HAL
open science

Un modèle d'environnement pour la simulation multi-agents des déplacements en milieu urbain

Jocelyn Buisson

To cite this version:

Jocelyn Buisson. Un modèle d'environnement pour la simulation multi-agents des déplacements en milieu urbain. Modélisation et simulation. Université de Technologie de Belfort-Montbeliard, 2014. Français. NNT : 2014BELF0252 . tel-01420744

HAL Id: tel-01420744

https://theses.hal.science/tel-01420744v1

Submitted on 21 Dec 2016

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.

SPIM

Thèse de Doctorat

utbm
université de technologie Belfort-Montbéliard

école doctorale **sciences pour l'ingénieur et microtechniques**

UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

Un modèle d'environnement pour la simulation multi-agents de déplacement en milieu urbain

JOCELYN BUISSON

IRTES
sciences & ingénierie
Institut de Recherche sur les Transports, l'Énergie et la Société

voxelia

SPIM

Thèse de Doctorat

école doctorale **sciences pour l'ingénieur et microtechniques**

UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

N° 2 5 2

THÈSE présentée par

JOCELYN BUISSON

pour obtenir le

Grade de Docteur de
l'Université de Technologie de Belfort-Montbéliard

Spécialité : **Informatique**

Un modèle d'environnement pour la simulation multi-agents de déplacement en milieu urbain

Unités de Recherche :
Institut de Recherche sur les Transports, l'Énergie et la Société (IRTES),
Laboratoire Systèmes et Transports (IRTES-SET)

Soutenue publiquement le 19 décembre 2014 devant le Jury composé de :

RENÉ MANDIAU	Rapporteur	Professeur à l'Université de Valenciennes et du Hainaut-Cambrésis
FRANCK BOOCHS	Rapporteur	Professeur à l'Université des Sciences Appliquées de Mayence
ZAHIA GUESSOUM	Examinateur	Maître de Conférences HDR à l'Université Pierre et Marie Curie
ADNAN YASSINE	Examinateur	Professeur à l'Université du Havre
MIKAËL MARTINS GONÇALVES	Examinateur	Président de la société Voxelia
ABDERRAFIAA KOUKAM	Co-directeur de thèse	Professeur à l'UTBM
STÉPHANE GALLAND	Co-directeur de thèse	Maître de Conférences HDR à l'UTBM

Remerciements

Je tiens à remercier René Mandiau et Franck Boochs pour avoir accepté d'être les rapporteurs de cette thèse. Je les remercie pour l'intérêt qu'ils ont porté à mes travaux, pour leurs remarques et leur participation à mon jury de thèse.

Je souhaite également remercier Zahia Guessoum et Adnan Yassine pour avoir accepté de faire partie de mon jury.

Je remercie Aberrafiâa Koukam et Stéphane Galland pour leur encadrement, leur patience et la pertinence de leurs conseils ayant permis à ce manuscrit d'aboutir.

Merci à Mikaël Martins Gonçalves et Renan Zéo pour la confiance qu'ils m'ont témoignée et pour m'avoir permis de faire cette thèse avec leur entreprise malgré certains obstacles.

Merci également à Nicolas Gaud, Abdeljalil Abbas-Turki, Olivier Lamotte, Philippe Descamps, Vincent Hilaire et Fabrice Lauri pour leurs conseils et les discutions que nous avons pu avoir.

Un merci à toute ma famille et tout particulièrement à ma sœur Loren qui a toujours été là pour me soutenir et m'encourager.

Enfin, merci à mes collègues et amis qui partagent mon quotidien et me permettent de travailler dans la bonne humeur : Guillaume Delanery, Charles Gaumet et Florent Perronnet.

Sommaire

Introduction générale **1**

1 Simulation multi-agent en environnement urbain **7**

1.1 Fondamentaux sur la simulation 8

1.1.1 Principes généraux 8

1.1.2 Simulation de trafic 9

1.1.3 Simulation de foules 10

1.2 Systèmes multi-agents 12

1.2.1 Définition de la notion d'agent 12

1.2.2 Définition d'un système multi-agent 13

1.3 Simulation orientée-agent 14

1.3.1 Principes généraux 14

1.3.2 Comportement d'agent 16

1.3.3 Environnement 17

1.3.4 Interaction entre un agent et l'environnement physique 21

1.4 Conclusion 26

2 Modèles d'environnement physique **29**

2.1 Modèles de simulation de trafic 30

2.1.1 Outils commerciaux 31

2.1.2 Approches académiques 32

2.2 Modèles de simulation de foules 37

2.2.1 Modèles basés sur des grilles régulières 37

2.2.2 Modèles basés sur des graphes 38

2.2.3 Les modèles basés sur des arbres de décomposition spatiale 40

2.3 Critères de comparaison 41

2.4 Comparaison des modèles d'environnement 43

2.5 Conclusion 43

3 Structure du modèle HEDGE **47**

3.1 Modélisation de l'environnement par un graphe 48
3.1.1 Contexte 48
3.1.2 Modèle de l'environnement 49
3.2 Composantes structurelles de l'environnement 52
3.2.1 Les nœuds 53
3.2.2 Les connecteurs 58
3.2.3 Les liens 60
3.3 Exemples de modélisation 61
3.3.1 L'intersection 61
3.3.2 Le cas du dépassement et du changement de voie 63
3.3.3 Environnement intérieur et extérieur 65
3.4 Composantes non-structurelles de l'environnement 66
3.4.1 Les corps des agents 66
3.4.2 Les obstacles 67
3.4.3 La signalétique 68
3.5 Vers un modèle informé de l'environnement 68
3.6 Conclusion 69

4 Comportement dynamique du modèle HEDGE 71
4.1 Réaction de l'environnement 72
4.1.1 Principe de la réaction inspirée des lois de la Physique 73
4.1.2 Modèles physiques des corps des agents 74
4.2 Localisation des entités 75
4.2.1 Principe de localisation 76
4.2.2 Adaptation du cycle de simulation 78
4.3 Processus endogènes dans le modèle HEDGE 80
4.4 Conclusion 81

5 Simulations pour l'aménagement de la ville de Belfort 83
5.1 Logiciel Voxelia Simulate 84
5.1.1 Architecture du logiciel Voxelia Simulate 84
5.1.2 Éditeur et Générateur de graphe HEDGE 85
5.2 Contexte et objectifs des études 86
5.2.1 Contexte général 87
5.2.2 Contexte des projets de simulation 87
5.3 Définition du comportement des agents 88

5.3.1 Comportement des piétons et son extension aux cyclistes 88
5.3.2 Comportement des conducteurs de véhicules routiers 91
5.4 Définition du modèle physique des véhicules 94
5.4.1 Architecture du modèle de véhicule 94
5.4.2 Modèles sélectionnés pour les composants d'un véhicule 95
5.5 Validation de HEDGE 98
5.5.1 La place Rabin 98
5.5.2 Le quartier Denfert-Thiers 102
5.5.3 La Place d'Armes 104
5.6 Conclusion 107

Conclusion générale **109**

A Publications de l'auteur **137**
A.1 Journal international avec comité de lecture 137
A.2 Conférences internationales avec comité de lecture 137
A.3 Conférences nationales sans comité de lecture 138
A.4 Lecture Notes en cours d'évaluation 138

B Modèle physique « simple » d'un véhicule **139**

C Les projets d'aménagement de la ville de Belfort **141**
C.1 MOSDAU 141
C.2 MISDAU 143
C.3 Modèle d'évitement de collision des agents piétons et cyclistes 144

INTRODUCTION GÉNÉRALE

1/ CONTEXTE

La simulation des systèmes urbains constitue un enjeu majeur pour l'aménagement des territoires. En effet, elle permet d'étudier plusieurs aspects impactant la dynamique de l'ensemble urbain, tels que l'offre de mobilité, l'aménagement des infrastructures, la gestion du trafic et de la sécurité. L'exploitation de maquettes 3D de zones urbaines (ville, quartiers, etc.) au sein de la simulation offre aussi une aide précieuse pour la prise de décisions et une assistance à la conception de zones urbaines. C'est dans ce cadre que s'inscrivent les travaux menés dans cette thèse. Plus précisément, nous nous intéressons à la simulation des déplacements de piétons et de véhicules terrestres dans un milieu urbain en vue d'analyser les divers scénarii d'aménagement et leurs impacts sur la dynamique urbaine.

La simulation multi-agent est l'une des approches les plus adaptées pour reproduire la dynamique d'une population. Elle se réfère aux modèles qualifiés « d'individus-centrés » (Amblard, 2003) et fournit un processus permettant de modéliser et de simuler la dynamique de populations composées d'entités en interaction.

La thèse défendue dans ce mémoire s'intéresse particulièrement à la modélisation de l'environnement, considéré comme la composante principale de notre approche de simulation. En effet, le modèle de l'environnement doit nous permettre la simulation de piétons et de véhicules terrestres se déplaçant dans des zones extérieures (par exemple les rues) et/ou intérieures (les sites clos et les bâtiments). Pour ce faire, nos travaux proposent de capitaliser l'expérience acquise dans les domaines de la simulation 3D et des systèmes multi-agents pour la construction et le développement de modèles de l'environnement.

Nos travaux s'inscrivent également dans un contexte industriel imposant la prise en compte de contraintes telles que la facilité et l'efficacité de la modélisation, et la latence du simulateur afin de permettre l'enrichissement de l'offre logicielle de simulation 3D de la société Voxelia [1].

1. Nos travaux ont été réalisés au sein de la société Voxelia, jeune entreprise universitaire, spécialisée dans les services et l'édition de composants logiciels pour la simulation 3D et l'imagerie de synthèse.

2/ Problématiques et objectifs de ces travaux

L'objectif de cette thèse peut être résumé ainsi :

> Proposer un modèle d'environnement physique pour permettre la simulation multi-agent des déplacements de piétons et de véhicules en milieu urbain.

