2.2.1 Définition

Dans la littérature, plusieurs définitions sont utilisées pour le terme « agent ». Ces définitions varient selon le contexte et l'utilisation. Nous choisissons deux définitions qui semblent les plus appropriées à notre contexte. La première est celle de Wooldridge qui dans son livre [Wooldridge2002] définit un agent en tant que « système informatique situé dans un environnement et capable d'ap-

pliquer des actions autonomes dans le but de satisfaire ses buts ». La deuxième est une définition plus détaillée que Ferber utilise dans son livre [Ferber1995] et dont il illustre 6 caractéristiques :

Définition AGENT LOGICIEL

on appelle agent purement communicant (ou agent logiciel) une entité informatique qui :

1. se trouve dans un système informatique ouvert (ensemble d'applications, de réseaux et de systèmes hétérogènes),
2. est mue par un ensemble d'objectifs propres,
3. possède des ressources propres,
4. ne dispose que d'une représentation partielle des autres agents,
5. possède des compétences (services) qu'elle peut offrir aux autres agents,
6. a un comportement tendant à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose et en fonction de ses représentations et des communications qu'elle reçoit.

Il ressort de ces définitions des propriétés clés, comme l'autonomie, la réactivité, la perception et la communication. D'autres propriétés peuvent être attribuées aux agents, comme la rationalité.

2.2.2 Caractéristiques

Les caractéristiques les plus communément admises pour décrire un agent sont l'autonomie, la réactivité et la sociabilité.

- L'autonomie : [Russell and Norvig2003] spécifient qu'un agent autonome est un agent qui agit sans l'intervention des humains ou des autres agents, et qui de ce fait possède un contrôle sur ces actions et sur ses états internes. Un agent est ainsi une entité informatique qui possède la liberté de répondre aux appels à ses compétences et de prendre des initiatives sans avoir recours à des interventions externes.
- La réactivité d'un agent est sa capacité de réagir en cas de changements de l'état de son environnement. Les choix d'actions se feront en fonction de sa perception des changements. Cette perception guidera dans la suite ses choix et sa gestion d'actions.
- La sociabilité est le degré d'interaction qu'un agent peut avoir avec les autres agents. L'interaction peut se faire pour un besoin propre à l'agent ou pour un besoin de coordination entre les agents du même système.

2.2.3 Types d'agents

Plusieurs travaux classent les agents selon leurs caractéristiques. [Weiss1999] distingue deux types d'agents en se basant sur leurs capacités de raisonnement : les agents réactifs qui sont fondés sur un mécanisme de stimuli-réactions, et les agents cognitifs qui ont la particularité de posséder un module de délibération qui intègre un processus décisionnel complexe.

- Les agents réactifs sont des agents qui perçoivent et réagissent aux changements de leur environnement. Leurs réactions se font selon des règles prédéfinies d'une façon rapide similaires à des réflexes. Ces caractéristiques rendent ce type d'agents adapté à la simulation de sociétés (humaines ou animales) ou d'un ensemble de machines dans le domaine robotique. Ce type de simulation est souvent utilisé pour analyser l'impact des comportements individuels des agents sur le cheminement du groupe.
- Les agents cognitifs se caractérisent par la possession d'un module de délibération qui intègre un processus décisionnel complexe et qui définit les actions de l'agent. Selon [Mandiau et al.02a], leurs actions sont souvent le résultat d'une planification et d'un raisonnement rationnel qui se base sur un ensemble de connaissances sur leur environnement, sur les autres agents et sur eux mêmes. Parmi les architectures d'agents cognitifs les plus connues est l'architecture BDI (*Belief, Desire, Intention*) [Rao and Georgeff1992]. Cette architecture intègre les notions de croyance, de désirs et d'intention dans le processus décisionnel et s'avère bien appropriée, par exemple, à l'environnement de la réalité virtuelle. Dans [Shendarkar et al.2008], l'architecture BDI est étendue pour proposer une simulation d'évacuation de foule. Les caractéristiques de l'agent cognitif le rend bien approprié pour ce type de simulations. En effet, l'influence d'éléments émotionnels sur son processus décisionnel favorise son utilisation pour étudier le comportement humain comme dans le cas de [Lyell and Becker2005] qui étudie le comportement de foules dans un contexte de crise à l'aide d'agents cognitifs.

D'autres critères peuvent être utilisés pour classer les agents. Un agent peut être statique ou mobile en fonction de sa capacité à parcourir les réseaux informatiques tel que Internet ou un réseau local afin d'exécuter ses tâches. Aussi, les degrés d'adaptation, d'autonomie et de coopération sont des critères importants qui permettent de classer les agents ainsi :

- Agent collaboratifs : ces agents favorisent la coopération à l'adaptation afin de résoudre une tâche globale qui nécessite l'association de plusieurs compétences individuelles. Ces agents sont généralement utilisés pour résoudre des problèmes distribués.
- Agents interfaces : ces agents jouent le rôle d'assistants personnels aux utilisateurs en collaborant et communiquant avec eux afin d'orienter l'exécution des tâches. Ces agents favorisent l'autonomie et l'apprentissage afin d'assister l'utilisateur et ils sont généralement utilisés dans le domaine des interfaces homme/machine.
- Agents d'information : ces agents se caractérisent par leurs autonomie en matière de recherche d'informations lorsque leurs tâches le nécessitent. Ils sont capables de collecter et de gérer des informations provenant de sources différentes. Selon [Adam2000], ces agents sont capables de lancer leurs activités de recherche et de collecte d'informations d'une manière autonome ; en fonction d'un raisonnement interne, d'un manque d'informations ou d'une nouvelle disponibilité d'informations ou de sources d'information.

2.3 Les systèmes multi-agents

2.3.1 Définition

[Ferber1995] donne la définition suivante d'un système multi-agent :

Définition SYSTÈME MULTI-AGENT

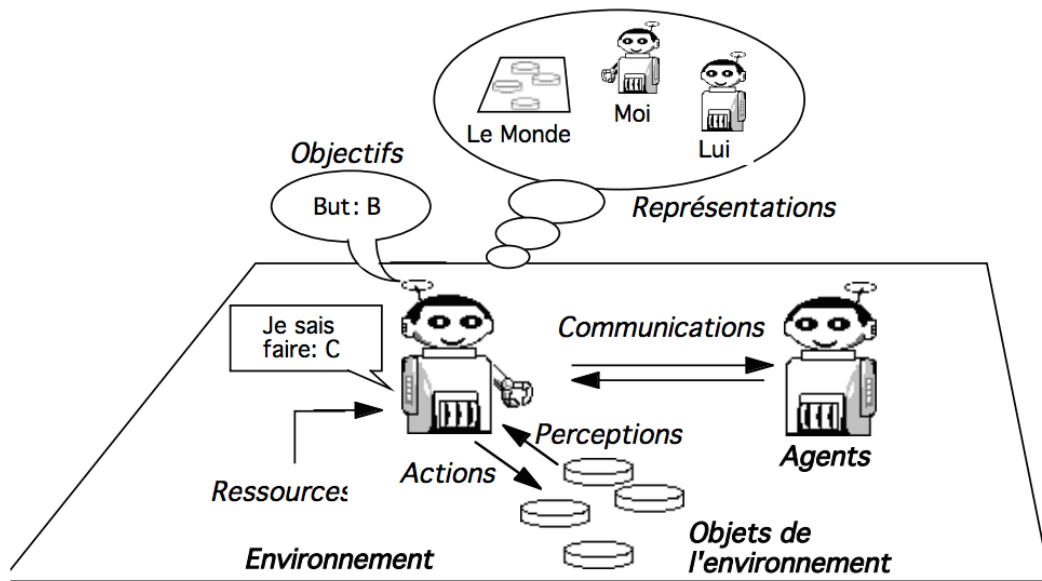


FIGURE 2.1 – Illustration de la notion de système multi-agent selon Ferber

On appelle système multi-agent, un système composé des éléments suivants :

1. Un environnement E , c'est-à-dire un espace disposant généralement d'une métrique.
2. Un ensemble d'objets O . Ces objets sont situés, c'est-à-dire que, pour tout objet, il est possible, à un moment donné, d'associer une position dans E . Ces objets sont passifs, c'est-à-dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents.
3. Un ensemble A d'agents, qui sont des objets particuliers ($A \subseteq O$), lesquels représentent les entités actives du système.
4. Un ensemble de relations R qui unissent des objets (et donc des agents) entre eux.
5. Un ensemble d'opérations Op permettant aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O .
6. Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, que l'on appellera les lois de l'univers.

La figure 2.1 donne une illustration de la notion de système multi-agent. [Fayech2003] définit les SMA ainsi : « Un système multi-agent est un réseau d'agents (solveurs) faiblement couplés qui coopèrent ensemble pour résoudre des problèmes qui dépassent les capacités ou les connaissances individuelles de chaque agent. Les agents sont autonomes et peuvent être de natures hétérogènes. »

Ainsi, les agents d'un SMA, n'ayant pas une visibilité globale sur leur environnement, ne peuvent avoir qu'un champ d'actions limité sur l'ensemble des objets de cet environnement. De

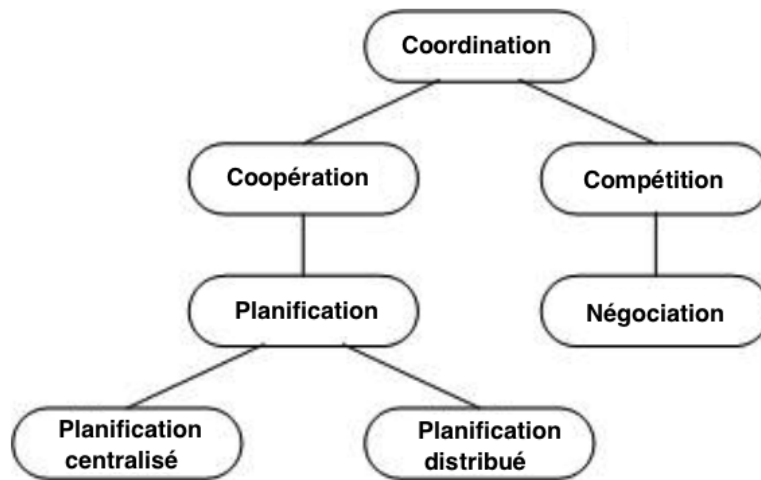


FIGURE 2.2 – Les formes d’interactions entre agents

ce fait pour résoudre un problème global, ces agents sont amenés à coopérer et à communiquer pour échanger des informations et pour mieux coordonner leurs actions individuelles et locales. Par conséquent, un système multi-agent doit avoir une forme d’organisation qui définit les règles de ces échanges.

2.3.2 Organisation des SMA

La notion d’organisation des agents dans un SMA, s’impose pour définir les rôles des agents et les mécanismes de coordination et de communication entre eux. [Weiss1999] propose le schéma de la figure 2.2 afin d’illustrer les différents types d’interaction qui peuvent avoir lieu entre les agents d’un SMA.

[Weiss1999] définit la coordination comme « la propriété d’un système composé d’au moins deux agents, exécutant des actions dans un environnement partagé ». Ainsi, la notion d’environnement partagé implique le besoin de coordination entre les différents agents du système. Deux types de coordinations existent :

- la négociation pour les agents antagonistes (ayant des buts et des objectifs contradictoires) afin d’éviter les situations de conflits.
- la coopération pour les agents non antagonistes afin d’améliorer l’efficacité et l’utilité de chacun d’eux.

La négociation peut être considérée comme un moyen de résolution de conflits entre plusieurs agents. Selon [Fayech2003], la négociation est une méthode de coordination qui permet à plusieurs agents d’atteindre un accord mutuel pour entreprendre une action donnée d’une certaine manière. Elle induit, par cette communication, des relaxations de buts initiaux, des concessions mutuelles, des mensonges ou des menaces. Ensuite, d’après [Weiss1999], la coopération est la coordination parmi des agents non antagonistes, qui cherchent à se satisfaire mutuellement sans se gêner. Ce type de coordination basé sur l’échange d’informations peut avoir la forme d’une

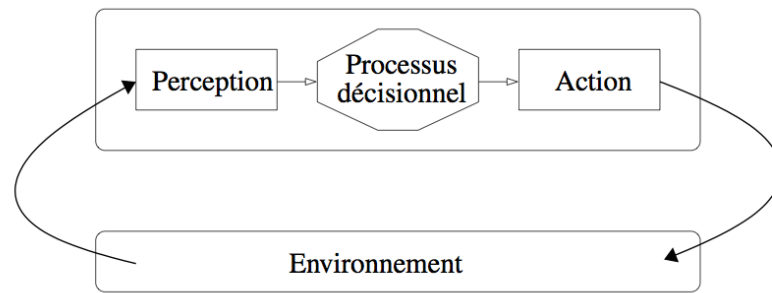


FIGURE 2.3 – Le lien entre l'environnement et l'agent

collaboration où la répartition des tâches entre les agents est étudiée.

2.3.3 Interaction

Pour se coordonner entre eux, pour négocier, et pour coopérer les agents ont besoin d'interagir entre eux. Dans les SMA, deux types d'interactions existent : la communication directe et la communication indirecte.

- La communication directe utilise un langage de communication pour pouvoir échanger des messages entre les agents. Les messages contiennent des informations que les agents désirent échanger afin d'aboutir aux besoins de leurs tâches. L'émetteur utilise le langage pour coder l'information à envoyer qui sera ensuite décodée par l'agent récepteur via le même langage. D'après [Roze2003], deux langages sont couramment utilisés : KQML (*Knowledge Query and Manipulation Language*) et ACL (*Agent Communication Language*).
- La communication indirecte utilise l'environnement comme support d'interaction. Pour transmettre une information, un agent émetteur effectue une modification sur l'environnement qui sera perçue puis interprétée par l'agent récepteur. Le meilleur exemple de ce type de communication est celui des fourmis qui communiquent entre eux le chemin vers la source de nourriture à travers le dépôt de phéromones.

2.3.4 Environnement

En se référant à la définition des systèmes multi-agents de Ferber présentée précédemment, nous constatons que l'environnement représente une composante majeure et essentielle dans ces systèmes. En effet, tous les états et les actions d'un agent dépendent de l'état de l'environnement. La figure 2.3 montre le rôle de l'environnement dans le processus décisionnel de l'agent. Aussi, il est la base des échanges et interactions entre les différents agents d'un SMA, ce qui représente le moteur du paradigme multi-agent.

L'environnement dans les systèmes multi-agents possède plusieurs propriétés, ce qui offre une richesse au processus de modélisation de ces derniers afin de répondre aux besoins de l'utilisateur. Voici les différentes propriétés selon la suggestion de [Russell and Norvig2003] de présenter chaque propriété en opposant ses deux extrêmes.

- Accessible ou inaccessible : un environnement accessible est un environnement dans lequel un agent peut obtenir une information complète, exacte et mise à jour de son état. La majorité des environnements modérément complexes (incluant, par exemple, le monde réel et Internet) sont inaccessibles. Plus l'environnement est accessible, plus simple est la construction d'un agent afin d'y opérer.
- Déterministe ou indéterministe : un environnement déterministe est un environnement dans lequel n'importe quelle action a un seul effet garanti : il n'y a aucune incertitude quant à l'état qui résultera de l'application d'une action. Les environnements non déterministes posent plus de problèmes pour le concepteur du système.
- Épisodique ou non épisodique : dans un environnement épisodique, la performance d'un agent est dépendante d'un nombre d'épisodes discrets, sans lien entre les performances d'un agent dans des scénarios différents. Un exemple d'environnement épisodique pourrait être un système de tri de courrier électronique. Les environnements épisodiques sont plus simples du point de vue du développeur agent parce que l'agent peut décider quelle action appliquer en se basant seulement sur l'épisode courant ; il n'a pas besoin d'interaction entre l'épisode courant et les épisodes à venir.
- Statique ou dynamique : un environnement statique est un environnement qui ne risque de changer d'état que lors de l'application d'actions par l'agent. Un environnement dynamique est un environnement qui a d'autres processus agissant dessus et qui risque de subir des changements au détriment de la volonté de l'agent. Le monde réel est un environnement hautement dynamique.
- Discret ou continu : un environnement est discret s'il comporte un nombre fixe et fini d'actions et de perceptions possibles. Russel et Norvig donnent un jeu d'échec comme exemple d'environnement discret et la conduite d'un taxi comme exemple d'environnement continu.

Les classes les plus complexes d'environnements concernent ceux qui sont inaccessibles, indéterministes, non-épisodiques, dynamiques et continus. Dans ce qui suit, nous nous intéressons à la simulation multi-agent, et aux plateformes de simulation existantes.

Dans la section qui suit, nous nous intéressons aux applications qui se fondent sur le paradigme multi-agent présenté dans cette section, pour la simulation des déplacements. Cette étude a pour objectif d'éventuellement choisir parmi ces applications, celle sur laquelle fonder notre proposition.

2.4 Simulateurs multi-agents des déplacements

On distingue principalement trois types de modèles de simulation de déplacements. Les modèles de déplacements macroscopiques considèrent une représentation des déplacements comme des flux, dont les comportements sont décrits par des équations. Les variables définies dans le modèle sont les débits, concentrations et vitesses sur les différents tronçons du réseau modélisé. Les modèles de déplacements mésoscopiques utilisent généralement les mêmes outils, mais représentent les déplacements sous forme de pelotons ou des groupes de voyageurs ou de véhicules (e.g. de [De Palma and Marchal2002]). Il s'agit d'une représentation intermédiaire entre la représentation microscopique et la représentation macroscopique. Enfin, les modèles microscopiques étudient des systèmes composés d'éléments individuels que sont les voyageurs ou les

véhicules. A cet égard, la simulation multi-agent des déplacements peut être vue comme un cas particulier des modèles microscopiques. Le recours à des modélisations agrégées est encouragé entre autres par la nature agrégée des données disponibles aux concepteurs, mais aussi par les limites des outils de modélisation et d'exécution [Michael *et al.*2009].

Cependant, la dernière décennie a été marquée par l'évolution continue des outils informatiques qui permettent de répondre au grand besoin de fournir des simulations microscopiques des réseaux de transport. Les données de calibration des modèles microscopiques sont également de plus en plus disponibles et de meilleure qualité. Les modèles de simulation microscopique et multi-agent ont plusieurs avantages. Face à la complexité continue de systèmes de transport, la simulation des déplacements à une échelle microscopique est pertinente pour les étudier finement et les analyser, pour en proposer une meilleure gestion. Parmi les avancées les plus marquantes, la capacité des équipements multi-processeurs à élaborer des calculs parallèles ainsi que les architectures à mémoires partagées permettant un accès rapide à la mémoire physique pour un nombre important de processeurs [Michel2015]. Cette simulation microscopique des modèles de transport offre ainsi aux gestionnaires un large choix de résultats qui peuvent aller des simples statistiques générales jusqu'à des informations individuelles de chaque voyageur simulé. Aussi, elle permet une modélisation explicite des processus individuels de prise de décision assurant une dynamique réelle à la simulation à travers l'hétérogénéité des comportements des voyageurs simulés.

Il existe plusieurs simulateurs de mobilité des voyageurs de ce type dans la littérature. Par exemple, Transims [Nagel and M.Rickert2001] simule des voyages multimodaux et évalue les impacts des changements de politique de mobilité dans les caractéristiques du trafic tandis que Miro [Chipeaux *et al.*2011] reproduit les dynamiques urbaines d'une ville française et propose un prototype de simulation multi-agent capable de tester la planification de scénarios en précisant les comportements des individus. AgentPolis [Jakob *et al.*2012] est également une plateforme multi-agent pour le transport multimodal et MATSim [Maciejewski and Nagel2012] est une plateforme bien connue de micro-simulation de mobilité. Sumo [Behrisch *et al.*2011] et Vissim [Fellendorf and Vortisch2010] sont des simulateurs microscopiques largement utilisés et principalement axés sur la simulation de trafic. Il existe également des simulateurs tels que Archisim [Doniec *et al.*2008] qui décrivent de manière très précise le comportement des conducteurs à l'échelle microscopique. Notre travail étant centré sur la simulation multi-agent des déplacements des voyageurs sur un réseau de transport en commun, nous choisissons d'étudier dans la section suivante les architectures des modèles de simulation à base d'agents pertinentes pour ce type de simulation : AgentPolis, Transims et MATSim.

2.4.1 AgentPolis

AgentPolis est une récente plateforme à base d'agents pour la modélisation de systèmes de transport multimodal. Les entités individuelles de ces systèmes sont représentées comme des agents autonomes ayant des modules de contrôle asynchrone et continu, ils interagissent librement avec leur environnement et entre eux. L'approche à base d'agents permet de concevoir des scénarios où les agents ajustent leurs plans à tout moment durant la journée en fonction de leur perception de l'environnement et/ou des coordinations avec les autres agents. AgentPolis est composé de trois composantes centrales : le coeur de la simulation, la librairie du domaine de transport et l'outil de simulation.

1. Le coeur de la simulation : la partie centrale d'AgentPolis contient une plateforme de simulation à événements discrets basée sur l'outil multi-agent ALITE⁸. Cette plateforme consiste en un moteur de traitement assurant la construction des modèles de larges systèmes d'événements discrets à base d'agents [Jakob *et al.*2012].
2. La librairie du domaine de transport : cette librairie fournit un ensemble de composantes spécifiques qui assurent une création rapide de modèles de transport et de mobilité. Elle est constituée des classes et modules suivantes :
 - Composants du réseau de transport : classes qui permettent de construire l'environnement de transport (routes, réseau ferroviaire et métro, carrefours, arrêt, correspondance).
 - Véhicules : classes représentant les véhicules (voitures, bus, trains) et leurs propriétés.
 - Actions et capteurs de transport : classes assurant les liens et accès entre les agents et leur environnement de transport comme l'embarquement et la descente des véhicules, le déplacement sur une route ou la détection d'arrivée des trains.
 - Activités de transport et cycles de vie : structures réactives de contrôle qui peuvent être composées pour créer un comportement d'agent désiré (e.g. voyager via le transport en commun ou conduire une voiture entre deux points du réseau).
 - Planificateur d'itinéraires et de voyage : module qui fournit un agent des capacités de raisonnement, en particulier planifier son itinéraire dans un réseau de transport multimodal.
3. Outils de simulation : cet ensemble d'outils permet la conception et l'exécution des modèles sur AgentPolis.
 - Interfaces de données et filtres : ils permettent de travailler directement avec des données en formats standards tels que OpenStreetMap pour les données de cartes et GTFS pour le transport public (réseau et tableau de marche).
 - Outil de génération de population : cet outil permet de générer un grand nombre d'agents avec une distribution réaliste d'attributs démographiques (âge, genre, salaires, domicile, voiture, etc.) basé sur des données réelles de recensements.
 - Outil de configuration, gestion et déploiement d'expériences : cet outil assure la mise en place des scénarios d'expérimentations et leur exécution automatique sur les ressources disponibles.
 - Outil de visualisation et de rapport : cet outil permet de visualiser le déroulement de la simulation ainsi que les résultats. Cette visualisation inclut l'aspect temporel et géospatial et l'agrégation suite aux multiples simulations d'une manière interactive basée sur GoogleEarth.

Ces outils permettent ensemble une construction rapide d'un modèle basé sur des données réelles. Ils permettent de définir une variété de scénarios de simulation, d'exécuter ces scénarios avec la possibilité d'exécution parallèle, d'analyse des résultats et d'affichage d'une manière interactive [Jakob *et al.*2012].

Exemples d'application :

AgentPolis a été utilisé pour développer un modèle de simulation de système de contrôle de titres de transport dans un réseau de transport en commun. Ce modèle permet d'évaluer

8. <http://agents.fel.cvut.cz/projects/alite>

l'efficacité des stratégies de contrôle mises en place par des experts humains ou par des outils informatiques. Ce modèle contient deux types d'agents : un agent voyageur et un agent contrôleur. En plus du processus de déplacement, l'agent voyageur dispose d'une logique d'achat de ticket qui détermine si un ticket doit être acheté ou non pour un voyage particulier. L'agent contrôleur contrôle les voyageurs dans certains trains et sur certaines stations suivant une stratégie d'inspection précise. La performance de chaque stratégie peut être testée sur des voyageurs avec différents niveaux de rationalité et de perception de leur environnement. Plusieurs mesures de performance peuvent être collectées par ce modèle comme la somme des amendes collectées, les revenus perdus et les coûts des opérations de contrôle. Le modèle a rapidement été développé pour le système de métro de Los Angeles et a simulé environ 400 milles trajets sur cinq lignes par jour.

Cependant, AgentPolis ne fournit pour le moment pas de code source de la plateforme, ni d'exécutable. Il s'agit en fait d'un projet de recherche parallèle au nôtre. Par ailleurs, l'impact de l'information de voyageurs n'a pas été traité dans cette plateforme, et les auteurs ne montrent pas la possibilité d'intégrer des couches d'informations permettant le guidage de ces agents pendant une situation particulière.

2.4.2 Transims

Transims est parmi les outils de modélisation de réseaux de transports les plus avancés. Transims est basé sur un système de simulation multi-agent capable de simuler instantanément les déplacements des voyageurs et des véhicules sur un réseau de transport d'une large zone urbaine. Il est composé de cinq modules : le synthétiseur de population, le générateur d'activités, le planificateur d'itinéraires, le microsimulateur et le module de rétroaction. Le dernier module fournit des outils d'analyse des résultats de simulations. Il s'agit essentiellement du système d'affichage qui intègre les informations géospatiales (GIS) et des statistiques ainsi qu'un estimateur d'énergie consommée et d'émission de gaz lors des simulations.

1. Le synthétiseur de population : ce module est utilisé pour créer une population artificielle de la zone urbaine étudiée. Il combine des données agrégées en provenance des tableaux de recensement démographique et des données désagrégées des recensements des échantillons de microdonnées à usage public afin de créer une base de population synthétique où chaque individu est associé à un domicile. Les statistiques agrégées de cette population reproduisent celles de la vraie population. Les attributs de la population synthétique contiennent des informations individuelles tels que l'âge, le genre, le travail/étude, voiture, etc.
2. Le générateur d'activités : ce deuxième module identifie la liste des activités journalières de chaque individu synthétique dans chaque domicile. Le nombre des voyages que chaque individu programme pendant sa journée est déterminé en comptant le nombre de changement de lieux dans sa liste d'activités journalières. Ainsi, la liste d'activité définit la séquence quotidienne de voyages désirée par chaque voyageur de la population. Les données d'entrées de ce module sont une liste d'informations élaborée par le synthétiseur de population.
3. Le planificateur d'itinéraires : l'itinéraire à emprunter pour chaque voyage généré lors du module 2 est calculé par ce module. Cet itinéraire inclut les liens à emprunter, le mode

de transport, les changements de modes, l'emplacement des parkings et d'autres informations. Ce module utilise une version modifiée de l'algorithme Dijkstra pour calculer le plus court chemin. Ce module prend comme entrées les informations des voyageurs et leurs activités issues du module 2 et les informations du réseau de transport (le graphe représentant le réseau, les modes de transport de chaque arc, les temps de voyage de chaque arc, etc.). Pour chaque itinéraire calculé, le temps de départ, les arcs à emprunter avec le mode correspondant et le temps total de voyage sont affichés.

4. Le microsimulateur : le module microsimulateur exécute les plans de voyages individuels de chaque voyageur générés par le module 3. Les déplacements des voyageurs et leurs interactions avec le réseau de transport sont simulés à une échelle microscopique. La première étape de la simulation consiste à lire les données du réseau telles que les arcs, les noeuds, les voies et leurs connexions, les parkings, etc. La deuxième étape intègre les véhicules et la localisation et plans des voyageurs puis les positionne sur les réseaux. La troisième étape gère les déplacements des véhicules et des voyageurs selon leurs plans de voyage en utilisant l'approche des automates cellulaires. Cette approche consiste à diviser les liens du réseau en cellules, une cellule permet de contenir un seul véhicule. Le mouvement d'un véhicule est représenté par un saut d'une cellule à une autre. Différentes vitesses de véhicules sont représentés par différentes distances de saut (e.g. un saut de 5 cellules par pas de temps correspond à une vitesse de 135 km/h). Les véhicules se déplacent selon les règles suivantes :
 - Accélération linéaire jusqu'à la vitesse maximale s'il n'y a pas de véhicule devant.
 - Si véhicule devant, la vitesse sera ajustée en fonction de la distance séparant les deux véhicules.
 - Quelquefois la vitesse est aléatoirement plus lente que ce qui résulte des deux règles précédentes.

Le microsimulateur permet d'utiliser plusieurs processeurs afin de supporter un grand nombre de voyageurs et des réseaux de transport de grande taille.

5. Le système de visualisation : ce module permet à l'utilisateur de visualiser les données d'entrées et de sortie de la simulation avec des outils d'analyses. Ces outils incluent le traçage, les statistiques, la visualisation spatiale et les animations. Les plans de voyage, les véhicules, les signaux, les carrefours ainsi que plusieurs autres données peuvent être affichées. L'interface graphique du système de visualisation permet de manipuler l'affichage en 3D et de filtrer l'affichage pour une requête individuelle. Le système permet aussi d'animer la simulation par l'envoi de clichés instantanés pendant le déroulement de la simulation. Une version commercialisée de ce système est créée par IBM Business Consulting qui permet d'afficher les données de la simulation en 2D et 3D et d'animer l'exécution de la simulation selon les attributs sélectionnés par l'utilisateur.
6. L'estimateur d'émission : ce module fournit une conversion des déplacements de voyageurs en termes de consommation d'énergie et émission d'oxyde d'azote, hydrocarbures, monoxyde de carbone, dioxyde de carbone et de matières particulaires. Pour cela, le module exploite les données sur la composition de la flotte de véhicules issue du module synthétiseur de population, les résultats de tests d'inspection et de maintenance issus des bases de données locales et nationales de la zone étudiée et les schémas du trafic à partir du microsimulateur de Transims. L'inventaire des émissions élaboré par ce modèle a pour

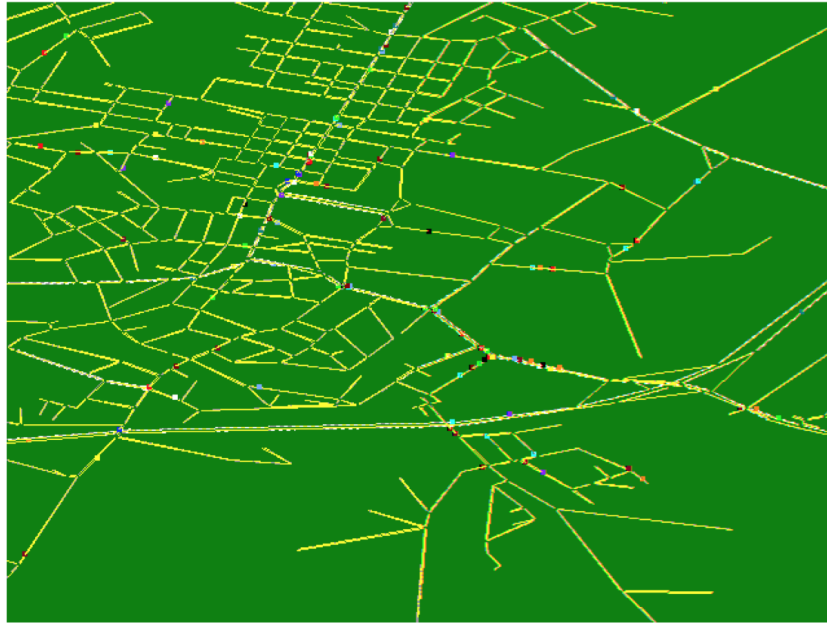


FIGURE 2.4 – Un cliché d’affichage d’une simulation Transims

but de calculer les concentrations des polluants dans une zone urbaine en se basant sur les conditions atmosphériques, le transport des dispersions locales et les réactions chimiques.

Transims est essentiellement utilisé pour étudier les déplacements sur des larges zones urbaines impliquant un grand nombre d’individus comme la microsimulation d’un réseau routier d’une région de la ville de Dallas (États-Unis) impliquant 200 000 voitures. Les résultats des simulations permettent aux autorités locales de calibrer leurs politiques de mobilité et offres de voyages. Cependant, Transims n’intègre pas de processus de prise de décision chez les voyageurs/conducteurs ainsi les réactions des usagers des réseaux de transport face à un événement externe ou à une information ne peuvent pas être modélisés dans Transims [Gu2004].

2.4.3 MATSim

MATSim est un système de modélisation de micro-simulation de réseaux de transport destiné à des applications à grande échelle. Il est développé par une équipe issue du domaine de transport et de l’ingénierie civile qui a exploité le paradigme multi-agent afin de répondre à leurs besoins. Dans MATSim, la demande est modélisée individuellement pour chaque agent. La modélisation concerne toute une journée et est appelée plan. La figure 2.5 montre un exemple de plan d’un agent écrit en xml qui décrit toute son activité journalière. Cette structure reste la même durant toute la modélisation et la simulation de la demande. lors de la phase d’affectation, tout le plan est exécuté et non pas chaque voyage séparément. Afin de fournir les plans de chaque agent, des données concernant l’infrastructure, la population et la demande de la région de la simulation sont créées par l’utilisateur. Ensuite, le système procède à la génération de la demande totale de voyage. Ce processus s’étire sur 4 parties :

- La création de scénario.

```
<plans name="example plans file">
...
<person id="393241" sex="f" age="27" license="yes" car_avail="always"
  employed="yes">
  <travelcard type="regional-abo" />
  <plan>
    <act type="home" link="58" start_time="00:00" dur="07:00" end_time="07:00" />
    <leg mode="car" dept_time="07:00" trav_time="00:25" arr_time="07:25">
      <route>1932 1933 1934 1947</route>
    </leg>
    <act type="work" link="844" start_time="07:25" dur="09:00" end_time="16:25"/>
    <leg mode="car" dept_time="16:25" trav_time="00:14" arr_time="16:39">
      <route>1934 1933</route>
    </leg>
    <act type="home" link="58" start_time="16:39" dur="07:21" end_time="24:00" />
  </plan>
</person>
...
</plans>
```

FIGURE 2.5 – Exemple d'un plan journalier d'un agent dans MATSim

- La modélisation des demandes individuelles initiales.
- L'optimisation itérative de la demande.
- L'analyse.

Le processus d'optimisation itérative de la demande est considéré comme la partie centrale de MATSim [Michael *et al.*2009]. Le processus s'exécute jusqu'à que chaque demande d'agent voyageur soit adéquate avec les restrictions du scénario (offre de voyage, réseau) et l'interaction avec les autres composantes de la simulation.

Généralement, une méthode de relaxation est utilisée afin de trouver un état d'équilibre. Pour le choix d'itinéraire l'équilibre de Wardrop [Wardrop1952] est défini comme état de repos. Cependant, l'optimisation concerne tout le plan journalier (routes, horaires, localisation, séquence d'activités et types d'activités).

Chaque agent essaie d'exécuter son plan avec l'utilité la plus élevée. L'utilité d'un plan journalier dépend de contraintes d'infrastructures (capacités des voies, horaires d'ouvertures des établissements, etc.) et des plans des autres agents dans le système. Cela implique que l'utilité effective d'un plan journalier ne peut être déterminée seulement par l'interaction de tous les agents. C'est ainsi que l'algorithme co-évolutif [Holland1992] joue son rôle. L'algorithme suit les étapes suivantes :

1. Initialiser les plans journaliers pour chaque agent du système.
2. Calculer l'utilité d'exécution des plans individuels de chaque agent.
3. Supprimer les mauvais plans (avec faible utilité).
4. Dupliquer et modifier les plans.
5. Faire de ces plans les plans pertinents pour l'itération suivante. Incrémenter le compteur.
6. Aller à 2.

Nous nous intéressons au processus de re-planification des plans journaliers des agents dans MATSim. Cette partie nous aide à comprendre comment l'activité d'un agent peut être modifiée suite aux changements d'état du réseau. D'après la description de cette phase dans [Michael *et al.*2009], nous constatons que les plans journaliers des agents sont modifiés par l'intermédiaire de modules centraux qui se chargent chacun d'une partie spécifique des plans et agissent indépendam-

ment pendant le processus d'optimisation. De ce fait, les agents n'ont pas de contrôle sur leurs choix et se montrent passifs dans leur processus de déplacement dans un réseau de transport. Parmi les modules qui contrôlent l'optimisation des plans le module « *Router* » qui calcule les chemins à emprunter en se basant sur les données de la simulation précédente. Aussi, le module *Time Allocation Mutator* qui modifie les temps de départ et les durées d'activités des plans journaliers. Les changements se font aléatoirement. Il existe aussi un module qui modifie tous les aspects d'un plan, il est nommé « *planomat* ». Ce module est capable de coordonner les plans d'agents appartenant au même foyer. Plusieurs autres modules peuvent être ajoutés et développés par l'utilisateur de MATSim. L'optimisation des plans des agents continue jusqu'à que le système atteigne un état d'équilibre, ce qui se traduit par une moyenne d'utilité constante. Dans le cas de la simulation du réseau national complet de la Suisse, avec 7 millions d'agents, un réseau de 24000 noeuds et 60000 et 22 millions de voyages générés dont 7,1 millions de voyage par transport routier, l'état de relaxation a été atteint après 70 itérations [Meister *et al.*2009].

Il est clair que MATSim permet de simuler des larges réseaux de transport multimodaux en intégrant un nombre très important d'agents voyageurs. Cependant, la centralisation des modules de modifications des plans de voyages ainsi que l'absence d'autonomie chez les agents limitent la flexibilité des modèles de simulation de MATSim à intégrer des couches d'informations voyageurs et de nouveaux types d'agents.

Après cette revue des simulateurs multi-agents des déplacements, nous concluons qu'aucune des propositions, très intéressantes par ailleurs, ne répond à notre problématique : représenter les flux d'information de voyageurs et les intégrer dans les comportements des voyageurs. Il nous faut donc concevoir et développer notre propre outil pour atteindre cet objectif. Néanmoins, nous ne sommes pas totalement démunis face à ce problème et nous ne sommes pas obligé de créer *ex nihilo* une nouvelle plateforme. En effet, il existe des plateformes multi-agents généralistes, *open source*, qui permettent de faciliter grandement le travail de conception et de développement. Ces plateformes font l'objet de la section suivante.

2.5 Plateformes de simulation multi-Agent

Pour concevoir et implémenter un simulateur multi-agent, il est possible de développer une application directement dans un langage de programmation hôte. Cependant, il est souvent plus rapide, plus utile et plus efficace de fonder le simulateur sur une plateforme multi-agent existante. Un critère central pour choisir la plateforme de simulation concerne sa capacité à créer des modèles d'agents avec des résultats reproductibles. Dans les perspectives que nous donnons à notre travail, nous désirons également pouvoir déployer les simulations sur plusieurs serveurs. Enfin, la troisième capacité que nous recherchons pour la plateforme de simulation est sa capacité à concevoir des modèles géo-spaciaux, i.e. sa capacité d'intégrer et de traiter des données géographiques.

Plusieurs autres critères interviennent dans le processus de choix de plateforme. Le premier groupe de critères concerne les facilités de développement offertes par la plateforme comme la taille de la communauté qui l'utilise, la disponibilité de supports d'aide (généralement animés par la communauté), la disponibilité de modèles de démonstration et de documentation technique. Le deuxième groupe concerne des critères en lien avec les fonctionnalités de la modélisation. Il s'agit du nombre d'agents que la plateforme permet de modéliser, les degrés

d'interaction entre les agents, la capacité de représenter différents niveaux d'organisation des agents, les différents modèles d'environnement possibles, la gestion des liens topologiques et spatiales entre agents, les mécanismes de planification, etc. Le critère *open source* est aussi important car il permet aux développeurs d'explorer le code permettant si besoin la modification et l'extension du système [Crooks and Castle2012].

Selon ces critères, nous avons choisis de nous focaliser sur cinq plateformes Swarm [Minar *et al.*1996], Mason [Luke *et al.*2004], Repast [Tatara and Ozik2009], NetLogo [Wilensky and Evanston1999] et Gama [Patrick *et al.*2012]. Elles remplissent la majorité des critères cités. Elles sont quotidiennement mises à jour, largement utilisées et supportées par une large communauté d'utilisateurs, accompagnées par une variété de modèles de démonstration, *open source* pour certaines et essentiellement capables de développer des modèles géospatiaux à travers l'intégration de fonctionnalités GIS. Ensuite, nous choisissons celle qui s'avère la plus compatible pour modéliser et simuler un système de transport multimodal en tenant en compte sa complexité et son aspect distribué et fortement dynamique.

2.5.1 Swarm

Swarm est une plateforme de simulation et de conception open source qui est conçue spécialement pour le développement de simulations multi-agents de systèmes complexes. Swarm permet aussi de développer des modèles à base d'agents [Patrick *et al.*2012] en proposant un ensemble de concepts standards pour la description et la conception de ces modèles. La plateforme fut développée en Objective-C d'abord, ensuite un ensemble de libraires ont été ajoutées pour permettre la réalisation de simulations de modèles d'agents en langage Java. Swarm intègre des fonctionnalités GIS à travers sa librairie Kerg qui permet le chargement de couches de données géographiques. Cependant, elle ne fournit pas de primitives spatiales, ni ne donne la possibilité de sauvegarder l'environnement géographique résultat [Patrick *et al.*2012].

Au début, Swarm a été utilisée pour essayer d'étudier des phénomènes biologiques [Minar *et al.*1996]. Ensuite, son utilisation a touché plusieurs autres domaines : l'anthropologie, l'écologie, l'économie, la géographie, les sciences politiques. Par exemple, [Railsback and Harvey2002], [Schelhorn *et al.*1999], [Haklay *et al.*2001] et [Batty *et al.*2003] représentent des modèles explicitement spatiaux développés sur Swarm. Les deux premiers constituent des simulations de piétons dans des centres urbains, et le troisième étudie la congestion de foule pendant le carnaval Notting Hill à Londres [Crooks and Castle2012].

2.5.2 Mason

Mason (*Multi Agent Simulation Of Neighbourhood*) est développé par le laboratoire ECLab et le centre de la complexité sociale à l'université George Mason [Luke *et al.*2004]. Mason dispose d'une grande base de documentation technique ainsi qu'une communauté active. Plusieurs publications détaillant son implémentation et son application et des modèles de démonstration sont disponibles afin d'évaluer ou de se familiariser avec cet outil. Mason est basé sur Java et offre plusieurs fonctionnalités comme le traçage dynamique (histogrammes, graphes, etc.) et la modélisation de l'affichage des simulations. Au début, Mason n'intégrait pas de données géospatiales jusqu'à l'addition de l'extension GeoMason. Cette extension permet l'importation et

l'exportation, la manipulation et l'affichage de données géographiques. Cependant, ces capacités restent limitées [Coletti2013]. Les fonctionnalités GIS de Mason sont utilisées dans plusieurs modèles d'agents géospaciaux afin d'étudier : le conflit entre les bergers et les fermiers dans l'est africain [Kennedy *et al.*2010], les bergers nomades en Inde Centrale [Cioffi-Revilla *et al.*2010], le dynamisme résidentiel en Arlington County - Virginia [Hailegiorgis2010] et l'industrie de drogue en Afghanistan [Latek *et al.*2010].

2.5.3 NetLogo

NetLogo [Wilensky and Evanston1999] est une version de StarLogo développée à l'université Northwestern initialement pour permettre aux modèles StarLogo d'être développés sur des systèmes d'exploitation Macintosh et ensuite pour le déploiement des modèles sur internet [Crooks and Castle2012]. Au début NetLogo et StarLogo fournissaient uniquement des fonctionnalités pour importer des fichiers images pour définir l'environnement dans lequel les agents sont localisés ce qui facilite le développement de modèles spatiaux. Aujourd'hui, NetLogo permet d'importer des données matricielles et vectorielles (*shapefiles*). Cette fonctionnalité multiplie les possibilités de création de modèles d'agents géospaciaux. Cependant, NetLogo ne fournit pas des opérations d'analyses géographiques avancées.

Deux modèles de démonstration accompagnent l'installation de NetLogo. Le premier est destiné aux données vectorielles et comporte trois ensembles de données : un fichier Points des villes du monde, un fichier Polygones représentant les rivières, un fichier polygone représentant les pays. Ces données sont importées dans un modèle NetLogo et converties en patches. Pour les données matricielles, un fichier Raster de l'élévation de la surface est chargé dans un modèle NetLogo pour montrer les possibilités de travail avec des données spatiales. Dans cet exemple, les agents suivent la surface à moindre élévation. Cet exemple peut être utilisé pour modéliser un processus qui s'articule sur l'évaluation de surfaces comme l'évacuation de bâtiments [Crooks *et al.*2008].

NetLogo a été utilisé pour développer des applications dans plusieurs domaines comme la biologie, la physique et les sciences sociales. Aussi, plusieurs modèles spatiaux ont été créés à l'aide de NetLogo comme l'étude de gentrification dans [Torrens and Nara2007], l'étude autour de la demande résidentielle dans [Fontaine and Rounsevell2009], l'émergence des modèles d'occupation des territoires [Graham and Steiner2006] et la réimplémentation artificielle du peuple amérindien Anasazi [Janssen2009].

Le site web de NetLogo offre une large base de documentation et de tutoriels, des modèles de démonstration ainsi que des API qui contiennent des fonctionnalités qui peuvent être extraites. A l'exception des plateformes déjà présentées, NetLogo utilise son propre langage de programmation et son code source n'est pas accessible.

2.5.4 Gama

Développée depuis 2007 [Taillandier2013], Gama est une plateforme de modélisation et de simulation à base d'agents. La plateforme est libre et permet à ses utilisateurs de modéliser des simulations à temps discret. Gama dispose de son propre langage de modélisation : le GAML, en plus d'un environnement de développement intégré. Ce langage est caractérisé

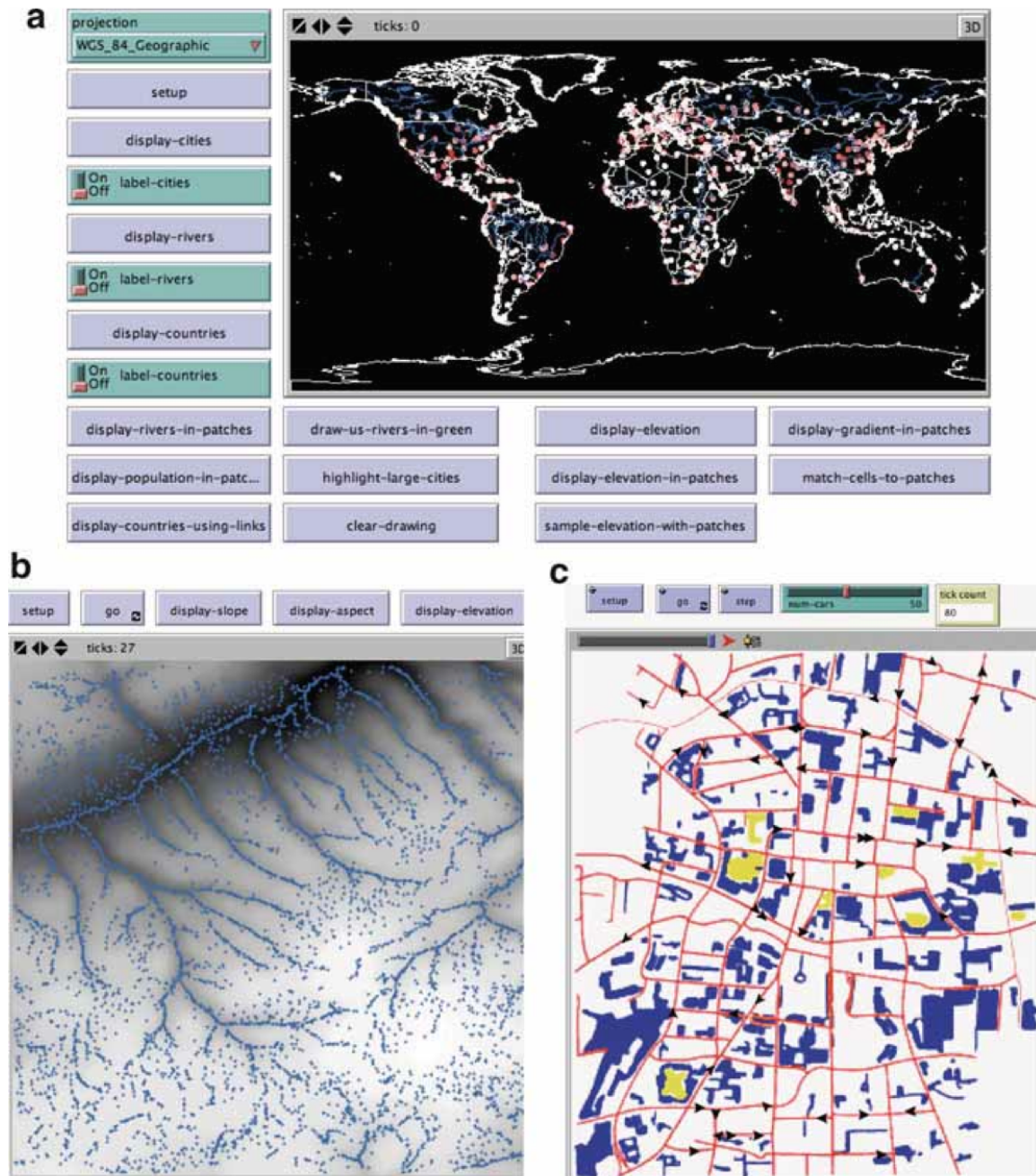


FIGURE 2.6 – Exemples d'intégration de GIS dans NetLogo (a) utilisation de shapefiles (b) gradient (c) le modèle de croisière

par son utilisation assez simple. Plusieurs projets de recherche dans des domaines différents ont utilisé Gama afin de répondre à des problématiques variées telles que les risques naturels, l'épidémiologie, l'évolution de l'occupation des sols, la gestion de l'eau avec le projet MAELIA [Gaudou *et al.*2013], la mobilité urbaine avec le projet MIRO 2 [Banos A.2013], ou encore la reconstitution d'événements géo-historiques [Taillandier *et al.*2014]. Plusieurs tutoriels, une documentation complète ainsi qu'une vidéo de démonstration complète sont disponibles sur le site web de la plateforme.

Plusieurs environnements peuvent être définis dans Gama avec un environnement continu de référence assurant la synchronisation entre eux. Gama permet de définir trois types de topologies : l'espace continu, la grille et le graphe. Une topologie est associée à chaque environnement afin de fournir aux agents des services tels que le calcul de voisinage et des distances. Chaque agent dans Gama dispose d'une variable de localisation de type point qui représente le centroïde de sa géométrie. Ainsi, modifier la géométrie d'un agent modifie automatiquement sa localisation. Gama offre plusieurs opérateurs pour effectuer des opérations sur les géométries. Gama est caractérisé par ses nombreux outils permettant l'intégration et la manipulation des données géographiques. Le langage GAML est riche d'instructions facilitant ce type de manipulations. Par exemple, par une simple instruction les agents sont instanciés à partir d'un *shapefile* contenant des données géographiques (les géométries des agents). Ou encore, les géométries des agents et autres variables peuvent être directement sauvegardées dans un *shapefile*. De nombreux opérateurs spatiaux sont aussi fournis tels que les tests géométriques (intersection, croisement, inclusion), les requêtes (calcul de voisinage), les transformations, la triangulation de polygones, la squelettisation et la création de graphes à partir de géométries. Ces opérations sont réalisées en utilisant la librairie JTS. Aussi, les bases de données géographiques telle que PostGis sont interrogeables via GAML [Taillandier *et al.*2014].

Concernant la visualisation des données dans Gama, la 3D est nativement intégrée. L'affichage est géré par la librairie OpenGL ce qui permet de bénéficier des avancées offertes par les cartes graphiques (GPU) en terme de rendu réaliste. Un utilisateur peut facilement centrer son affichage sur un agent particulier et le suivre au cours de la simulation tout en définissant différents points de vue. L'utilisation de calques facilite l'abstraction à travers la superposition des indicateurs. Concernant les données en 2D, Gama permet de rajouter des hauteurs aux géométries rendant ces derniers en 3D d'une manière simple. La figure 2.7 illustre un exemple de cette transformation selon deux visions. Gama permet aussi de représenter un modèles numériques de terrain (MNT) dans un environnement 3D et d'exploiter les données d'altitude et de les intégrer dans le comportement d'un agent. La figure 2.7 montre le rendu visuel obtenu par l'ajout d'un MNT à l'exemple précédent.

2.5.5 Repast

Repast (*Recursive Porous Agent Simulation Toolkit*) est développé à l'université de Chicago et actuellement maintenu par le laboratoire Argonne National. Repast est inspiré de la plateforme Swarm et a été initialement implementé sous trois langages : Python (RepastPy), Java (RepastJ) et Microsoft .Net (Repast .Net). RepastPy permet le développement de modèles basiques par des utilisateurs ayant un faible niveau de programmation à travers une interface graphique. Les modèles RepastPy peuvent être exportés/convertis en Java pour des développements avancés sur RepastJ. En effet, RepastJ et Repast Net permettent de développer des

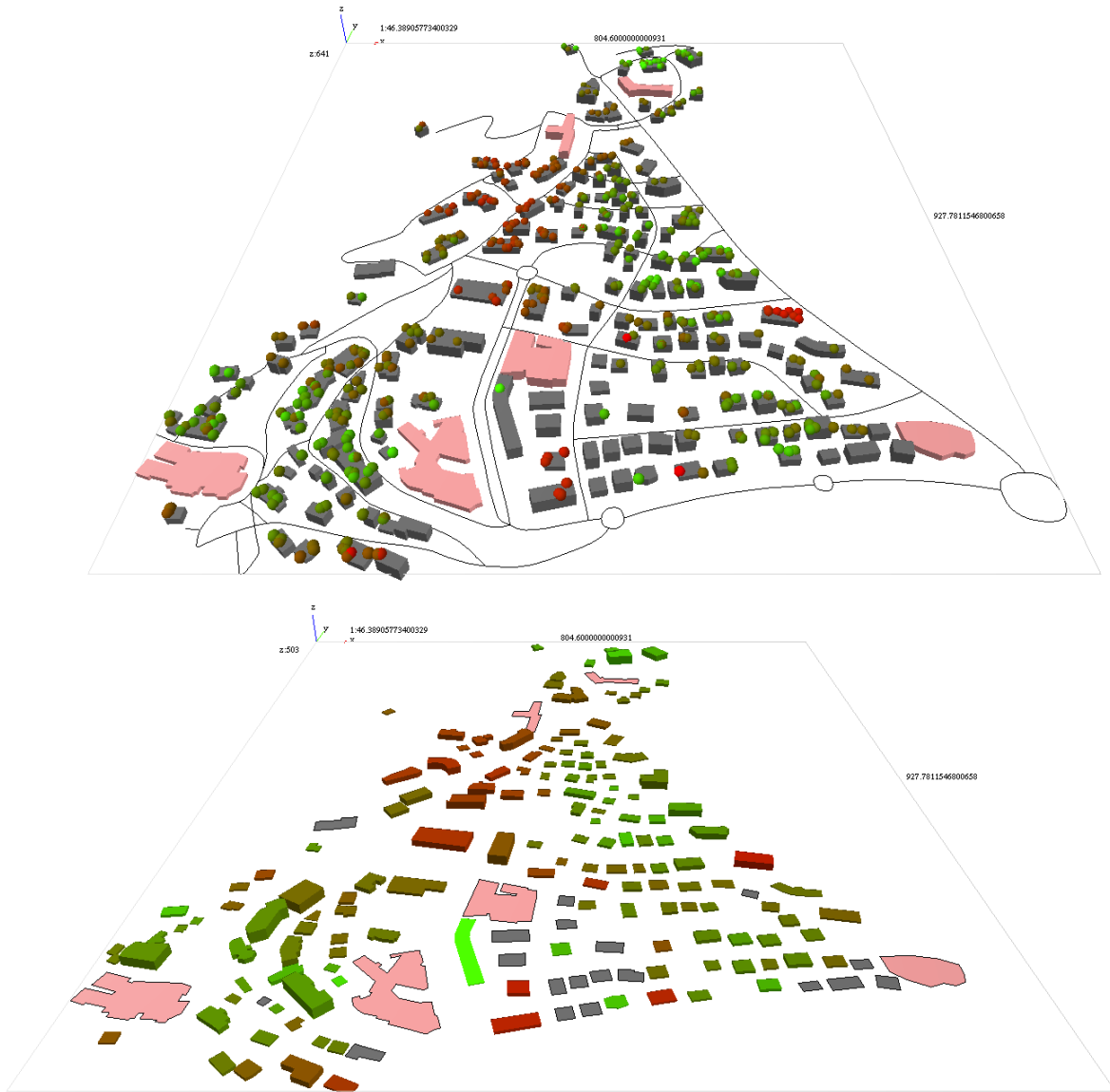


FIGURE 2.7 – Fenêtres de visualisation d’une simulation Gama

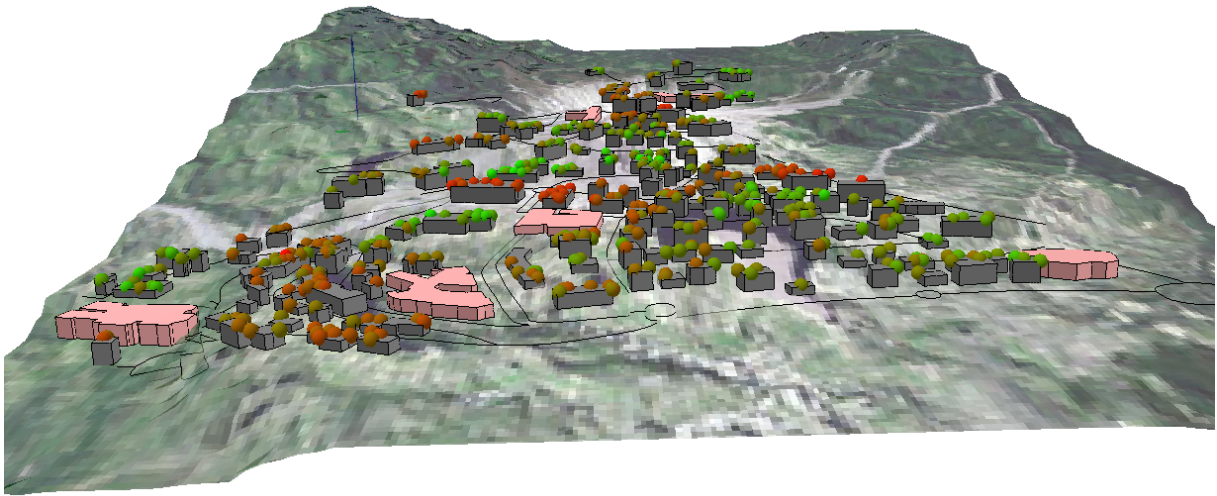


FIGURE 2.8 – Exemple de résultat obtenu après l’ajout d’un MNT

modèles avancées vu qu’ils offrent des fonctionnalités plus complexes que RepsatPy. Ces trois versions ont rapidement laissé la place en 2006 à Repast Symphony (RepastS) qui fournit toutes les fonctionnalités de RepastJ et Repast Net et est implémentée en Java. Cette nouvelle version assure les mêmes fonctionnalisés que les autres et rajoute un environnement d’interfaces graphiques pour des modèles de développement. Aussi, une amélioration de l’interface graphique d’exécution qui permet de créer des affichages en 2D et 3D, des graphiques, interroger les agents et des interfaces avec d’autres programmes (e.g. le logiciel de statistique R). Repast Symphony intègre une librairie d’information géographique (Geotools), et fournit des services additionnels, tels que la modélisation du réseau comme un graphe, le calcul de plus courts chemins, la visualisation et la gestion de données à deux et trois dimensions [Tatara and Ozik2009]. L’extension Agent Analyst permet d’éditer des modèles d’agents Repast à travers ArcGIS⁹.

Des méthodes sont aussi fournies pour visualiser les modèles d’agents avec des images satellites, des surfaces élevées et d’autres données scientifiques à travers la bibliothèque interactive du système d’informations géographiques 3D NASA World Wind¹⁰. La figure 2.9(b) est un exemple de cette visualisation [Crooks and Castle2012]. [Malleon2008] propose un tutoriel qui décrit l’intégration d’autres fonctionnalités GIS à travers la démonstration de la création d’une ville virtuelle via l’importation de *shapefiles* puis le déplacement d’agents sur le réseau de routes (figure 2.9(a)). RepastS permet l’importation de modèles d’agents NetLogo sur sa plateforme via ReLogo. Cette fonctionnalité vise à faciliter le prototypage de modèles d’agents en créant des simples modèles sur NetLogo pour ensuite les étendre sur RepastS [Crooks and Castle2012]. Aussi, RepastS aide au déploiement de simulateurs sur plusieurs hôtes via Repast HPC [Collier2010]. Plusieurs modèles géospatiaux sont développés avec Repast. Nous citons les modèle d’évacuation de piétons d’une station sous-terreine [Castle2007], d’étude de la dynamique résidentielle [Jackson *et al.*2008], d’étude de la criminalité

9. Agent Analyst : Agent Based Modeling Extension for ArcGIS. Available at <http://resources.arcgis.com/en/help/agent-analyst/>

10. <http://goworldwind.org/demos/>

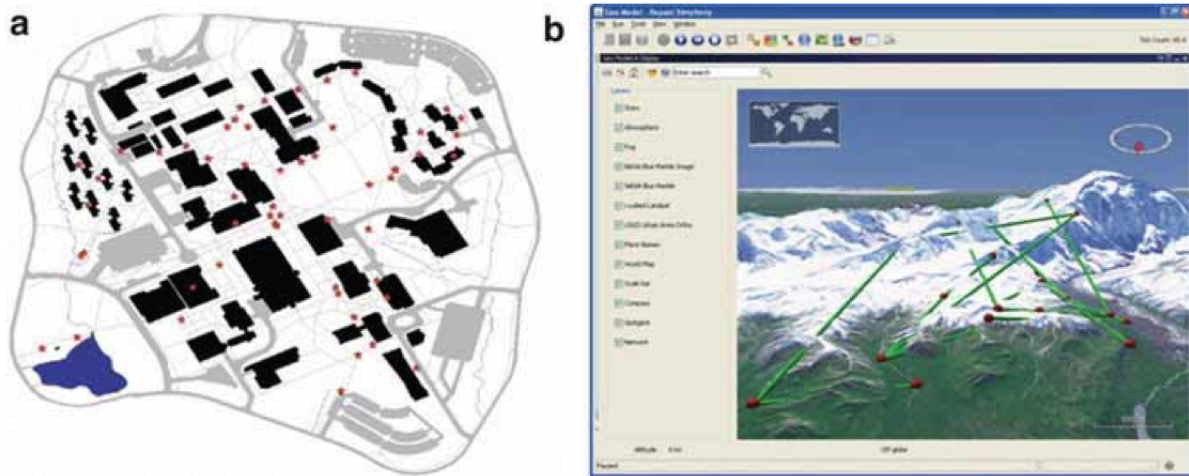


FIGURE 2.9 – Exemples d'un modèle vectoriel à base d'agent sur RepastS. (a) Des agents se déplaçant sur un réseau (b) Un modèle superposé sur *Nasa World Wind*

[Malleon *et al.*2010] et d'étude de la ségrégation [Crooks2010]. L'équipe de développement de Repast a fourni une série d'articles à propos de RepastS. L'architecture et les fonctionnalités de base sont introduites par [North *et al.*2005], et l'environnement de développement est discuté dans [Howe *et al.*2006]. L'enregistrement, l'affichage et le comportement et interaction des agents ainsi que des fonctions pour l'analyse de données (exemple : l'intégration du package de statistique R) sont exposés dans [Macal and North2005]. [Tatara *et al.*2006] fournit une discussion qui explique les capacités de RepastS à travers le développement d'un modèle simple de prédateur. Aussi, Repast dispose d'une large communauté active et une importante base de documentation et de modèles de démonstration disponibles sur le site Web.

2.5.6 Discussion

Toutes les plateformes multi-agents présentées dans cette section pourraient être utilisées dans la modélisation et la simulation de déplacements, dans l'objectif d'évaluer l'impact des informations sur les voyageurs. Cependant, quelques critiques ont guidé notre choix final vers la plateforme Repast Symphony. En effet, Swarm, Mason et Netlogo ne fournissent pas de bibliothèque géospatiale complète, incluant des primitives spatiales. Swarm ni ne donne pas la possibilité de sauvegarder l'environnement géographique résultat [Patrick *et al.*2012].

Les deux plateformes Gama et Repast Symphony sont très proches en termes de fonctionnalités. Notre choix a été guidé par deux considérations. La première concerne la taille de la communauté autour de la plateforme et la réactivité des personnes en charge de la plateforme aux questions des développeurs. La deuxième concerne la facilité de déploiement physiquement distribué de la simulation, qui est l'un de nos axes de recherche en cours. Ces ultimes critères ont fait pencher la balance vers Repast Symphony. C'est pour cette raison que nous fondons notre simulateur multi-agent des déplacements sur cette plateforme (cf. chapitre 3).

2.6 Conclusion

Dans ce chapitre, nous avons présenté les concepts de base du paradigme multi-agent que nous avons choisis comme approche pour concevoir notre simulateur. Cette approche est particulièrement pertinente pour la simulation de mobilité de voyageurs sur un réseau de transport. Le concept d'agent est bien adapté pour la représentation de voyageurs dans les réseaux de transport. Ce sont des entités autonomes, situées dans un environnement, qui adaptent leurs comportements à la dynamique de l'environnement qu'elles perçoivent et interagissent avec les autres agents pour réaliser des buts spécifiques. Ensuite, nous avons mené une étude comparative des plateformes existantes afin de choisir celle qui correspond à nos besoins. Nous n'avons pas trouvé de plateforme « clé en main » permettant de simuler notre système. Nous sommes donc passé à l'étude des plateformes généralistes de simulation multi-agents. Dans notre contexte, deux critères centraux pour choisir la plateforme de simulation concernent d'abord sa capacité à créer des modèles d'agents géospaciaux, i.e. sa capacité d'intégrer et de traiter des données géographiques. Puis, sa capacité à déployer la simulation sur plusieurs hôtes pour pouvoir intégrer un nombre important de voyageurs simulés interagissant dans un environnement distribué et complexe. À l'issue de cette étude, nous choisissons la plateforme Repast Symphony.

Deuxième partie

Contributions

Chapitre 3

Un simulateur multi-agent des déplacements sur les réseaux de transport en commun

Sommaire

3.1	Introduction	58
3.2	La plateforme de voyage multimodal	58
3.2.1	Scénario et hypothèses	59
3.2.2	Interface publique	59
3.2.3	Modèle	61
3.3	La plateforme de simulation Repast Symphony	62
3.3.1	Les contextes	63
3.3.2	Les projections	64
3.3.3	L'ordonnancement	64
3.4	Le simulateur	66
3.4.1	Données et paramètres	66
3.4.2	Ordonnanceur parallèle	67
3.4.3	Données et paramètres du simulateur	69
3.5	Déplacement dans la géographie	76
3.5.1	Contextes et projection	76
3.5.2	Planification des itinéraires	76
3.5.3	Le déplacement	77
3.6	Le système multi-agent	78
3.6.1	Les agents	78
3.6.2	Packages	80
3.7	Optimisations	80
3.7.1	Des données erronées ou manquantes	80
3.7.2	Problème de performance	82
3.8	Exécution	83
3.9	Conclusion	83

3.1 Introduction

Comme nous l'avons vu dans le chapitre 1, le Web représente aujourd'hui l'un des principaux supports d'informations destinées aux voyageurs. Avec l'évolution de la version mobile du Web, les nouveaux SIV permettent d'intégrer une nouvelle forme d'information personnalisée, rendant possible le partage d'informations adaptées aux besoins dynamiques individuels des usagers. C'est dans ce contexte que le projet Européen FP7 *Instant Mobility*¹¹ a vu le jour. Ce projet suppose la localisation permanente des voyageurs et une interaction permanente avec eux. Dans ce cadre, une plateforme de voyage multimodal (MMT pour *multimodal travel platform* [Zargayouna *et al.*2012]) a été conçue et implémentée. Cette plateforme guide les voyageurs suivant les meilleurs itinéraires possibles jusqu'à leurs destinations, quelque soit le mode de transport qu'ils empruntent. Afin de fonctionner, la plateforme doit-être alimentée avec des demandes continues de déplacement. Ceci est le contexte originel du développement des premières versions du simulateur de déplacements présenté dans ce chapitre. L'objectif étant de générer des voyageurs et des moyens de transport et de les faire interagir avec la plateforme en lui fournissant les demandes et la localisation continues de ces entités.

Cependant, bien que le simulateur a été développé originellement pour interagir avec la plateforme, nous en faisons un outil indépendant qui fonctionne en autonomie sur n'importe quelle zone géographique, du moment que nous disposons des données nécessaires à son exécution. Désormais, le simulateur est une base pour le prototypage rapide et l'exécution de simulations pour divers types d'applications en ligne. Ce chapitre présente rapidement les principales briques de la plateforme avant de présenter le simulateur. Le simulateur est fondé sur la plateforme Repast Symphony, suivant notre étude du chapitre 2.

Le simulateur est conçu comme un système multi-agent. Nous nous fondons donc sur le paradigme multi-agent présenté dans le chapitre 2, qui est comme nous l'avons vu un outil puissant pour la conception et l'implantation d'applications de transport.

Ce chapitre est structuré ainsi. La section 3.2 présente la plateforme MMT et décrit son fonctionnement ainsi que les quatre modules qui la composent. Dans la section 3.3, nous décrivons les briques de base que nous avons utilisées de la plateforme Repast Symphony. Dans la section 3.4, nous détaillons la conception du simulateur : ses paramètres et données, la représentation du réseau, la planification d'itinéraires et la méthode de déplacement. La section 3.6 décrit les agents du système et leurs comportements. Dans la section 3.7 nous détaillons l'ensemble des optimisations que nous avons effectuées pour améliorer les performances du système. Nous concluons ce chapitre avec la section 3.9.

3.2 La plateforme de voyage multimodal

Dans cette section, nous présentons une description générale de la plateforme MMT pour mieux comprendre le scénario considéré et le développement du simulateur. Les détails techniques et les spécifications complètes de la plateforme sont disponibles sur le site du projet *Instant Mobility* (et ici [Zargayouna2012]).

11. Ce travail est le résultat d'un travail auquel ont participé : Gérard Scemama, Mustapha Tendjaoui, Mahdi Zargayouna, Besma Zedini et moi-même.

3.2.1 Scénario et hypothèses

Dans le scénario considéré dans le projet, les services en ligne fournissent aux voyageurs un large éventail d'options personnalisées de voyages selon leurs préférences. Un voyage peut utiliser différents modes tels que les transports en commun (TC), les voitures et le co-voiturage pour une même déplacement. Pendant la planification d'un voyage, le voyageur reçoit une liste d'itinéraires alternatives. Cette liste représente les plans optimaux qui respectent ses préférences et qui intègrent les dernières informations temps-réel des différents modes de transport. Une fois que l'application d'assistance de voyage reçoit l'itinéraire choisi par le voyageur, elle réserve le service correspondant et lui fournit un reçu en échange d'un paiement à partir de son compte voyageur. Ensuite, le voyageur reçoit son itinéraire choisi via Internet et sera guidé pendant son déplacement afin de trouver les bons arrêts, avancer pendant les correspondances ou dans les terminaux complexes, etc. Pendant le voyage, l'itinéraire est continuellement contrôlé en temps-réel et le voyageur est alerté à chaque changement. L'itinéraire est ainsi mis-à-jour en fonction du choix du voyageur. Le paiement est effectué à la fin du voyage, ainsi les problèmes survenus pendant ce voyage sont pris en compte et le voyageur ne sera pas amené à demander un remboursement. Aussi, il paiera seulement le service demandé.

Deux hypothèses sont énoncées pour la plateforme MMT. Premièrement, nous supposons que l'équipement Internet-mobile de chaque voyageur connaît sa localisation et chaque véhicule (public et privé) est continuellement positionné. Deuxièmement, la plateforme est capable de gérer simultanément les demandes de réservations de voyage ainsi que le contrôle en temps-réel des déplacements en cours, qui reste le premier objectif de la plateforme, en tenant en compte des changements qui surviennent sur les réseaux (accidents, perturbations, etc).

Le diagramme d'activité dans la figure 3.1 décrit le scénario. Le défi principal avec ce scénario et ces deux hypothèses est double :

- D'un point de vue technologique, trouver la meilleure architecture et les fonctionnalités du système pour profiter de tous les avantages des technologies de l'Internet du futur tout en respectant les préférences des voyageurs et les besoins de confidentialité des données.
- D'un point de vue algorithmique, trouver les bonnes procédures pour assurer l'équilibre du système entre les demandes des voyageurs et l'offre de voyage.