L'environnement est communément considéré comme l'une des parties essentielles d'une simulation multi-agent. Cependant, différentes perspectives existent quant au rôle qu'il joue dans un système multi-agent. Les environnements utilisés dans cette thèse peuvent également être considérés comme des cas particuliers « d'environnement physique » (Odell et al., 2002). La notion d'environnement physique se réfère à la classe de systèmes dans laquelle les agents, ainsi que les objets, disposent d'une position explicite et produisent des actions elles-aussi localisées (Ferber et al., 1996). Nous allons maintenant présenter nos contributions pour atteindre cet objectif.

2.1/ Modèle de l'environnement physique

La réflexion sur les déplacements ainsi que les modalités de leur gestion reposent sur la prise en compte de la complexité des politiques de mobilité en ville (Allemand et al., 2004). Cette complexité amène naturellement à considérer la diversité des modes de déplacement mis à la disposition des individus. Plusieurs modèles ont été proposés pour la simulation des déplacements. Cependant, aucun ne répond pleinement à nos exigences. En effet, la plupart de ces modèles ne considère qu'un seul type d'entité (piéton ou véhicule), ou ne permettent de simuler qu'une seule topologie d'environnement (intérieur de bâtiment ou extérieur uniquement). Parmi ceux spécialement adaptés aux véhicules routiers, très peu sont susceptibles de permettre l'incorporation d'un modèle physique. Par conséquent, nous proposons un nouveau modèle d'environnement physique, nommé HEDGE (HEterogenous Dual Graph Environment), répondant aux problématiques posées. HEDGE structure l'environnement selon deux aspects distincts : la navigation et la perception. Cette structuration permet aux agents de percevoir toutes les entités, quel que soit leur type, tout en leur attribuant des zones de déplacement spécifiques. L'ensemble des abstractions proposées permet de modéliser un large panel de topologies comprenant à la fois l'intérieur et l'extérieur des bâtiments. Cette décomposition permet également la cohabitation d'agents hétérogènes dans le même environnement et une gestion simplifiée de zones mixtes telles que les passages piétons. Notre modèle permet de simuler la dynamique des objets de l'environnement (incluant les corps des agents) en utilisant une approche inspirée de la Physique. L'intérêt de réaliser des calculs inspirés de la Physique dans l'environnement est partagé par (Gechter et al., 2012). En effet, elle facilite la reproduction réaliste des mouvements des mobiles dans l'espace, notamment des véhicules (Hoogendoorn et al., 2001). Cette approche répond également à la problématique de détermination de la réaction de l'environnement face aux actions initiées par les agents dans le cadre d'une simulation.

Enfin, le modèle proposé est conçu pour permettre une génération manuelle et rapide du graphe représentant l'environnement à partir d'un ensemble restreint de sources de

données numériques. Il s'agit d'une contrainte imposée par le cadre industriel dans lequel s'inscrit cette thèse. Pour ce faire, nous avons développé des outils permettant d'assister l'utilisateur dans la création du modèle.

2.2/ Extension de la plate-forme Simulate et application

Afin de concrétiser notre approche de modélisation de l'environnement, nous avons contribué au développement de la plate-forme de simulation Simulate. En particulier, nous avons implanté les outils nécessaires à la mise en œuvre du modèle HEDGE. L'objectif de cette plate-forme est de fournir un environnement logiciel permettant de simuler les différents modes de déplacement dans une ville (piétons, véhicules, bus, vélos). Cette plate-forme s'inscrit dans le cadre du partenariat avec la société Voxelia qui conçoit, développe et utilise ces outils pour réaliser des études d'aménagement urbain.

En s'appuyant sur cette plateforme, nous avons expérimenté notre modèle dans le cadre de la simulation des déplacements de piétons et des véhicules dans les zones névralgiques de la ville de Belfort et de sa communauté d'agglomérations. L'objectif est d'étudier les différents scénarii d'aménagement des infrastructures pour la mise en place d'un nouveau réseau de transports en commun. Les simulations réalisées ont mis en évidence des problèmes structurels (géométrie de voie inappropriée, etc.) et des problèmes fonctionnels (congestion, synchronisation des feux, etc.).

3/ Description du plan de la thèse

Après la présentation de nos propositions dans le contexte de cette thèse, cette section introduit le plan de ce mémoire. Selon les objectifs énoncés dans la section 2, cette thèse est organisée en six chapitres. La figure 1 résume l'organisation du mémoire.

Le **chapitre 1** présente les éléments théoriques des deux domaines sous-jacents aux travaux présentés dans cette thèse : la simulation des déplacements en milieu urbain et les systèmes multi-agents.

Le **chapitre 2** présente et compare différents modèles de l'environnement physique dans le cadre des applications réalisées dans cette thèse.

Le **chapitre 3** introduit les concepts fondamentaux de notre modèle de l'environnement fondé sur les graphes. Dans l'optique d'autoriser la modélisation de différentes topologies et de rendre cette modélisation facile et efficace, nous proposons des types de nœuds et d'arcs permettant de construire une représentation de l'environnement intelligible et performante dans le cadre de projets de simulation.

Le **chapitre 4** décrit la dynamique de l'environnement. L'un des problèmes récurrents des modèles de l'environnement concerne la détermination de la réaction de l'environnement aux actions initiées par les agents. Nous adoptons une approche de modélisation inspirée de la Physique afin de répondre spécifiquement à cette problématique.

Le **chapitre 5** présente la mise en œuvre des travaux présentés dans le cadre de deux projets d'aménagement de la ville de Belfort et de sa communauté d'agglomérations. Ces deux projets ont permis d'instancier notre modèle de l'environnement et de réaliser des simulations interactives afin de mettre en place un nouveau système de transport urbain.

Enfin, une conclusion présente un bilan de nos travaux de recherche et adresse quelques perspectives possibles.

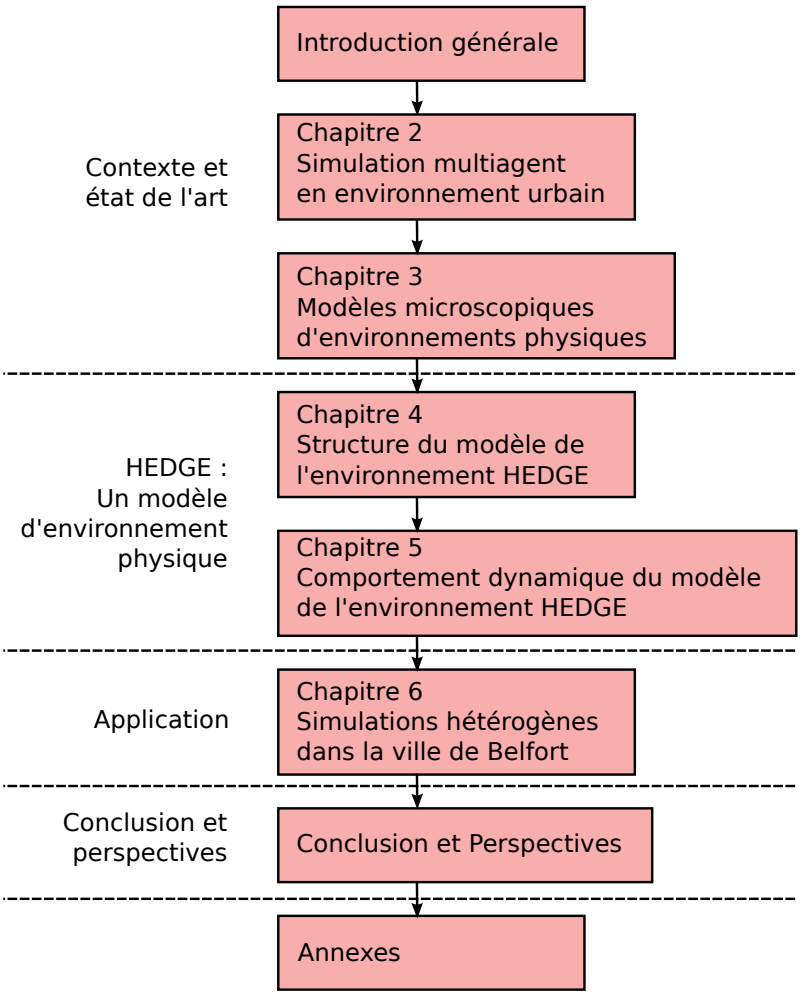

FIGURE 1 – Plan de la thèse

1

Simulation multi-agent en environnement urbain

L'étude des phénomènes liés aux déplacements individuels suscite un intérêt important dans l'aménagement des sites urbains, l'étude de la sécurité, l'architecture, et l'analyse des flux, où les deux classes d'applications dominantes sont la simulation d'évacuations et l'analyse des flux des déplacements d'un grand nombre d'individus dans un environnement intérieur ou extérieur (Thalmann et al., 2013). L'objectif de ces simulations est d'aider les décideurs et les experts à comprendre la « relation entre l'organisation de l'espace et les comportements humains » (Okazaki et al., 1993). Les environnements considérés sont hétéroclites : les rues (Farenc et al., 1999; Thomas et al., 2000b), les bâtiments (Braun et al., 2005a; Thompson et al., 1995b), les métros (Hareesh, 2000), les bateaux (Klüpfel et al., 2000), les avions (Owen et al., 1998), les stades (Still, 2000) ou encore les aéroports (Szymanezyk et al., 2011).