3.2.2 Interface publique

La figure 3.2 présente la plateforme de voyage multimodal MMT d'un point de vue extérieur. Elle spécifie les besoins pour installer le service *Instant Mobility* dans une région ainsi que les différents échanges nécessaires avec la plateforme. La plateforme interagit avec trois types d'acteurs : les opérateurs de transport public, les opérateurs du transport routier et les voyageurs.

Chaque opérateur de transport public doit fournir à la plateforme une description de ses réseaux et les tableaux de marches théoriques. Comme indique la figure, la plateforme n'exige pas aux opérateurs de transport public d'avoir une base de donnée commune qui intègre tout les données des réseaux de transport en commun. A travers la plateforme ClaireSiti [Scemama and Carles2004], MMT interagit simultanément avec les différentes bases de données des opérateurs. Ensuite, l'opérateur doit fournir en temps réel les trois types d'informations suivantes :

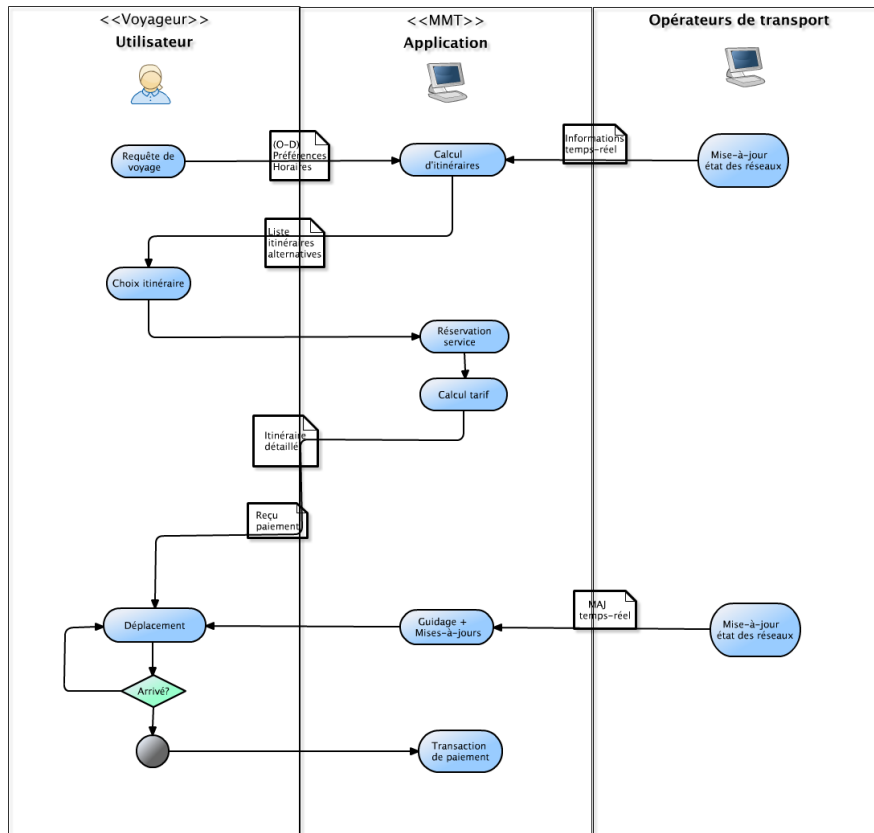


FIGURE 3.1 – Scénario MMT

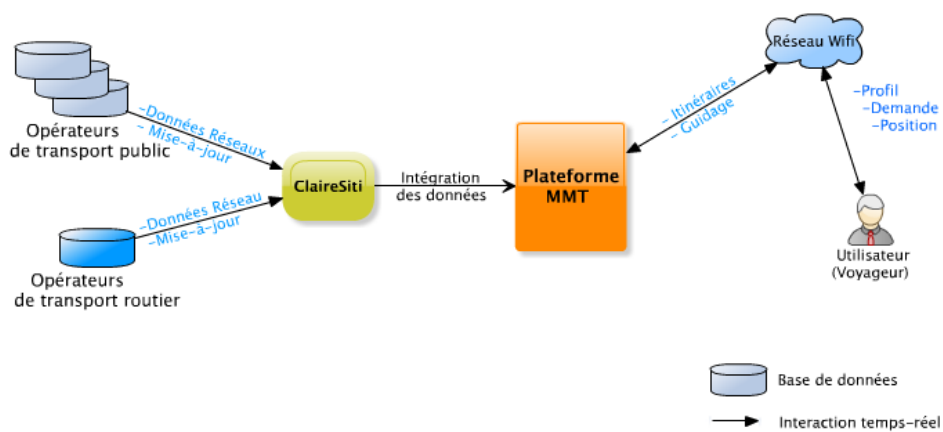


FIGURE 3.2 – Vue extérieure de la plateforme multimodale de voyage MMT

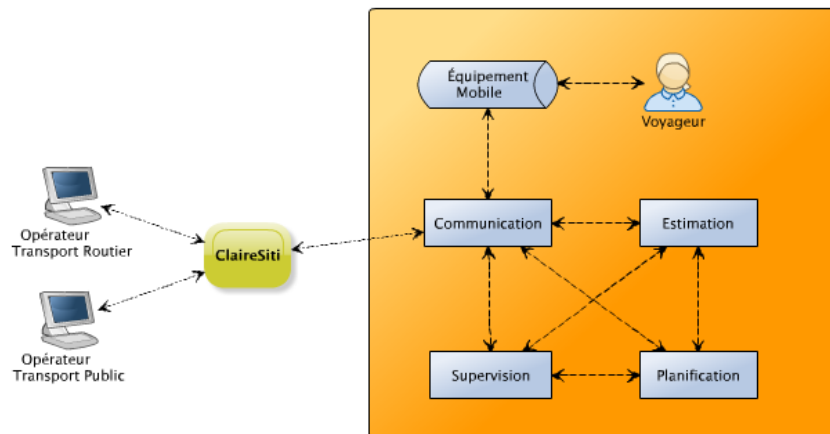


FIGURE 3.3 – Le modèle de la plateforme MMT

- La description de l’offre dynamique de transport.
- Les positions et les avancements/retards de toute sa flotte de véhicules.
- Les événements qui provoquent des perturbations du service.

Toutes les données échangées doivent être conformes aux dernières normes européennes. Les voyageurs interagissent avec la plateforme en utilisant les protocoles de communication standards. Chaque voyageur, fournit son profil au système qui inclut des informations détaillées à propos de ses propriétés et ses préférences. Une fois un plan est reçu par le voyageur, son équipement mobile envoie dynamiquement sa position actuelle à la plateforme. Si la plateforme détecte une grande différence entre la position actuelle et la position planifiée (i.e. la position planifiée dans le plan proposé au voyageur), le voyageur reçoit un nouveau plan selon sa nouvelle position en considérant ses préférences.

3.2.3 Modèle

D’un point de vue fonctionnel, la plateforme MMT est composée de quatre modules conçus en tant que composantes indépendantes et faiblement couplées (cf. Figure 3.3) :

- Le module de communication est responsable des interactions entre la plateforme MMT et les acteurs externes (les opérateurs de transport public, les opérateurs du transport routier et les voyageurs). (cf. Figure 3.4) ;
- Le module de planification responsable de planifier et re-planifier les itinéraires des voyageurs (cf. Figure 3.5) ;
- Le module d’estimation maintient la meilleure vision possible de l’état futur des réseaux. Cette vision est la base des calculs du module de planification ;
- Finalement, le module de supervision supervise l’exécution des plans des voyageurs.

Les équipements mobiles des voyageurs peuvent sauvegarder des informations qui peuvent accélérer les calculs d’itinéraires et/ou améliorer la qualité des solutions qui leur sont présentées. Par exemple, ils peuvent sauvegarder des informations sur les habitudes des voyageurs.

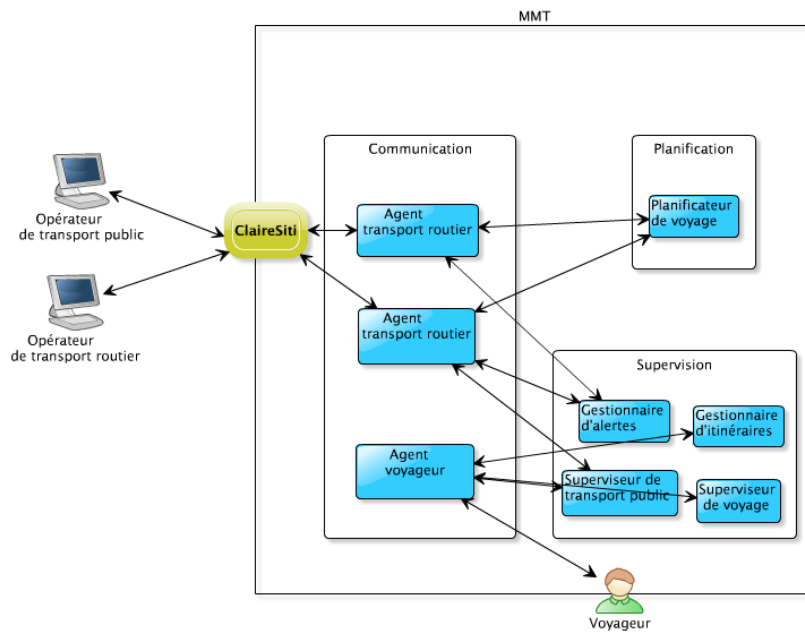


FIGURE 3.4 – Le module de communication

Cela peut économiser un temps important en émettant immédiatement des suggestions et des recommandations aux voyageurs.

Afin que la plateforme MMT fonctionne correctement, elle doit interagir avec les acteurs externes à travers son module de communication décrit plus haut. Vu la difficulté de tester ce scénario sous des conditions réelles avec notamment un nombre significatif de voyageurs multimodaux équipés de *smartphones* avec l'application *Instant Mobility*, nous développons un simulateur de déplacement afin de remplacer l'environnement de la plateforme. Notre simulateur doit générer des voyageurs et des moyens de transport afin de fournir à la plateforme toutes les données nécessaires (statiques et dynamiques) concernant l'offre et les demandes de voyage et la localisation continue de ces entités.

3.3 La plateforme de simulation Repast Symphony

Nous avons choisi la plateforme de la simulation Repast pour les raisons invoquées dans le chapitre 2. Dans ce qui suit, nous en donnons les éléments de base nécessaires à la compréhension des développements que nous avons menés dans le cadre de ce travail. Trois notions importantes dans Repast Symphony sont : les contextes, les projections et comme tout simulateur, l'ordonnancement.

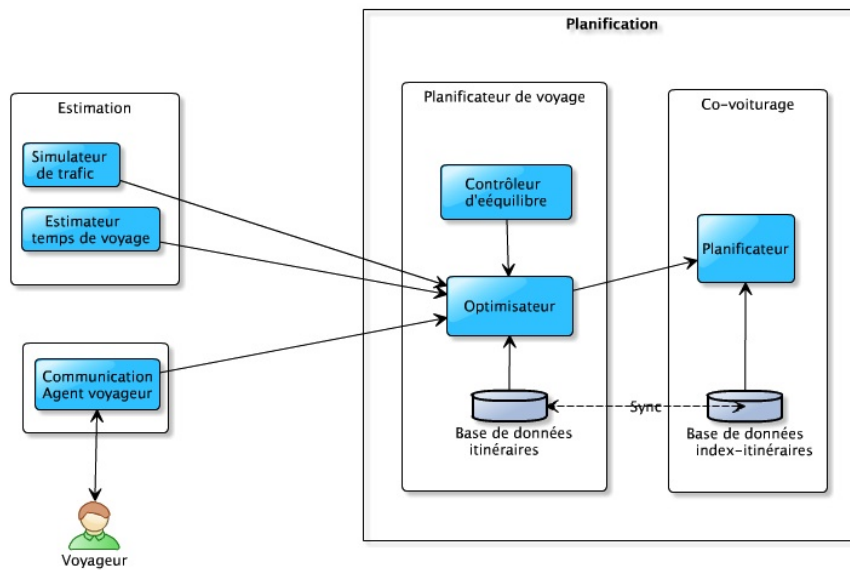


FIGURE 3.5 – Le module de planification

3.3.1 Les contextes

Les contextes sont les concepts de base dans Repast Symphony. Un contexte fournit une structure de données pour organiser les agents autant pour la modélisation que pour l'implémentation. Concrètement, un contexte est une collection d'agents, mais un peu plus riche qu'une simple collection. Les contextes dans Repast sont hiérarchiquement imbriqués, chacun étant un conteneur de composants du modèle. Les composants du modèle peuvent être des simples objets POJO¹², y compris d'autres contextes, mais on s'attend souvent à ce qu'ils soient des objets représentant des agents ayant un comportement. Chaque composant du modèle peut être présent dans de nombreux contextes selon les besoins du concepteur. Un composant du modèle, qui est présent dans un contexte, est également présent dans tous les contextes parents de ce contexte (l'inverse n'est évidemment pas vrai).

L'un des intérêts principaux des contextes est qu'ils servent comme support à des projections (voir plus bas). Les projections dans Repast sont des relations définies sur les membres d'un contexte. Par exemple, la projection *Network* stocke une relation du réseau ou d'un graphe entre les membres de son contexte. Les membres de ce contexte peuvent alors savoir à qui ils sont liés et qui sont liés à eux. De même, la projection *Grid* stocke un ensemble de coordonnées cartésiennes pour chaque membre du contexte. Les membres de ce contexte peuvent savoir où ils se trouvent dans la grille.

Les contextes sont également en charge des couches de données. Les couches de données représentent les données numériques qui peuvent être accessibles en utilisant un ensemble de coordonnées. Les couches de données permettent aux concepteurs de modèles de fournir des

12. Plain Old Java Object

valeurs numériques avec lesquels leurs agents peuvent interagir.

3.3.2 Les projections

Alors que les contextes créent une collection pour organiser les agents, les projections imposent une structure sur ces agents. En utilisant les seuls contextes, on ne peut écrire un modèle qui fournit plus qu'un simple conteneur pour les agents. Les projections permettent de créer une structure qui définit les relations, qu'elles soient spatiales, sémantiques, etc. Une projection est attachée à un contexte particulier et s'applique à tous les agents dans ce contexte. Un point important est à signaler ici : un objet doit exister dans un contexte avant d'être utilisé dans une projection. On peut appliquer autant de projections qu'on désire à un même contexte. On peut donc avoir un contexte qui contient une grille et deux espaces géographiques (deux manières de représenter un réseau de transport par exemple). Les agents peuvent obtenir une référence à diverses projections à travers le contexte auquel ils appartiennent. Un agent peut faire ce qui suit (**this** se réfère à l'agent) :

```
Context<Agent> context = ContextUtils.getContext(this);
Projection projection = context.getProjection("PublicTransportNetwork");
```

Cette projection ne serait pas utile à moins qu'on « caste » la projection à une représentation plus spécifique (un *Network* par exemple).

```
Context<Agent> context = ContextUtils.getContext(this);
Network<Agent> network = context.getProjection(Network.class, "PublicTransportNetwork");
```

En général, les projections sont créés en utilisant le mécanisme de trouver la Factory ainsi.

- Trouver la Factory
- Utiliser la Factory pour créer la projection

Par exemple, pour créer une grille :

```
Context<Agent> context = Contexts.createContext(Agent.class, "MasterContext");
GridFactory factory = GridFactoryFinder.createGridFactory(new HashMap());
Grid grid = factory.createGrid("WorldGrid", context, params);
```

Chaque Factory crée une projection d'un type spécifique et nécessite le contexte auquel la projection est associée et le nom des projections ainsi que des arguments supplémentaires notamment pour le type de projection. Ces arguments supplémentaires ne sont pas définis ci-dessus (le paramètre *params*).

3.3.3 L'ordonnancement

L'ordonnancement est un composant clé de toute simulation. Il définit la manière avec laquelle les actions des agents sont exécutés pendant la simulation. Il y a trois manières de travailler avec l'ordonnanceur de Repast.

Ordonnancer directement une action Pour ordonnancer directement une action dans Repast, on récupère une référence vers un *Schedule* et on l’informe du moment où l’action doit être exécutée. Ci-après un exemple d’ajout d’une action au *Schedule*.

```
//Spécifier que l'action commence au tick 1 et s'exécute tous les deux ticks
ScheduleParameters params = ScheduleParameters.createRepeating(1, 2);

//Récupérer l'ordonnanceur
ISchedule schedule = RunEnvironment.getInstance().getCurrentSchedule();
//Ordonnancer l'action move pour l'agent myAgent avec les paramètres params
schedule.schedule(params, myAgent, "move");
```

Ordonnancer avec les annotations

Les annotations sont des métadonnées qui peuvent être associées aux classes, méthodes ou attributs qui sont disponibles pendant l’exécution pour donner au système plus d’informations. Pour Repast, les annotations sont utilisées entre autres pour associer des informations d’ordonnancement aux méthodes.

```
@ScheduledMethod(start= 1, interval = 2, priority = 2)
public void move()
{
```

Ici la méthode *move* est ordonnancée à partir du tick 1, elle s’exécute tous les deux ticks, et a une priorité de 2. Cela signifie que si une autre action de priorité inférieur (1 par exemple) est ordonnancée, *move* sera exécutée avant elle. L’usage des annotations est le plus simple lorsque les paramètres d’ordonnancement sont connues lors de la compilation. Les arguments des annotations sont similaires aux propriétés d’un objet *ScheduleParameters*. La plupart du temps, les objets avec annotations sont automatiquement ajoutés au *schedule*.

Ordonnancer avec le *Watcher*

Les *Watchers* ou observateurs sont conçus pour être utilisés pour l’ordonnancement dynamique où un *workflow* typique est bien compris par le concepteur du modèle. Un observateur permet à un agent d’être notifié d’un changement d’état dans un autre agent et planifier un événement de se produire à la suite. Le *watcher* est mis en place en utilisant une annotation. Mais au lieu d’utiliser les temps statiques pour les paramètres de planification, l’utilisateur spécifie une requête définissant les agents à observer et une requête définissant une condition de déclenchement qui doivent être remplies pour exécuter l’action.

```
@Watch(watcheeClassName = "com.sm4t.agents.busagent",
        watcheeFieldNames = "position",
        whenToTrigger = WatcherTriggerSchedule.IMMEDIATE,
        scheduleTriggerPriority = 3)
public void positionChanged(BusAgent busAgent){
    if(busAgent.equals(myBus) )
        position = busAgent.getPosition();
}
```

L'annotation `@Watch` avant la méthode indique à Repast Symphony d'observer les autres objets pour ordonnancer la méthode `move()`. Ici, l'agent voyageur voudrait mettre à jour sa position à chaque fois que son bus change de position. Le premier paramètre de l'annotation est le `watcheeClassName`. Il définit le type d'agents que cet agent va surveiller. Le second argument, `watcheeFieldName`, définit l'attribut à surveiller. Cela signifie qu'il y a une variable dans la classe `busAgent`, dont nous voulons surveiller les changements. Quand elle change, cet agent sera notifié. L'argument `whenToTrigger` spécifie s'il faut exécuter l'action immédiatement (avant que les autres actions planifiées à cette instant ne soient exécutées) ou d'attendre jusqu'au prochain tick. Enfin, le `scheduleTriggerPriority` permet de définir l'ordre d'exécution de cette action.

3.4 Le simulateur

Dans cette section, nous décrivons le système multi-agent pour la simulation de l'impact de l'information de voyageurs. Il ne s'agit pas du même simulateur développé pour la plateforme MMT mais d'une version adaptée pour qu'elle s'exécute sur un réseau de transport en commun. Comme nous le verrons dans le chapitre suivant, nous l'avons également enrichi avec des dispositifs d'information de voyageurs, autant au niveau des arrêts qu'avec de l'information personnalisée. Dans ce qui suit, nous donnons une description générale du simulateur, qui sera détaillée dans la suite de ce chapitre et dans le chapitre suivant.

La Figure 3.6 présente le système multi-agent de la simulation, les agents qui le composent, l'environnement ainsi que les interactions entre agents. Les agents qui se déplacent sur le réseau de transport sont les agents voyageurs et les agents véhicules. Les agents voyageurs se décomposent en deux types : des agents connectés (à une source d'information temps-réel de l'opérateur de transport) et les agents non connectés. Le troisième type d'agent évoluant dans le réseau, mais sans se déplacer, est l'agent d'information locale. Il est responsable de la diffusion des informations sur les perturbations au niveau des arrêts du réseau. La plateforme offre un service de planification pour les voyageurs connectés, assurant le suivi de leurs voyages, en se fondant sur une représentation spatiotemporelle du réseau de transport.

Le simulateur représente des passagers, des véhicules de transport public et de moyens d'information à un niveau microscopique et simule leurs mouvements dynamiques. A cette fin, le simulateur a besoin de certains types de données et quelques paramètres additionnels.

3.4.1 Données et paramètres

Le *workflow* de la Figure 3.7 détaille une exécution de simulation et structure le reste de la présentation. La simulation démarre avec le chargement des paramètres (durée de la simulation, nombre d'agents de chaque type, les vitesses par défaut de chaque type d'agent, etc.). Le programme principal crée les graphes logiques (décrits dans la section 3.5.2) et lance l'ordonnanceur. L'ordonnanceur est responsable de la synchronisation de l'exécution des agents à chaque tick de temps simulé. Nous avons défini un ordonnanceur qui lance les agents sur tous les cœurs disponibles (cf. la section 3.4.2 qui suit). Une fois lancé, chaque agent exécute une

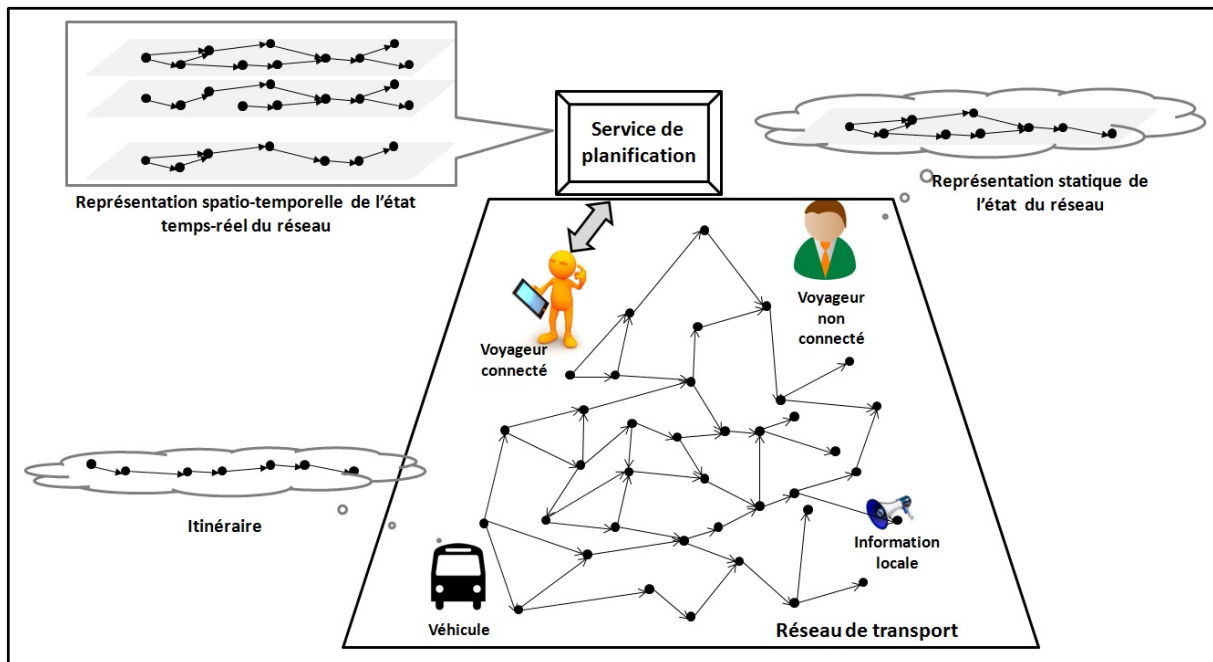


FIGURE 3.6 – Le modèle du système multi-agent

méthode *step*. Le comportement de chaque type d'agent est décrit en section 3.6. Enfin, quand la durée de simulation est atteinte, les résultats sont collectés et la simulation se termine.

3.4.2 Ordonnanceur parallèle

Les algorithmes 1, 2 et 3 détaillent l'ordonnanceur parallèle que nous avons défini.

Algorithm 1 Ordonnanceur parallèle

ENTRÉES: Liste d'agents à ordonnancer \mathcal{A}
 Lancer un Contrôleur de processus $C(\mathcal{A})$ [1]
Tant que \neg AgentsTerminés **faire**
 Attendre [2]
Fin tant que

L'algorithme 1 relatif à l'ordonnanceur parallèle lance un *contrôleur de process* (instruction [1]) et reste en attente qu'il ait terminé son exécution (instruction [2]), i.e. que tous les agents aient exécuté un pas de simulation en parallèle. Lorsque ce sera le cas, il sera réveillé par le contrôleur de processus. Ce dernier, dès son lancement, parcourt la liste d'agents, et pour chaque agent, il essaie de le lancer sur un cœur disponible.

L'état courant des cœurs disponibles est représenté par un tableau de booléens CPU . Lorsque le cœur i est disponible, $CPU[i]$ est égal à *vrai*. Si le contrôleur de processus trouve un cœur disponible, il lance un *processus Agent* représentant l'agent courant (instruction [3]), i.e. il lance un processus léger (*Thread*) en parallèle avec les autres. S'il ne trouve aucun cœur disponible,

Algorithm 2 Contrôleur de processus

ENTRÉES: Liste d'agents à ordonnancer \mathcal{A}

Pour tout $a \in \mathcal{A}$ **faire**

Pour tout $i \in \{1, \dots, nbCPU\}$ **faire**

si $CPU[i]$ **alors**

 Lancer un processus agent $PA(i)$ [3]

 trouveCPU \leftarrow vrai

fin si

fin pour

si \neg trouveCPU **alors**

 Attendre [4]

fin si

fin pour

Tant que \neg tousFinis **faire**

 tousFinis \leftarrow vrai

Pour tout $i \in \{1, \dots, nbCPU\}$ **faire**

si $\neg CPU[i]$ **alors**

 tousFinis \leftarrow faux

 Attendre [5]

fin si

fin pour

Fin tant que

Réveiller l'ordonnanceur parallèle [6]

Algorithm 3 Processus Agent

ENTRÉES: un a à exécuter

 Exécuter le comportement *step* de a

 Réveiller le contrôleur de processus [7]

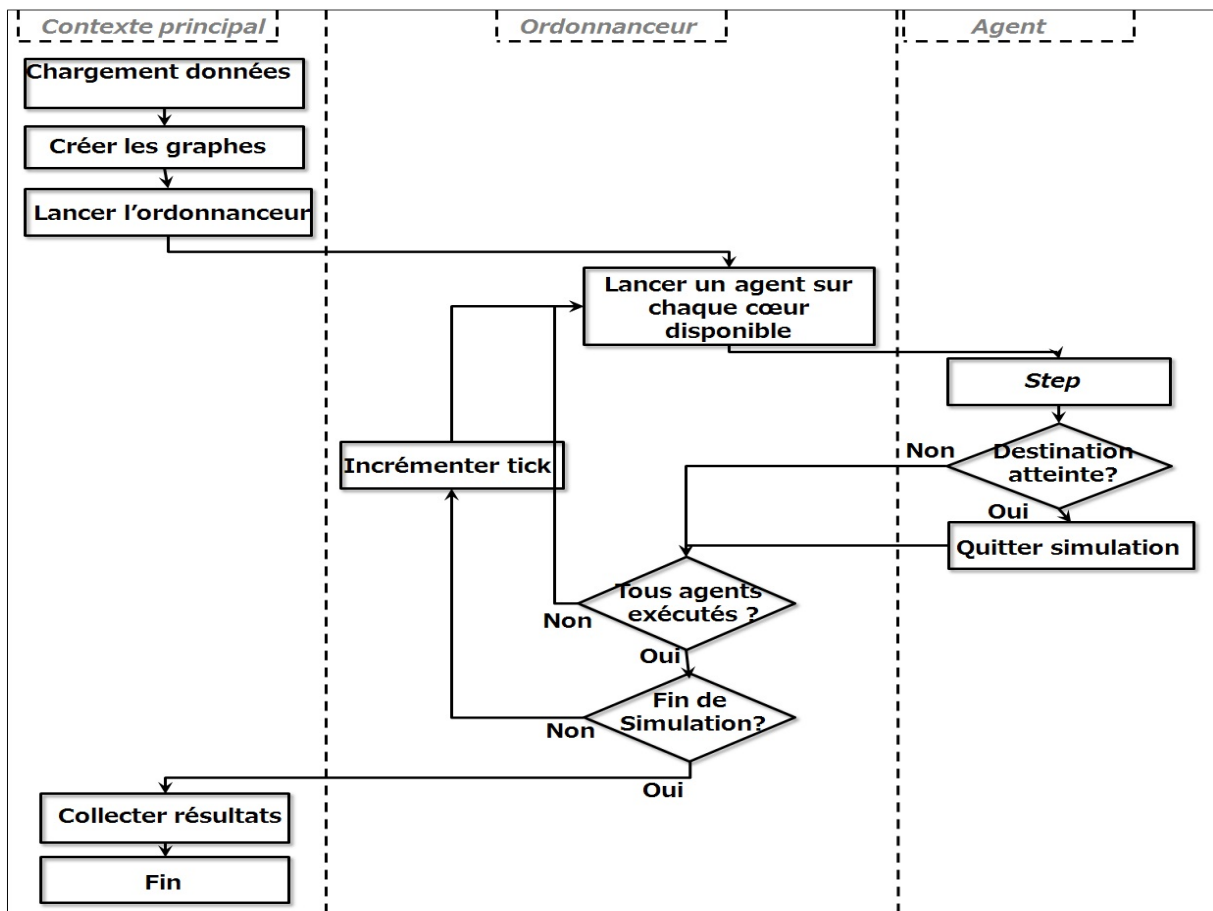


FIGURE 3.7 – Le workflow de la simulation

le processus se met en attente (instruction [4]). Il sera réveillé dès qu'un agent libère un cœur. Lorsque tous les agents ont été lancés, le contrôleur de processus se met en attente jusqu'à ce que les derniers agents terminent (instruction [5]). Une fois que tous les agents ont terminé, le contrôleur de processus réveille l'ordonnanceur parallèle. Enfin, le *processus Agent* est responsable de l'exécution en parallèle des actions par défaut des agents (*step*) avant de réveiller le contrôleur de processus.

L'exécution en parallèle des agents posent des problèmes de synchronisation et d'accès concurrent à l'environnement. Nous discuterons de ces questions lorsque nous aurons présenté l'ensemble du SMA, à la fin du chapitre suivant.

3.4.3 Données et paramètres du simulateur

Les données manipulées par le simulateur sont les informations nécessaires à son exécution. Les paramètres sont les valeurs servant de calibrage du simulateur, qui agissent sur son fonctionnement.

Données

Les données en entrée pour notre simulateur de déplacements sont les suivants :

1. Le réseau piéton,
2. Le réseau de transports en commun,
3. La description des correspondances,
4. Le tableau de marche des véhicules de transport en commun,
5. Le modèle de déplacement,
6. Les profils des voyageurs.

Il existe des standards internationaux pour la définition des réseaux piétons et des réseaux de transport en commun. Néanmoins, beaucoup d'informations contenues dans un fichier de description d'un réseau sont inutiles pour notre problématique. Nous avons donc décidé, par souci d'optimisation, de créer un schéma XSD de description de réseau qui est utilisé autant pour le réseau piéton que pour le réseau de transport en commun. Ce schéma ne contient que les informations dont nous avons besoin pour exécuter le simulateur. Nous créons donc nous-mêmes les objets à afficher et le graphe logique correspondant à partir des définitions XML des réseaux reçus en entrée.

Ces données nous ont été rendues disponibles dans le cadre de l'Open Data de Toulouse Métropole et les API fournies par Tisséo SMTC, l'autorité organisatrice des transports publics Toulousains. Leur consommation, filtrage et mise à disposition de ces données dans le format que nous avons défini ont été effectués avec les outils de la plateforme Claire-Siti [Scemama and Carles2004].

Les réseaux piéton et transport en commun Les réseaux piétons et de transport en commun sont définis de la même manière et obéissent au schéma XSD des figures A.1 et A.2 de l'annexe A.

La description des correspondances Le fichier des correspondances est un tableau contenant des informations sur les arrêts du réseau pour lesquels une correspondance à pied est possible et les nœuds du réseau routier qui sont accessibles à pied depuis les arrêts. Cette correspondance entre nœuds du réseau routier et réseau de transport en commun est nécessaire, bien que nous ne représentons pas les véhicules privés dans la simulation, parce que les voyageurs démarrent et arrivent sur le réseau piéton. Ils doivent donc à un moment passer d'un réseau à un autre, en utilisant ces arcs de correspondance.

Le tableau de marche Les tableaux de marche obéissent au schéma XSD de la figure A.3 de l'annexe A.

Les tableaux de marche des véhicules sont composés d'un ensemble de missions. Chaque mission correspond à un itinéraire spécifique et décrit le chemin d'un véhicule et les temps de visite correspondants. Chaque tableau de marche est donc une séquence de couples $\langle \text{arrêt}, \text{temps} \rangle$.

Le modèle de déplacement Optionnel, le modèle de déplacement découpe la région géographique considérée en zones et décrit le nombre de personnes demandant à quitter chaque zone ainsi que leurs zones de destination. Les modèles de déplacement sont très importants si nous désirons avoir une simulation qui mime le comportement de mobilité d'une manière réaliste. Afin que le simulateur intègre le modèle de déplacement, nous adoptons la procédure suivante [kso2016].

	Z_1	Z_2	Z_3
Z_1	0	8	4
Z_2	4	0	2
Z_3	2	0	0

TABLE 3.1 – Exemple de matrice D (avec $N = 3$)

Soit un modèle de déplacement pour la région géographique et la période de temps considérée sous la forme d'une matrice $D = \{(d_{ij})\}$ de dimension $N \times N$ avec N le nombre de zones de la région; d_{ij} est le nombre de personnes se déplaçant de la zone i vers la zone j . Par exemple, dans la Table 3.1, le nombre de voyageurs dans le modèle de déplacement est $M = 20$, 8 partent de la zone 1 à la zone 2, 4 partent de la zone 2 à la zone 1, 2 partent de la zone 3 à la zone 1, etc.

Cependant, le nombre d'agents réellement simulés $|A|$ n'est pas nécessairement égal au nombre de personnes $|M|$ dans le modèle de déplacement. Notre objectif est de générer, d'une manière non-déterministes, des origines et destinations simulées qui sont proportionnelles au modèle de déplacement. Pour ce faire, nous créons d'abord une matrice $S = \{(s_{ij})\}$ de dimension $N \times N$; s_{ij} est le nombre d'agents simulés qui partiront de la zone i vers la zone j , $s_{ij} = d_{ij} \times \frac{|A|}{|M|}$, avec $|A|$ le nombre d'agents simulés et $|M|$ le nombre de voyageurs réels dans le modèle de déplacement, i.e. $M = \sum_{i=0}^N \sum_{j=0}^N d_{ij}$.

En nous fondant sur S , une matrice MP de proportions de voyageurs est créée (cf. Table 3.3).

Enfin, une table de correspondance dynamique P est créée (cf. Table 3.4) sur la base de MP . Le tableau P fait correspondre des intervalles avec des couples ($zone_{origine}, zone_{destination}$). La longueur de l'intervalle est proportionnelle au poids relatif courant des couples de zones dans S . Lorsqu'un agent est généré avec une origine appartenant à Z_1 , et une destination appartenant à Z_2 , la cellule (Z_1, Z_2) dans S est décrémentée et P est mise à jour en conséquence (cf. Table 3.4).

A chaque pas de temps, le simulateur génère un nombre de nouveaux voyageurs pour lesquels nous devons définir une origine et une destination. Pour chaque nouvel agent voyageur,

	Z_1	Z_2	Z_3
Z_1	0	4	2
Z_2	2	0	1
Z_3	1	0	0

TABLE 3.2 – Exemple de matrice S avec $|A| = 10$ et $|M| = 20$

	Z_1	Z_2	Z_3
Z_1	0%	40%	20%
Z_2	20%	0%	10%
Z_3	10%	0%	0%

TABLE 3.3 – Exemple de matrice MP correspondant aux proportions dans S

Intervalle	Origine-Destination	Calcul de la Longueur d'Intervalle
$[0\%, 40.00\%[$	$Z_1 - Z_2$	$\frac{4}{10}$
$[40.00\%, 60.00\%[$	$Z_1 - Z_3$	$\frac{2}{10}$
$[60.00\%, 80.00\%[$	$Z_2 - Z_1$	$\frac{2}{10}$
$[80.00\%, 90.00\%[$	$Z_2 - Z_3$	$\frac{1}{10}$
$[90.00\%, 100\%[$	$Z_3 - Z_1$	$\frac{1}{10}$

TABLE 3.4 – Exemple de table P

un nombre aléatoire $\rho \in [0, \dots, 1]$ est choisi et P est utilisé pour choisir les zones origine et destination. Soit $\rho = 0.15$, alors suivant la table P dans la table 3.4, le couple $Z_1 - Z_2$ est choisi pour l'agent voyageur courant. Le nœud origine est alors choisi aléatoirement parmi les nœuds de la zone Z_1 alors que le nœud destination est choisi aléatoirement dans la zone Z_2 . La nouvelle table P est donnée dans la table 3.5 et la probabilité de choisir une nouvelle fois ce même couple de zones $Z_1 - Z_2$ devient plus faible ($\frac{3}{9}$).

De cette manière, même si les origines et les destinations sont choisis d'une manière aléatoire, les origines et destinations choisies demeurent proportionnelles au modèle de déplacement tout au long de la simulation.

Les profils des voyageurs Optionnels, les profils des voyageurs définissent les propriétés et les préférences des voyageurs. Elles peuvent concerner des contraintes d'accessibilité pour les personnes à mobilité réduite, des vitesses de piétons (profils de vitesses associés à l'âge par exemple), des préférences en termes de critères à optimiser (nombre de correspondances, plus

Intervalle	Origine-Destination	Calcul de la Longueur d'Intervalle
$[0\%, 33.33\%[$	$Z_1 - Z_2$	$\frac{3}{9}$
$[33.33\%, 55.55\%[$	$Z_1 - Z_3$	$\frac{2}{9}$
$[55.55\%, 77.77\%[$	$Z_2 - Z_1$	$\frac{2}{9}$
$[77.77\%, 88.88\%[$	$Z_2 - Z_3$	$\frac{1}{9}$
$[88.88\%, 100\%[$	$Z_3 - Z_1$	$\frac{1}{9}$

TABLE 3.5 – P après le choix de (Z_1, Z_2)

court chemin en temps ou en distance, le moins de marche à pied, le moins d'émissions CO₂, etc.).

Paramètres

Durée de la simulation Deux valeurs définissent la durée d'une simulation. Le premier est un intervalle définissant le premier instant et le dernier instant simulés (notés τ^- et τ^+ respectivement). En l'absence de ces deux paramètres, nous utilisons respectivement deux valeurs associées au temps maximal et au temps minimal dans les tableaux de marche des véhicules de transport en commun. La seconde valeur qui doit être définie est le nombre de ticks discrets de temps que la simulation exécutera avant de terminer (notée δ dorénavant). A chaque tick, tous les agents sont activés pour une action particulière définie par leurs comportements.

Chaque agent ordonnancé a une méthode *step* qui est appelée à chaque pas de temps. La durée de la simulation est importante. Plus la durée est longue, et plus les déplacements des agents sont fluides et synchronisés (sans saut brusque d'une position à une autre, et sans la sensation d'un traitement séquentiel des agents). La contrepartie est une simulation plus longue, puisque plus il y a de pas de simulation, plus l'ordonnanceur aura de tâches à faire. Ce paramètre est très important pour la définition des autres données du modèle que nous définissons dans ce qui suit.

Vitesses des agents Comme évoqué plus haut, la durée de la simulation (paramètre Δ) est très important pour le calibrage de toute la simulation. L'un des paramètres clé de la simulation est la vitesse des agents. Les vitesses des agents sont définies de deux manières :

- Véhicules de transport en commun : fondés sur les données en entrée, les vitesses sont déduites depuis les couples $\langle \text{arrêt}, \text{temps} \rangle$.
- Défaut : l'utilisateur du simulateur définit les vitesses des piétons et des véhicules de transport en commun comme paramètres. Les vitesses de piétons définies par l'utilisateur sont utilisées si les profils des utilisateurs sont absents ou si les vitesses n'y sont pas définies. Les vitesses de véhicules de transport en commun définies par l'utilisateur sont utilisées si deux couples $\langle \text{arrêt}, \text{temps} \rangle$ successifs donnent une vitesse non consistante due à des erreurs dans les données (E.g. vitesse infinie due à deux temps de visite identiques de deux arrêts successifs.).

Transformation des unités Toutes les données sont exprimées en fonction du temps (e.g. les temps de passage des véhicules de transport en commun) doivent être exprimées en termes de ticks de simulation. Cependant, puisque les données originales relatives au temps sont exprimées en termes de dates, elles doivent être transformées comme suit. Soit δ le nombre de ticks pendant la simulation, t le temps originel et τ^- et τ^+ le premier instant et le dernier instant simulés. t' est le nouveau temps (en ticks de simulation) est calculé comme suit :

$$t' = \frac{t - \tau^-}{\gamma} \times \delta, \text{ avec } \gamma = \tau^+ - \tau^-$$

De plus, toutes les vitesses sont originellement définies en termes de Km/h. Elles doivent être transformées en mètres/tick comme suit (σ_{mode} est la vitesse du mode de transport, et $\sigma_{modeKmH}$ est sa vitesse originelle exprimée en Km/h).

```

class TickCountFormatterClass implements TickCountFormatter{
    private double initial;
    private long gap;
    private long minTime;

    public TickCountFormatterClass(double initial, long gap, long minTime){
        this.initial = initial;
        this.gap = gap;
        this.minTime = minTime;
    }

    @Override
    public String format(double tick) {
        double x = (tick*1000*gap/simulationDuration)+(minTime*1000);
        DateFormat dateFormat = new SimpleDateFormat("EEE d MMM yyyy 'at' HH:mm:ss", Locale.UK);
        Date date = new Date(new Long (Math.round(x)));
        return cityName + ": " + dateFormat.format(date);
    }

    @Override
    public String getInitialValue() {
        // TODO Auto-generated method stub
        DateFormat dateFormat = new SimpleDateFormat("EEE d MMM yyyy 'at' HH:mm:ss", Locale.UK);
        Date date = new Date(Math.round(initial)*1000);
        return cityName + ": " + dateFormat.format(date);
    }
}

TickCountFormatter tcf = new TickCountFormatterClass(minTime, gap, minTime);
repast.simphony.ui.RSApplication.getRSApplicationInstance().getGui().setTickCountFormatter(tcf);

```

FIGURE 3.8 – Le formatage du temps

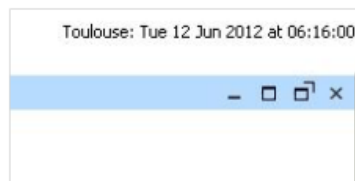


FIGURE 3.9 – Le formatage du temps - Résultat

$$\sigma_{mode} = \frac{\sigma_{modeKmH} \times \gamma}{3.6 \times \delta}$$

Prenons le cas du mode véhicule de transport en commun comme exemple. $\frac{\gamma}{\delta}$ donne le nombre de secondes passées en un tick de temps de simulation. Ainsi, $\sigma_{car} = \frac{\sigma_{carKmH} \times \gamma}{3.6 \times \delta}$ donne le nombre de mètres qu'un véhicule peut parcourir en un tick de temps de simulation. Le même principe s'applique pour les piétons.

L'affichage par défaut dans Repast Symphony donne le tick courant de la simulation. Il est donc difficile de pouvoir se repérer par rapport au moment de la journée que nous sommes en train d'observer à l'écran. Pour avoir non plus des ticks, mais une date et une heure affichées à l'écran, il faut définir une classe interne qui formate les ticks d'une manière correcte et informer Repast qu'un objet de cette classe servira pour le formatage des ticks de temps (cf. figure 3.8). Le résultat de ce formatage pendant l'exécution est donné par la figure 3.9.



FIGURE 3.10 – La définition des paramètres dans l'interface du simulateur

```

simulationDuration = getParameter("simDur");
cityName = getParameter("cityName");
nbDrivers = getParameter("nbDrivers");
nbTravellers = getParameter("nbTravellers");
busSpeedKMH = getParameter("busSpeedKMH");
carSpeedKMH = getParameter("carSpeedKMH");
pedestrianSpeedKMH = getParameter("pedestrianSpeedKMH");

busSpeed = busSpeedKMH / (3.6 * simulationDuration); // A multiplier par gap
carSpeed = carSpeedKMH / (3.6 * simulationDuration); // A multiplier par gap
pedestrianSpeed = pedestrianSpeedKMH / (3.6 * simulationDuration); // A multiplier par gap

```

FIGURE 3.11 – La récupération des paramètres

3.5 Déplacement dans la géographie

3.5.1 Contextes et projection

Il y a deux types d'objets que nous utilisons par type de réseau de transport. Pour le réseau piéton, nous utilisons les *Roads* (routes) et les *Junctions* (intersections). Les *Junctions* sont créés dans les points où deux *roads* se rencontrent et ils forment les nœuds dans le graphe logique de déplacement. Nous utilisons deux contextes séparés pour stocker ces objets appelés : *RoadContext* et *JunctionContext*. Nous avons également créé deux projections de type GIS (projection géographique 2D) et une projection *Network* pour ces contextes. Pareillement pour le réseau de transport en commun, nous utilisons les *Links* (inter-arrêts) et les *Stops* (les arrêts) avec leurs contextes associés : *LinkContext* et *StopContext*. Nous utilisons la même projection pour le réseau routier et le réseau de transport en commun de type GIS, que nous avons appelé *mainGeography*. Tous les objets à afficher sont ajoutés dans cette projection (les *roads*, *junctions*, *links*, *stops* et les agents).

Les mêmes traitements sont effectués pour le réseau piéton et le réseau TC. Pour chaque intersection ou arrêt, nous en créons le correspondant graphique (*Junction* ou *Stop*) et logique (nœud dans le graphe logique). Pour chaque route ou inter-arrêt, nous en créons le correspondant graphique (*Road* ou *Link*) et logique (arc dans le graphe logique). Nous avons défini deux graphes logiques, un pour le réseau piéton, l'autre pour le réseau TC. Nous n'utilisons pas le même graphe pour les deux réseaux afin d'avoir des coûts de déplacements corrects. En effet, une même portion de route n'a pas le même coût de déplacement selon qu'elle est traversée par un bus ou par un piéton. Utiliser le même graphe pour les deux réseaux entraînerait des calculs de plus courts chemin erronés. Ainsi, le coût associé à un arc du graphe piéton est égal à sa distance multipliée par la vitesse d'un piéton (paramètre défini plus haut). Le coût associé à un arc du graphe TC est quant à lui égal à sa distance multipliée par la vitesse d'un bus.

C'est tout ce qu'il est nécessaire d'avoir pour créer le réseau de transport d'une ville. Maintenant, nous avons les informations spatiales (la *mainGeography*) et les informations topographiques (les deux projections *Network* (TC et piétons)).

3.5.2 Planification des itinéraires

Pour pouvoir calculer les itinéraires des voyageurs, nous devons modifier le graphe de transport en commun vu que la notion de ligne de transport en commun est perdue au sein du graphe logique. En effet, si nous fondons nos calculs sur ce graphe, les plus courts chemins résultats risquent de comporter un très grand nombre de correspondances. Ainsi, pour éviter cet effet nous modifions le graphe de transport en commun comme suit (cf. Figure 3.12). Soit $\langle s_1, s_2 \rangle$ un arc dans le graphe de transport en commun. Soit it_1 et it_2 deux itinéraires (suivis par deux véhicules différents donc) passant par cet arc. Nous créons quatre nouveaux nœuds : $it_1 : s_1$, $it_1 : s_2$ que nous relient avec l'arc appartenant à l'itinéraire it_1 ; et $it_2 : s_1$ et $it_2 : s_2$ que nous relient avec l'arc appartenant à l'itinéraire it_2 . Nous créons aussi quatre nouveaux arcs $\langle it_1 : s_1, it_2 : s_1 \rangle$, $\langle it_2 : s_1, it_1 : s_1 \rangle$, $\langle it_1 : s_2, it_2 : s_2 \rangle$ et $\langle it_2 : s_2, it_1 : s_2 \rangle$, que nous

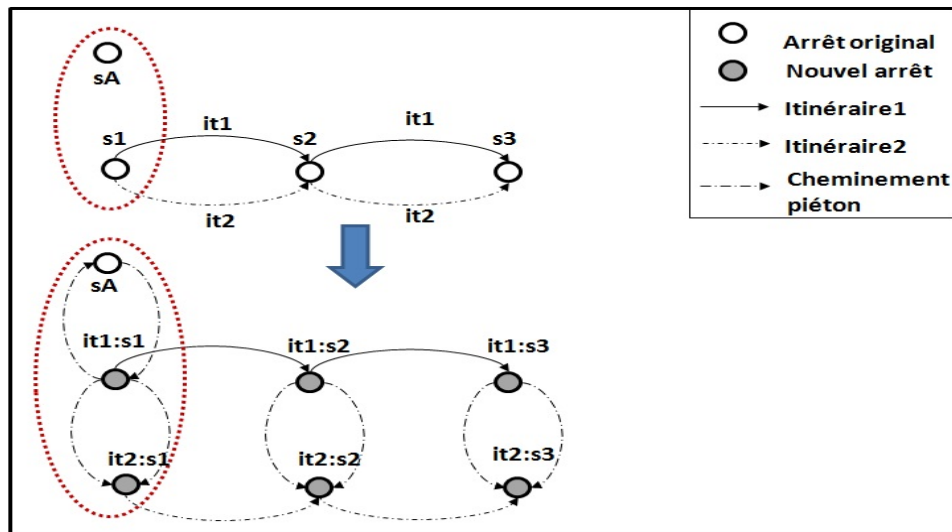


FIGURE 3.12 – La transformation du réseau de transport en commun

appelons « arcs piétons » et qui sont pénalisés. Lorsque nous exécutons un plus court chemin sur ce nouveau graphe, les itinéraires qui appartiennent au même véhicule/ligne sont encouragés et une correspondance n'est proposée que s'il est impossible ou très cher de rester dans le même véhicule ou sur la même ligne. Après l'exécution d'un plus court chemin A^* sur le graphe résultat, le service de planification interroge les arrêts du meilleur itinéraire trouvé afin de calculer la séquence de véhicules que l'agent voyageur aura à prendre et renvoie le résultat à l'agent voyageur.

Les coûts associés aux arcs de ce réseau sont également différents. En effet, les itinéraires en transport en commun ont une fréquence différente. Le coût d'un arc est défini de manière à refléter cette fréquence. Le temps de parcours t_{ij}^k entre deux nœuds i et j du même itinéraire k est égale au temps de parcours d'un véhicule $\tau(i, j)$ à laquelle est ajoutée le temps d'attente moyen en secondes égal à $\frac{3600}{\phi(k)}$, avec $\phi(k)$ la fréquence horaire de l'itinéraire k . L'ensemble des fréquences et des coûts des arcs sont mis à jour lors du passage des heures creuses aux heures pleines et inversement.

3.5.3 Le déplacement

Les agents sont autorisés à se déplacer une certaine distance à chaque tick de simulation. La dernière étape est donc de savoir si l'agent peut aller jusqu'à la prochaine coordonnée dans sa liste ou pas. Si ce n'est pas le cas, on doit savoir jusqu'où il peut aller. Dans la figure 3.13, si l'agent ne peut pas aller de la coordonnée (b) à la coordonnée (a), il doit avancer dans la direction de (a) du maximum de distance lui restant à parcourir. Ainsi, au prochain tick, il peut continuer vers (a) normalement. La distance maximale autorisée par tick de temps est calculée comme défini dans le paragraphe 3.4.3.

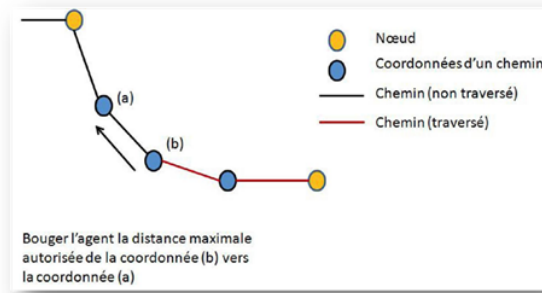


FIGURE 3.13 – Le déplacement d’un agent

3.6 Le système multi-agent

3.6.1 Les agents

Notre simulateur est composé de 2 types d’agents (d’autres agents seront définis au chapitre suivant) : les agents véhicules (*BusAgent*) et les agents voyageurs (*TravellerAgent*) constituent les éléments mobiles qui se déplacent sur le réseau. Le service de planification assure une fonction d’assistance aux agents voyageurs. Dans cette section, nous présentons les deux types d’agents ainsi que le service de planification.

Les agents véhicules (*BusAgent*)

Les *BusAgent* ne choisissent pas leur destination d’une manière autonome, mais obéissent à un tableau de marche. Chaque *BusAgent*, dès sa création, retrouve son chemin à partir de son tableau de marche. Comme explicité plus haut et en annexe A, un tableau de marche est composé d’une séquence de couples $\langle arrt, temps \rangle$. Le *BusAgent* transforme ce chemin en une succession d’arcs. Ceci est fait par un accès à la définition du réseau de transport public à la recherche d’un arc de la même ligne que le bus qui relie les deux nœuds successifs du tableau de marche. Une fois tous les arcs du tableau de marche retrouvés, l’agent *BusAgent* retrouve les vitesses associées à ces arcs. La vitesse doit prendre en compte le fait que le bus doit s’arrêter pour que les voyageurs montent à bord. L’équation retenue, et qui est utilisée en général, est la suivante : si le bus visite deux arrêts successifs i et j aux instants t et $t + 3$, il s’arrête au niveau de i de t à $t + 1$ avant de démarrer. Un tiers du temps de déplacement entre deux arrêts est donc alloué à la montée et à la descente des voyageurs. Si, pour une raison ou une autre, la vitesse du bus ainsi calculée se révèle farfelue (e.g. $120km/h$), le paramètre *BusSpeedKmH* défini plus haut est utilisé pour lisser les valeurs des temps de passage vers des quantités raisonnables.

Tant que le *BusAgent* n’est pas arrivé à destination, il se déplace à chaque pas de temps de la distance permise, suivant sa vitesse courante. S’il arrive à un arrêt, il déclenche un compteur d’arrêt à l’expiration duquel il va redémarrer. Durant ce temps, il fait descendre les agents

voyageurs inscrits dans la liste de descente, ensuite, il embarque ceux inscrits dans la liste d'attente de l'arrêt (figure 3.14). S'il a atteint sa destination, le *BusAgent* quitte la simulation.

Le service de planification

Le service de planification répond aux requêtes de déplacement des agents voyageurs. Il est responsable du calcul du meilleur itinéraire pour ces agents voyageurs. La méthode de calcul des itinéraires est décrite dans la section 3.5.2. Le résultat du calcul est constitué d'une alternance entre cheminement piétons et itinéraires en réseau de transport en commun. Un cheminement piéton est composé d'une séquence d'arcs du réseau piéton avec les temps de passage correspondants. Un itinéraire en transport en commun est quant à lui composé d'une séquence de couples $\langle id_{vehicule}, itinéraire \rangle$, avec $id_{vehicule}$ l'identifiant du véhicule à prendre et la partie correspondante de l'itinéraire.

Les agents voyageurs (*TravelerAgent*)

L'origine et destination de l'agent voyageur sont soit déduites depuis le modèle de déplacement s'il existe, soit choisies d'une manière aléatoire sinon. Ensuite, le voyageur reçoit un itinéraire composé d'une liste de coordonnées de la part du service de planification. Pour parcourir cet itinéraire, un agent voyageur alterne entre la marche, l'attente d'un véhicule et le déplacement à bord du véhicule jusqu'à arriver à destination. Quand il marche, l'agent voyageur est autorisé à parcourir une distance égal à $\sigma_{passager}$ à chaque pas de temps. Quand il est à bord d'un véhicule, il délègue la responsabilité de son mouvement à l'agent véhicule qu'il emprunte.

Une question s'est posée à ce niveau. Qui doit vérifier la compatibilité entre un agent voyageur et un agent véhicule? Intuitivement, c'est le voyageur qui vérifie que le véhicule arrivé à l'arrêt dans lequel il est positionné l'emmène bien à sa destination (i.e. le prochain point de correspondance) ou pas. Néanmoins, dans le cadre du simulateur, la question se pose différemment. Si nous définissons une durée de simulation trop petite, de telle manière qu'un véhicule a le temps de s'arrêter au niveau d'un arrêt et de redémarrer dans un même pas de temps, l'agent voyageur n'aura jamais la main pour vérifier s'il doit prendre le véhicule ou pas. Pour cette raison, nous procédons de la manière suivante pour faire correspondre les voyageurs et les véhicules de transport public.

Dès qu'un agent voyageur arrive à un arrêt pour attendre un véhicule de transport public, il s'ajoute dans une liste prévue à cet effet dans les attributs de l'arrêt et il met son propre statut à « en attente d'un véhicule ». C'est l'agent véhicule qui, en arrivant au niveau de l'arrêt en question vérifie la liste des agents voyageurs en attente et vérifie qu'il les emmène bien à leur prochaine destination. Si tel est le cas, il se charge de leur changement de coordonnées (i.e. leur déplacement) jusqu'au prochain point de correspondance. Durant ce temps, les agents voyageurs concernés ont un statut égal à « à bord ». Ainsi, les agents voyageurs passent par une succession d'états :

1. soit ils marchent vers leur prochain arrêt,
2. soit ils attendent un véhicule,
3. soit ils sont à bord d'un véhicule,

4. soit ils sont arrivés à destination et dans ce cas ils quittent la simulation.

La figure 3.14 illustre le diagramme d'activité du déplacement de l'agent voyageur et son interaction avec l'agent véhicule.

L'usage de l'ordonnanceur parallèle fait que l'exécution des agents risquent d'interférer, et qu'ils risquent d'accéder d'une manière concurrente à la projection *mainGeography*. Pour éviter que l'exécution aboutisse à un état incohérent de la simulation, nous avons recours à deux procédures :

1. l'accès à *mainGeography* est systématiquement synchronisé.
2. les actions des agents véhicules sont ordonnancées avec une priorité plus grande que celle des agents voyageurs.

3.6.2 Packages

Les packages `public_transport.agent` et `common.agents` contiennent respectivement les agents de transport public (principalement *BusAgent*) et les agents piétons (principalement *TravelerAgent*).

Le package `common.environment.generated.network` contient les classes générées à partir du XSD de définition des réseaux (avec l'Api JAXB). Le package `common.environment.generated.timetable` contient les classes générées à partir du XSD de définition des tableaux de marche des véhicules de transport en commun. Les packages `public_transport.environment`, `road_transport.environment` et `common.environment` contiennent les classes définissant l'environnement des réseaux, respectivement de transport public, des piétons et de transport multimodal. Ils contiennent les définitions de *Road*, *Junction*, *Link*, *Stop*, *Edge*, etc. et toutes les classes nécessaires au déplacement dans la géographie. Le package `common.main` contient la classe principale qui initialise la simulation. Dans *Repast*, il s'agit d'une classe qui implémente `repast.simphony.dataLoader.ContextBuilder`, et qui définit une méthode `build` qui renvoie le contexte principale de la simulation, contenant tous les agents et tous les objets géographiques créés initialement. Les autres packages sont assez classiques : `common.exceptions` pour les exceptions et `common.util` pour les fonctionnalités utilitaires manipulées dans la simulation, se présentant généralement sous la forme de méthodes de classe. Le package `common.main.schedulers` contient la classe qui sert à paralléliser l'exécution des agents.

3.7 Optimisations

Nous avons rencontré un grand nombre de problèmes pendant le développement du simulateur de déplacements. Beaucoup de ces problèmes étaient liés à de mauvaises performances, relatives à l'usage de la mémoire ainsi qu'aux temps d'exécution. Nous avons également dû faire face à des erreurs dans les données manipulées, que nous avons dû corriger quasi-manuellement.

3.7.1 Des données erronées ou manquantes

Les données géographiques en entrée dans le simulateur nous avaient été fournies par Tisséo SMTC, via la plateforme *ClaireSiti*. Ces données ne sont pas destinées à la base pour faire une

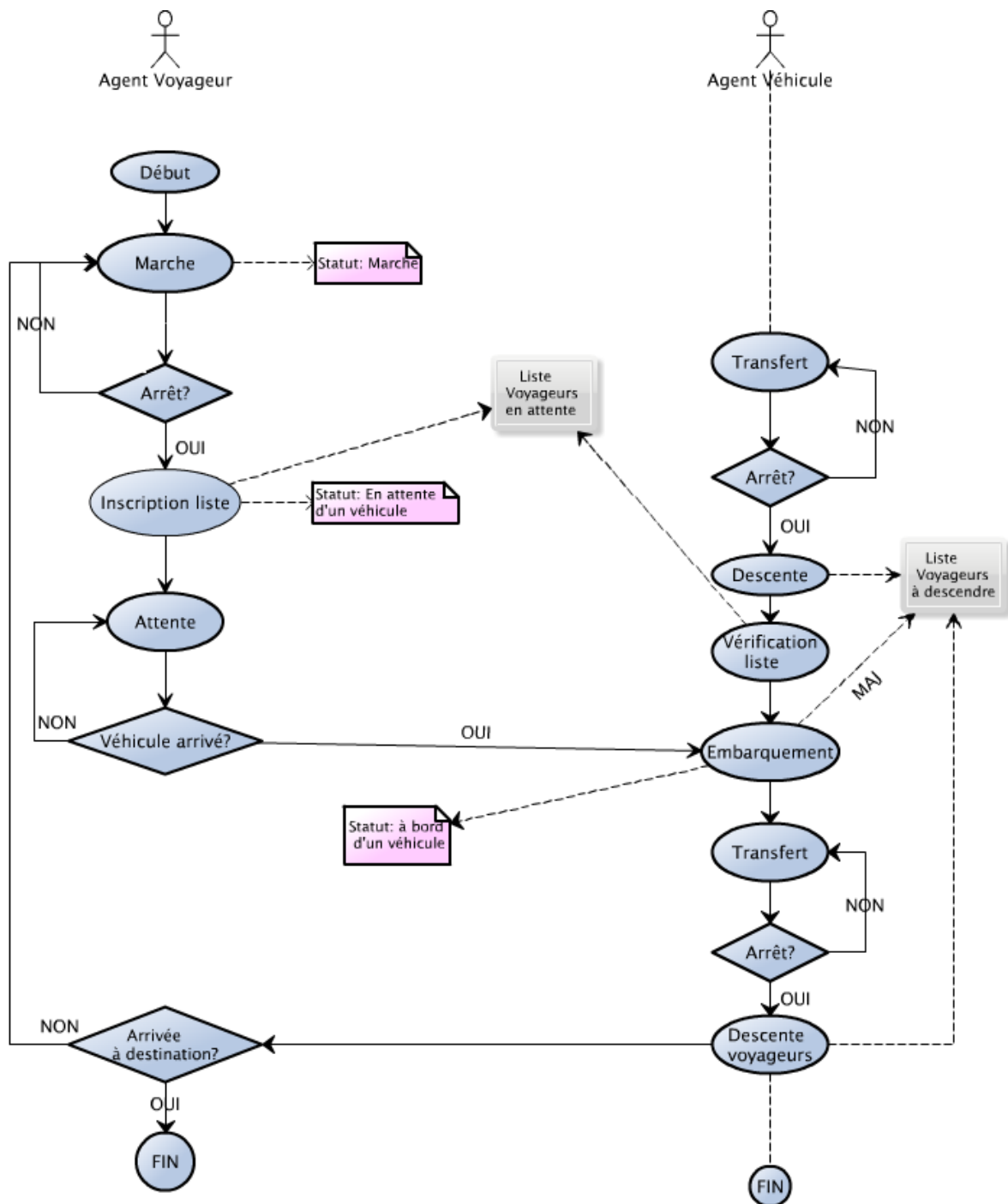


FIGURE 3.14 – Le diagramme d'activité du déplacement de l'agent voyageur.

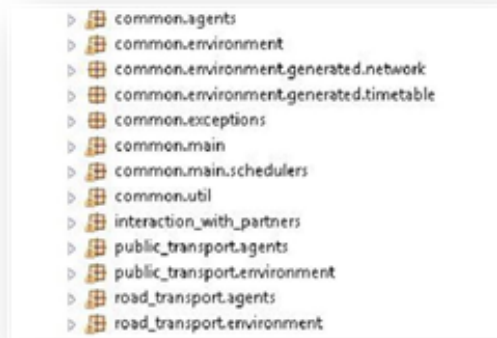


FIGURE 3.15 – Les packages

simulation détaillée de déplacements, mais pour faire de la régulation. Que les tracés géographiques des lignes ne soient pas exacts n'est pas un problème. Or c'est bien un problème pour faire une simulation de déplacement. Nous avons donc dû corriger ces données en revenant à la description géographique détaillée du réseau (Base Navteq dont dispose l'unité de recherche GRETTIA dans le cadre d'un partenariat de recherche). Une fois les arrêts localisés sur la carte géographique, il nous a suffi de faire tourner un plus court chemin entre deux arrêts successifs pour retrouver le tracé géographique correct des inter-arrêts.

Les données sur les tableaux de marche des véhicules nous avaient été également fournies via la plateforme ClaireSiti. Elles résultent de la sauvegarde de journées d'exploitations archivées. Les tableaux de marche sur lesquelles nous travaillons concernent donc des temps de passages réels aux arrêts, enregistrés à posteriori. Cela à l'avantage d'avoir une circulation réaliste des bus. Cependant, ces données proviennent des données du système d'aide à l'exploitation, qui se fondent sur une consultation des positions des bus après un certain temps de scrutation (toutes les 30 secondes, toutes les minutes, etc.). Il suffit donc que le temps de scrutation soit assez grand, ou qu'il y ait un problème de communication pour que le tableau de marche devienne incohérent. C'est pour cette raison que nous avons procédé à deux corrections :

- nous avons corrigé certaines données en revenant au tableau de marche théorique de l'opérateur
- nous avons utilisé le paramètre vitesse des véhicules comme décrit plus haut

3.7.2 Problème de performance

L'étape d'initialisation de la simulation prenait un très grand temps. Afin d'alléger l'initialisation de la simulation, nous avons sérialisé les objets représentant les réseaux de transport en commun et piéton, que nous chargeons à chaque nouvelle simulation. Pareillement pour les tableaux de marche. Du moment que les tableaux de marche étaient stabilisés et corrigés, nous les avons également sérialisés, et nous les chargeons au début de la simulation.

Un autre traitement qui ralentissait l'initialisation ainsi que l'ensemble de la simulation et augmentait la taille de la mémoire nécessaire était le dessin du réseau de transport en commun

à chaque démarrage de simulation. Ce dessin s'effectue une seule fois au début de la simulation (les arrêtes et les arcs ne bougent pas). Pour pallier ces inconvénients, nous avons créé un petit programme qui transforme la description du réseau en un *shapefile*. Désormais, nous utilisons ce *shapefile* comme simple fond de carte de la simulation, ce qui ne prend pas de temps à l'initialisation.

Grâce à ces optimisations, nous avons assisté à une nette amélioration du temps d'initialisation, qui prenait jusqu'à 16 minutes, et qui ne prend plus que 45 secondes environ. Les temps d'exécutions sont reportés dans le chapitre .

3.8 Exécution

Comme souvent répété dans les forums spécialisés sur Internet, la courbe d'apprentissage de Repast est assez élevée. Il a fallu douze mois de développement intensif pour maîtriser la plateforme d'une manière satisfaisante et arriver au résultat que nous nous étions fixé. La figure 3.16 présente l'interface graphique après le chargement des réseaux de transport et l'initialisation de toute l'application.

À gauche de l'interface, l'utilisateur peut définir l'ensemble des paramètres du modèle. L'interface du réseau en elle-même est composée du réseau piéton (en gris) et du réseau de transport en commun (en violet). Un autre fond de carte, représentant les zones urbaines, les bâtiments, les espaces verts etc. est également intégré dans le simulateur. Il provient de *shapefiles* gratuites disponibles sur Internet . Pendant l'exécution, les véhicules de transport en commun sont dessinés sous la forme de rectangles bleus. Les voyageurs sont dessinés sous la forme de croix noires. Nous pouvons observer dans la figure 3.16 les voyageurs empruntant un bus, qui se charge de les faire bouger sur le réseau.

3.9 Conclusion

Dans ce chapitre, nous avons d'abord présenté la plateforme MMT afin d'expliquer le contexte de développement de notre simulateur. Puis, nous avons décrit les briques de base de la plateforme de développement Repast Symphony que nous avons choisie pour développer notre simulateur. Ensuite, une description détaillée de la conception du simulateur met l'accent sur ses paramètres et données, la représentation du réseau, la planification d'itinéraires, la méthode choisie pour gérer les déplacements des agents ainsi que les comportements de ses agents.

À ce niveau, le simulateur que nous avons développé assure la fonctionnalité de base qui est le déplacement de voyageurs sur un réseau de transport en commun. Or, pour répondre à notre objectif, qui est l'évaluation de l'impact de l'information des voyageurs sur le réseau particulièrement en cas de perturbation, notre simulateur nécessite l'intégration de l'information des voyageurs et les taux d'équipement des voyageurs. En effet, afin de mesurer l'impact de l'information des voyageurs, notre simulateur doit être capable de fournir aux voyageurs les informations nécessaires pour la préparation de leurs voyages et pour les guider pendant leurs déplacements à travers les informations personnalisées en fonction de leurs niveaux d'équipement. Le niveau d'équipement des voyageurs s'avère un paramètre principal dans le transport

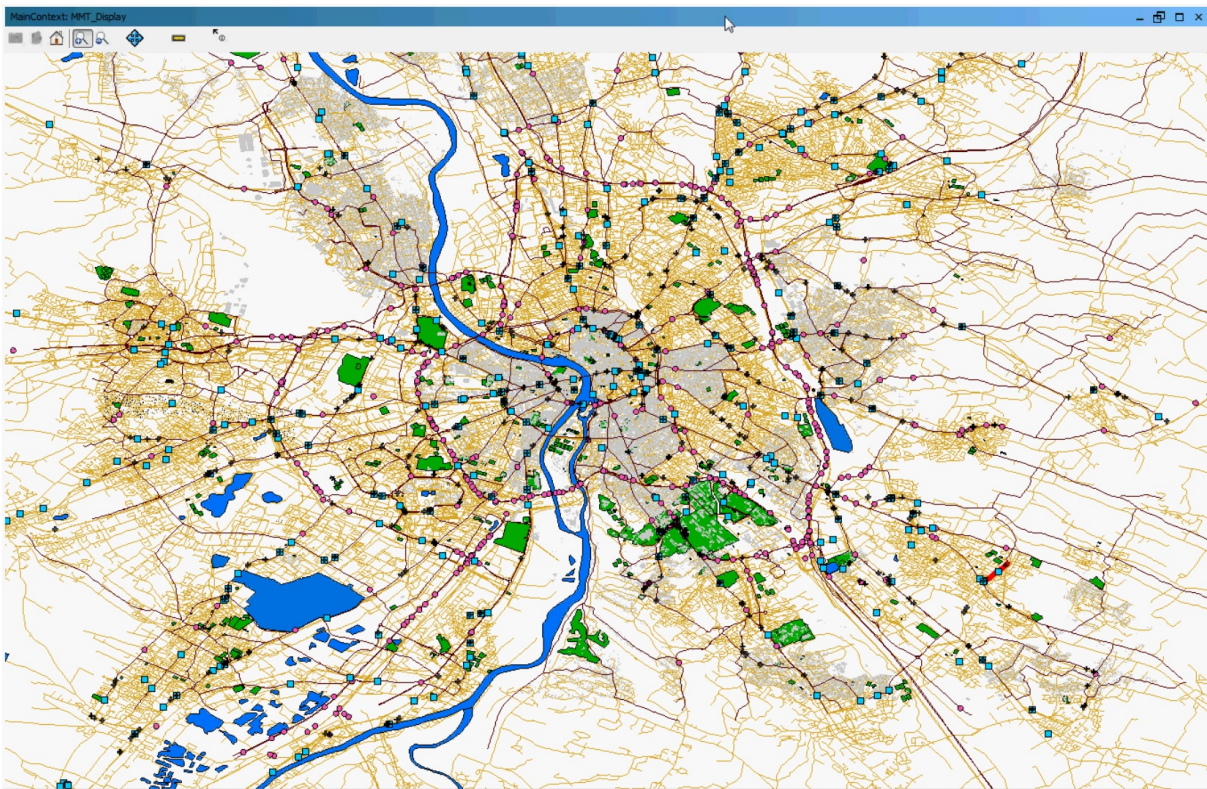


FIGURE 3.16 – Exécution de la simulation

moderne qui nécessite d'être pris en considération dans une telle étude. Dans le chapitre suivant, nous décrivons l'approche utilisée pour intégrer l'information des voyageurs dans notre simulateur.