Les systèmes multi-agents (SMA) revêtent de plus en plus d'importance pour leur capacité à aborder les systèmes complexes, et notamment les systèmes urbains. Le domaine des SMA peut être vu comme le confluent de plusieurs disciplines de recherche : l'intelligence artificielle (Weiss, 1999) pour les aspects décisionnels de l'agent, l'intelligence artificielle distribuée (Fagin, 1995) et plus généralement les systèmes distribués pour les interactions entre agents et la distribution de la résolution et de l'exécution, et enfin le

Sommaire

1.1 Fondamentaux sur la simulation 8
1.1.1 Principes généraux 8
1.1.2 Simulation de trafic 9
1.1.3 Simulation de foules 10
1.2 Systèmes multi-agents 12
1.2.1 Définition de la notion d'agent 12
1.2.2 Définition d'un système multi-agent 13
1.3 Simulation orientée-agent 14
1.3.1 Principes généraux 14
1.3.2 Comportement d'agent 16
1.3.3 Environnement 17
1.3.4 Interaction entre un agent et l'environnement physique 21
1.4 Conclusion 26

génie logiciel (Jennings, 2000; Shoham, 1994) pour l'approche de modélisation orientée-agent et la création de composants logiciels autonomes.

L'intérêt de ces systèmes est double. D'une part, ils contribuent à l'analyse des mécanismes produits lorsque des entités autonomes interagissent entre elles. Les SMA permettent alors de modéliser, expliquer et simuler des phénomènes naturels. Ils suscitent également des modèles d'auto-organisation. Par exemple, ils peuvent être utilisés pour modéliser l'ensemble des foyers au sein d'une ville, et ainsi déterminer l'évolution des migrations intra-urbaines (Vanbergue, 2003a).

D'autre part, ils sont utilisés pour la réalisation de systèmes complexes via les concepts d'agent, de communication, de coopération et de coordination d'actions (Jarras et al., 2002). Le travail en collaboration d'un groupe d'individus (exemples : des chefs de projets, des ingénieurs, des experts, etc.) sur un projet est un exemple de système complexe pouvant être modélisé par les SMA.

L'objectif de ce chapitre est de définir les concepts et les principes de base de la simulation multi-agent d'environnements urbains. Il se divise en quatre sections. Nous commençons par rappeler les fondamentaux de la simulation, et plus particulièrement de la simulation de trafic et de foules. Dans la section 1.2 nous présentons les définitions et principes liés aux SMA. Le chapitre se poursuit par la présentation des principes et des fondamentaux de la simulation orientée-agent. Enfin, nous concluons ce chapitre en décrivant les problématiques qui n'ont pas été résolues par les approches exposées.

1.1/ Fondamentaux sur la simulation

Dans cette section, nous rappelons les principes et les fondamentaux de la simulation. Ensuite, nous présentons les points clés de la simulation de trafic et de foules.

1.1.1/ Principes généraux

Shannon (1977) définit la simulation comme « le processus permettant de concevoir un modèle d'un système réel et de mener des expérimentations sur la base de ce modèle pour comprendre le comportement du système ou évaluer différentes stratégies pour son fonctionnement (dans les limites imposées par un critère ou un ensemble de critères) » [1].

L'objectif de la simulation est de faciliter la compréhension de la dynamique d'un système et tenter d'en prédire l'évolution. Satisfaire cet objectif nécessite l'élaboration d'un modèle du système à étudier, son exécution sur un calculateur, et l'analyse des résultats de cette exécution (Fishwick, 1997).

Le modèle de simulation désigne généralement l'ensemble des mécanismes qui gèrent les changements d'état du système. Il correspond à l'ensemble des lois, conditions ou contraintes qui définissent le comportement du système, ainsi que la manière dont ses composantes sont agrégées. L'exécution, quant à elle, doit faire évoluer dans le temps le modèle du système (Coquillard et al., 1997). Pour y parvenir, elle est généralement associée à un ensemble d'outils qui constituent le simulateur.

1. la citation originelle est : ``*the process of designing a model of a real system and conducting experiments with this model for the purpose either of understanding the behaviour of the system or of evaluating various strategies (within the limits imposed by a criterion or a set of criteria) for the operation of the system.*´´

Les modèles de simulation peuvent être classés en trois grandes familles : macroscopique, microscopique et mésoscopique.

Les modèles de simulation macroscopiques sont basés sur des relations déterministes entre la circulation, la vitesse et la densité de population, constituant la foule ou le trafic routier simulé (Helbing et al., 1998). La simulation d'un modèle macroscopique se focalise principalement sur des régions ou des populations plutôt que sur les comportements individuels. Ces modèles ont été initialement développés pour modéliser le trafic dans les réseaux de transport, tels que les autoroutes, les réseaux de rues, et les routes rurales. Cette approche permet la simulation d'une très grande population avec un relativement faible coût de calcul. Toutefois, en raison de son haut niveau de représentation, les résultats sont agrégés, imprécis et liés à la taille de la population simulée.

Les modèles de simulation microscopiques s'intéressent aux mouvements des personnes sur la base de comportements dynamiques et individuels. Citons par exemple les comportements de véhicule-suiveur et de changement de voie pour représenter des conducteurs, ou les comportements basés sur des forces pour les piétons (Dey et al., 2007; Galland et al., 2009; Razavi et al., 2011b; Reynolds, 1987; Thalmann et al., 2013). Ces modèles sont efficaces pour l'évaluation des conditions d'encombrement et de saturation d'un système, l'étude de la configuration topologique du système, et l'évaluation des impacts des comportements individuels sur ce système. Cependant, ces modèles sont difficiles à mettre en œuvre, coûteux en termes de temps de calcul, et peuvent être difficiles à calibrer.

Les modèles mésoscopiques combinent les propriétés des modèles de simulation microscopique et macroscopique. Par exemple, ils peuvent se focaliser sur des entités composant le système en utilisant des modèles ne permettant pas de distinguer les individus les uns des autres comme les modèles particulaires (Schaefer et al., 1998) ; en regroupant les individus au sein d'entités de plus haut niveau comme des groupes de piétons (Gaud et al., 2008) ; ou encore en utilisant un modèle discret de l'environnement comme les automates cellulaires (Karafyllidis et al., 1997).

1.1.2/ Simulation de trafic

La simulation de trafic est la classe d'applications permettant la modélisation et la reproduction de la dynamique de systèmes de transport. Elle est généralement utilisée dans le cadre de la conception, de l'aide à la planification, et l'exploitation des systèmes de transport (Barceló, 2010). Elle permet l'étude de modèles trop complexes pour le traitement analytique ou numérique. De plus, elle peut être utilisée lors d'études expérimentales pour mettre en avant les comportements dynamiques qui ne peuvent être considérés par les approches analytiques. Un autre point fort des modèles et des outils de simulation de trafic est leur capacité à produire des résultats pouvant être visualisés dans des outils attractifs et intuitifs (visualisation en trois dimensions, etc.).

Que ce soit pour la planification ou l'exploitation, de nombreux modes de transport peuvent être considérés dans un modèle de la simulation de trafic. Dans le cadre de cette thèse, nous nous intéresserons uniquement aux modes de transport terrestres. Toutefois, nous considérons de le modèle présenté dans cette thèse peut être adapté au transport fluvial et aérien.

Les modèles de simulation de trafic peuvent utiliser des approches théoriques différentes

(Barceló, 2010). Dans le cadre de cette thèse, nous n'abordons pas les modèles basés sur la méthode de Monté Carlo (Du et al., 2009; Rubinstein et al., 2011), les automates cellulaires ou la simulation à événements discrets (Wainer, 2009).

Les modèles orientés-agent (``*Agent-based model*´´ - ABM) sont maintenant largement utilisés pour la simulation de trafic (Macal et al., 2005; Niazi et al., 2011). Ils permettent d'analyser et comprendre les phénomènes complexes liés aux interactions entre les différentes entités constituant le trafic. Ainsi, ils réduisent la complexité du système à la modélisation d'agents autonomes en interaction et en déduisent les règles régissant un tel système. Nous détaillons les principes des systèmes multi-agents [2] dans la section 1.2.

1.1.3/ Simulation de foules

La simulation de foules est la classe d'applications dans laquelle les comportements de déplacements d'une foule d'individus sont reproduits. Ces deux dernières décennies, les chercheurs d'un large éventail de domaines tels que l'architecture (Chelhorn et al., 1999; Penn et al., 2001; Turner et al., 2002), l'infographie (Bouvier et al., 1996; Braun et al., 2003; Hodgins et al., 1994; Musse et al., 2001; Tecchia et al., 2002; Ulicny et al., 2002), la physique (Farkas et al., 2002; Helbing et al., 2000, 1995), la robotique (P. et al., 2001), la science de la sécurité (Still, 2000; Thompson et al., 1995a), les systèmes de formation (Bottaci, 1995; Varner et al., 1998; Williams, 1995), et la sociologie (Jager et al., 2001; Tucker et al., 1999) ont utilisé des simulations impliquant des collections d'individus. Plusieurs techniques de modélisation de la foule existent déjà. Toutefois, des problématiques scientifiques et technologiques restent d'actualité (Thalmann et al., 2013; Ulicny et al., 2006) : (i) Les approches existantes sont souvent axées sur les situations de panique plutôt que les comportements habituels des individus ; (ii) Les comportements sont généralement ceux des individus, les groupes ou les familles sont rarement traitées ; (iii) Il est nécessaire d'utiliser des méthodes distinctes du modèle de simulation de foules pour calibrer a priori les déplacements des individus au sein de la foule, et pour affecter les objectifs aux individus ; (iv) Les méthodes existantes sont souvent complexes et nécessitent de nombreux ajustements des paramètres pour obtenir des résultats probants.