Chapitre 4

L'impact de l'information des voyageurs sur les réseaux de transport en commun

Sommaire

4.1	Introduction	87
4.2	Gestion de l'information des voyageurs	88
4.2.1	Flux d'information des voyageurs	88
4.2.2	Flux d'information dans le simulateur	89
4.3	Système multi-agent	90
4.3.1	Agents voyageurs	90
4.3.2	Agents d'information locale	92
4.3.3	Service de planification	93
4.3.4	Agents véhicules	93
4.4	Les perturbations	93
4.4.1	Injection de perturbations	93
4.4.2	Impact des perturbations sur le comportement des agents véhicules	93
4.5	Représentation spatiotemporelle de l'environnement	94
4.5.1	Problématique	94
4.5.2	Proposition	95
4.5.3	Les perturbations dans l'environnement spatiotemporel	97
4.5.4	Usage de l'environnement spatiotemporel par les agents d'information locale	97
4.5.5	Usage de l'environnement spatiotemporel par le service de planification	98
4.6	Modèle temporel de la simulation	99
4.7	Conclusion	100

4.1 Introduction

Dans ce chapitre, nous décrivons notre méthode pour intégrer l'information des voyageurs dans le simulateur afin de pouvoir évaluer son impact sur le déplacement des agents voyageurs.

Dans le contexte du transport moderne, les quantités d'informations échangées entre les gestionnaires et les voyageurs s'avèrent importantes et proviennent de plusieurs sources géographiquement distribuées et interconnectées. Aussi, ces informations sont fortement dynamiques vu que l'état des réseaux est susceptible de changer très rapidement. Nous avons choisi de nous fonder sur le simulateur multi-agent du chapitre précédent pour vérifier l'impact de l'information des voyageurs sur les déplacements des voyageurs. En effet, la dynamique des informations, leurs différentes sources et natures (locales en diffusion, personnalisée, etc.) fait que l'analyse d'un tel système d'une manière statique et agrégée est une tâche très complexe. Il nous faut donc nous fonder sur un système dynamique et une représentation individualisée des voyageurs afin d'analyser finement le comportement des voyageurs face aux informations reçues.

Nous proposons des méthodes qui optimisent au mieux ces échanges d'information. Nous désirons représenter la fourniture d'informations temps-réel pour les voyageurs, et de plus personnalisées pour les voyageurs équipés. Ainsi, nous avons choisi d'utiliser l'environnement du système multi-agent comme support d'échange. Nous avons opté pour une représentation spatiotemporelle de cet environnement pour représenter l'état du réseau dans le temps. Cette méthode permet de suivre la dynamique du réseau tout en respectant l'aspect temporel de l'information et assure l'optimisation des échanges.

La suite de ce chapitre est structurée comme suit. La section 4.2 illustre notre méthode pour intégrer l'information des voyageurs dans notre simulateur. La section 4.3 décrit le comportement des agents additionnels de notre système, notamment les deux types d'agents voyageurs connectés et non-connectés et le rôle des informations perçues dans leurs processus de déplacement. La section 4.4 présente l'injection des perturbations sur le réseau et leur prise en compte dans le comportement des agents véhicules. La section 4.5 décrit notre méthode pour cibler les informations individuelles aux voyageurs connectés à travers leurs *smartphones*, en utilisant les réseaux espace-temps. Nous décrivons le modèle temporel de la simulation en section 4.6. Enfin, la section 4.7 est la conclusion de ce chapitre.

4.2 Gestion de l'information des voyageurs

Dans cette section, nous décrivons l'intégration de l'information des voyageurs dans le simulateur.

4.2.1 Flux d'information des voyageurs

Dans un système moderne d'information des voyageurs, il existe plusieurs types d'information des voyageurs.

1. Le premier type concerne les informations statiques sur la structure spatiale du réseau. Ces informations concernent la structuration du réseau en lignes, itinéraires et arrêts. Elles sont disponibles sous format papier ou sur les sites Internet des opérateurs concernés.
2. Le deuxième type concerne les informations statiques sur la structure temporelle du réseau, i.e. les tableaux de marche théoriques de véhicules de transport en commun. Cette information est généralement présente sous format papier aux arrêts de transport en commun ou sur les sites des opérateurs concernés.

3. Le troisième type d'information concerne les informations dynamiques sur les positions des véhicules de transport en commun. Il s'agit de modifications dynamiques des tableaux de marche des véhicules. Cette information est parfois disponible aux arrêts concernés du réseau et sur les sites Internet des opérateurs. Lorsque les modifications des tableaux de marche concernent les véhicules de toute une ligne ou de tout un itinéraire, ces modifications concernent la structure dynamique du réseau. Il s'agit du quatrième type d'information.
4. Le quatrième type d'information concerne la modification systématique des missions ou des courses des véhicules d'une ligne ou d'un itinéraire. Ces informations sont disponibles sur les arrêts du réseau, sur les sites Internet, mais également sur les autres arrêts du réseau, généralement les grands pôles d'échange, ou les arrêts de la même ligne que la partie du réseau touché.
5. Le cinquième type d'information concerne les informations dynamiques sur la structure spatiale ou temporelle du réseau, mais reçues de la part d'autres voyageurs. Ces échanges point à point peuvent survenir lorsque les voyageurs sont aux mêmes endroits par communication directe, ou via les réseaux sociaux, entre voyageurs de la même communauté (mêmes intérêts, mêmes lignes de transports habituels, mêmes zones origines et/ ou destinations).

Dans le cadre de cette thèse, nous nous intéressons aux quatre premiers types d'information, que nous désirons intégrer dans le simulateur. Le dernier type d'information fera l'objet de travaux futurs.

4.2.2 Flux d'information dans le simulateur

En l'état, il existe peu de flux d'informations dans le simulateur. Les interactions dans le simulateur sont de deux types :

1. Les agents voyageurs communiquent avec le service de planification afin d'obtenir leurs itinéraires.
2. Les agents véhicules perçoivent les agents voyageurs aux arrêts pour décider s'ils peuvent les embarquer à bord.

Afin de simuler des scénarios perturbés, et d'intégrer les quatre types d'information listés plus haut, il faut doter le simulateur des capacités suivantes :

1. Les agents véhicules doivent être capables d'adapter leurs mouvements à l'état du réseau. Ils doivent donc être capables de percevoir des changements de vitesses sur les arcs qu'ils empruntent et adapter leur déplacement en conséquence. Ces changements de vitesses peuvent aller jusqu'à la rupture totale (en cas de panne ou de crise).
2. Les agents voyageurs doivent pouvoir être notifiés des changements survenus sur leur itinéraire et adapter leurs plans en conséquence.

Concernant la deuxième capacité, les voyageurs peuvent être notifiés de différentes manières des modifications survenues sur leur environnement spatiotemporel. Selon qu'ils soient connectés à une plateforme d'information temps-réel ou non, les voyageurs se comportent d'une manière différente. C'est la raison pour laquelle nous créons deux types d'agents voyageurs. Les

agents voyageurs connectés disposent d'un service de *monitoring* de leurs itinéraires, leur permettant de disposer en permanence du meilleur itinéraire possible étant donné l'état courant du réseau. Les voyageurs non connectés quant à eux ne disposent que des informations statiques sur le réseau, mis à jour avec les informations reçues localement pendant leur voyage.

Afin d'intégrer la diffusion des informations sur le réseau de transport d'une manière réaliste et efficace, il faut répondre aux deux questions suivantes :

1. comment faire pour effectuer le suivi des itinéraires des voyageurs, et ne les garder informés que des changements survenus sur les parties du réseau qui les intéressent ?
2. comment faire pour représenter les flux d'information locale, leurs mises à jour et leur diffusion aux seuls agents voyageurs qui peuvent y accéder (contrainte spatiotemporelle) ?

Afin de répondre à ces deux questions, nous avons procédé à la modification du comportement du service de planification, et à l'intégration d'un nouveau type d'« agent d'information locale ». L'agent d'information locale est responsable de l'information des agents voyageurs dans son voisinage. Il doit donc disposer des informations pertinentes à diffuser et avoir accès aux agents voyageurs pertinents.

4.3 Système multi-agent

4.3.1 Agents voyageurs

Les agents voyageurs représentent les usagers du réseau de transport en commun. Nous définissons deux types d'agents voyageurs, selon leur équipement en *smartphones* et leur connexion à une source d'information temps-réel ou non.

Les voyageurs non connectés

Les voyageurs non connectés à un système d'information temps réel fondent leur calcul sur une vue statique du réseau. Leur connaissance est donc faite du graphe de transport avec un ensemble de nœuds et un ensemble d'arcs. L'agent voyageur connaît également les itinéraires k des véhicules et leur fréquence moyenne théorique $\phi(k)$ telle que définie dans le chapitre précédent ainsi que les temps de parcours théoriques moyens entre les arrêts. Cette représentation de l'environnement provient de la définition statique des réseaux de transport en commun telle que décrite par les opérateurs de transport. Elle représenterait les connaissances d'un voyageur en visite courte dans une ville, ou un voyageur habitué qui ne regarde pas ou plus les sources d'informations de transport temps-réel. Comme le montre la figure 4.1, l'agent a comme connaissance du monde un sous-ensemble du réseau statique de transport.

L'agent non connecté fonde son calcul de plus court chemin sur cette vue statique et partielle du réseau. Une fois son itinéraire calculé, il est complètement autonome. Il va attendre les véhicules dans les arrêts planifiés et ne va changer d'itinéraire que lorsqu'il est bloqué dans une perturbation (retard ou déconnexion de ligne) ou lorsqu'il reçoit une information locale (de la part d'un agent d'information locale, voir plus loin) à propos d'une perturbation.

Quand un agent non connecté reçoit une information locale concernant une perturbation sur le réseau, il déduit une approximation du nouvel état du réseau (cf. Figure 4.2). Il s'agit d'une

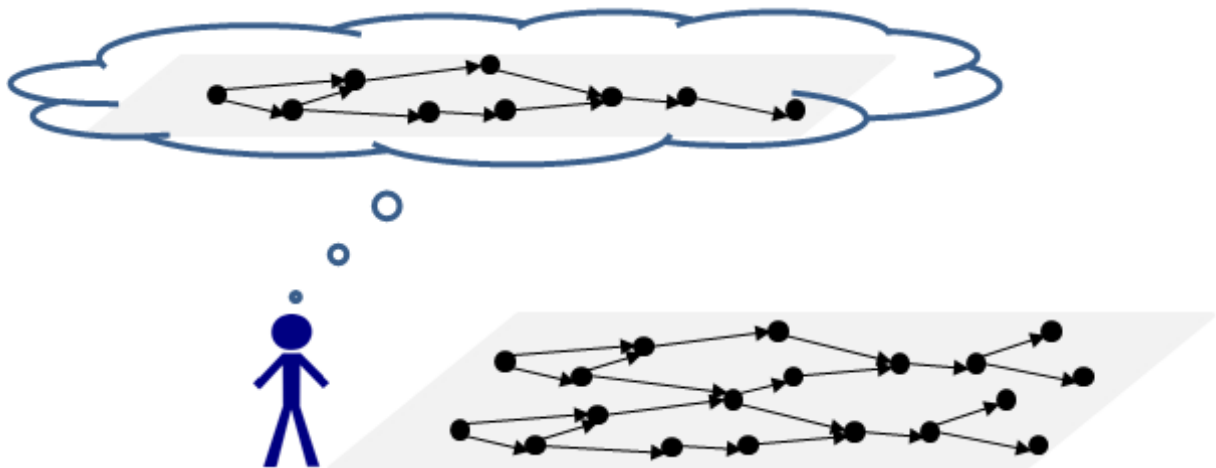


FIGURE 4.1 – La représentation mentale d'un voyageur non connecté

approximation car l'information ne concerne qu'un sous-ensemble du réseau. D'autres perturbations peuvent avoir lieu ailleurs sur le réseau, mais sa représentation mentale du réseau ne le reflète pas car il n'a pas l'information. En appliquant les modifications reçues à sa représentation mentale - et statique - du réseau de transport, l'agent non connecté calcule un plus court chemin fondé sur cette représentation.

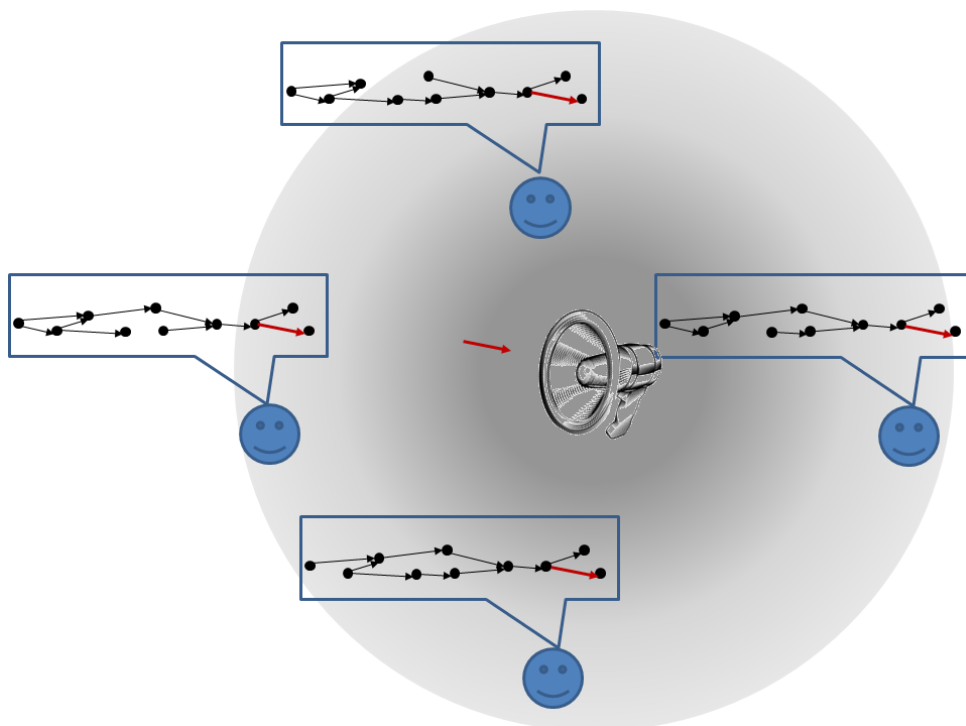


FIGURE 4.2 – L'impact des informations locales sur les voyageurs non connectés

Ainsi, les voyageurs non connectés n'interagissent pas avec le service de planification comme

le faisaient les agents voyageurs dans le simulateur. Tous leurs calculs sont fondés sur des connaissances locales sur une vision propre du réseau de transport en commun. Les agents voyageurs non connectés ne partagent pas tous la même perception du réseau de transport en commun. Même si la connaissance initiale du réseau de départ est la même, les voyageurs se forgent une vision propre du réseau, résultat de leur mouvement et de leur interaction locale avec les informations diffusées tout au long du parcours. A moins que les voyageurs aient une même origine, une même destination et qu'ils partent au même moment, la vision dynamique du réseau par les agents voyageurs est différente. Contrairement aux agents voyageurs connectés que nous décrivons dans la section suivante.

Les voyageurs connectés

Les voyageurs connectés sont en interaction en permanence avec une source d'information temps-réel. Leurs itinéraires sont suivis en permanence et de nouveaux itinéraires leur sont proposés lorsqu'une perturbation qui concerne leur chemin est détectée. Pour simuler ce type d'agents voyageurs, nous réutilisons le service de planification du simulateur, pour lequel nous ajoutons une fonction de *monitoring* des itinéraires. Quand les passagers sont connectés, ils ont des préférences qui définissent, entre autres, l'écart temporel accepté entre leur itinéraire calculé et leur situation réelle. Lorsque l'écart observé par le voyageur connecté dépasse ce seuil, le service de planification lui recalcule un nouvel itinéraire. Le service de planification envoie au voyageur connecté un itinéraire composé d'une liste de coordonnées. L'itinéraire de chaque voyageur connecté est donc suivi. L'objectif du suivi est de demander un nouvel itinéraire pour l'agent dans deux cas :

1. la position réelle de l'agent est différente de la position planifiée avec un certain écart Δ_a (défini dans les préférences de l'agent) ;
2. il y a un événement sur le réseau de transport qui a un impact sur l'itinéraire courant du voyageur. Le voyageur dans ce cas n'a pas encore perçu les effets de la perturbation, mais il les percevra s'il ne change pas de chemin.

Le premier cas est donc traité par le comportement des agents, puisque autant la position de l'agent que son itinéraire planifié font partie de ses connaissances. L'agent voyageur demandera un nouvel itinéraire s'il remarque un écart par rapport au trajet planifié. Le deuxième cas est traité par le service de planification à travers la fonction de *monitoring* : le service de planification s'occupe de rester à l'écoute des seules perturbations qui concernent l'agent voyageur connecté. Le détail de cette écoute et de cette fonction de *monitoring* sera détaillé plus loin.

4.3.2 Agents d'information locale

Les agents d'information locale représentent des appareils qui diffusent de l'information trafic sur les écrans ou via des annonces vocales au niveau des arrêts et des stations du réseau. L'information diffusée par les agents d'information locale concernent des événements survenant dans le réseau, ils doivent donc en être informés, comme les agents connectés.

4.3.3 Service de planification

Le service de planification répond aux requêtes de déplacement des agents voyageurs connectés. Il est responsable de calculer le meilleur itinéraire pour cet agent voyageur. La méthode de calcul des itinéraires est la même que celle décrite dans le chapitre précédent. Il devient également responsable du *monitoring* des itinéraires des voyageurs connectés.

4.3.4 Agents véhicules

Les agents véhicules tels que décrits dans le chapitre précédent calculent une fois pour toutes leurs itinéraires, les coordonnées géographiques qu'ils doivent avoir ainsi que la succession de vitesses qu'ils adopteront au début de la simulation. Dans ce chapitre, les agents véhicules voient survenir des perturbations dans leurs parcours, et cela doit être reflété dans leur comportement. Dans la sous-section suivante, nous décrivons la manière avec laquelle les perturbations sont injectées dans la simulation et l'impact que cela a sur le comportement des agents véhicules.

4.4 Les perturbations

4.4.1 Injection de perturbations

Pour valider l'impact de l'information temps-réel, autant individuelle qu'au niveau des arrêts, nous devons modéliser les perturbations. En effet, l'information devient très importante dans le cas de perturbations, puisqu'il s'agit dans ces cas de diriger les flux passagers sur les réseaux. Les perturbations sont exclusivement injectés en tant que modifications dynamiques des tableaux de marche des véhicules. En effet :

1. Les retards sont injectés dans le modèle en ajoutant du temps additionnels aux temps de passage aux arrêts.
2. Les ruptures ou les pannes sont modélisées par une suppression de la partie en aval de l'endroit concerné de la mission de l'agent véhicule.
3. Pour modéliser la rupture sur toute une ligne ou un itinéraire, les tableaux de marche des véhicules restants sont tous supprimés, en aval de l'endroit concerné par la rupture.

Les fréquences $\phi(k)$ des itinéraires k sont également modifiées en conséquence. Le graphe logique représentant l'état du réseau de transport est modifié en conséquence, comme spécifié dans le chapitre précédent. De cette manière, le service de planification peut effectuer ces calculs sur le bon état du réseau.

4.4.2 Impact des perturbations sur le comportement des agents véhicules

Les agents véhicules calculent au début de la simulation un itinéraire, des temps de passages, une séquence de coordonnées géographiques et des vitesses. Cependant, contrairement au chapitre précédent, les agents véhicules ne se contentent pas de suivre leur plan du début à la fin de la simulation. En effet, les tableaux de marche ayant servi au calcul de leurs itinéraires

initiaux sont amenés à changer comme nous venons de le décrire plus haut. Les agents véhicules, à chaque pas de simulation vérifient si les temps de visites des arrêts restant dans leur tableau de marche ont été modifiés. Si tel est le cas, le reste des vitesses et temps de passages sont recalculés. S'il s'agit d'une interruption, l'agent véhicule délétera l'ensemble des agents voyageurs à son bord au dernier point du réseau précédant l'interruption. Les échanges d'information concernant ces retards et interruptions, afin de les transférer aux bons agents (connectés et non connectés), sont décrits dans les sections qui suivent.

4.5 Représentation spatiotemporelle de l'environnement

La notification des perturbations dans les réseaux de transport en commun et la re planification des trajets des voyageurs est une tâche complexe. En effet, les changements dans les itinéraires et temps de parcours sont exogènes et les agents ne savent pas *a priori* où elles peuvent survenir. Nous avons donc d'une part les fournisseurs d'information géolocalisée temps réel sur l'état des réseaux et d'autre part des agents voyageurs potentiellement intéressés par cette information.

4.5.1 Problématique

Ce problème de mise en relation dynamique des agents est connu sous le nom de « problème de connexion » [Davis and Smith1983] : comment trouver les agents qui pourraient avoir les informations ou les capacités recherchées. Dans notre simulation, les agents rejoignent et quittent le système librement et leurs besoins en informations dépendent de leur parcours. Ces besoins sont donc variables par essence. Pour pouvoir gérer cette variabilité, deux familles d'approches ont été adoptées dans la littérature. D'une part, des auteurs ont proposé une gestion centralisée des échanges. Elle consiste en la délégation de la mise en relation des agents à un type spécial d'agents [Payne *et al.*2002, Wong and Sycara2000, Decker *et al.*1997]. Ces agents responsables de la mise en relation, et qui ne sont ni des demandeurs ni des fournisseurs, sont appelés les agents intermédiaires (*middle agents*). Suivant la nature des fonctions qui lui sont attribuées, ces agents peuvent être :

- un facilitateur qui connaît les caractéristiques de tous les agents du SMA et qui tient cette connaissance à disposition des autres agents en fournissant un service de pages jaunes.
- un courtier (*broker*) qui joue le rôle d'intermédiaire entre les agents du SMA et par lequel passe toute recherche de partenaires pour une coopération.

Dans les deux cas, tout agent doit connaître l'intermédiaire, et être connu de celui-ci par une représentation de ses caractéristiques, pour avoir accès à des connaissances sur les autres agents, nécessaires au raisonnement social ou à la mise en place de coopération. Les modifications de l'état du SMA peuvent détruire certaines propriétés de la connaissance de l'intermédiaire. Celui-ci doit alors gérer cette modification de manière à retrouver les propriétés perdues. Tout agent sujet à un ajout, retrait ou à une évolution doit signaler cette modification à l'intermédiaire. Les avantages de ce mode de gestion de l'ouverture tient principalement à sa simplicité de conception et de mise en oeuvre. Cependant, la centralisation de cette gestion présente des inconvénients : l'agent intermédiaire est indispensable à la recherche de partenaires pour coopérer et risque d'être très sollicité par les agents d'où une surcharge de communication à traiter ce

qui nuit aux performances du système.

D'autre part, d'autres auteurs proposent une gestion distribuée des échanges pour le problème de connexion [Vercouter2001]. Le principe est fondé sur la notion de recommandation. En effet, les agents s'échangent mutuellement des recommandations parmi leurs accointances, sans avoir besoin d'un agent intermédiaire. Cette approche permet la constitution progressive et distribuée d'un carnet d'adresses aux agents.

Cependant, ni l'une ni l'autre des deux approches ne convient, telle quelle, à notre problématique. Dans ces propositions, la mise en relation est fondée sur la compatibilité de capacité ou de préférences entre agents. Or, dans le cas des informations voyageurs, le problème n'est pas de retrouver l'émetteur d'une information, puisque les sources d'information sont connues (les opérateurs de transport) et c'est le contenu de l'information échangée qui doit être géré. En effet, l'intérêt porté à une information dans le simulateur ne dépend pas de sa nature mais de sa valeur. C'est également le cas, par exemple, dans le cadre de la gestion des informations dans un poste de régulation aérienne, où il a été montré [Cahour and Salembier1996] qu'une partie des interactions relevait du concept de l'écoute flottante ou *mutual awareness* [Dugdale *et al.*2000, Balbo and Zargayouna2004]. Dans ce contexte, les opérateurs ont l'ensemble des informations du poste de régulation à leur disposition et ils dirigent leur attention en fonction de l'intérêt d'une information par rapport à leur activité. Notre problématique d'information des voyageurs dans le cadre d'une simulation des déplacements s'inscrit dans ce cadre.

4.5.2 Proposition

Notre proposition est une extension des travaux sur la représentation explicite de l'environnement multi-agent (e.g. [Saunier *et al.*2006, Weyns *et al.*2004, Weyns and Michel2014, Balbo2000, Saunier2008]). Il s'agit également d'une extension des travaux sur les graphes spatio-temporels pour les problématiques de transport dynamique (e.g. [Zargayouna2005], [Zargayouna2007] et [Zargayouna and Zeddini2012]). Le principe général est celui de la mise en commun d'informations puis d'une sélection locale par les agents en fonction de leur activité. Cette méthode a également été testée dans le cadre de la résolution de problèmes [Chevaillier *et al.*2003]. L'application choisie concerne la gestion distribuée d'emploi du temps et la mise en commun des informations est réalisée par diffusion. Les différents tests réalisés montrent que si les objectifs de résolution du problème sont atteints, le recours aux diffusions pénalise fortement l'efficacité de la proposition [Balbo and Zargayouna2004]. Dans notre cas également, la diffusion de toute l'information trafic à tous les agents voyageurs est une méthode simple et intuitive pour la mise en relation des demandeurs et fournisseurs d'informations, mais elle est coûteuse car générant des échanges et des traitements inutiles.

Nous adoptons donc une approche centrée-environnement, car elle se focalise sur les données partagées et permet des sélections sur celles-ci sans avoir besoin de connaître ou de maintenir une connaissance sur les émetteurs de ces données. Cette représentation a été utilisée par le passé dans différentes applications : le transport à la demande, les tournées de véhicules, la gestion de crise, etc. [Zargayouna2005, Zargayouna and Zeddini2012]. Dans notre application, l'environnement spatio-temporel multi-agent représente l'état du réseau de transport en commun dans le temps. Cet environnement est l'interlocuteur privilégié des agents voyageurs et est actif, dans le sens où il sauvegarde des informations et réagit en fonction de leur dynamique.

Soit le réseau de transport $G = (V, E)$, avec l'ensemble de nœuds $V = \{(v_i)\}, i =$

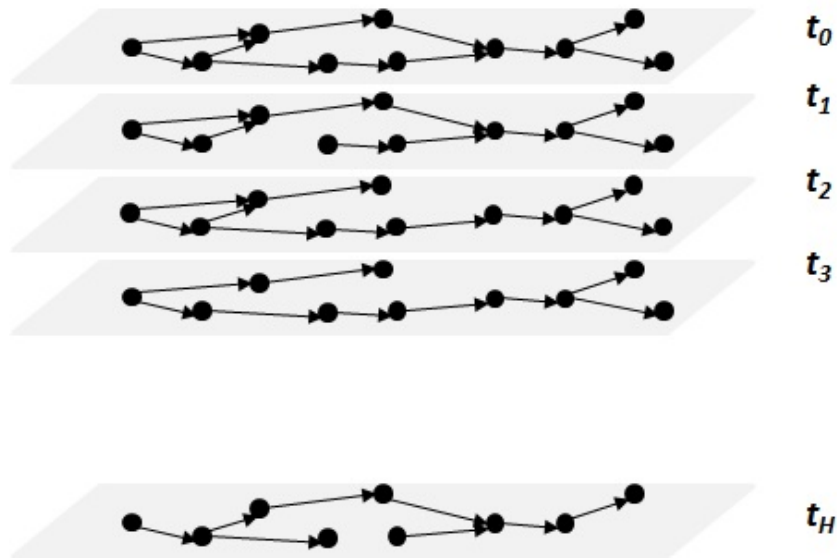
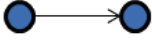


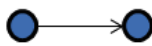
FIGURE 4.3 – Le graphe spatiotemporel

$\{0, \dots, N\}$ et un ensemble d'arcs $E = \{(v_i, v_j) | v_i \in V, v_j \in V, v_i \neq v_j\}$. Soit deux matrices $D = \{(d_{ij})\}$ et $T = \{(t_{ij})\}$ de coûts, de dimensions $N \times N$ (l'arc (v_i, v_j) a une distance de d_{ij} et un temps de parcours de t_{ij}). La représentation spatiotemporelle de l'environnement multi-agent est faite d'une duplication de G , H fois, avec H le temps maximal permis par l'application considérée : $G(t) = (N(t), E(t))$, avec $N(t)$ un ensemble de nœuds au temps t et $E(t)$ un ensemble de liens orientés au temps t , et $0 \leq t \leq H$ (cf. Figure 4.3). Les copies temporelles de G ne sont pas nécessairement identiques. En effet, nous pouvons avoir différents temps de parcours entre deux copies afin de refléter l'état du trafic. Des nœuds peuvent être présents dans une copie et absents dans une autre afin de refléter l'expansion d'une situation de crise. Les arcs peuvent également être absents afin de refléter les tableaux de marche des véhicules de transport en commun.

Ce réseau espace-temps représente à tout moment l'état actuel du réseau : les positions des véhicules de transport en commun, ainsi que leurs trajets planifiés et mis à jour. Un agent qui désirerait être informé des seuls événements survenant à un arc (v_1, v_2) du réseau pendant une période allant de t_1 à t_2 doit s'inscrire auprès des arcs $\{((v_1, v_2), t_1), \dots, ((v_1, v_2), t_2)\}$. Comme nous le montrons dans la suite, avec cette représentation de l'environnement, toute modification dynamique de l'offre de transport est répercutée directement sur les seuls agents qui sont concernés, ce qui évite les diffusions massives d'information à tous les agents du système.

Dans le simulateur, nous avons implémenté le réseau espace-temps sous la forme d'une table de correspondances (cf. Figure. 4.4). La clé d'accès à la table est un arc du graphe spatial. La valeur correspondant à l'arc est une liste triée d'instant, correspondant à des temps de démarrage des véhicules au nœud origine de cet arc. A chaque valeur de temps est associée un couple $\langle \text{véhicule}, \langle \text{agents} \rangle \rangle$, correspondant au véhicule concerné et tous les agents inscrits à l'arc espace-temps correspondant. Lorsque le tableau de marche d'un véhicule est modifié, les couples $\langle \text{véhicule}, \langle \text{agents} \rangle \rangle$ concernés sont retrouvés dans la table de correspondance, et tous

Arc	Temps ↑	< véhicule, < voyageurs,... >>
	t1	< véhicule1, < voyageur1, voyageur3,...>>

	t2	< véhicule2, < voyageur4, voyageur8,...>>

...

FIGURE 4.4 – L'implémentation du graphe spatiotemporel dans le simulateur

les agents sont notifiés du changement.

4.5.3 Les perturbations dans l'environnement spatiotemporel

Les perturbations sont modélisées en modifiant la représentation spatiotemporelle de l'environnement, en corollaire de la modification des tableaux de marche des agents véhicules. En effet, les retards sont injectés dans le modèle en modifiant dynamiquement les temps de passage des véhicules. Soit l'arc $((v_1, v_2), t)$ représentant le passage d'un véhicule au nœud v_1 en direction du nœud v_2 à l'instant t . Si le véhicule a un retard de Δ_t , l'arc $((v_1, v_2), t)$ est supprimé et remplacé par l'arc $((v_1, v_2), t + \Delta_t)$. Les pannes et les déconnexions de tout ou une partie d'une ligne sont modélisées également en supprimant l'ensemble des arcs spatiotemporels correspondants. Comme nous le verrons plus bas, dès qu'un arc spatiotemporel est modifié, l'information est diffusée à tous les agents d'information locale pertinents ainsi qu'aux voyageurs connectés concernés.

4.5.4 Usage de l'environnement spatiotemporel par les agents d'information locale

Les agents d'information locale s'inscrivent auprès de tous les arcs espace-temps des lignes qui passent par leurs arrêts. En effet, nous faisons le choix, réaliste, de ne diffuser au niveau des arrêts que les événements concernant les lignes passant par ces arrêts et non tous les événements survenant sur le réseau de transport. Autrement, la masse d'information aux arrêts deviendra vite difficile à diffuser sur les écrans ou par des annonces vocales. Ces informations dans les arrêts ne peuvent être perçues par un agent voyageur que s'il se trouve dans la station ou l'arrêt

concerné. Dans les simulations en cours, un seul agent d'information locale est présent par arrêt du réseau, et les informations qu'il diffuse sont perçues par tout agent voyageur en transit dans cet arrêt.

La délivrance de ces informations aux agents voyageurs passant par l'arrêt suit le même principe que celui de l'attente des voyageurs des véhicules au niveau des arrêts. En effet, l'information perçue par l'agent d'information locale est stockée comme un attribut de l'arrêt concerné. Les agents voyageurs, arrivés au niveau de l'arrêt, consultent les informations qui y sont stockées.

4.5.5 Usage de l'environnement spatiotemporel par le service de planification

Le service de planification est responsable de l'information des voyageurs connectés. Pour être au courant des seuls événements qui le concernent, le service de planification inscrit l'agent voyageur aux seuls arcs spatiotemporels de l'environnement qui forment son itinéraire (Figure 4.5). Quand le temps de passage ou le temps de parcours change, le nouveau temps de passage et de parcours est diffusé aux voyageurs connectés concernés. Le processus de planification est donc lancé avec le nouvel état du réseau.

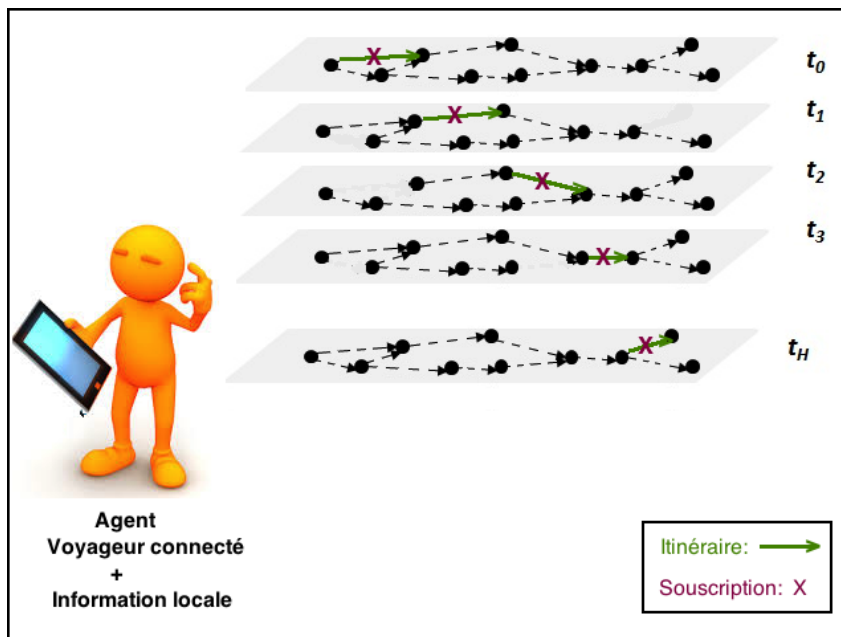


FIGURE 4.5 – La souscription de l'agent voyageur connecté sur le graphe spatiotemporel.

Il existe donc deux types de réseau dans notre simulation. Le premier type est un graphe spatial utilisé comme représentation mentale des voyageurs non connectés. Il est mis à jour par l'expérience du voyageur et les informations locales reçues. Il existe autant de graphes de ce type que de voyageurs non connectés dans le système. Le second type est le graphe spatiotemporel. Il en existe une seule copie. Ce graphe est utilisé par le service de planification

(pour le compte des agents voyageurs connectés), par les agents d'information locale et par l'opérateur qui y injecte les perturbations quand elles surviennent (Figure 4.6).

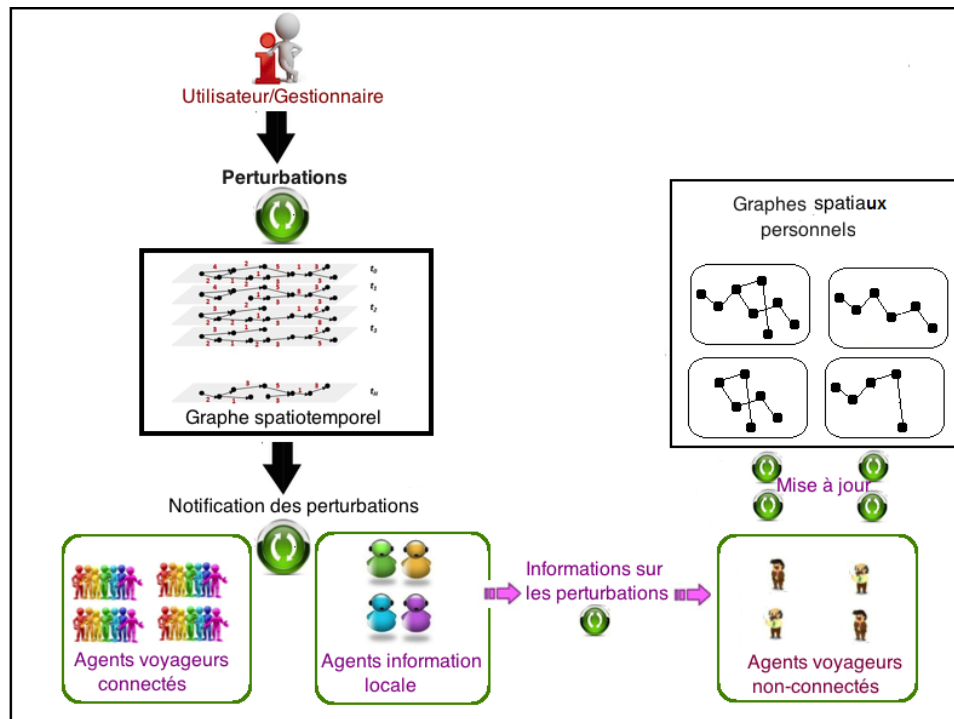


FIGURE 4.6 – Les graphes du simulateur.

4.6 Modèle temporel de la simulation

L'exécution concurrente des agents nécessite la synchronisation de leurs accès à l'environnement de transport, afin d'éviter des états incohérents du réseau dus à des tentatives de modifications simultanées. Ainsi, l'appel de la méthode de déplacement dans la géographie est séquentielisé, les agents voyageurs et véhicules, une fois leur décision de déplacement prise, se déplacent donc un par un.

D'autres considérations quant à l'ordre d'exécution des agents sont également à prendre en compte. En effet, les principales interactions entre agents sont effectués entre :

- les agents voyageurs et les agents véhicules lors de la montée dans les arrêts ;
- les agents connectés et le service de planification ;
- les agents d'information locale et les agents voyageurs dans les arrêts.

Puisque les véhicules arrivent au niveau des arrêts et recherchent les agents voyageurs désirant monter à bord, il y a un risque qu'un agent voyageur désirant effectivement prendre ce véhicule ne soit pas encore à l'arrêt, non pas parce qu'il est en retard, mais parce que l'ordonnanceur ne lui a pas encore donné la main dans le même pas de temps simulé. Pour cette raison, les agents voyageurs sont exécutés en priorité, avant les agents véhicules.

Des incohérences de réaction des voyageurs aux perturbations peuvent également survenir. Par exemple, nous pourrions avoir des voyageurs qui réagissent immédiatement aux perturbations, alors que d'autres ne réagissent qu'au pas de simulation suivant, car l'injection de la perturbation aura été effectué au milieu de l'exécution des agents pour un pas de temps donné. Pour éviter cette situation, les perturbations et leur impact sur le réseau espace-temps, sur le service de planification, ainsi que sur les agents d'information locale sont effectués avant l'exécution des agents voyageurs et véhicules.

4.7 Conclusion

Dans ce chapitre, nous avons décrit les enrichissements nécessaires au simulateur pour intégrer les flux d'informations sur les réseaux de transport en commun. Nous y avons également décrit les changements introduits dans le système multi-agent et les comportements des voyageurs afin de mieux représenter l'impact de l'information temps-réel, en diffusion ou personnalisée, sur l'état dynamique des réseaux.

Dans le chapitre suivant, nous donnons les résultats des simulations que nous avons tournées et nous en donnons notre interprétation.

Chapitre 5

Résultats

Sommaire

5.1	Introduction	101
5.2	Configuration	101
5.2.1	Environnement informatique	101
5.2.2	Données en entrée	102
5.2.3	Injection des perturbations	102
5.2.4	Tirage aléatoire selon le type d'agent	102
5.3	Scénarios	104
5.4	Résultats	106
5.5	Temps d'exécution	110
5.6	Discussion	114
5.7	Conclusion	116

5.1 Introduction

L'objectif de ce chapitre est de tester l'utilisation du simulateur défini dans le chapitre 3 et enrichi par la couche informationnelle du chapitre 4 dans un contexte réaliste de déplacement. Ce chapitre sera aussi l'occasion de comparer les interprétations et les conclusions des approches de la littérature avec notre travail. Comme nous l'avons énoncé dans le chapitre 3, les données concernant le réseau de transport en commun que nous utilisons proviennent de Tisséo SMTC, l'autorité organisatrice des transports publics Toulousains, avec lesquels nous avons un partenariat de recherche.

5.2 Configuration

5.2.1 Environnement informatique

Nos expérimentations ont été exécutées sur un ordinateur tournant sous Windows 7 Professional™ 64 bits, avec 12 cœurs (2 microprocesseurs de 6 cœurs chacun) tournant à 2,7 GHz. L'ordinateur est équipé d'une mémoire vive de 50 Go.

5.2.2 Données en entrée

Nous exécutons nos expérimentations sur la ville de Toulouse (cf. Figure 5.1), pour laquelle nous avons des données détaillées, y compris un modèle de déplacement [Othman *et al.*2014]. Nous avons considéré le réseau de transport public de Toulouse avec 80 lignes, 359 itinéraires et 3.887 arcs. Dans nos simulations courantes, notre système multi-agent est fait de 18.180 bus¹³ et de 5.000 à 30.000 passagers. Nous injectons des perturbations d'une manière stochastique dans la simulation. Le nombre de ticks par simulation est défini à 5.000, pour une journée de 6h du matin à 2h du matin le lendemain. Chaque tick de temps simulé correspond donc à 14 secondes environ.

5.2.3 Injection des perturbations

Nous avons donc créé des perturbations aléatoires sur le réseau de transport en commun afin de valider l'impact de ces événements sur les différentes catégories de voyageurs. Pour chaque simulation, nous avons généré cinq coupures d'arcs aléatoires sur le réseau de transport pendant toute la simulation (une coupure toutes les 233 minutes de temps réel environ, en moyenne). Chaque coupure dure pendant 250 ticks de temps simulé (un peu moins d'une heure de temps réel). Les coupures des arcs font que certains voyageurs ne retrouvent plus d'itinéraire vers leur destination. Ces voyageurs ne sont pas considérés dans les résultats de simulation qui suivent¹⁴. Les perturbations sont aléatoires, mais ne peuvent concerner qu'un certain nombre d'arcs que nous avons jugés intéressants à couper. Les arcs que nous jugeons intéressants sont ceux par lesquels passent cinq itinéraires différents ou plus. Cela pour éviter de couper des arcs en bout de ligne avec très peu de voyageurs connectés. Les cinq arcs coupés aléatoirement pendant la durée de la simulation sont donc tirés parmi les 21 arcs candidats qui obéissent à cette contrainte.

5.2.4 Tirage aléatoire selon le type d'agent

Les origines et les destinations des voyageurs sont tirées aléatoirement, d'une manière cohérente par rapport au modèle de déplacement de la ville. Cela signifie que le nombre de voyageurs partant d'une zone vers une autre est toujours le même pour une simulation, mais que les nœuds spécifiques d'origine et de destination sont différents à chaque simulation.

Nous générons nos agents au fur et à mesure de l'avancement de la simulation. Selon la configuration, nous devons générer un certain nombre de voyageurs de chaque type. Il faut éviter de générer trop d'agents d'un type ou d'un autre. Idéalement, il faudrait qu'à tout moment de la simulation, nous ayons plus ou moins la même proportion dictée par les paramètres d'agents connectés et non connectés. Afin de tourner nos simulations avec le bon nombre d'agents voyageurs connectés et non connectés, tout en gardant l'aspect stochastique des tirages, nous procédons ainsi.

Soit $p_{connect}$ le paramètre de la simulation définissant le pourcentage de voyageurs connectés et $p_{non-connect}$ le paramètre définissant le pourcentage de voyageurs non connectés. Les deux

13. Dans nos simulations, nous créons un bus par mission.

14. Le pourcentage de voyageurs sans itinéraire est stable, autour de 5% sur toutes les simulations que nous avons tournées

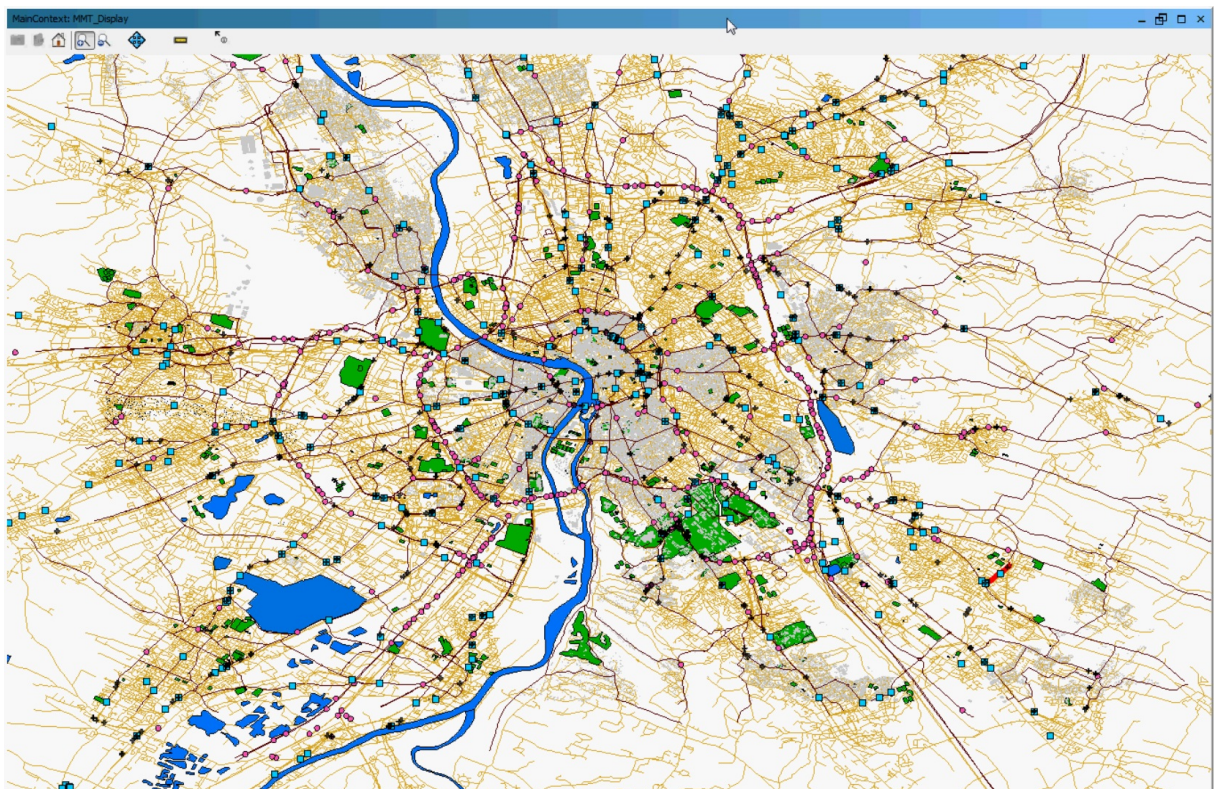


FIGURE 5.1 – Exécution d'une simulation

variables $p_{connect}$ et $p_{non-connect}$ seront en permanence mis à jour tout au long de la simulation pour refléter les proportions courantes de voyageurs connectés et non connectés dans la simulation. Au départ

$$p_{connect} = \frac{\text{nombre de voyageurs connectés restant à tirer}}{\text{nombre de voyageurs restant à tirer}}$$

A chaque pas de simulation, nous effectuons un certain nombre de tirages aléatoires d'un nombre $\rho \in [0, \dots, 1]$. Si $\rho \in [0, \dots, p_{connect}]$, l'agent voyageur créé sera connecté, sinon il sera déconnecté. Une fois le choix effectué, $p_{connect}$ et $p_{non-connect}$ seront mis à jour selon la formule. Comme le montre la Figure 5.2 représentant une exécution exemple, cette manière de procéder nous garantit d'avoir en permanence la bonne proportion d'agents voyageurs connectés et non connectés tout au long de la simulation. Dans la figure, la proportion cible est de 20% – 80%. A la toute fin de simulation, le nombre d'agent restant à tirer devient trop faible pour pouvoir garder la bonne proportion.

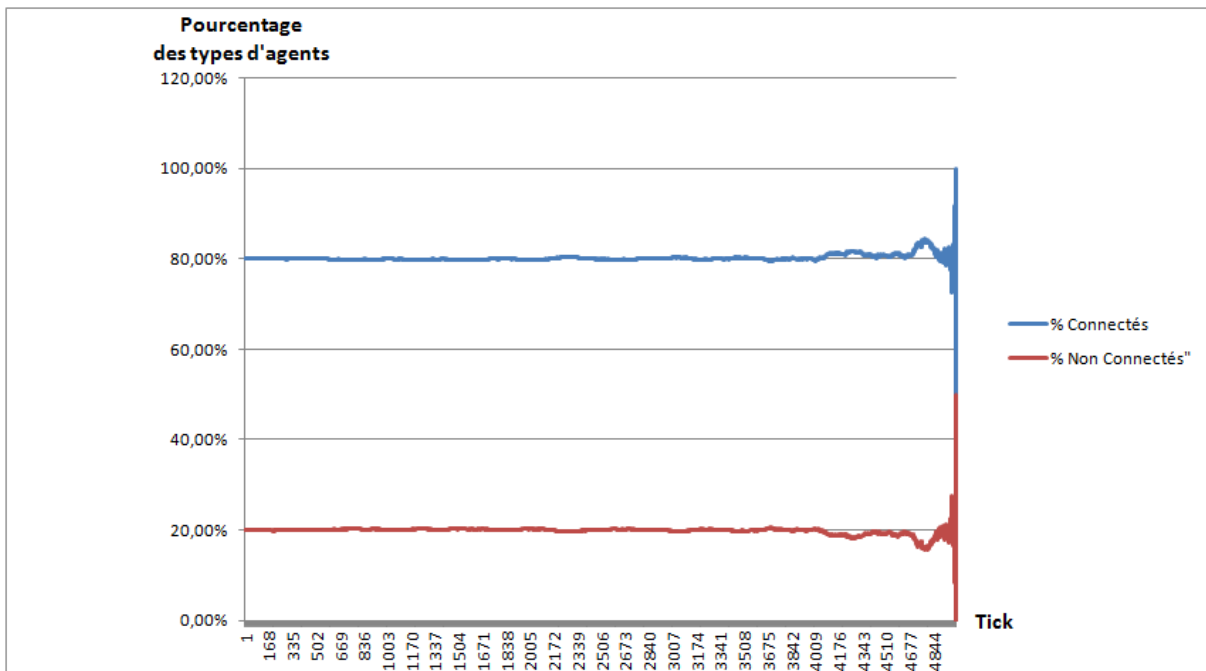


FIGURE 5.2 – Génération des agents de chaque type

5.3 Scénarios

Nous exécutons six scénarios différents, avec cinq ensembles d'agents voyageurs pour chaque scénario (cf. table 5.1).

Scénarios	Informations locales	Informations personnalisées
1	Non	Non
2	Oui	Non
3	Oui	20%
4	Oui	50%
5	Oui	80%
6	Oui	100%

TABLE 5.1 – Les six scénarios en fonction des niveaux d’informations.

Le premier scénario est notre configuration dite de référence, à laquelle nous comparons toutes les autres. Dans ce scénario, il n’existe aucun type d’information pour les voyageurs (ni information locale, ni information personnalisée). Les voyageurs n’ont qu’une description statique du réseau et des tableaux de marche. Dans ce scénario, le voyageur suppose que sa ligne est coupée s’il attend dans un arrêt au-delà de 40 minutes (l’attente théorique maximale dans nos données est de 30 minutes). Dans ce cas, le voyageur met à jour sa représentation du réseau en supprimant l’itinéraire en question de sa représentation du réseau et recalcule un plus court chemin.

Les autres scénarios correspondent à une augmentation progressive du niveau d’information dans le réseau. Dans le deuxième scénario, il n’y a qu’une information locale qui est fournie aux voyageurs et pas d’information personnalisée (0% de voyageurs connectés). L’information n’est donc disponible que pour les passagers présents dans les arrêts concernés.

Les scénarios 3 à 6 représentent les situations où de l’information locale est donnée aux voyageurs, et l’information personnelle est donnée aux voyageurs connectés : 20%, 50%, 80% et 100% des voyageurs sont connectés dans ces quatre scénarios, respectivement.

Dans chacun de ces six scénarios, nous exécutons cinq simulations avec 1.000, 5.000, 10.000, 20.000 et 30.000 passagers. Pour chacune de ces 30 configurations, nous reportons le temps de parcours moyen des passagers.

Chacune de ces 30 configurations a été tournée 5 fois. L’objectif est de vérifier que les simulations ne comportent pas de biais, que les résultats sont fiables et que les interprétations que nous en faisons sont correctes. Nous reportons les résultats moyens ainsi que les écarts-types observés.

Les dépassements de capacité des bus est l’une des causes principales de retards pris par les voyageurs en cas de perturbations. En effet, le report des flux de voyageurs sur certains itinéraires fait que les capacités des bus peuvent ne plus suffire et donner lieu à des retards pris par les voyageurs car ils ne peuvent plus monter dans les véhicules. Les capacités des bus doivent donc être choisies avec précaution pour ne pas introduire un biais dans la simulation. En effet, si la capacité des véhicules est trop grande, les perturbations risquent d’influer marginalement les temps de parcours, surtout dans les réseaux fortement maillés. En revanche, une capacité de véhicules trop petite risque d’amplifier artificiellement l’effet des perturbations, en introduisant une trop grande attente au niveau des arrêts. Comme nous ne simulons pas le nombre réel des voyageurs sur le territoire considéré, nous ne pouvons pas considérer la capacité réelle des bus.

Ainsi, nous nous sommes fondés sur les chiffres annuels du réseau bus de Tisseo de l’année 2010 [Tisseo2011], qui fait état d’une moyenne de 116.380 déplacements par jour. La majo-

rité des bus du réseau ont une capacité totale de 97 places (31 places assises et 68 places debout) [Bus Tisseo2013]. Puisque nous simulons une partie des volumes réels transportés, nous adaptons la capacité des bus en conséquence. L'objectif est d'avoir un rapport équivalent entre le nombre de voyageurs par journée d'une part et le nombre de places disponibles par bus fois le nombre de courses par journée d'autre part. Le résultat de cette adaptation est reporté dans le tableau 5.2.

Nombre de voyageurs	Capacités des véhicules
1.000	1
5.000	4
10.000	8
20.000	17
30.000	25

TABLE 5.2 – Les capacités des bus dans la simulation

5.4 Résultats

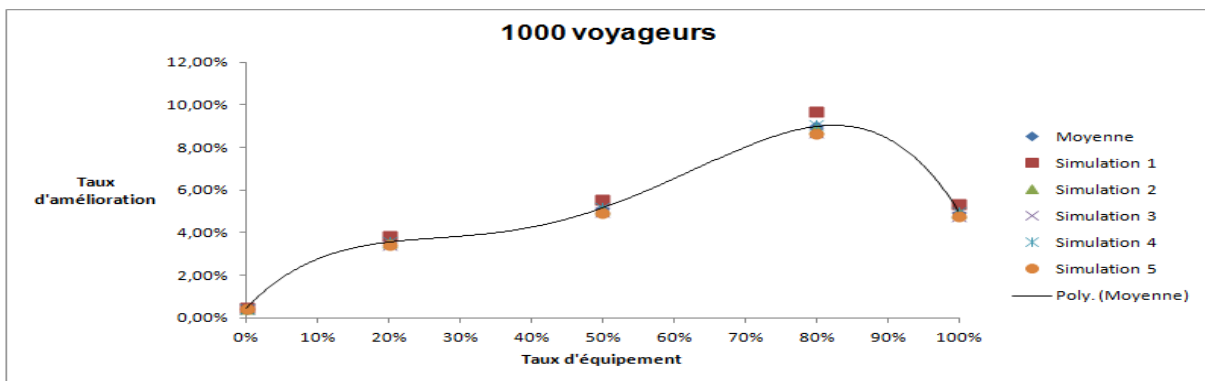


FIGURE 5.3 – Améliorations par rapport à un scénario sans information (1.000 voyageurs)

Les résultats sont reportés dans les Figures de 5.3 à 5.7. Les courbes représentent le taux d'amélioration des temps de parcours en comparaison avec la configuration de référence. Nous reportons l'ensemble des résultats pour chacune des cinq simulations tournées par configuration. L'objectif de cette multiplication des simulations avec les mêmes paramètres est de vérifier que la différence de résultats selon les scénarios ne provient pas simplement des choix des origines et des destinations ou des perturbations injectées, qui sont stochastiques. Rappelons que le tirage des origines et destinations des voyageurs n'est pas tout à fait aléatoire, puisqu'il dépend des zones origines et destinations spécifiées par le modèle de déplacement. Dans chaque courbe, nous ajoutons une courbe de tendance polynomiale associée à la moyenne, afin de faciliter l'interprétation. Nous donnons une synthèse des résultats sous la forme d'un tableau dans la table 5.3, où nous rappelons la moyenne calculée ainsi que les écarts-types observés. Ces

écarts sont très faibles et ne remettent pas en cause la tendance des courbes comme nous pouvons l'observer visuellement dans les figures (les courbes sont croissantes jusqu'à un certain seuil puis décroissantes).

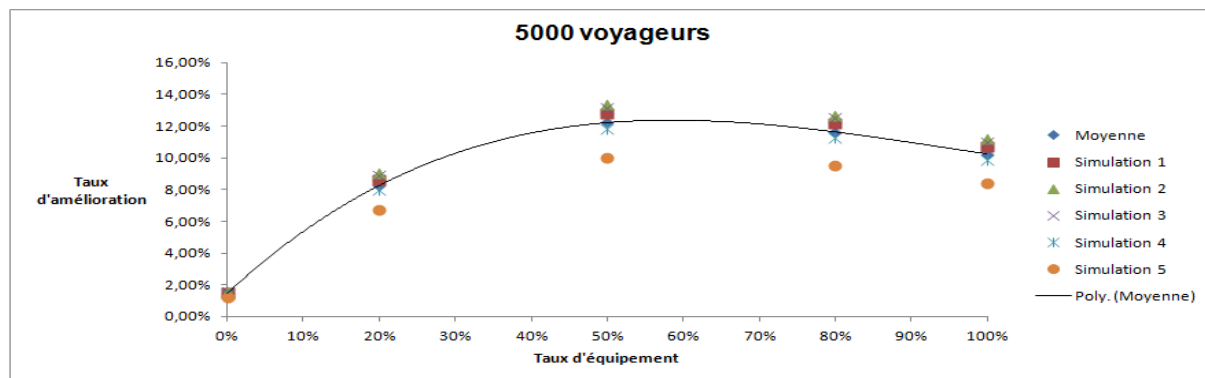


FIGURE 5.4 – Améliorations par rapport à un scénario sans information (5.000 voyageurs)

Tous les scénarios améliorent les temps de parcours par rapport au scénario sans information, avec une amélioration minimale de 0,45% pour le scénario à 1.000 voyageurs et 0% de voyageurs connectés et une amélioration maximale de 16,3% pour le scénario à 10.000 voyageurs et 80% de voyageurs connectés. Ils montrent donc d'abord que le scénario de référence sans information fournit le pire temps de parcours moyen. En effet, dans ce scénario, les passagers ont une information pour un retard ou une panne quand ils sont déjà bloqués dans un point du réseau. En recherchant une alternative, ils vont essayer d'utiliser l'infrastructure restante en concurrence et en ayant comme nouvelle information la seule perturbation dans laquelle ils sont engagés. Ils perdent donc un temps maximal en ne contournant pas la perturbation, mais peuvent également en perdre davantage s'ils se retrouvent pris dans une autre perturbation.

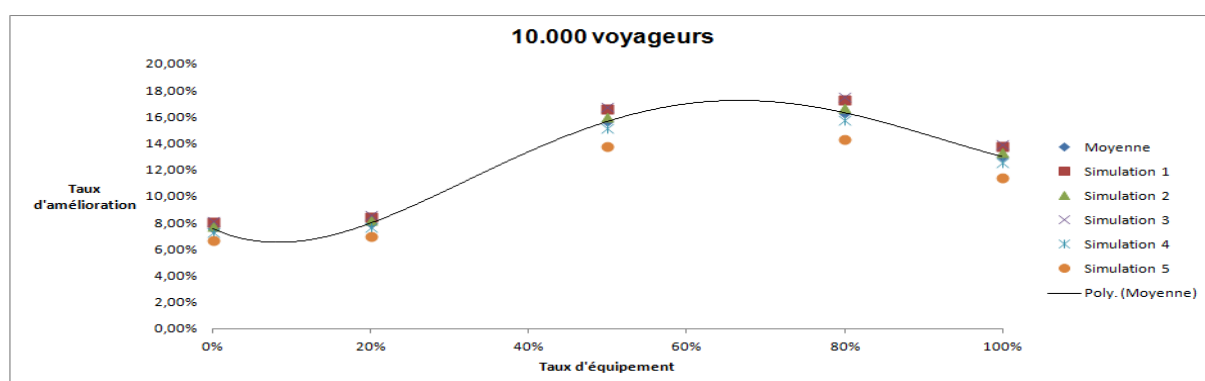


FIGURE 5.5 – Améliorations par rapport à un scénario sans information (10.000 voyageurs)

Le scénario 2 (avec exclusivement de l'information locale) fournit de meilleurs résultats que le scénario de référence. L'impact positif de l'information locale est du au fait que le voyageur n'a pas à atteindre la perturbation pour être informé, il a l'information dès qu'il passe par un

arrêt avec de l'information locale. Cependant, il peut toujours se diriger vers une autre perturbation, puisque son itinéraire est fondé sur sa vision mentale du réseau, essentiellement statique hormis l'information reçue aux arrêts.

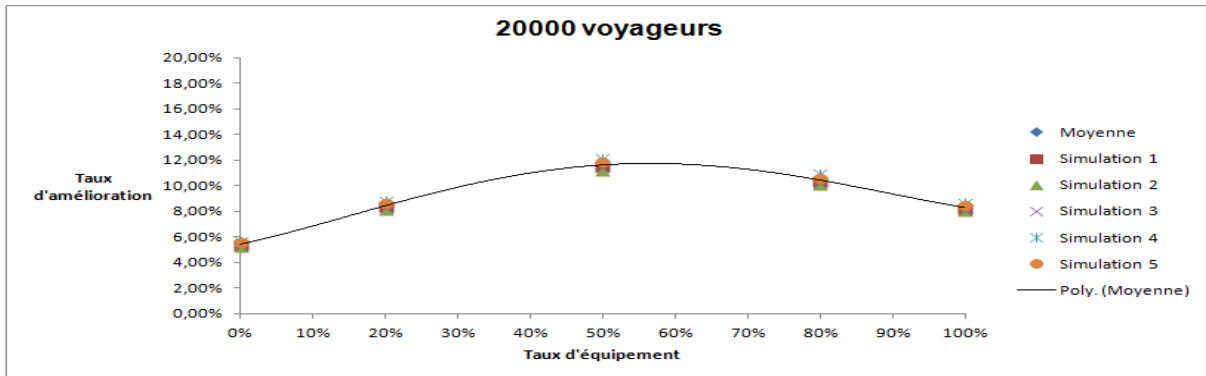


FIGURE 5.6 – Améliorations par rapport à un scénario sans information (20.000 voyageurs)

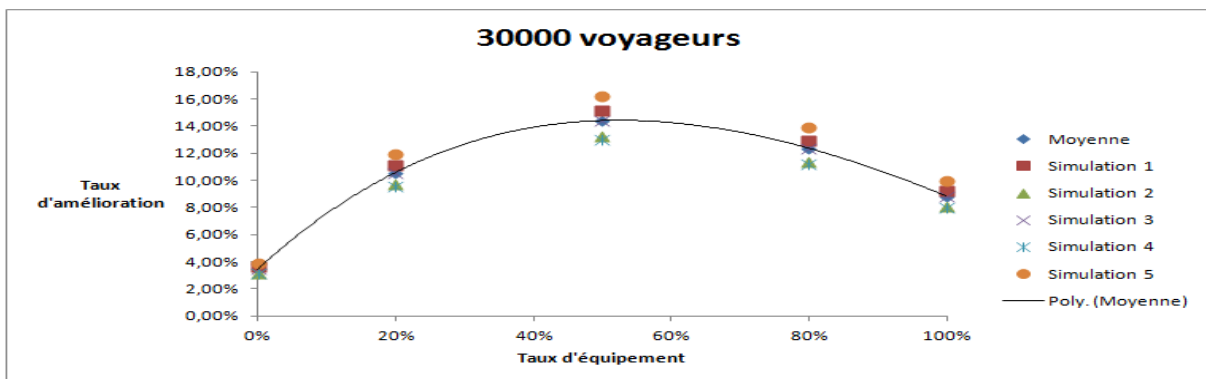


FIGURE 5.7 – Améliorations par rapport à un scénario sans information (30.000 voyageurs)

L'intégration d'un certain pourcentage de voyageurs connectés améliore les temps de parcours moyens jusqu'à un pourcentage de 50% de voyageurs connectés pour les scénarios à 5.000, 20.000 et 30.000 voyageurs et jusqu'à 80% pour les scénarios à 1.000 et 10.000 voyageurs. Cette fois, le parcours d'un voyageur est calculé en se fondant sur l'état réel du réseau, et les passagers évitent les zones perturbées très tôt. Le passager n'a pas à passer par un arrêt avec de l'information locale diffusée, il reçoit une notification et un nouveau plan immédiatement.

Cependant, cette amélioration plafonne au taux d'équipement de 50% pour certains scénarios et à 80% de voyageurs connectés pour d'autres, avant que les résultats deviennent unanimement moins bons dans les scénarios à 100% de voyageurs connectés. Dans les scénarios avec une grande majorité de voyageurs connectés, les voyageurs concernés par une perturbation reçoivent de nouveaux plans mis à jour générés simultanément. La conséquence est que les voyageurs, en appliquant leurs nouveaux plans, se retrouvent devant des contraintes de capacité des véhicules et voient leurs temps de parcours moyen augmenter.

	1.000 voy	5.000 voy	10.000 voy	20.000 voy	30.000 voy
0%	0,45% (0,03%)	1,45% (0,1%)	7,56% (0,25%)	5,43% (0,52%)	3,44% (0,33%)
20%	3,56% (0,23%)	8,25% (0,57%)	7,97% (0,26%)	8,43% (0,8%)	10,57% (1%)
50%	5,15% (0,33%)	12,21% (0,84%)	15,64% (0,52%)	11,62% (1,1%)	14,40% (1,36%)
80%	8,99% (0,58%)	11,63% (0,8%)	16,3% (0,54%)	10,43% (0,99%)	12,35% (1,17%)
100%	4,95% (0,32%)	10,24% (0,71%)	12,99% (0,43%)	8,2% (0,79%)	8,82% (0,84%)

TABLE 5.3 – Synthèse des temps de parcours

La provision d'information personnalisée en temps réel est supposée avoir un impact positif, en priorité pour les voyageurs connectés. Pour vérifier cet état de fait, nous reprenons les mêmes résultats que nous discriminons selon le type d'agent voyageur. Les résultats sont reportés dans les Figures de 5.8 à 5.12. Nous reportons ici les valeurs moyennes observées sans montrer les valeurs des simulations individuelles. En effet, la participation relative du type de voyageurs dans l'amélioration est par définition la même ¹⁵.

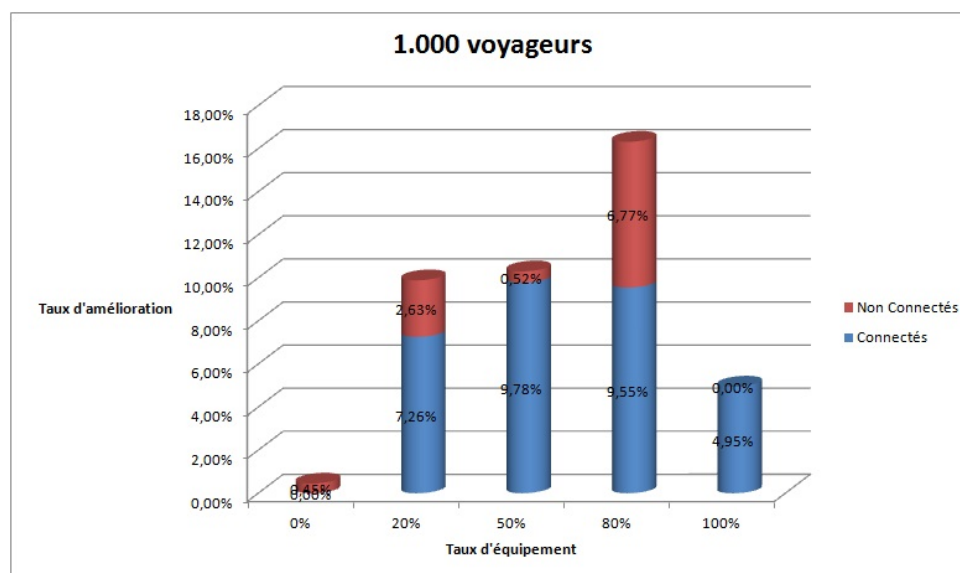


FIGURE 5.8 – Améliorations comparées des temps de parcours (voyageurs connectés vs. voyageurs non connectés) avec 1.000 voyageurs

Chaque figure représente les améliorations de temps de parcours pour un nombre de voyageurs concernés. La partie rouge des barres de chaque figure représente les améliorations pour les voyageurs non connectés et la partie bleue représente les améliorations pour les voyageurs connectés. Ainsi, pour les scénarios à 0% de voyageurs connectés, l'ensemble de l'amélioration est assurée par les voyageurs non connectés et pour les scénarios à 100% de voyageurs connectés, l'ensemble de l'amélioration est assurée par les voyageurs connectés.