Afin d'obtenir une application convaincante basée sur la simulation de foules dans un environnement virtuel, divers aspects de la simulation doivent être abordés : l'animation comportementale, la modélisation de l'environnement et la modélisation des représentations graphiques. Les travaux récents ont montré qu'il est nécessaire de produire un modèle de simulation approximant correctement le système réel étudié et une représentation graphique réaliste des entités constituant le système (Fuchs et al., 2006; Thalmann et al., 2013). En effet, si le rendu n'est pas satisfaisant, même le meilleur modèle de comportement ne pourra pas être convaincant pour l'utilisateur. A l'opposé, si le modèle de comportement n'approxime pas correctement le comportement de l'individu réel, les entités simulées seront perçues comme stupides et cela même si le module d'affichage produit des images de qualité photo-réaliste. Enfin, si le modèle de l'environnement n'approxime pas de manière suffisamment précise l'environnement réel, l'utilisateur et les entités simulées pourront réaliser des actions semblant être incohérentes.

2. La communauté en Science du Transport utilise le terme « Modèle orienté-agent ». La communauté en Science Informatique utilise le terme « Système multi-agent ».

L'utilisation la plus commune des simulateurs de foules est la modélisation du comportement des foules en cas d'évacuation forcée d'un environnement confiné en raison d'une menace. Dans une telle situation, un certain nombre de personnes doivent évacuer la zone donnée, généralement par un nombre relativement restreint de sorties fixes. Ces simulations tentent de donner une réponse aux questions telle que : La région peut-elle être évacuée dans un délai prescrit ? Où sont les zones susceptibles de contenir une densité d'individus trop importante (Robbins, 1999) ? L'approche de modélisation la plus connue dans ce domaine est l'utilisation des automates cellulaires servant à la fois de représentation des individus et de l'environnement.

Simulex (Thompson et al., 1995b) est un modèle simulant les déplacements d'individus pour la simulation de l'évacuation de bâtiments. Le modèle de l'environnement représente de grands espaces ouverts, géométriquement complexes qui sont définis par les plans 2D des étages et un ensemble d'escaliers les reliant. Chaque individu possède des attributs tels que sa position, son orientation, sa taille corporelle, et sa vitesse de marche. Différents algorithmes sont utilisés pour calculer les déplacements : carte mentale des distances entre deux points, recherche du plus court chemin, dépassement, changement de voie, et ajustement de la vitesse.

Still (2000) a proposé une collection de programmes nommée « Legion » pour la simulation et l'analyse de la dynamique d'une foule lors l'évacuation d'environnements contraints (stade, bâtiment, etc.). La dynamique des mouvements est modélisée par un automate cellulaire. Toutefois, chaque personne dans la foule est traitée comme un individu autonome qui détermine sa position en percevant son environnement proche et choisit une action appropriée à réaliser.

Helbing et al. (1995) a proposé un modèle inspiré des lois de la Physique et utilisant des forces socio-psychologiques pour décrire les comportements des individus lors de situations de panique (Helbing et al., 2000; Werner et al., 2003). Le modèle est basé sur un système de particules dans lequel chaque particule i de masse m_i a une vitesse prédéfinie v_i^0 (sa vitesse de consigne) dans une direction donnée $\mathbf{e}_i^0$. Pour atteindre cette consigne, chaque particule tend à adapter sa vitesse instantanée $\mathbf{v}_i$ dans l'intervalle de temps τ (premier terme de l'équation 1.1). Dans le même temps, chaque particule essaie de garder une distance minimale avec chacune des autres particules j et des murs w se trouvant dans son voisinage. Pour cela les forces de répulsion $\mathbf{f}_{ij}$ et $\mathbf{f}_{iw}$ sont calculées (deuxième et troisième termes de l'équation 1.1). La modification de la vitesse en fonction du temps τ est donnée par l'équation :

$$m_i \frac{dv_i}{dt} = F_i^{(H)} = m_i \frac{v_i^0 \mathbf{e}_i^0 - \mathbf{v}_i(t)}{\tau_i} + \sum_{j \neq i} \mathbf{f}_{ij} + \sum_{w} \mathbf{f}_{iw} \tag{1.1}$$

Braun et al. (2003) a étendu le modèle de Helbing et al. (1995), $F_i^{(H)}$, afin de pouvoir simuler des comportements d'individus différents, et de reproduire des comportements de groupes (Braun et al., 2005b). Dans ce travail, la population des agents est composée d'individus aux comportements hétérogènes et possédants des propriétés physiques et comportementales différentes.

Buisson et al. (2013) a proposé un modèle inspiré des lois de la Physique utilisant des forces de glissement pour remplacer les forces de répulsion proposées par Helbing et al. (1995). Cette approche permet de mieux prendre en compte l'inertie dans le mouvement de l'individu. Il répond également au problème de la détermination de $\mathbf{e}_i^0$. Ce modèle est

détaillé dans l'annexe C.3.

Van den Berg et al. (2008) considère le cas dans lequel chaque agent navigue indépendamment sans communication explicite avec d'autres agents. La formulation adoptée prend en compte le comportement réactif des autres agents en supposant implicitement qu'ils tenteront également d'éviter une collision. Chaque agent construit une représentation de son environnement dans laquelle ce dernier est décomposé en zones « sures » et en zones « non sures ». Chaque agent calcule les enveloppes géométriques g_j représentant les mouvements de ses voisins j. Les zones non sures sont celles qui possèdent une intersection avec l'enveloppe géométrique v_i du déplacement de l'agent. Dans ce cas, le modèle considère qu'il y a un potentiel de collisions avec l'un des voisins (équation 1.2).

$$z_{\neg sure} = \bigcup_{c=intersect(g_j,v_i)} c \tag{1.2}$$

Cette méthode garantit des mouvements sûrs et sans oscillation dans les déplacements d'un agent.

Une grande majorité des travaux traitant de la simulation de foules considérant les déplacements de chaque individu utilise une approche orientée-agent (Braun et al., 2003; Buisson et al., 2013; Gaud et al., 2008; Helbing et al., 2005; Jian et al., 2005; Karamouzas et al., 2009b; Paris et al., 2007b; Van den Berg et al., 2008). L'approche multi-agent est particulièrement bien adaptée à l'étude des dynamiques de déplacement de populations. Les systèmes multi-agents s'avèrent plus flexibles que les modèles macroscopiques à base d'équations différentielles pour simuler des phénomènes spatiaux et évolutifs (Bretagnolle et al., 2003).

Nous décrivons les principes des systèmes multi-agents dans la section suivante.

1.2/ Systèmes multi-agents

Les systèmes multi-agents (SMA) sont considérés comme des sociétés composées d'entités autonomes et indépendantes, appelées agents, qui interagissent en vue de résoudre un problème ou de réaliser collectivement une tâche. Grâce à la généricité de ces concepts, les domaines d'application des SMA sont vastes (Jennings et al., 1998). Les SMA offrent aussi un cadre pour la modélisation et la simulation de systèmes complexes. Notamment, les travaux dans le domaine de la simulation de trafic et de foules illustrent l'usage important des SMA.

Dans la suite de cette section, nous définissons les notions d'agent et de système multi-agent. Les principes de la simulation orientée-agent sont présentés dans la section 1.3.

1.2.1/ Définition de la notion d'agent

L'une des définitions les plus célèbres de la notion d'agent a été formulée par Russell et al. (1995). Ils considèrent un agent comme « Tout ce qui peut être vu comme percevant son environnement à l'aide de capteurs et agissant sur cet environnement à l'aide

d'effecteurs, de façon autonome ». Cette définition très générale et volontairement minimaliste dans sa formulation (uniquement) a été étendue notamment par Ferber (1995) (définition 1) pour, en outre, accentuer l'importance de l'environnement (qui demeurait rarement spécifié).

Définition 1 : Agent (Ferber, 1995)

L'agent est une entité physique ou virtuelle :

1. qui est capable d'agir dans un environnement ;
2. qui peut communiquer directement avec d'autres agents ;
3. qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou de fonctions de satisfaction, voire de survie, qu'elle cherche à optimiser) ;
4. qui possède des ressources propres ;
5. qui est capable de percevoir son environnement (mais de manière limitée) ;
6. qui ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune) ;
7. qui possède des compétences et offre des services ;
8. qui peut éventuellement se reproduire ;
9. et dont le comportement tend à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit.

1.2.2/ Définition d'un système multi-agent

La plupart des auteurs s'accordent généralement pour définir un système multi-agent (SMA) comme un système composé d'agents qui communiquent et collaborent pour achever des objectifs spécifiques personnels ou collectifs. La communication implique l'existence d'un espace partagé support de cette communication. Cet espace est généralement qualifié d'Environnement. Nous considérons particulièrement la définition 2 proposée par Ferber (1995). En effet, nous pensons que cette définition met particulièrement en avant le concept d'environnement.

Un SMA est une société organisée d'agents dans laquelle un certain nombre de phénomènes peuvent émerger comme la résultante des interactions entre les agents [3]. Cette notion d'émergence est essentielle dans les SMA, car c'est l'une des propriétés qui les rendent si aptes à modéliser les systèmes complexes.

3. Cette résultante n'est d'ailleurs pas forcément prévisible.

Définition 2 : Système multi-agent (Ferber, 1995)

Un système multi-agent est un système composé des éléments suivants :

- Un environnement E, c'est-à-dire un espace disposant généralement d'une métrique.
- 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.
- Un ensemble A d'agents qui sont des objets particuliers ($A \subseteq O$), lesquels représentent les entités actives du système.
- Un ensemble de relations R qui unissent des objets (et donc des agents) entre eux.
- Un ensemble d'opérations Op permettant aux agents de A de percevoir, produire, consommer, transformer, et manipuler des objets de O.[a]
- 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.

a. Cela correspond entre autres à la faculté des agents de percevoir leur environnement, de manger, etc.

1.3/ Simulation orientée-agent

La simulation multi-agent se réfère aux modèles microscopiques ou qualifiés individu-centrés (Amblard, 2003) et fournit un processus permettant de modéliser et simuler la dynamique de populations composées d'individus en interaction.