15. Pour retrouver les valeurs d'amélioration moyenne des Figures 5.3 à 5.7, il suffit d'effectuer la somme des améliorations pondérée par la proportion du type de voyageurs concernés. Par exemple, le taux d'amélioration du scénario à 5.000 voyageurs, dont 20% connectés = $5,7\% \times \frac{4.000}{5.000} + 18,3\% \times \frac{1.000}{5.000} = 8,25\%$

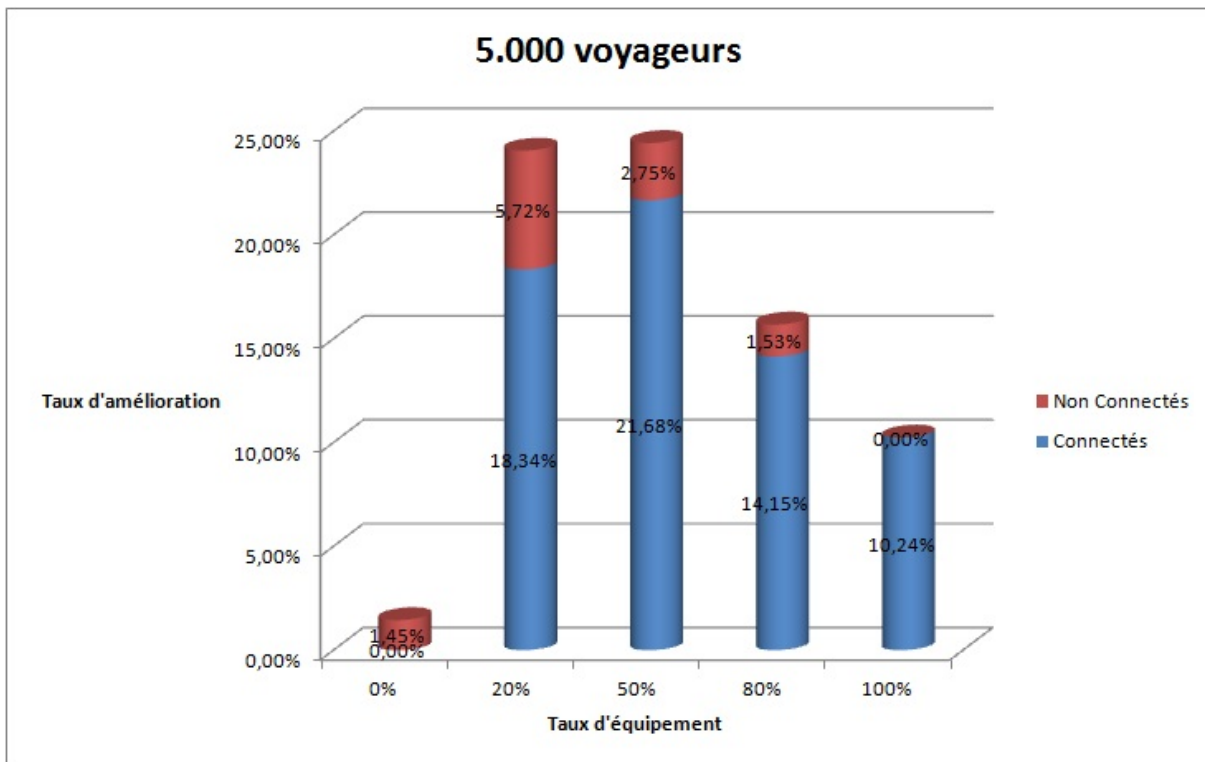


FIGURE 5.9 – Améliorations comparées des temps de parcours (voyageurs connectés vs. voyageurs non connectés) avec 5.000 voyageurs

Les résultats montrent que l'amélioration de temps de parcours pour les voyageurs connectés suit la même tendance que les courbes dans les Figures 5.3 à 5.7, i.e. une amélioration croissante jusqu'à un seuil, puis décroissante.

Pendant, le maximum d'amélioration est atteint cette fois systématiquement à 50% de voyageurs connectés. L'excédent d'amélioration pour les scénarios à 1.000 et 10.000 voyageurs et 80% de voyageurs connectés est en fait apporté par les voyageurs non connectés.

Le taux d'amélioration pour les voyageurs non connectés est quant à lui assez stable entre les scénarios avec le même nombre de voyageurs. Ainsi, il est entre 0,45% et 6,8% pour 1.000 voyageurs, entre 1,4% et 5,7% pour 5.000 voyageurs, entre 5,5% et 7,6% pour 10.000 voyageurs, entre 0,1% et 5,9% pour 20.000 voyageurs, et enfin entre 2,2% et 8% pour 30.000 voyageurs. Leurs courbes ne suivent pas de tendance particulière, leur comportement n'étant pas impacté par l'augmentation du niveau d'information des autres agents.

5.5 Temps d'exécution

Nous avons très rapidement constaté un problème de performance dans nos expérimentations quand la simulation durait très longtemps. Après une exécution fluide et relativement rapide, le simulateur avait des sauts de performances chaotiques. Comme le montre la Figure 5.13, la simulation affiche des temps d'exécution avec une très petite variation d'un tick à l'autre,

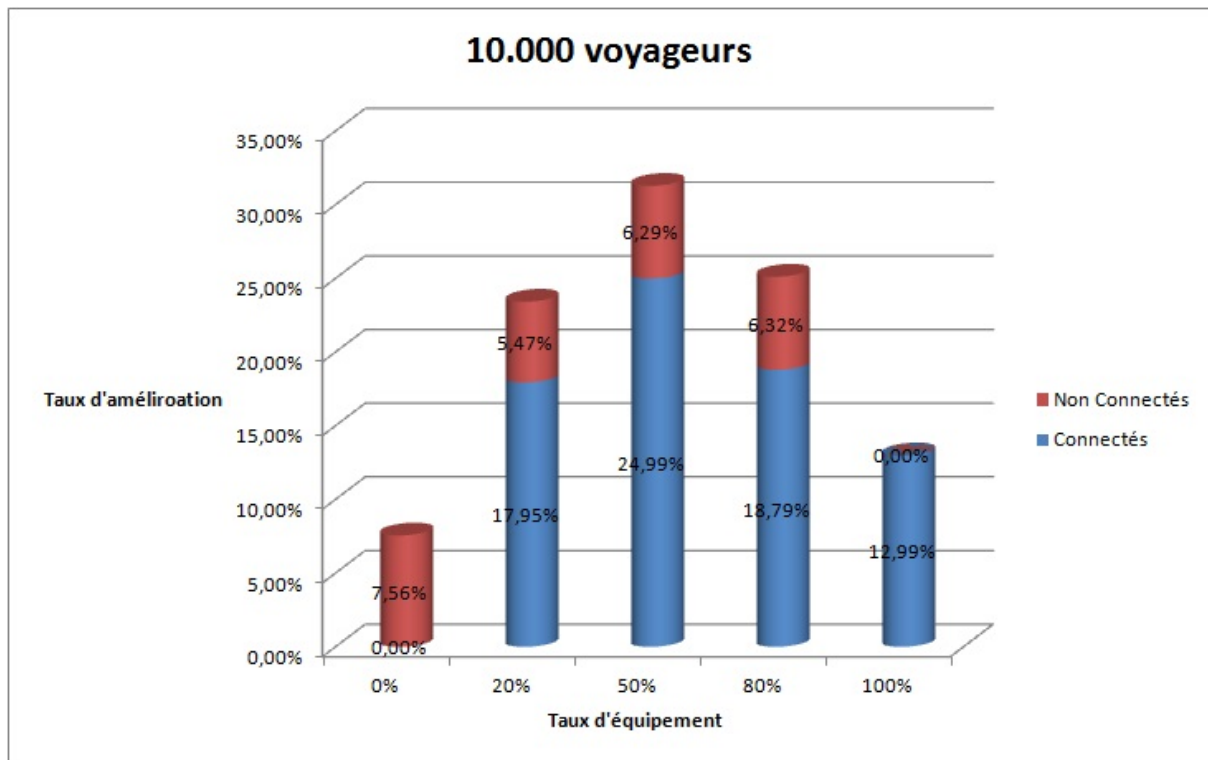


FIGURE 5.10 – Améliorations comparées des temps de parcours (voyageurs connectés vs. voyageurs non connectés) avec 10.000 voyageurs

jusqu'à 1.200 ticks environ où elle commence à afficher des temps d'exécution allant de moins d'une seconde jusqu'à 70 secondes entre deux ticks successifs, avec le même nombre d'agents.

Après diverses investigations, il s'est avéré que le système d'exploitation, observant qu'un processus prenait trop de temps d'exécution et ralentissait ses performances, abaissait sa priorité, ce qui générait ces longs sauts. Ce problème a été réglé grâce à un outil de gestion des processus¹⁶ qui permet d'allouer une priorité haute permanente à la machine virtuelle Java. La figure 5.14, relative à une simulation à 5.000 voyageurs montre que cette solution diminue grandement la variation dans les temps d'exécution par pas de temps de simulation.

Malgré nos diverses optimisations, présentées dans le chapitre 3, les temps d'exécution affichés par nos simulations restent assez grands. Cette lenteur est exclusivement liée au nombre d'agents voyageurs simulés. En effet, la taille du réseau considéré influe seulement sur le temps d'initialisation de la simulation. Le nombre d'agents véhicules considéré quant à lui impacte très peu la durée de la simulation. Nous avons effectué des simulations avec des dizaines de milliers de bus à vide, et la simulation demeure très rapide et très fluide : la durée totale de la simulation ne dépasse jamais la demi-heure. En revanche, et comme le montre la Figure 5.15, dès que nous introduisons les agents voyageurs dans la simulation, les temps d'exécution deviennent corrélés au nombre d'agents voyageurs, et les performances commencent à se détériorer, avant de franchement ralentir à partir de 20.000 voyageurs.

16. *Process Lasso, Bitsum*TM

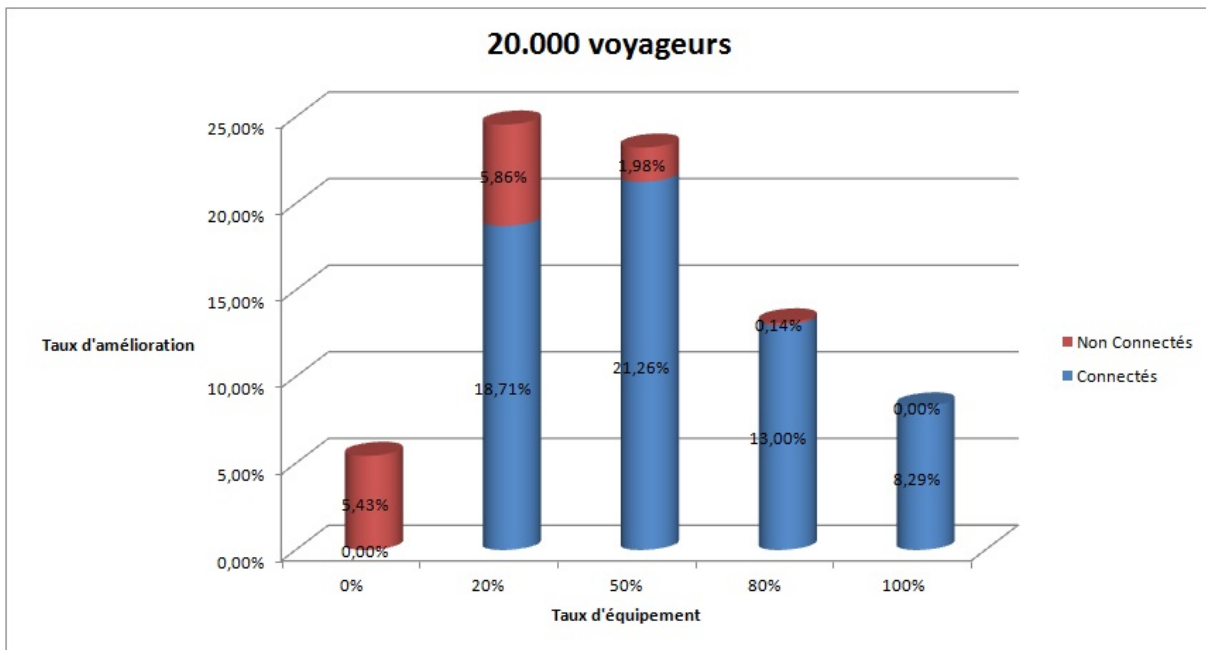


FIGURE 5.11 – Améliorations comparées des temps de parcours (voyageurs connectés vs. voyageurs non connectés) avec 20.000 voyageurs

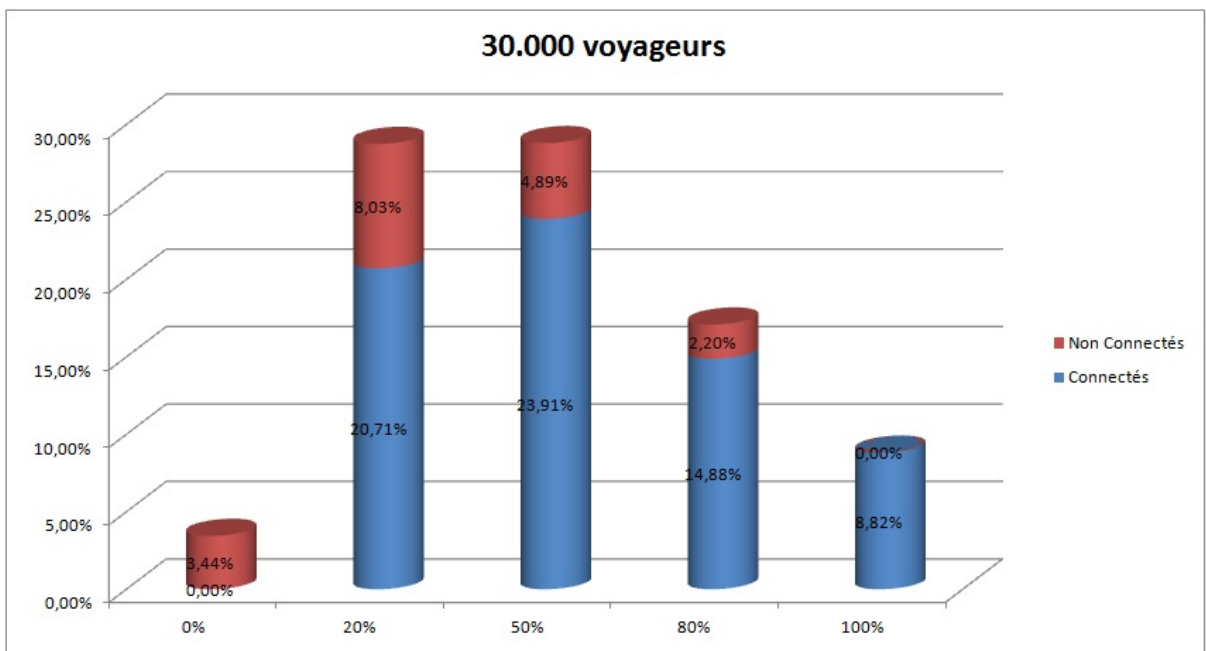


FIGURE 5.12 – Améliorations comparées des temps de parcours (voyageurs connectés vs. voyageurs non connectés) avec 30.000 voyageurs

Nous pensons que le système d'exploitation a recours à des *swap* entre mémoire vive et

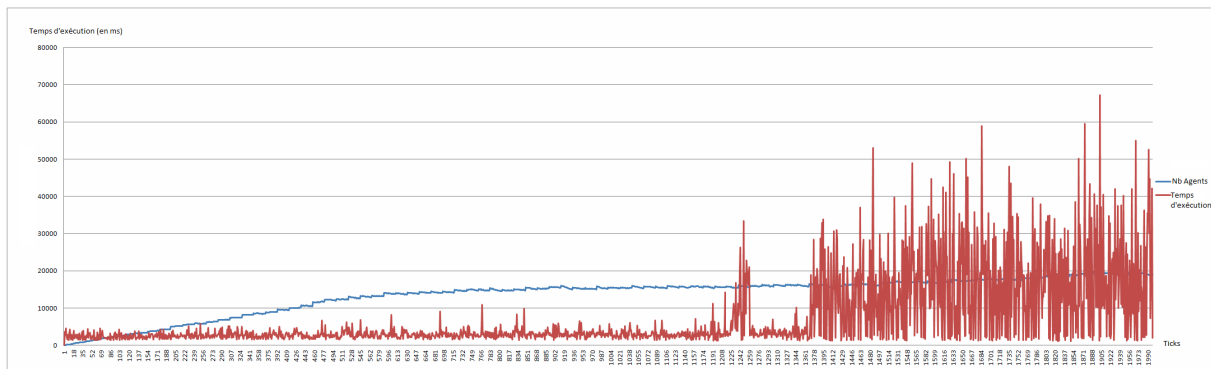


FIGURE 5.13 – Détérioration des temps d'exécution pour les simulations longues

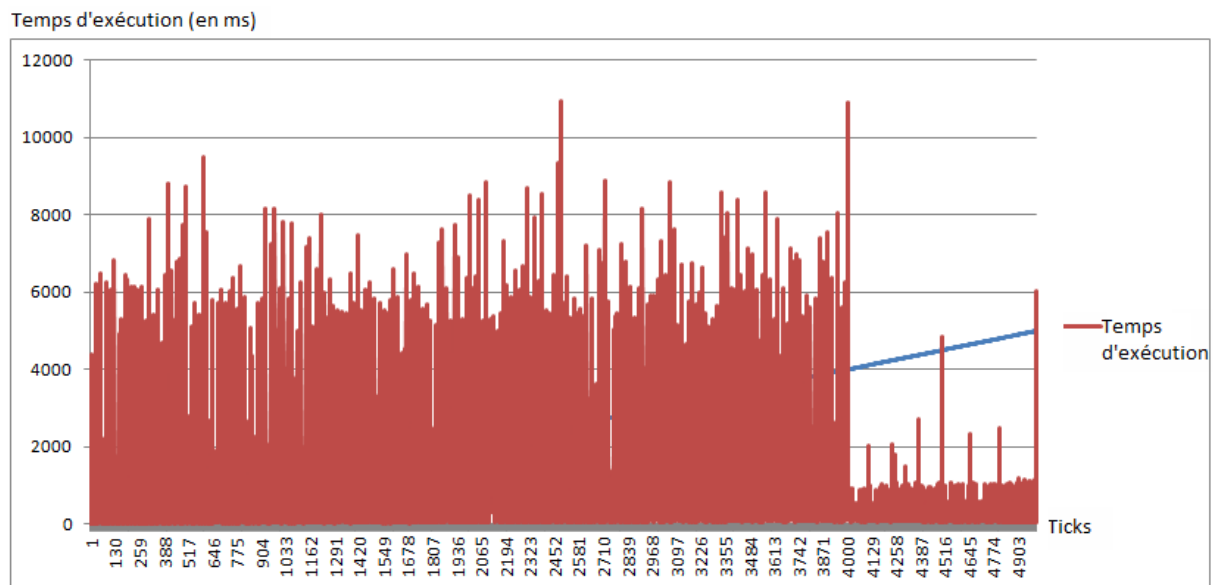


FIGURE 5.14 – Détérioration des temps d'exécution pour les simulations longues

disque dur à partir de 20.000 voyageurs (cf. Figure 5.16).

Sur la base de cette série de figures, nous concluons qu'il nous faut continuer à optimiser notre implémentation des agents voyageurs qui semblent consommer beaucoup de mémoire et ralentir la simulation. Un de nos travaux en cours concerne également la distribution des simulations de voyageurs sur divers serveurs. Ces deux travaux nous permettront dans un futur proche de monter en échelle concernant le nombre de voyageurs considérés, pour arriver à en traiter des millions dans une même simulation, et pouvoir ainsi simuler des volumes réels de voyageurs (tels que définis dans les modèles de déplacement) sur des grandes villes et région.

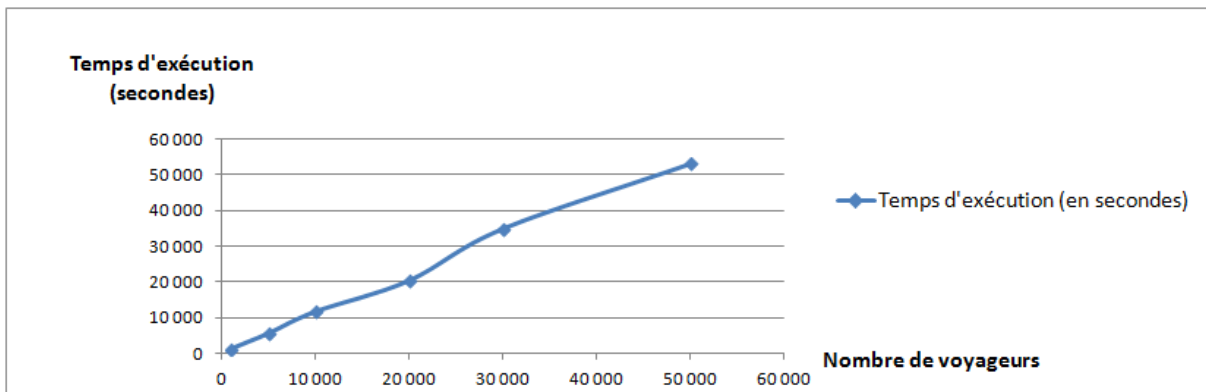


FIGURE 5.15 – Variation des temps d'exécution par rapport au nombre de voyageurs

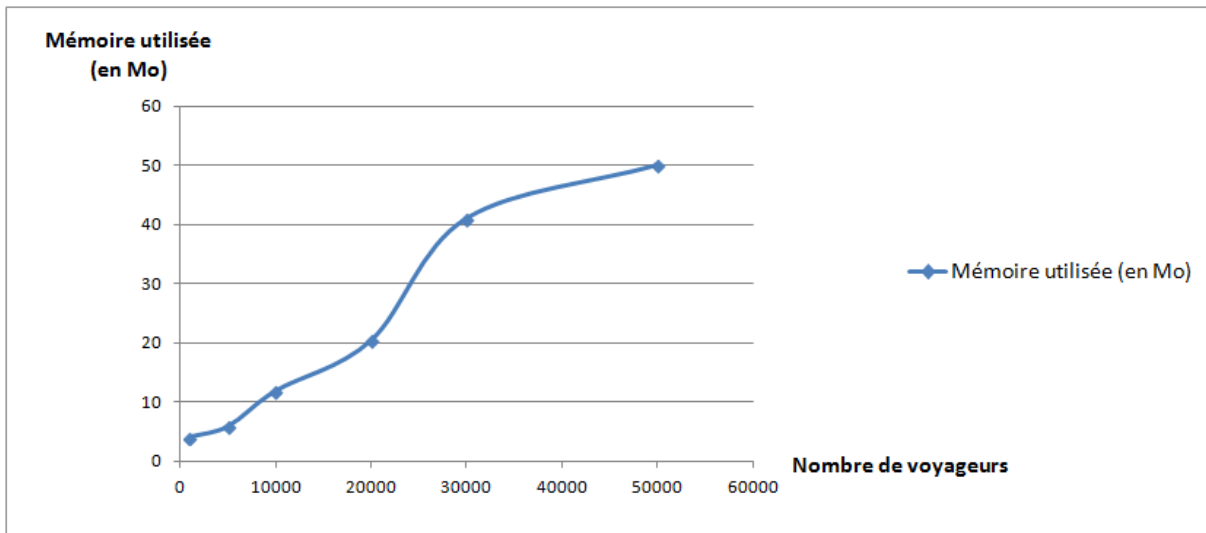


FIGURE 5.16 – Usage de la mémoire par rapport au nombre de voyageurs

5.6 Discussion

Nous avons vu dans le chapitre 1 les résultats reportés dans la littérature concernant notre problématique de validation de l'impact des informations voyageurs sur les réseaux de transport en commun. Les résultats et les conclusions de ces travaux sont différents des nôtres.

En effet, la nature des informations évaluées dans ce travail diffère de celles dans la littérature. Dans tous les travaux étudiés, l'évaluation ne concerne que des informations générales autour des temps d'attente diffusées à tous les voyageurs ou une partie d'eux (en fonction des scénarios), tandis que, dans notre travail nous avons évalué l'impact des informations personnalisées diffusées aux voyageurs à travers les *smartphones* et ne concernant que leurs itinéraires. Nous avons comparé différents scénarios où les taux d'équipements des voyageurs varient entre 0% et 100% afin d'analyser l'effet de ces informations sur le temps de parcours moyen des voyageurs. Par conséquent, les niveaux d'information étudiés dans nos expérimentations dif-

diffèrent aussi de ceux dans la littérature. Vu qu'on étudie les informations personnelles, les niveaux d'information portent sur les taux d'équipement des voyageurs, alors que dans les travaux de la littérature, ils portent sur la localisation des supports de diffusion des informations et la disponibilité géographique de ces informations. Nous avons fait le choix de nous concentrer sur les informations personnelles vu l'absence de travaux évaluant leur impact malgré le rôle majeur qu'elles jouent dans les systèmes d'information des voyageurs avancés.

Une autre différence entre notre travail et la littérature concerne le type de perturbations intégrées dans les expérimentations. Dans chacune de nos simulations, nous avons généré cinq coupures d'arcs aléatoires sur le réseau de transport pendant toute la simulation. Ces coupures des arcs font que certains voyageurs ne retrouvent plus d'itinéraires vers leur destination. Dans les travaux de la littérature, les perturbations générées sont généralement des retards de passages des véhicules, des réductions de fréquences de lignes de transport ou des effets de congestion sur le réseau (i.e capacité moyenne des véhicules). Nous considérons que dans les situations où des arcs d'un réseau sont coupés que les informations des voyageurs ont le plus d'impact et sont plus utiles aux voyageurs essentiellement lorsqu'ils leur permettent d'éviter de se retrouver bloqués pendant leurs déplacements.

L'envergure des échantillons considérés est aussi différente. Les travaux en simulation de la littérature considèrent quelques itinéraires jouets (3 itinéraires reliant une même paire origine-destination (origine-destination) pour [Hickman and Wilson1995] par exemple), alors que nous considérons l'ensemble du réseau avec des dizaines de milliers d'origines-destinations. A cet égard, notre travail est le premier avec une configuration aussi importante. Cela est rendu possible par le développement d'un vrai simulateur de déplacements multi-agent pour l'évaluation de l'impact, au contraire des travaux de la littérature.

Enfin, concernant les résultats, nos résultats diffèrent de ceux de la littérature. En effet, dans [Coppola and Rosati2009] et dans [Cats *et al.*2011], les auteurs concluent que la provision des informations a un petit impact en cas de service irrégulier. Les résultats présentés ici sont à cet égard tout à fait originaux, concluant que la provision d'information temps-réel a un impact positif qui atteint son optimum avant le taux d'équipement de 100%.

Ainsi, le résultat le plus intéressant reporté concerne la moindre amélioration des temps de parcours des voyageurs à partir d'un certain taux d'équipement des voyageurs. Ce résultat peut sembler paradoxal, dans la mesure où on s'attend intuitivement à ce que les temps de parcours moyens s'améliorent continuellement à mesure que le pourcentage de voyageurs connectés augmente. En réalité, ce résultat met en évidence la différence entre optimum individuel et optimum collectif énoncée par Wardrop [Wardrop1952]. En effet, le second principe de Wardrop énonce qu'à l'équilibre, le temps moyen de parcours de tous les voyageurs est minimal. Cela implique que tous les voyageurs se comportent d'une manière coopérative dans le choix de leurs itinéraires pour s'assurer du comportement efficace de l'ensemble du système. Dans nos expérimentations, les voyageurs connectés optimisent leurs itinéraires sans prendre en compte les choix des autres voyageurs, ce qui génère des dépassements de capacités des véhicules, surtout en cas de perturbations. Ce résultat suggère donc que les instructions données par le service planification devraient prendre en compte l'ensemble des voyageurs connectés et leur proposer des itinéraires différenciés, en prenant en compte les capacités courantes des véhicules.

5.7 Conclusion

Dans ce chapitre, nous avons présenté la configuration des expérimentations que nous avons effectuées pour valider le fonctionnement de notre simulateur. Afin d'évaluer l'impact de la provision d'information voyageur sur le réseau, nous avons testé différents scénarios en fonction du pourcentage de voyageurs connectés représentés par des agents. Dans chacun de ces scénarios, nous avons exécuté cinq simulations avec 1.000, 5.000, 10.000, 20.000 et 30.000 passagers. Ces scénarios simulés sont analysés suivant leur impact sur les temps de parcours moyens des voyageurs, connectés et non connectés. Le résultat le plus intéressant montre que le nombre de voyageurs connectés a un impact positif sur les temps de parcours jusqu'à un certain seuil, au delà duquel l'impact devient relativement négatif. Ce résultat suggère donc, qu'à partir de ce seuil, les instructions données par le service planification devraient prendre en compte l'ensemble des voyageurs connectés et leur proposer des itinéraires différenciés, en prenant en compte les capacités courantes des véhicules.

Conclusion générale et perspectives

Bilan

Dans cette thèse, nous proposons d'étudier et de comprendre l'impact des informations reçues par les voyageurs sur leurs temps de parcours et sur l'état des réseaux de transport en commun. Après un état de l'art sur les systèmes d'information de voyageurs ainsi que les méthodes d'évaluation de l'impact de l'information des voyageurs (cf. chapitre 1), nous avons conclu que la simulation est la méthode la plus intéressante pour l'évaluation. L'évaluation de l'impact de l'information voyageur par simulation offre aux gestionnaires des réseaux de transport un outil de gestion pertinent. En effet, cela permet de tester et de calibrer les stratégies d'informations opérantes ou celles que les opérateurs désirent mettre en place. Ces stratégies sont un outil indispensable pour la gestion d'un réseau de transport essentiellement en cas de dysfonctionnement : ils permettent de diriger les flux de voyageurs en cas de congestion sur le réseau et participent en tant qu'outil de gestion de crise dans certains cas (accidents, catastrophes, attentat, etc.).

Partant de ce choix, nous avons étudié dans le chapitre 2 la simulation des déplacements. Nous avons évalué les simulateurs de déplacements existants et les plateformes de simulation multi-agent existantes. Notre choix s'est porté sur la plateforme généraliste Repast Symphony pour la conception et le développement d'un nouveau simulateur des déplacements.

Fondé sur Repast Symphony, nous avons conçu un simulateur multi-agent de déplacement de voyageurs sur un réseau de transport en commun intégrant un système d'information des voyageurs. Le simulateur est divisé en deux parties. La première partie permet le suivi, à un niveau microscopique, des déplacements des voyageurs sur le réseau. Tandis que la deuxième partie assure la diffusion en temps-réel des informations concernant l'état du réseau. Trois agents et un service composent notre système multi-agent : les agents véhicules et les agents voyageurs constituent les éléments mobiles, le service de planification et les agents information locale assurent une fonction d'assistance aux agents voyageurs. Nous avons défini deux types d'agents voyageurs : connecté et non-connecté au système d'information temps-réel à l'aide d'un *smartphone*. Cet équipement permet à ses utilisateurs de recevoir des informations personnalisées pendant leur déplacements afin de les guider en cas de changements sur l'état du réseau, notamment en cas de perturbations. Les voyageurs non-connectés ne reçoivent que des informations générales diffusées au niveau des arrêts et des stations et perçues par tous les voyageurs présents.

Pour mettre en place ce système d'information, nous avons adopté une approche centrée-environnement et nous avons repris un modèle de représentation de l'environnement physique fondé sur les réseaux espace-temps. Cet environnement multi-agent spatiotemporel représente à tout moment l'état actuel du réseau : les positions des véhicules de transport en commun,

ainsi que leurs trajets planifiés. En modifiant cette représentation spatiotemporelle de l'environnement, les perturbations peuvent être intégrées. Avec cette représentation de l'environnement, toute modification dynamique de l'offre de transport est répercutée directement sur les seuls agents qui sont concernés, ce qui évite les diffusions massives d'information à tous les agents du système. Il suffit à un agent de s'inscrire auprès des arcs espace-temps de son itinéraire pour être informé des changements en lien avec cet itinéraire.

Pour valider notre simulateur, nous avons exécuté nos expérimentations sur la ville de Toulouse, pour laquelle nous avons des données détaillées, y compris un modèle de déplacement. Nous avons exécuté six scénarios différents dans lesquels le niveau d'équipement des voyageurs varie entre 0% et 100%. Ces scénarios simulés sont analysés suivant leur impact sur les temps de parcours moyens des voyageurs, connectés et non connectés. Les résultats montrent que le nombre de voyageurs connectés a un impact positif sur les temps de parcours jusqu'à un certain seuil, au delà duquel l'impact devient relativement négatif.

Publications : Ce travail de thèse a fait l'objet de six publications (quatre publiées et deux soumises) dans :

- *Agent-Based Simulator for Travelers Multimodal Mobility. KES-AMSTA 2013 : pp.81-90;*
- *Simulating the impact of future Internet on multimodal mobility. IEEE AICCSA 2014 : pp. 230-237;*
- *Impact of travelers information level on disturbed transit networks : a multiagent simulation. IEEE ITSC 2015 : pp. 2889-2894;*
- Impact de l'information voyageurs : simulation multi-agent. JFSMA'2016 : 10 pages.
- Simulation multi-agent de l'information voyageurs en transport en commun. Soumis à Technique et Science Informatique : 28 pages.
- *Multiagent simulation of personalized travelers information on transit networks. Soumis à IEEE Transactions on Intelligent Transport Systems : 15 pages.*

Perspectives

Le simulateur de déplacements enrichi par l'information des voyageurs nous offre « un terrain de jeu » qui nous permet désormais d'envisager le test rapide d'un grand nombre d'applications de transport. Au delà de cette perspective, nos travaux futurs concerneront en priorité les points suivants.

Stratégies d'information

Une perspective à ce travail est relative à l'optimisation des stratégies d'information des voyageurs par les opérateurs de transport. En effet, à partir du seuil d'équipement critique, les opérateurs peuvent décider d'optimiser les informations qu'ils fournissent aux voyageurs afin de mieux les distribuer sur leurs réseaux. Au-delà du problème d'optimisation, des questions intéressantes, relatives à l'obéissance des voyageurs et à leur confiance dans le système d'information, sont à traiter.

Impact de l'information multimodale

Nous sommes aussi en train d'étendre ce travail vers les réseaux multimodaux (réseaux de transport en commun, réseau routiers et modes alternatifs de transport). Cette problématique est très intéressante puisque les échanges de flux de voyageurs entre les réseaux routiers et en commun peuvent faire en sorte qu'une situation se déclenche sur un réseau et qu'elle se prolonge sur l'autre. Ou au contraire que le second réseau résorbe la perturbation du premier. Outre le partage de la voirie entre véhicules des deux modes, les interactions entre réseaux routiers et en commun peuvent avoir lieu suite à des actions de *Park & Ride* des conducteurs (qui laissent leur voiture dans un parking et continuent leur chemin en transport en commun). Ou à des actions de co-voiturage pour les voyageurs des transports en commun.

Une feuille de route pour le simulateur

Outre les optimisations de performances énoncées dans le chapitre précédent, nous prévoyons d'améliorer le simulateur de diverses manières.

Réutilisabilité du simulateur

Afin de faciliter l'utilisation du simulateur en dehors de notre équipe, il faudrait qu'il soit capable d'intégrer facilement les différents types de réseaux en entrée, en lieu et place des fichiers xml que nous avons spécifiés. En cela, nous pouvons nous inspirer des mécanismes de consommation des réseaux des simulateurs commerciaux.

Passage à l'échelle

Dans le futur proche, nous projetons de développer des scénarios avec des centaines de milliers de passagers pour vérifier que les résultats restent valides avec des volumes réels de voyageurs. Pour ce faire, la distribution du simulateur sur le *cloud* est en cours de développement.

Cas d'utilisation du simulateur

Le simulateur doit pouvoir être utilisé en remplacement des données dynamiques de l'opérateur de transport. En effet, une interruption ponctuelle ou pérenne de la fourniture de données par l'opérateur de transport peut survenir. Dans ces cas, lorsque les données sont indisponibles concernant une zone géographique ou une plage temporelle donnée, le simulateur devrait générer des données de déplacement cohérentes. Ces données peuvent alimenter une plateforme de régulation comme ClaireSiti [Scemama and Carles2004].

Bibliographie

- [Adam2000] E. Adam. *Modèle d'organisation multi-agent pour l'aide au travail coopératif dans les processus d'entreprise : application aux systèmes administratifs complexes*. PhD thesis, Université de Valenciennes et du Hainaut Cambrésis, 2000.
- [Ambrosino *et al.*2010] G. Ambrosino, M. Boero, J.D. Nelson, and M. Romanazzo. *INFOMOBILITY SYSTEMS AND SUSTAINABLE TRANSPORT SERVICES*. ENEA, 2010.
- [Balbo and Zargayouna2004] Flavien Balbo and Mahdi Zargayouna. Serveur agent d'information : application à l'information des voyageurs. In *Actes du XXIIème Congrès INFORSID, Biarritz, France, 25-28 mai, 2004*, pages 249–265, 2004.
- [Balbo2000] Flavien Balbo. *ESAC : un modèle d'interaction multi-agent utilisant l'environnement comme support actif de communication : application à la gestion des transports*. PhD dissertation, Paris (France), 2000.
- [Banos A.2013] Maps Team. Banos A. Miro 2 – compte-rendu de fin de projet. Technical report, UMR Géographie-Cités, 2013.
- [Barbeau *et al.*2010] S.J Barbeau, Winters P.L., Georggi N.L., and Labrador M.A. Travel assistance device : utilising global positioning system-enabled mobile phones to aid transit riders with special needs. *Intelligent Transport Systems, IET*, 4(1), 2010.
- [Batty *et al.*2003] M. Batty, J. Desyllas, and E. Duxbury. Safety in numbers ? modelling crowds and designing control for the notting hill carnival. *Urban Studies*, 40(8) :1573–1590, 2003.
- [Bazzan and Klügl2014] A. L. Bazzan and F Klügl. A review on agent-based technology for traffic and transportation. *The Knowledge Engineering Review*, 29(03) :375–403, 2014.
- [Behrisch *et al.*2011] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz. Sumo - simulation of urban mobility - an overview. *SIMUL 2011, The Third International Conference on Advances in System Simulation*, pages 55–60, 2011.
- [Ben-Akiva *et al.*1991] Moshe Ben-Akiva, Andre De Palma, and Kaysi Isam. Dynamic network models and driver information systems. *Transportation Research Part A : General*, 25(5) :251 – 266, 1991.
- [Bessghaier *et al.*2012] Nesrine Bessghaier, Mahdi Zargayouna, and Flavien Balbo. Management of urban parking : an agent-based approach. In *Artificial Intelligence : Methodology, Systems, and Applications*, pages 276–285. Springer Berlin Heidelberg, 2012.
- [Brakewood *et al.*2014] Candace Brakewood, Sean Barbeau, and Kari Watkins. An experiment evaluating the impacts of real-time transit information on bus riders in tampa, florida. *Transportation Research Part A*, 69 :409–422, 2014.