La simulation orientée-agent assimile un individu à un agent. La simulation multi-agent a été appliquée à un grand nombre de domaines tels que la robotique (Drogoul, 1993; Kitano et al., 1997), l'éthologie (Drogoul et al., 1999), l'écologie et la biologie, ou les sciences sociales (Conte et al., 1998; Gilbert et al., 2005).

La perspective adoptée se situe au niveau de l'individu ; la dynamique du système est issue des interactions entre les individus. La structure du système est considérée comme émergente de ces interactions.

De manière générale, on peut considérer que les techniques de simulation microscopique offrent un niveau de précision important dans les comportements simulés mais au détriment d'un coût calculatoire plus important. En outre, elles nécessitent des données précises et nombreuses pour leur initialisation, ce qui les rend de fait difficilement applicables à des systèmes de grande échelle.

1.3.1/ Principes généraux

Le modèle proposé par Michel (2004) pour la modélisation et la simulation de systèmes multi-agents constitue la base pour la conception d'un simulateur multi-agent. Il adopte une approche multi-vue et distingue quatre aspects fondamentaux dans un modèle de simulation multi-agent :

— **Comportements :** cet aspect traite de la modélisation des processus de délibération des agents (leurs « esprits »).
— **Environnement :** ce point de vue vise à définir les différents objets physiques du monde simulé (l'environnement situé et le « corps » des agents) ainsi que la dynamique endogène de l'environnement.
— **Ordonnancement :** cet aspect concerne la modélisation de l'écoulement du temps et la définition de l'ordonnancement utilisé pour exécuter le comportement des agents.
— **Interaction :** cette vue s'intéresse plus particulièrement à la modélisation du résultat des actions et des interactions à un instant donné.

La figure 1.1 illustre les relations entre ces quatre aspects fondamentaux d'une simulation multi-agent.

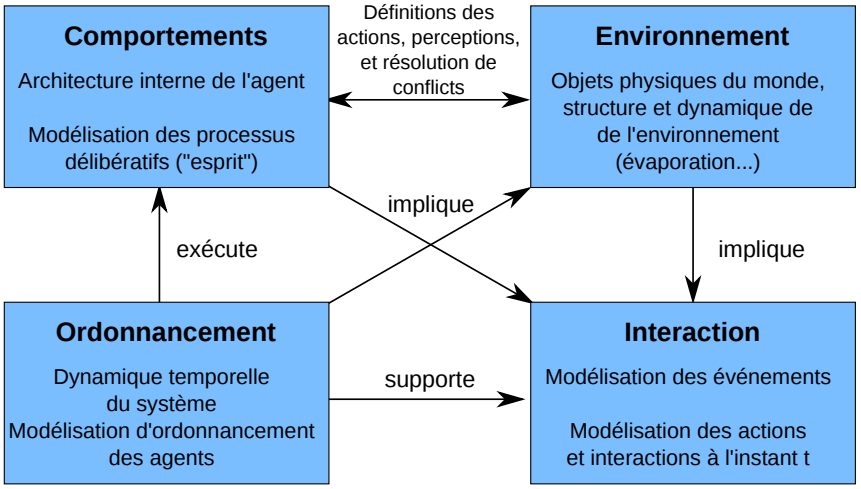

FIGURE 1.1 – Les quatre aspects d'un modèle de simulation multi-agent selon Michel (2004)

La modélisation et l'implantation de chacun de ces aspects et de leurs relations sont autant de points délicats qui soulèvent les problématiques suivantes :

— **Respecter la contrainte de localité :** un agent est une entité dont les perceptions et les actions n'ont qu'une portée locale. Deux approches principales existent pour respecter cette contrainte :

  a) L'approche discrète (centrée environnement) où la discrétisation de l'environnement sous forme de zones définit la granularité des perceptions et des actions des agents.

  b) L'approche continue (centrée agent) où la portée de chaque perception et de chaque action fait l'objet d'un traitement particulier qui est fonction de la nature et des caractéristiques de l'agent concerné (Weyns et al., 2004b).

  Ces deux approches peuvent également être combinées.

— **Respecter la contrainte d'intégrité environnementale :** un agent ne doit pas être en mesure de modifier directement les variables d'état de l'environnement.
— **Biais de simulation et simultanéité des actions :** pour éviter d'introduire des biais dans la simulation, il est nécessaire de disposer d'un modèle de gestion de l'action des agents et du temps, qui permettent de modéliser la simultanéité de deux événements. Un modèle de simulation ne doit pas être lié à une implantation

particulière (Zeigler et al., 2000). L'ordre dans lequel les agents sont activés influe sur la dynamique du système et peut entraîner des biais de simulation.

Dans la suite de cette section, nous nous intéressons particulièrement au comportement des agents et à la modélisation de l'environnement.

1.3.2/ Comportement d'agent

Le comportement d'un agent tend à satisfaire les objectifs de celui-ci, en tenant compte des ressources et des compétences à sa disposition, et des représentations et des perceptions accessibles. La principale hypothèse dans la théorie normative proposée par Hoogendoorn et al. (2002) est que toutes les actions d'un agent, exerçant une activité ou suivant un trajet, auront une utilité (ou dés-utilité) pour lui. Les individus pourront prévoir et optimiser l'utilité totale, compte tenu de l'incertitude sur les conditions de déplacement attendues (semblable à la théorie du consommateur micro-économique (Franck, 2002)). En choisissant parmi les options disponibles, ils tentent de réaliser une utilité nette maximale, qui est la somme des utilités des activités d'interprétation et des dés-utilités des efforts liés aux déplacements vers les lieux d'activités. Cette notion est appelée maximisation de l'utilité subjective : chaque individu a sa propre vision subjective de la situation. Notre hypothèse est que tous les agents se déplaçant dans l'univers virtuel possèdent des comportements tendant à cette maximisation.

L'architecture d'un agent dans notre cadre d'application peut être composée de trois couches, illustrées sur la figure 1.2 (Hoogendoorn et al., 2001; Thalmann et al., 2013; Van den Berg et al., 2008) :

— **Couche stratégique :** Au niveau stratégique, les individus décident des activités à réaliser dans l'univers. Si certaines de ces activités peuvent être discrétionnaires (par exemple l'achat d'un journal), d'autres peuvent être obligatoires (validation d'un billet avant d'accéder à un train). L'ensemble des choix peut être lié aux caractéristiques environnementales (type et emplacement des magasins, etc.). Dans cette couche, les agents utilisent en général une architecture de sélection des actions comme l'architecture ``*Belief-Desire-Intention*´´ (BDI), ou les modèles ``*Goal Oriented Action Planning*´´ (Orkin, 2003)

— **Couche tactique :** Le niveau tactique concerne les décisions à court ou moyen termes devant être prises par les agents en utilisant comme référence les décisions du niveau stratégique. À partir des objectifs donnés par ce dernier, le modèle de la couche tactique doit construire un plan d'action détaillé. Les localisations des différentes activités et les chemins pour atteindre ces localisations sont déterminés à ce niveau. L'architecture BDI peut être utilisée pour construire la séquence d'actions. Les algorithmes de recherche de plus court chemin peuvent former la base de calcul des trajets à emprunter sous la forme d'une séquence de zones de l'environnement : A* (Dechter et al., 1985), D* (Stentz, 1995).

— **Couche opérationnelle :** Au niveau opérationnel, les individus prennent des décisions à très court terme. Ces décisions sont orientées par celles fournies par la couche précédente. Les modèles appartenant à cette couche décident de la trajectoire d'un individu, de sa vitesse ou de son accélération. Contrairement à Hoogendoorn et al. (2001) qui ne considère que le choix de la meilleure vitesse pour un individu, nous pensons que le modèle opérationnel d'un individu a pour principal objectif d'éviter les collisions avec les objets de l'environnement ou les autres individus l'entourant. En d'autres termes, nous considérons que les modèles de

la couche opérationnelle ont pour objectif de déterminer les mouvements et les actions que devra réaliser le corps d'un individu.

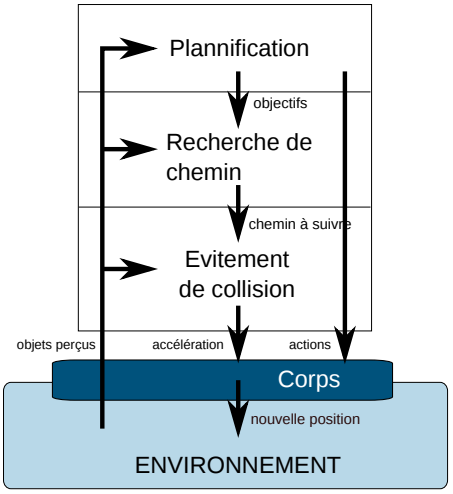

FIGURE 1.2 – Architecture en couches d'un agent

Il existe des interactions entre les modèles de chaque niveau de décision. La figure 1.2 donne un exemple d'interactions entre les couches constituant un modèle de simulation de piéton. Cette architecture en couches est entre autres utilisée pour la simulation de piétons et de cyclistes dans des villes virtuelles (Buisson et al., 2013; Paris et al., 2007a; Thalmann et al., 2013; Van den Berg et al., 2008), dans des gares et des aéroports (Daamen, 2004; Demange, 2012), pour la simulation de l'évacuation d'un bâtiment en cas d'incendie (Geramifard et al., 2005).

Nous ne fournissons pas plus de détails dans cette section sur les modèles de comportements d'agents. En effet, le chapitre 5 présente et illustre différents modèles d'agents pour simuler des piétons, des cyclistes et des conducteurs de véhicules en accord avec l'architecture présentée dans cette section.

1.3.3/ ENVIRONNEMENT

Les travaux provenant du groupe de travail sur l'environnement pour les systèmes multi-agents (issu de la conférence E4MAS) ont largement contribué à la prise de conscience de l'importance de tout ce qui est « extérieur » aux agents. La définition du concept d'environnement dans un système multi-agent s'est alors précisée.

1.3.3.1/ DÉFINITION DU CONCEPT D'ENVIRONNEMENT

L'environnement est communément défini par tout ce qui entoure un agent. Weyns et al. (2005a) donnent une définition globale de l'environnement (*cf.* définition 3), sans pour autant faire référence à un type d'agent particulier. L'une des principales idées est de déléguer une partie des responsabilités du SMA à l'environnement, qui intègre des mécanismes fournissant des services tels que l'observabilité et l'accessibilité à des ressources partagées (voir définition 2, page 14).

Définition 3 : Environnement (Weyns et al., 2005a)

L'environnement est une abstraction de premier ordre qui fournit les conditions environnantes aux agents pour exister et qui sert d'intermédiaire à la fois pour les interactions entre agents et l'accès aux ressources.

1.3.3.2/ Missions de l'environnement

Les missions de l'environnement sont définies par Weyns et al. (2005a) comme suit :

- **Partager les informations :** L'environnement est une structure commune pour les agents, où chacun d'eux perçoit et agit. Le principe de stigmergie[4] est au cœur de cette mission. Il exprime la notion que les actions d'un agent laissent des signes dans l'environnement, signes perçus par lui-même et les autres agents et qui déterminent leurs prochaines actions Parunak (2003).
- **Gérer les actions des agents et les interactions :** Cet aspect est lié à la gestion des actions simultanées et conjointes des agents, et à la préservation de l'intégrité de l'environnement. Par exemple, lorsque deux agents poussent la même boîte à l'opposé l'un de l'autre, l'environnement peut calculer la localisation réelle de la boîte selon les lois de la Physique, et les forces appliquées.
- **Générer des perceptions et des observations :** L'environnement doit être localement et partiellement observable. Ainsi, les agents peuvent également gérer l'accès aux informations environnementales et garantir le caractère partiel et local de la perception.
- **Maintenir les dynamiques endogènes :** L'environnement est une entité active. Il peut avoir ses propres processus indépendamment des activités réalisés par les agents. Un exemple typique est l'évaporation des phéromones artificielles dans les algorithmes à base de colonies de fourmis.

Ces missions sont exécutées dans un ordre prédéterminé. La figure 1.3 illustre un cycle de simulation issu de la synthèse des propositions de Ferber et al. (1996) et Weyns et al. (2004a). Ce cycle tente de répondre aux deux problématiques suivantes (Weyns et al., 2004a) :

- **Synchronisation des agents et de la dynamique endogène :** Le problème est, que dans de nombreux cas, les activités en cours dans l'environnement sont associées au temps physique. L'évaporation de phéromones, par exemple, est principalement mise en œuvre en fonction du temps physique (Sauter et al., 2002). Toutefois, la charge sur un système informatique fluctue continuellement. En conséquence, le déclin de l'intensité des phéromones n'évolue pas au prorata du temps de calcul disponible pour les agents.
- **Évolution du temps :** L'évolution temporelle d'un SMA peut être considérée comme une machine à états abstraite qui exécute, à chaque cycle, les activités des agents et l'activité dans l'environnement au même instant. De cette façon, le modèle assure la simultanéité conceptuelle entre les agents et l'environnement. Toutefois, l'exécution sur un calculateur étant séquentielle, la mise en œuvre pratique pose également le problème de la simultanéité des actions dans l'environnement.

4. Le terme fut introduit par le biologiste français Pierre-Paul Grassé en 1959, en référence au comportement des termites. Il le définit comme : « Stimulation des travailleurs par l'œuvre qu'ils réalisent. »

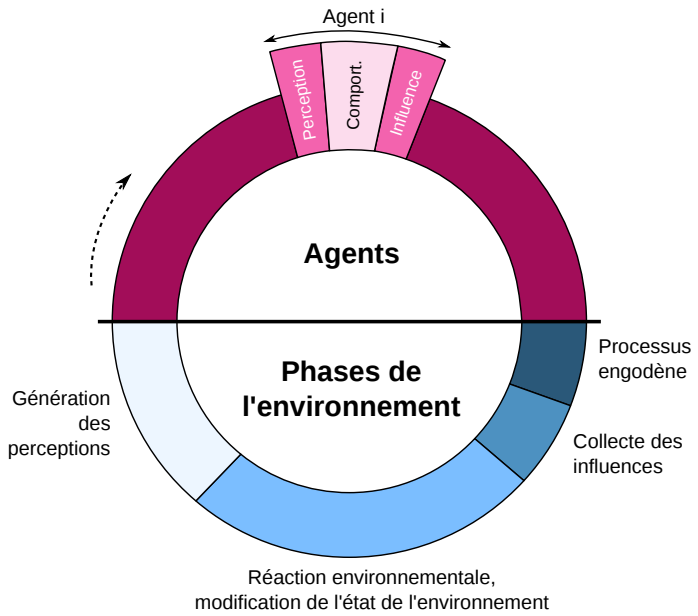

FIGURE 1.3 – Cycle d'exécution d'un système multi-agent durant une simulation (Galland et al., 2009)

1.3.3.3/ TYPES D'ENVIRONNEMENTS

Dans une synthèse des différents travaux relatifs aux modèles d'environnement dans les systèmes multi-agents, Odell et al. (2002) définissent trois types d'environnements :

- L'environnement de communication (*cf.* section 1.3.3.4),
- L'environnement social (*cf.* section 1.3.3.5) et
- L'environnement physique (*cf.* section 1.3.3.6).

Des travaux récents tentent de caractériser tous les types d'environnements pour définir des patrons de conception de modèles d'environnement (Mathieu et al., 2014). La définition des fonctions $voisin_t(a, r)$ et $position_t(a, p)$ est au cœur de l'approche adoptée par Mathieu et al. (2014). La première fonction permet de déterminer les agents qui se trouvent dans le voisinage (distance $\leq r$) de a à l'instant t. La seconde fonction permet de modifier la position de l'agent a. Quatre patrons d'environnement ont été proposés :

- L'`AgentSet` est une collection (non structurée) d'agents possédant une position dans l'environnement. Ce type d'environnement est plutôt destiné à un environnement contenant un petit nombre d'agents. En effet, l'environnement n'étant pas structuré, la fonction $voisin_t(a, r)$ possède une complexité importante de l'ordre de $O(n)$.
- Le `StandardGrid` correspond à un environnement possédant un espace discret de coordonnées (grille, graphe, etc.). Ce type d'environnement permet de diminuer la complexité de calcul du voisinage d'un agent en structurant l'espace de l'environnement. Cette structuration est généralement basée sur une grille ou un graphe.
- L'`AggregateGrid` correspond à un environnement possédant un espace continu de coordonnées (plan, arbre spatial, etc.) qui est ensuite discrétisé par une structure de données choisie. Cette discrétisation peut être réalisée soit par une grille, soit

par un arbre de décomposition spatiale (Galland et al., 2009).
— Le `SocialNet` est le patron permettant de définir l'environnement social. Mathieu et al. (2014) considère que ce patron permet de modéliser la plupart des types d'environnement social, et des interactions entre les agents dans ces types d'environnements. Le voisinage est défini par des relations d'accointances entre les agents. La position d'un agent est une position sociale (rôle, etc.).

Les trois premiers patrons sont relatifs à l'environnement physique, et le dernier à l'environnement social.

Galland et al. (2014c) propose de considérer les interactions directes entre les différents types d'environnements. Le modèle proposé permet de définir les influences de la dimension sociale sur la dimension physique, et vice-versa. Il est alors possible de décrire le comportement endogène de l'environnement en considérant ses différentes dimensions et leurs interactions au sein d'un même modèle.

Les différents types d'environnements ont fait l'objet de formalisations décrivant leurs composants et leurs propriétés (Béhé et al., 2014; Gouaïch et al., 2005; Weyns et al., 2004a).

Dans les sous-sections suivantes, nous détaillons les trois types d'environnements.

1.3.3.4/ Environnement de communication

Odell et al. (2002) proposent la définition 4 pour la notion d'environnement de communication.

Définition 4 : Environnement de communication (Odell et al., 2002)

L'environnement de communication fournit les principes, les processus et les structures qui permettent à une infrastructure ou une plateforme de transporter de l'information entre les agents.

L'environnement de communication sert uniquement de support à la définition des interactions entre les agents. Les interactions constituent une part essentielle à la mise en œuvre des SMA. En effet, elles permettent aux agents de collaborer et de coopérer pour produire des comportements collectifs complexes.

1.3.3.5/ Environnement social

Odell et al. (2002) proposent la définition 5 pour la notion d'environnement social dans un SMA.

Définition 5 : Environnement social (Odell et al., 2002)

Un environnement social est un environnement de communication dans lequel les agents interagissent de manière coordonnée.

L'environnement social est donc une extension de l'environnement de communication. Ils fournissent des protocoles de communications évolués permettant aux agents d'interagir de manière coordonnée. Ainsi, la communication est un cas particulier d'interaction. Des travaux proposent des protocoles de communications entre agents (Barbuceanu et al., 1995; Boissier et al., 1994; Finin et al., 1994; FIPA, 1998).

1.3.3.6/ ENVIRONNEMENT PHYSIQUE

L'environnement physique est un environnement spatial dans lesquels des agents sont immergés. Ces agents peuvent se déplacer, percevoir et agir dans ce type d'environnement. Odell et al. (2002) proposent la définition 6 pour la notion d'environnement physique.

Définition 6 : Environnement physique (Odell et al., 2002)

L'environnement physique fournit les principes et les processus qui régissent et supportent une population d'entités.

L'environnement physique est considéré comme inaccessible, non déterministe, dynamique et continu (Russell et al., 1995).