- [Bus Tisseo2013] 2013.
- [Cahour and Salembier1996] Beatrice Cahour and Pascal Salembier. Cooperation and cooperator modeling. *Computer Supported Cooperative Work (CSCW)*, 5(2) :285–297, 1996.
- [Cambridge Systematics2009] Inc. Cambridge Systematics. Advanced traveler information system study. Technical report, Mid-Ohio Regional Planning Commission, (2009).
- [Castle2007] C. J. E. Castle. *Agent-based modelling of pedestrian evacuation : A study of London's king's cross underground station*. PhD thesis, University College London, London., 2007.
- [Cats *et al.*2011] O Cats, Haris N. Koutsopoulos, Wilco Burghout, and Tomer Toledo. Effect of real-time transit information on dynamic path choice of passengers. *Transportation Research Record : Journal of the Transportation Research Board*, 2217 :46 – 54, 2011.
- [Chevaillier *et al.*2003] Pierre Chevaillier, Pierre De Loor, and Cyril Septseault. *Technique et Science Informatiques*, 22(4) :331–344, 2003.
- [Chipeaux *et al.*2011] S. Chipeaux, F. Bouquet, C. Lang, and N. Marilleau. Modelling of complex systems with aml as realized in miro projec. *LAFLang 2011, workshop of the Int. Conf. WI/IAT (Web Intelligence and Intelligent Agent Technology)*, IEEE Computer Society, Lyon, France, pages 159– 162, 2011.
- [Cioffi-Revilla *et al.*2010] C. Cioffi-Revilla, J. D. Rogers, and M. Latek. *Simulating interacting agents and social phenomena : The second world congress in social simulation.*, volume 7, chapter The MASON householdWorlds of pastoral nomad societies. Springer, 2010.
- [Coletti2013] Mark Coletti. Geo Mason. <https://cs.gmu.edu/eclab/projects/mason/extensions/geomason/geomason.pdf>, 2013. [Online ; accessed 9-march-2016].
- [Collier2010] N. T. Collier. *Repast HPC manual*. Available at <http://repast.sourceforge.net/docs.html>, 2010.
- [Coppola and Rosati2009] Pierluigi Coppola and Luca Rosati. *Simulation-Based Evaluation of Advanced Public Transportation Information Systems (APTIS)*, volume 46 of *Operations Research/Computer Science Interfaces Series*, chapter Schedule-Based Modeling of Transportation Networks, pages 1–21. Springer, 2009.
- [Crooks and Castle2012] Andrew T Crooks and Christian JE Castle. The integration of agent-based modelling and geographical information for geospatial simulation. In *Agent-based models of geographical systems*, chapter 12, pages 219–251. Springer, 2012.
- [Crooks *et al.*2008] A. T. Crooks, C. J. E. Castle, and M. Batty. Key challenges in agent-based modelling for geo-spatial simulation. *Computer, Environment and Urban Systems.*, 32(6) :417–430, 2008.
- [Crooks2010] A. T. Crooks. Constructing and implementing an agent-based model of residential segregation through vector gis. *International Journal of GIS*, 24(5) :661–675, 2010.
- [Davis and Smith1983] Randall Davis and Reid G. Smith. Negotiation as a metaphor for distributed problem solving. *Artificial Intelligence*, 20 :63–109, 1983.
- [De Palma and Marchal2002] A. De Palma and F. Marchal. Real case applications of the fully dynamic metropolis tool-box : An advocacy for large-scale mesoscopic transportation systems. *Networks and Spatial Economics*, 2(4) :347–369, 2002.

-
- [Decker *et al.*1997] Keith Decker, Katia Sycara, and Mike Williamson. Middle-agents for the internet. In *In Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI-97, 1997)*.
- [Doniec *et al.*2008] A. Doniec, R. Mandiau, S. Piechowiak, and S. Espié. A behavioral multi-agent model for road traffic simulation. *Engineering Applications of Artificial Intelligence*, 21(8), 2008.
- [Dugdale *et al.*2000] Julie Dugdale, Bernard Pavard, and Jean-Luc Soubie. A Pragmatic Development of a Computer Simulation of an Emergency Call Center (regular paper). In *Designing Cooperative Systems : The Use of Theories and Models, Sophia Antipolis, 23/05/2000-26/05/2000*, pages 241–256, Amsterdam, mai 2000. IOS Press Ohmsha.
- [Dziekan and Kottenhoff2007] K. Dziekan and K. Kottenhoff. Dynamic at-stop real-time information displays for public transport : effects on customers. *Transportation Research Part A : Policy and Practice*, 41(6) :489–501, 2007.
- [El2012] Mageret El. Urban rail networks in world cities. *Journeys*, 2012.
- [Estrada *et al.*2015] Matías Estrada, Ricardo Giesen, Antonio Mauttone, Emilio Nacelle, and Leandro Segura. Experimental evaluation of real-time information services in transit systems from the perspective of users. *CASPT 2015, Rotterdam, The Netherlands*, 19-23 July 2015.
- [Fayech2003] B. Fayech. *Régulation des réseaux de transport multimodal : Systèmes multi-agents et algorithmes évolutionnistes*. PhD thesis, Ecole Centrale de Lille/ Université des Sciences et Technologies de Lille, Octobre 2003.
- [Fellendorf and Vortisch2010] Martin Fellendorf and Peter Vortisch. Microscopic traffic flow simulator vissim. In *Fundamentals of Traffic Simulation*, pages 63–93. Springer, 2010.
- [Ferber1995] Jacques Ferber. *Multigent Systems*. Inter-Editions, 1995. 522 pages.
- [Ferris *et al.*2009] B. Ferris, Watkins K., and A. Borning. Onebusaway : a transit traveller information system. in : International conference on mobile computing, applications and services (mobicase). ferris, b., watkins, k., borning, a. (2009) onebusaway : a transit traveller information system. In : *International Conference on Mobile Computing, Applications and Services (MobiCase)*, 2009.
- [Fontaine and Rounsevell2009] C. M. Fontaine and M. Rounsevell. An agent-based approach to model future residential pressure on a regional landscape. *Landscape Ecology*, 24(9) :1237–1254, 2009.
- [Fonzone and Schmöcker2014] A. Fonzone and J Schmöcker. Effects of transit real-time information usage strategies. *Transportation Research Board, Washington, D.C.*, 2014.
- [Gaudou *et al.*2013] B. Gaudou, C. Sibertin-Blanc, O. Therond, F. Amblard, J.P. Arcangeli, M. Balestrat, M.H. Charron-Moirez, E. Gondet, Y. Hong, T. Louail, E. Mayor, D. Panzoli, S. Sauvage, J. Sanchez-Perez, P. Taillandier, V.B. Nguyen, M. Vavasseur, and P. Mazzega. The maelia multi-agent platform for integrated assessment of low-water management issues. *International Workshop on Multi-Agent-Based Simulation (MABS), Saint-Paul, USA*, May 2013.
- [Graham and Steiner2006] S. Graham and J. Steiner. Travellersim : Growing settlement structures and territories with agent-based modelling. *Proceedings of the 34th Conference on Computer Applications and Quantitative Methods in Archaeology*, 2006.

- [Gu2004] Yahong Gu. Transims review. Master's thesis, VirginiaTech, 2004.
- [Hailegiorgis2010] A. B. Hailegiorgis. Changing residence in the city : An agent based model of residential mobility in arlington county. *The Association of American Geographers (AAG) Annual Meeting, Washington , DC.*, pages 14–18, 2010.
- [Haklay et al.2001] M. Haklay, D. O'Sullivan, M. Thurstain-Goodwin, and T. Schelhorn. So go downtown” :simulating pedestrian movement in town centres. *Environment and Planning B*, 28(3) :343–359, 2001.
- [Hickman and Wilson1995] D. Hickman and H. Wilson. Passenger travel time and path choice implications of real-time transit information. *Transportation Research Part C*, 3(4) :221–226, 1995.
- [Hillsman and Barbeau2011] E. Hillsman and S. Barbeau. Enabling cost-effective multimodal trip planners through open transit data. final report. Technical report, Florida Department of Transportation. National Centre for Transit Research, Tempa., 2011.
- [Holland1992] J. Holland. *Adaptation in Natural and Artificial Systems*. Bradford Books, 1992.
- [Howe et al.2006] T Howe, N Collier, M North, M Parker, J Vos, et al. Containing agents : Contexts, projections, and agents. In *Proceedings of the Agent 2006 Conference on Social Agents : Results and Prospects*. Argonne, Illinois, USA : Argonne National Laboratory. Citeseer, 2006.
- [Hu2009] Hyejung Hu. *Measuring the Effectiveness of Advanced Traveler Information Systems (ATIS)*. ProQuest, 2009.
- [Infopolis1998] Infopolis. Review of current passenger information systems,. Technical report, Commission of the European Communities, 1998.
- [Jackson et al.2008] J. Jackson, B. Forest, and R. Sengupta. Agent-based simulation of urban residential dynamics and land rent change in a gentrifying area of boston. *Transactions in GIS*, 12(4) :475–491, 2008.
- [Jain et al.2014] Gaurav V. Jain, S. S. Jain, and Manoranjan Parida. Web-based passenger information system for ahmedabad city. *International Journal of Remote Sensing and GIS*, 3(2) :18–29, 2014.
- [Jakob et al.2012] M. Jakob, Z. Moler, A. Komenda, Z. Yin, A. X. Jiang, M. P. Johnson, M. Pechoucek, and M. Tambe. Agentpolis : towards a platform for fully agent-based modeling of multi-modal transportation (demonstration). *International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2012*, 3 :1501–1502, 2012.
- [Janssen2009] M. A. Janssen. Understanding artificial anasazi. *Journal of Artificial Societies and Social Simulation*, 12(4), 2009.
- [Jariyasunant et al.2010] J. Jariyasunant, D. Work, B. Kerkez and R. Sengupta, S. Glaser, and A. Bayen. Mobile transit trip planning with real-time data. *Transportation Research Board 89th Annual Meeting, Washington, D.C.*, 2010.
- [Kennedy et al.2010] W. B. Kennedy, A. B. Hailegiorgis, Rouleau M., Bassett J. K., M. Coletti, G. C. Balan, and T. Gulden. An agent-based model of conflict in east africa and the effect of wateringholes. *Behavior Representation in Modeling and Simulation (BRiMS) Conference, Charleston*, 2010.

-
- [kso2016] Building a realistic data environment for multiagent mobility simulation. *Smart Innovation, Systems and Technologies*, page 10, 2016.
- [Kumar *et al.*2005] Praveen Kumar, Varun Singh, and Dhanunjaya Reddy. Advanced traveler information system for hyderabad city. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, 6(1), March 2005.
- [Łatek *et al.*2010] M. M. Łatek, S. M. M. Rizi, and A. Geller. Persistence in the political economy of conflict : The case of the afghan drug industry. *Complex Adaptive Systems : Resilience, Robustness, and Evolvability Workshop, Fall Symposium of Association for the Advancement of Artificial Intelligence*,, 2010.
- [Lathia *et al.*2012] Neal Lathia, Licia Carpa, and Daniele Magliocchetti. Personalizing mobile travel information system” ., *Transport Research Arena-Europe vol.*, 48, 2012.
- [Li *et al.*2012] Jing-Quan Li Kun Zhou Kun Zhou Liping Zhang Jing-Quan Li, Liping Zhang, Kun Zhou, and Liping Zhang. A multimodal trip planning system incorporating the park-and-ride mode and real-time traffic/transit information. *Journal of Intelligent Transportation Systems*, 16(2), 2012.
- [Luke *et al.*2004] S. Luke, C. Cioffi-Revilla, L. Panait, and K. Sullivan. Mason : A new multi-agent simulation toolkit. *SwarmFest Workshop*, 2004.
- [Lyell and Becker2005] M. Lyell and M. Becker. Simulation of cognitive pedestrian agents crowds in crisis situations. *Systemics, Cybernetics and Informatics*, 4(3) :79–84, 2005.
- [Macal and North2005] Charles M Macal and Michael J North. Tutorial on agent-based modeling and simulation. In *Proceedings of the 37th conference on Winter simulation*, pages 2–15. Winter Simulation Conference, 2005.
- [Maciejewski and Nagel2012] M. Maciejewski and K. Nagel. Towards multi-agent simulation of the dynamic vehicle routing problem in matsim. *the 9th International Conference on Parallel 20 Ire soumission à RSTI - RIA Processing and Applied Mathematics, Part II, PPAM’11*, :561–560, 2012.
- [Malleon *et al.*2010] N. Malleon, A. Heppenstall, and L. See. Crime reduction through simulation : An agent- based model of burglary. *Computers, Environment and Urban Systems*, 34(3) :236–250, 2010.
- [Malleon2008] N. Malleon. *RepastCity – A demo virtual city*. Available at <http://portal.ncess.ac.uk/access/wiki/site/mass/repastcity.html>., 2008.
- [M.Ben-Akiva *et al.*1991] M.Ben-Akiva, A. D. Palma, and K. Isam. Dynamic network models and driver information systems. *Transportation Research Part A : General*,, 25(5) :251 – 266, 1991.
- [Meister *et al.*2009] K. Meister, M. Rieser, F. Ciari, A. Horni, M. Balmer, and K. W. Axhausen. Anwendung eines agentenbasierten modells der verkehrsnachfrage auf die schweiz. *Straßenverkehrstechnik*, 53(5) :269–280, 2009.
- [Michael *et al.*2009] Balmer Michael, Marcel Rieser, Konrad Meister, David Charypar, Nicolas Lefebvre, and Kai Nagel. Matsim-t : Architecture and simulation times. *Multi-Agent Systems for Traffic and Transportation Engineering*. IGI Global, pages 57–78, 2009.
- [Michel2015] Fabien Michel. *Approches environnement-centrées pour la simulation de systèmes multi-agents*. PhD thesis, Montpellier II, 2015. Habilitation thesis.

- [Minar *et al.*1996] N. Minar, R. Burkhart, C. Langton, and M. Askenazi. The swarm simulation system : A toolkit for building multi-agent simulations. Technical report, Santa Fe Institute, 1996.
- [Nagel and M.Rickert2001] K. Nagel and M.Rickert. Parallel implementation of the transims micro-simulation. *Parallel Computing*, 27(12) :1611–1639, 2001.
- [North *et al.*2005] M North, Thomas R Howe, Nick T Collier, and J Vos. The repast symphony runtime system. In *Agent 2005 Conference on Generative Social Processes, Models, and Mechanisms. Argonne, Illinois, USA : Argonne National Laboratory*. Citeseer, 2005.
- [Othman *et al.*2014] Amine Othman, Mahdi Zargayouna, Besma Zeddini, and Gérard Scemama. Simulating the impact of future internet on multimodal mobility. In *The 11th ACS/IEEE International Conference on Computer Systems and Applications AICCSA'2014*, page 10 pages. IEEE Computer Society, 2014.
- [Pajor2009] Thomas Pajor. Multi-modal route planning. Master's thesis, Karlsruhe Institute of Technology, 2009.
- [Patrick *et al.*2012] Taillandier Patrick, Vo Duc-An, Amouroux Edouard, and Drogoul Alexis. *Principles and Practice of Multi-Agent Systems : 13th International Conference, PRIMA 2010, Kolkata, India, November 12-15, 2010, Revised Selected Papers*, chapter GAMA : A Simulation Platform That Integrates Geographical Information Data, Agent-Based Modeling and Multi-scale Control, pages 242–258. Springer Berlin Heidelberg, 2012.
- [Payne *et al.*2002] T. Payne, M. Paolucci, R. Singh, and K. Sycara. Communicating agents in open multi agent systems. In *First GSFC/JPL Workshop on Radical Agent Concepts (WRAC)*, 2002.
- [Railsback and Harvey2002] S. F. Railsback and B. C. Harvey. Analysis of habitat selection rules using an individual- based model. *Ecology*, 83(7) :1817–1830, 2002.
- [Rao and Georgeff1992] A. S. Rao and M. P. Georgeff. An abstract architecture for rational agents. In *Nebel, B., Rich, C., Swartout, W.R., eds. : KR, Morgan Kaufmann*, pages 439–449, 1992.
- [Roth *et al.*2012] Camille Roth, Soong Moon Kang, Michael Batty, and Marc Barthelemy. A long-time limit for world subway networks. *The Royal Society*, 2012.
- [Roze2003] C. P. Roze. *Organisation multi-agents au service de la personnalisation de l'information*. PhD thesis, Université de valenciennes et du Hainaut-Cambrésis, 2003.
- [Russell and Norvig2003] S. Russell and P. Norvig. *Artificial Intelligence : A Modern Approach*. Prentice Hall, 2003. 1057 pages.
- [Saunier *et al.*2006] Julien Saunier, Flavien Balbo, and Fabien Badeig. Environment as active support of interaction. In *Proceeding of the third workshop on Environment for Multiagent Systems (E4MAS'06)*, Hakodate (Japan), 2006.
- [Saunier2008] Julien Saunier. *Les communications multi-parties et leur régulation dans les systemes multi-agents : modele et support*. PhD thesis, University of Paris-Dauphine, 2008.
- [Scemama and Carles2004] Gérard Scemama and Olivier Carles. Claire-siti, public road transport network management control : a unified approach. In *Proc. of 12th IEE Int. Conf. on Road Transport Information & Control (RTIC'04)*, pages 11–18, London (UK), April 2004.

-
- [Schelhorn *et al.*1999] Thorsten Schelhorn, David O’Sullivan, Mordechai Haklay, and Mark Thurstain-Goodwin. Streets : An agent-based pedestrian model. Technical report, Centre for Advanced Spatial Analysis UCL, 1999.
- [Schweiger2003] C. L. Schweiger. Customer and media reactions to real-time bus arrival information systems. *Transportation Research Board*, 48, 2003.
- [Shendarkar *et al.*2008] Ameya Shendarkar, Karthik Vasudevan, Seungho Lee, and Young-Jun Son. Crowd simulation for emergency response using {BDI} agents based on immersive virtual reality. *Simulation Modelling Practice and Theory*, 16(9) :1415 – 1429, 2008.
- [Taillandier *et al.*2014] Patrick Taillandier, Arnaud Grignard, Benoit Gaudou, and Alexis Drogoul. Des données géographiques à la simulation à base d’agents : application de la plateforme gama. *Cybergeo : European Journal of Geography [En ligne], Systèmes, Modélisation, Géostatistiques*, 2014.
- [Taillandier2013] Taillandier. Gama 2013. <https://code.google.com/p/gama-platform/>, 2013. [Online ; accessed 9-march-2016].
- [Tatara and Ozik2009] E. Tatara and J. Ozik. How to build an agent-based model iii – repast symphony. *Applied Agent-based Modeling in Management Research, Academy of Management Annual Meeting, Chicago*, 2009.
- [Tatara *et al.*2006] E. Tatara, M. J. North, T. R. Howe, N. T. Collier, and J. R. Vos. An introduction to repast symphony modelling using a simple predator-prey example. In M. J. North (Eds.) D. Sallach, C. M. Macal, editor, *the Agent 2006 Conference on Social Agents : Results and Prospects*, Chicago, 2006. University of Chicago and Argonne National Laboratory.
- [Tisseo2011] 2011.
- [Torrens and Nara2007] P. M. Torrens and A. Nara. Modelling gentrification dynamics : A hybrid approach. *Computers, Environment and Urban Systems*, 31(3) :337–361, 2007.
- [Tumas and Ricci2009] Gytis Tumas and Francesco Ricci. Personalized mobile city transport advisory system. *Information and Communication Technologies in Tourism*, 2009.
- [Vercouter2001] Laurent Vercouter. A distributed approach to design open multi-agent systems. In Andrea Omicini, Paolo Petta, and Robert Tolksdorf, editors, *Engineering Societies in the Agents World II : Second International Workshop, ESAW 2001 Prague, Czech Republic, July 7, 2001 Revised Papers*, pages 25–38, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.
- [Wardrop1952] J. Wardrop. Some theoretical aspects of road traffic research. *Proceedings of the Institution of Civil Engineers, Part II*, 1(36) :352–362, 1952.
- [Watkins *et al.*2011] Kari Watkins, Brian Ferris, Alan Borning, G. Scott Rutherford, and David Layton. Where is my bus? impact of mobile real-time information on the perceived and actual wait time of transit riders. *Transportation Research Part A*, 45 :839–848, 2011.
- [Weiss1999] G. Weiss. *Multiagent Systems, A Modern Approach to distributed Artificial Intelligence*. MIT Press, 1999.
- [Weyns and Michel2014] Danny Weyns and Fabien Michel. Agent environments for multi-agent systems - A research roadmap. In *Agent Environments for Multi-Agent Systems IV*

- 4th International Workshop, E4MAS 2014 - 10 Years Later, Paris, France, May 6, 2014, Revised Selected and Invited Papers, pages 3–21, 2014.
- [Weyns *et al.*2004] Danny Weyns, H.V.D Parunak, Fabien Michel, Tom Holvoet, and Jacques Ferber. Environments for multiagent systems, state-of-the-art and research challenges. In *Proceedings of Workshop on Environments for Multi-Agent Systems (E4MAS)*, pages 1–47, New York, NY (USA), 2004. Springer-Verlag.
- [Wilensky and Evanston1999] Uri Wilensky and I Evanston. Netlogo : Center for connected learning and computer-based modeling. *Northwestern University, Evanston, IL*, pages 49–52, 1999.
- [Wong and Sycara2000] H. C. Wong and K. Sycara. A taxonomy of middle-agents for the internet. In *Proceedings of the Fourth International Conference on MultiAgent Systems*, pages 465 – 466, 2000.
- [Wooldridge2002] Michael Wooldridge. *An Introduction to MultiAgent Systems*. John Wiley & Sons, Chichester (UK), February 2002. 256 pages.
- [Yan2012] Y Yan. Bus transit travel time reliability evaluation based on automatic vehicle location data. *Journal of Southeast University*, 28(1) :100–105, 2012.
- [Zargayouna and Zeddini2012] Mahdi Zargayouna and Besma Zeddini. Fleet organization models for online vehicle routing problems. In *Transactions on Computational Collective Intelligence VII*, pages 82–102. Springer Berlin Heidelberg, 2012.
- [Zargayouna *et al.*2012] Mahdi Zargayouna, Gerard Scemama, Besma Zeddini, Paul Kompfner, Patrick Gatellier, Patrick Constant, and Dirk Beckman. Future internet for a personal travel companion service. In *19th ITS World Congress, 2012*.
- [Zargayouna2005] Mahdi Zargayouna. Une représentation spatio-temporelle de l’environnement pour le transport a la demande. *Atelier : Représentation sur le temps et l’espace (RTE 2005), Plate-forme AFIA*, 2005.
- [Zargayouna2007] Mahdi Zargayouna. *Modèle et langage de coordination pour les systèmes multi-agents ouverts. Application au problème du transport à la demande*. PhD dissertation, University of Paris-Dauphine, Paris (France), 2007. In french.
- [Zargayouna2012] Mahdi Zargayouna. Multimodal journey optimisation enablers specifications. Technical report, Instant Mobility Project, 2012.
- [Zenker and Ludwig2010] Bjørn Zenker and Bernd Ludwig. Rose – an intelligent mobile assistant discovering preferred events and finding comfortable transportation links. In *ICAART 2010 - Proceedings of the International Conference on Agents and Artificial Intelligence*, 2010.
- [Zhang *et al.*2011] Zhang, Liping Li, Jing-Quan Zhou, and Kun Gupta. Traveler information tool with integrated real-time transit information and multimodal trip planning : Design and implementation. *Journal of the Transportation Research Board*, 2011.
- [Zhang *et al.*2012] Zhang, Feixiong Liao, Theo Arentze, and Harry Timmermans. A multimodal transport network model for advanced traveler information systems. *Journal of Ubiquitous Systems and Pervasive Networks*, 4(1) :21–27, 2012.

[Zografos and Androutsopoulos2008] K.G. Zografos and K.N. Androutsopoulos. Algorithms for itinerary planning in multimodal transportation networks. *IEEE Transactions on Intelligent Transportation Systems*, 9(1) :175–184, 2008.

Troisième partie

Annexes

Annexe A

Formats de données en entrée du simulateur

Sommaire

A.1	Formats des données du réseau de transport	135
A.2	Formats des données d'un tableau de marche	135

A.1 Formats des données du réseau de transport

Les Figures A.1 et A.1 présentent le schéma XSD d'un réseau de transport. Un réseau est composé de lignes de transport. Une ligne est composée d'un ensemble d'itinéraires. Un itinéraire est composé d'un ensemble d'arcs. Un arc a un traçage sous forme de longitude-latitude, et est composé d'un nœud de départ et d'un nœud d'arrivée. Enfin, chaque nœud est défini par son nom et ses coordonnées.

Si l'interprétation de ces données pour le réseau de transport public est intuitive, elle l'est moins pour le réseau piéton, où tout ce dont nous aurions besoin seraient des nœuds (les intersections, les carrefours) et les arcs (les rues, avenues, etc.). Nous avons décidé néanmoins de garder le même schéma pour les deux types de réseau, pour ne pas compliquer inutilement la définition des données en entrée. Les itinéraires pour le réseau piéton peuvent servir à donner de grandes directions pour des axes de transport, qui peuvent servir lors de l'information de voyageur. En effet, au lieu de dire au voyageur de prendre telle rue, on lui dira d'emprunter la rue en direction d'un itinéraire particulier. Ce schéma a donc un sens et une utilité, même pour les réseaux piétons.

Le fichier de correspondances suit également le même schéma. Si un cheminement piéton entre un nœud TC et un nœud piéton existe, alors un arc les reliant est ajouté au fichier XML. L'ensemble de ces arcs forme le graphe de correspondances.

A.2 Formats des données d'un tableau de marche

La Figure A.3 présente le schéma XSD d'un tableau de marche d'un véhicule de transport en commun.


```

- <xs:schema version="1.0">
- <xs:complexType name="label">
- <xs:sequence>
  <xs:element name="attribut" type="xs:string"/>
  <xs:element name="valeur" type="xs:string"/>
</xs:sequence>
</xs:complexType>
- <xs:complexType name="coordinate">
- <xs:sequence>
  <xs:element name="longitude" type="xs:double"/>
  <xs:element name="latitude" type="xs:double"/>
</xs:sequence>
</xs:complexType>
- <xs:complexType name="link">
- <xs:sequence>
  <xs:element name="idlink" type="xs:string"/>
  <xs:element name="nameLink" type="xs:string"/>
  <xs:element name="originStop" type="stop"/>
  <xs:element name="destinationStop" type="stop"/>
  <xs:element name="tracing" type="coordinate" nillable="true" minOccurs="0" maxOccurs="unbounded"/>
  <xs:element name="attributs" type="label" nillable="true" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
- <xs:complexType name="stop">
- <xs:sequence>
  <xs:element name="idStop" type="xs:string"/>
  <xs:element name="nameStop" type="xs:string"/>
  <xs:element name="location" type="coordinate" nillable="true" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
- <xs:element name="network">
- <xs:complexType>
- <xs:sequence>
  <xs:element name="idNetwork" type="xs:string"/>
  <xs:element name="nameNetwork" type="xs:string" minOccurs="0"/>
  <xs:element name="lines" type="line" nillable="true" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>

```

FIGURE A.1 – Schéma XSD d'un réseau de transport (1/2)

```

- <xs:complexType name="line">
- <xs:sequence>
  <xs:element name="idLine" type="xs:string" minOccurs="0"/>
  <xs:element name="nameLine" type="xs:string" minOccurs="0"/>
  <xs:element name="modus" type="xs:string" minOccurs="0"/>
  <xs:element name="itineraries" type="itinerary" nillable="true" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
- <xs:complexType name="itinerary">
- <xs:sequence>
  <xs:element name="idItinerary" type="xs:string"/>
  <xs:element name="nameItinerary" type="xs:string" minOccurs="0"/>
  <xs:element name="links" type="link" nillable="true" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:schema>

```

FIGURE A.2 – Schéma XSD d'un réseau de transport (2/2)

```

- <xs:schema version="1.0">
- <xs:complexType name="stopTime">
- <xs:sequence>
  <xs:element name="stop" type="stop" minOccurs="0"/>
  <xs:element name="visitTime" type="xs:dateTime" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
- <xs:complexType name="coordinate">
- <xs:sequence>
  <xs:element name="latitude" type="xs:double"/>
  <xs:element name="longitude" type="xs:double"/>
</xs:sequence>
</xs:complexType>
+ <xs:complexType name="stop"></xs:complexType>
- <xs:complexType name="mLine">
- <xs:sequence>
  <xs:element name="idMLine" type="xs:string"/>
  <xs:element name="nameMLine" type="xs:string" minOccurs="0"/>
  <xs:element name="missions" type="mission" nillable="true" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
- <xs:complexType name="mission">
- <xs:sequence>
  <xs:element name="courses" type="course" nillable="true" minOccurs="0" maxOccurs="unbounded"/>
  <xs:element name="idMission" type="xs:string"/>
  <xs:element name="nameMission" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
- <xs:complexType name="course">
- <xs:sequence>
  <xs:element name="idCourse" type="xs:string"/>
  <xs:element name="nameCourse" type="xs:string" minOccurs="0"/>
  <xs:element name="stopTimes" type="stopTime" nillable="true" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
- <xs:element name="timeTable">
- <xs:complexType>
- <xs:sequence>
  <xs:element name="idTimeTable" type="xs:string"/>
  <xs:element name="nameTimeTable" type="xs:string" minOccurs="0"/>
  <xs:element name="mLines" type="mLine" nillable="true" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

FIGURE A.3 – Schéma XSD d'un tableau de marche

Le tableau de marche est défini comme suit : Un *TimeTable* est composé d'un ensemble de lignes (appelés *MLine* pour les différencier des lignes de réseau définies plus haut). Une ligne est composée de missions. Une mission est la somme des trajets assignés à un véhicule de TC particulier. Une mission est donc composée de courses. Et chaque course est composée d'une séquence de couples ⟨arrêt, temps de passage⟩.

Annexe B

Résultats détaillés

Sommaire

B.1 Temps de parcours	139
--	------------

B.1 Temps de parcours

Cette annexe présente l'ensemble des résultats des temps de parcours, qui ont été présentés d'une manière synthétique dans le chapitre 5.

TABLE B.1 – Temps de parcours (1.000 voyageurs)

Scénario	Moyenne	Simulation 1	Simulation 2	Simulation 3	Simulation 4	Simulation 5
0%	0,45%	0,48%	0,45%	0,44%	0,45%	0,43%
20%	3,56%	3,82%	3,53%	3,44%	3,58%	3,41%
50%	5,15%	5,54%	5,11%	4,98%	5,19%	4,95%
80%	8,99%	9,67%	8,92%	8,69%	9,05%	8,64%
100%	4,95%	5,33%	4,91%	4,79%	4,99%	4,76%

TABLE B.2 – Temps de parcours (5.000 voyageurs)

Scénario	Moyenne	Simulation 1	Simulation 2	Simulation 3	Simulation 4	Simulation 5
0%	1,45%	1,52%	1,58%	1,55%	1,40%	1,18%
20%	8,25%	8,64%	9,00%	8,86%	7,98%	6,75%
50%	12,21%	12,80%	13,34%	13,13%	11,81%	9,99%
80%	11,63%	12,19%	12,69%	12,49%	11,24%	9,51%
100%	10,24%	10,74%	11,18%	11,01%	9,90%	8,38%

TABLE B.3 – Temps de parcours (10.000 voyageurs)

Scénario	Moyenne	Simulation 1	Simulation 2	Simulation 3	Simulation 4	Simulation 5
0%	7,56%	8,01%	7,74%	8,10%	7,30%	6,64%
20%	7,97%	8,45%	8,16%	8,54%	7,69%	7,00%
50%	15,64%	16,58%	16,01%	16,77%	15,10%	13,75%
80%	16,30%	17,28%	16,68%	17,47%	15,74%	14,32%
100%	12,99%	13,78%	13,30%	13,93%	12,54%	11,42%

TABLE B.4 – Temps de parcours (20.000 voyageurs)

Scénario	Moyenne	Simulation 1	Simulation 2	Simulation 3	Simulation 4	Simulation 5
0%	5,43%	5,39%	5,24%	5,32%	5,67%	5,53%
20%	8,43%	8,36%	8,12%	8,26%	8,80%	8,58%
50%	11,62%	11,53%	11,20%	11,39%	12,13%	11,83%
80%	10,43%	10,35%	10,05%	10,22%	10,89%	10,62%
100%	8,29%	8,23%	7,99%	8,12%	8,65%	8,44%

TABLE B.5 – Temps de parcours (30.000 voyageurs)

Scénario	Moyenne	Simulation 1	Simulation 2	Simulation 3	Simulation 4	Simulation 5
0%	3,44%	3,61%	3,17%	3,44%	3,12%	3,88%
20%	10,57%	11,08%	9,72%	10,57%	9,57%	11,91%
50%	14,40%	15,10%	13,24%	14,40%	13,03%	16,23%
80%	12,35%	12,94%	11,35%	12,35%	11,18%	13,91%
100%	8,82%	9,24%	8,11%	8,82%	7,98%	9,93%

Résumé :

Dans ce travail de thèse, nous proposons une simulation multi-agent pour mesurer l'impact de la fourniture d'informations sur la qualité des voyages en transports en commun, notamment dans des situations perturbées. D'une part, nous concevons une simulation qui assure le déplacement de voyageurs sur un réseau de transport en commun. D'autre part, nous l'enrichissons par l'intégration de l'information des voyageurs, y compris des voyageurs connectés à des sources d'information en temps-réel. Pour ce faire, nous nous fondons sur le paradigme multi-agent, qui est un modèle puissant pour la conception et l'implantation d'applications de transport. Pour répondre aux besoins de l'intégration de l'information des voyageurs, en particulier l'information individuelle, nous adoptons une approche centrée-environnement où l'environnement spatio-temporel multi-agent est l'interlocuteur des agents voyageurs et représente l'évolution dans le temps de l'état du réseau de transport en commun.

Dans nos expérimentations, nous considérons un réseau réel et des dizaines de milliers de voyageurs. Pour évaluer l'impact de la provision de l'information aux voyageurs, nous testons différents scénarios en fonction du pourcentage de voyageurs connectés. Ces scénarios simulés sont analysés suivant leur impact sur les temps de parcours moyens des voyageurs, connectés et non connectés. Les résultats montrent que le nombre de voyageurs connectés a un impact positif sur les temps de parcours jusqu'à un certain seuil, au delà duquel l'impact devient relativement négatif.

Abstract :

In this PhD work, we propose a multiagent simulation to measure the impact of information provision on the quality of passengers' travels, notably in case of disturbances. On the one side, we design and implement a simulation that ensure travelers movements in a transit network. On the other side, we enrich our model to integrate traveler information, including travelers that are connected to real-time information sources. To this end, we use the multi-agent paradigm, which is proven to be a powerful model to design and implement transportation applications. To deal with the integration of the traveler information system in the simulator, we adopt an environment-centered approach, where the space-time multiagent environment is the privileged interlocutor of the agents and represent the evolution of the transit network state over time.

In our experiments, we use real network data and tens of thousands of travelers. To assess the impact of information provision, we simulate different scenarios in function of the percentage of connected travelers. These simulated scenarios are analyzed following their impact on the average travel times of the travelers (connected and no-connected). Results show that the number of connected travelers has a positive impact on overall travel times up until a certain threshold before becoming relatively negative.