Les agents (ou entités autonomes) possèdent un corps correspondant à une représentation physique de l'agent à laquelle est associé son comportement (Michel, 2004). On dira alors qu'ils sont situés dans l'environnement. Le concept de corps a été raffiné et adapté par Galland et al. (2009) à la simulation dans des environnements virtuels 3D. Le corps sert d'interface à l'agent, lui permettant de percevoir et d'agir dans l'environnement (lire la section 1.3.4 pour plus de détails).

La perception est la capacité d'un agent à collecter des informations sur le système dans lequel il évolue. Généralement, la perception est vue comme un simple capteur récupérant des informations spécifiques. Pour notre part et en référence au domaine de la robotique (Begum et al., 2011; Elfes, 1989), nous définissons la perception d'un agent comme un processus complexe fondé sur trois étapes : (i) l'acquisition d'informations, (ii) leur extrapolation et (iii) leur filtrage. Cette vision est également partagée par Béhé et al. (2014) qui définit un méta-modèle ontologique pour la simulation orientée-agent en environnement virtuel.

L'acquisition a pour but d'acquérir les informations du système dans lequel évolue un agent. Vient ensuite l'extrapolation de l'ensemble des informations pour fournir plus de robustesse dans la collecte des informations. Finalement, des filtrages permettent à l'agent d'analyser les informations dont il a conscience et besoin. Autrement dit, par analogie à la perception humaine, un agent peut voir un objet sans le percevoir (en avoir conscience).

D'après Jarras et al. (2002), les agents peuvent interagir en communiquant directement entre eux ou indirectement à travers l'environnement physique (la stigmergie). La stigmergie est une méthode de communication indirecte où les individus communiquent entre eux en modifiant leur environnement et en percevant ces modifications. Nous pouvons citer comme exemple de ce type d'interaction celui des fourmis déposant des phéromones dans l'environnement. Une fourmi vagabonde dans l'environnement à la recherche de nourriture, puis dépose des phéromones tout au long du chemin lorsqu'elle en trouve. Toutes les fourmis sont attirées par ces phéromones en fonction de la quantité déposée. Ainsi, les fourmis interagissent par le biais de dépôts de phéromones dans l'environnement.

1.3.4/ INTERACTION ENTRE UN AGENT ET L'ENVIRONNEMENT PHYSIQUE

Dans cette section, nous nous intéressons particulièrement aux mécanismes qui permettent à un agent d'interagir avec l'environnement physique. Deux principes d'interac-

tion sont envisagés dans les définitions présentées dans les sections précédentes : la perception de son environnement par un agent et l'action d'un agent dans l'environnement.

Dans la suite de cette section, nous présentons les principes de la perception de l'environnement. Nous poursuivons par une présentation des modèles d'artéfacts et de ``*smart objects*´´ en nous focalisant sur leur capacité à fournir un mécanisme d'action. Nous poursuivons cette section en présentant le concept de corps d'un agent. Il permet notamment de distinguer les propriétés d'un agent relatives à l'environnement de celles qui ne le sont pas. Enfin, nous concluons cette section en présentant le modèle Influence-Réaction. Nous pensons qu'il est incontournable pour garantir un état cohérent de l'environnement.

1.3.4.1/ PERCEPTION DE L'ENVIRONNEMENT

Le mécanisme de perception est utilisé par les agents pour obtenir des informations à partir de l'environnement dans lequel ils progressent. Il s'agit d'extraire les caractéristiques environnementales des éléments situés dans le champ de perception d'un agent. Le calcul des perceptions d'un agent i est exprimé par l'équation 1.3 (Béhé et al., 2014; Weyns et al., 2004a).

$$\begin{array}{rccl} Perception_i : & \Sigma & \longrightarrow & \mathcal{P}\Sigma \\ & \sigma_t^r & \longmapsto & Assimilation_i \circ Filter_i \circ Alter_i \circ Extract_i(\sigma_t^r) \end{array} \tag{1.3}$$

Où :

- $Extract_i$: permet de parcourir les structures de données du `World Model` afin d'en extraire les objets de l'environnement ayant une intersection avec le champ de perception. Dans la formulation de l'équation 1.3, le champ de perception n'est pas considéré comme un paramètre. En effet, nous supposons que la fonction $Extract_i$ contient la description mathématisée ou algorithmique de ce champ. La fonction $Extract_i$ est en général fournie par le corps de l'agent : le champ de perception dépend uniquement des capacités sensorielles du corps.
- $Alter_i$: permet d'altérer les perceptions extraites par $Extract_i$. L'altération ne prend en compte que les propriétés physiques de l'agent, c'est-à-dire les propriétés de son corps. Ainsi, il est possible de modéliser des agents ayant des défaillances sensorielles (astigmatisme, surdité partielle, ...). Tout comme pour la fonction précédente, $Alter_i$ est définie dans le corps de l'agent, et ce dernier ne devrait pas altérer sa définition.
- $Filter_i$: permet de filtrer les perceptions fournies par son corps. Contrairement à $Alter_i$, cette fonction ne dépend pas des propriétés environnementales de l'agent.
- $Assimilation_i$: permet à l'agent d'assimiler les informations perçues. Cette fonction est fortement dépendante de l'architecture de l'agent. Par exemple, elle peut modifier la représentation mentale des objets perçus dans une base de règles représentant les connaissances de l'agent.

Les fonctions $Extract_i$ et $Alter_i$ sont des composantes de la représentation physique de l'agent : le corps. Les fonctions $Filter_i$ et $Assimilation_i$ sont quant à elles parties prenantes de l'architecture interne de l'agent.

La définition concrète de ces fonctions est dépendante du cadre d'application. Par exemple, Reynolds (1987), Van den Berg et al. (2008) et Buisson et al. (2012) définissent $Extract_i$ à l'aide d'un champ de perception circulaire autour de l'agent. Galland et al. (2009) propose d'utiliser une pyramide tronquée pour cette même fonction.

1.3.4.2/ ARTEFACTS DE COORDINATION

Dans (Ricci et al., 2003, 2005) le concept d'artéfact de coordination a été proposé. Il est basé sur l'idée d'adapter le concept d'artéfact de médiation de la théorie de l'activité au domaine des systèmes multi-agents. Les artéfacts de coordination sont des entités fournies par l'infrastructure SMA et jouant un rôle similaire aux objets médiateurs de la théorie de l'activité : l'objectif est de servir de médiateur dans l'interaction entre des agents en réalisant une tâche de coordination spécialement conçue dans cet objectif. Tucson (Omicini et al., 1999; Ricci et al., 2003) et CArtAgO (Ricci et al., 2007) sont des exemples de cadriciels SMA proposant la notion d'artéfact de coordination.

D'après (Ricci et al., 2005), les artéfacts de coordination ne peuvent être modélisés à l'aide du concept d'agent. En effet, leurs comportements ne sont pas caractérisés par « la réalisation d'un objectif de manière autonome ». De plus, leurs caractéristiques ne comprennent pas la pro-activité ou la rationalité. Au contraire, un artéfact de coordination réalise les lois qui caractérisent une tâche donnée de coordination sans avoir une liberté d'autonomie. Ces lois peuvent être inspectées par l'agent qui peut les adapter en fonction de ses besoins.

Outre la notion d'artéfact de médiation issu de la théorie de l'activité, les artéfacts de coordination sont inspirés de la notion de milieu de coordination (Ciancarini, 1996) développée dans le domaine des modèles et des langages de coordination (Omicini et al., 2001). Les médiateurs de coordination sont des abstractions conçues pour régir et gouverner l'interaction entre les processus d'un système distribué. Ils traitent principalement des problématiques de synchronisation et de communication entre ces processus. La méthode de conception des artéfacts de coordination est construite en partie sur les méthodologies d'ingénierie dédiées aux médiateurs de coordination et basées sur la notion d'interaction, sur des outils formels tels que les algèbres de processus, et sur les techniques de développement comme les infrastructures de coordination (Viroli et al., 2003). Les cadriciels comme Tucson (Omicini et al., 1999; Ricci et al., 2003) et MARS (Cabri et al., 2000) proposent des outils supportant les concepts et le processus de modélisation des artéfacts de coordination.

Chaque artéfact de coordination définit une interface d'utilisation décrivant les interactions permises avec les agents. Elle est exprimée par un ensemble d'opérations, qui peuvent être exécutées. Lorsqu'elles s'achèvent, ces opérations fournissent des informations sur leurs résultats. Ainsi, le service de coordination définit dans un artéfact est exploité par l'agent en suivant deux étapes : (i) une action est exécutée par l'agent sur l'artéfact, en précisant quelle opération est impliquée ; (ii) lorsque l'opération se termine, l'agent devient capable de percevoir l'achèvement de l'action. En conséquence, l'interaction allant de l'agent à l'artéfact de coordination est supportée par les actions sur cet artéfact. L'interaction opposée, de l'artéfact à l'agent, est définie par les notifications d'achèvement des opérations.

Le concept d'artéfact de coordination est une abstraction élégante qui peut être utilisée à la fois dans un environnement physique et social. Nous considérons que les artéfacts de coordination sont bien adaptés pour simuler les interactions entre les agents et les objets de l'environnement. Toutefois, si nous considérons que chaque objet dans l'environnement physique est représenté par un artéfact, il reste difficile de décrire leurs propriétés dynamiques et les interactions entre eux. Le concept d'objet intelligent, décrit dans la section suivante, permet de compléter celui de l'artéfact et de répondre partiellement à

ces deux problématiques.

1.3.4.3/ Objets Intelligents

Dans les domaines de la psychologie et de l'interaction homme-machine (IHM), le concept d'affordance (Gibson, 1977) suggère que la conception de dispositifs du monde réel doit informer les utilisateurs sur la façon de faire fonctionner un appareil. Un exemple de-facto est la forme d'une poignée de porte qui correspond à la forme d'une main humaine. Cette poignée suggère à un être humain que la poignée peut être saisie et tournée. L'avantage le plus convaincant de cette approche est qu'elle diminue la complexité des tâches d'interactions entre un être humain (ou un agent représentant cet être) et les objets constituant un univers virtuels (Peters et al., 2003).

La mise en œuvre la plus connue de cette approche concerne les objets intelligents proposés par Kallman et al. (1998) et étendue par Manovich (2005). Dans cette approche, les objets contiennent des informations et des conseils sur la façon dont ils devraient être en interaction avec les agents. Typiquement, cette information décrit les facteurs tels que les conditions préalables qui doivent être remplies avant qu'un objet puisse être utilisé, comment les parties d'un objet doivent être saisies et comment la représentation graphique de l'agent doit être animée lors de l'utilisation de l'objet.

À l'instar des artéfacts proposés par Ricci et al. (2003), les objets intelligents fournissent un ensemble d'actions, et des comportements associés, pouvant être exécutés sur les objets. Toutefois, Kallman et al. (1998) ont proposé une définition plus complète en ajoutant les informations suivantes dans un objet :

- **Informations d'interaction :** la position de poignées, boutons, leviers, etc. ;
- **Comportement de l'agent :** la description du comportement qu'un agent doit suivre lors de l'utilisation de l'objet.

Certaines versions de modèles d'objets intelligents comprennent également des informations d'animation graphique. Mais cette approche n'est pas considérée comme efficace (Jorissen et al., 2005). La taille des objets dans la mémoire du calculateur devient inappropriée et surdimensionnée.

Peters et al. (2003) présente deux contributions au modèle d'objet intelligent. La première permet aux objets intelligents de contrôler l'attitude de l'agent, et notamment son regard. Cette approche permet de reproduire des animations squelettales plus réalistes pour les observateurs de l'objet intelligent. La seconde contribution place les objets intelligents au centre de la stigmergie entre les agents. L'objet devient alors médiateur de l'interaction entre deux agents et fournit une description des comportements que doivent suivre les agents pour interagir par son intermédiaire.

1.3.4.4/ Modèle Influence - Réaction

L'action est généralement modélisée comme la transformation d'un état global (Ferber, 1995). Par exemple, dans le formalisme de Genesereth et al. (1987), où $\sigma \in \Sigma$ représente l'état du monde, le cycle comportemental (perception/délibération/action) d'un agent a est représenté par une fonction $Behaviour_a : \Sigma \rightarrow \Sigma$ qui correspond à l'application successive des trois fonctions suivantes :

— $Percept_a : \Sigma \rightarrow P_a$, qui calcule un percept à partir de l'état du système ;
— $Mem_a : P_a \times S_a \rightarrow S_a$, qui calcule le nouvel état interne de l'agent S_a ;
— $Decision_a : P_a \times S_a \rightarrow \Sigma$, qui modifie le monde suivant l'action de a.

Pour les agents tropiques (sans mémoire), les deux dernières fonctions se résument à une application : $Reflexe_a : P_a \rightarrow \Sigma$. Ainsi, la modification directe des variables de l'environnement est le moyen de signifier le résultat de la délibération d'un agent, par exemple $\sigma = door(closed) \rightarrow \sigma = door(open)$.

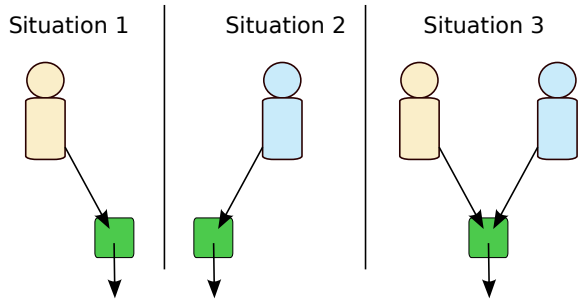

FIGURE 1.4 – Problème de simultanéité dans l'application des robots footballeurs (Michel, 2006)

La modélisation précédente ne permet pas de représenter simplement la simultanéité des actions des agents. Prenons l'exemple des robots footballeurs. Idéalement, il doit être possible de simuler que deux robots frappent une balle simultanément (situation 3, figure 1.4). Malgré son apparente simplicité, cette situation est difficile à modéliser car les actions sont considérées individuellement. Soit l'action de l'agent 1 est validée, soit l'action de l'agent 2 (situations 1 et 2, figure 1.4). Comme le souligne Ferber (1995), cette représentation de l'action, ainsi que ses dérivés, ne permet de traiter la simultanéité qu'au prix de programmes complexes qui s'apparentent plus à des artifices de programmation ponctuels qu'à une véritable modélisation de la simultanéité.

Pour prendre en considération ces différentes contraintes et concevoir des simulations multi-agents, le modèle Influence-Réaction a été proposé par Ferber et al. (1996) et étendu à la SOA par Michel (2001, 2006). Le cœur du modèle Influence-Réaction consiste à séparer l'action d'un agent de l'effet qu'elle produit. Dans ce modèle, un agent produit des influences sur son environnement et non des actions au sens vu précédemment. L'influence représente le désir d'un agent de voir modifier l'environnement d'une certaine façon. Le résultat effectif de cette tentative de modification de l'environnement par un agent ne peut être calculé sans connaître l'ensemble des influences produites au même instant. Ce modèle se base sur la distinction claire entre deux dynamiques qui sont combinées dans un système multi-agent (Galland et al., 2009; Michel, 2006) :

1. La dynamique au niveau de l'agent qui produit des influences.
2. La dynamique au niveau du système qui calcule la réaction de l'environnement compte tenu de l'ensemble des influences émises simultanément. Pour calculer cette réaction, les influences sont considérées en fonction des lois de l'univers (Ferber et al., 1996).

Adopter le modèle Influence-Réaction implique également de distinguer deux composantes de l'agent : son corps et son esprit (Galland et al., 2009; Michel, 2004) (voir sec-

tion 1.3.4.5 concernant la distinction entre le corps et l'esprit d'un agent). Cette distinction permet de clairement distinguer les propriétés relatives à un environnement et pouvant être influencées (au sens du modèle Influence-Réaction) de celles pouvant être directement manipulées par les agents. La figure 1.5 illustre ces deux types de propriétés.

1.3.4.5/ CORPS D'UN AGENT

L'interface entre les agents et leur environnement est une problématique encore d'actualité (Saunier, 2014). Comme cela a été démontré dans plusieurs modèles de simulation, l'ajout d'un élément à cette interface permet d'encapsuler des fonctionnalités telles que les mécanismes d'action (Michel, 2007; Simonin et al., 2000) et d'observabilité et de gestion de la perception (Gouaïch et al., 2005).

Un concept adapté pour gérer cette interface est le corps de l'agent. Il s'agit d'un élément qui est associé à chaque agent pour gérer son interface avec l'environnement (Galland et al., 2009; Michel, 2006; Saunier, 2014). La distinction entre l'esprit et le corps dans le cadre de l'intelligence artificielle a été principalement proposée en robotique (Brooks, 1991). Son application aux systèmes multi-agents, où il n'y a pas nécessairement de corps physique, n'a pas été examinée en détail dans la littérature Saunier (2014). Toutefois, ce concept a été utilisé ponctuellement pour modéliser et contraindre les interactions entre les agents et l'environnement Barella et al. (2012); Béhé et al. (2014); Demange (2012); Galland et al. (2014b); Michel (2007); Rincon et al. (2014); Saunier et al. (2014); Thiebaux et al. (2008). Dans ces travaux, le corps est un objet de l'environnement, ayant une dynamique qui ne peut être contrôlées directement par l'agent. L'environnement contrôle les propriétés dynamiques du corps (position, orientation, etc.) en garantissant qu'elles respectent les règles et les lois de l'univers Okuyama et al. (2005); Platon et al. (2006). Toutefois, l'agent peut influencer son corps en utilisant un mécanisme tel que le modèle influence/réaction présenté dans la section précédente.

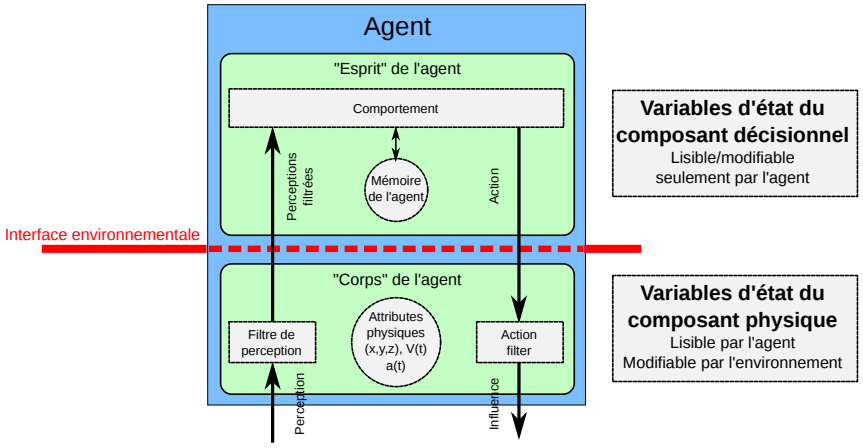

FIGURE 1.5 – Distinction entre l'esprit et le corps d'un agent dans le contexte du principe Influence/Réaction (Galland et al., 2014b)

Dans le cadre de cette thèse, nous considérons que le corps est un élément incontournable. Nous pensons que cette distinction est nécessaire afin de clairement séparer le modèle d'environnement des modèles des agents. En effet, l'esprit de l'agent est

dépendant du système cible (c'est-à-dire de l'application), alors que son corps est intégré au modèle d'environnement (celui-ci pouvant être défini pour une classe d'application, c'est-à-dire partagé par plusieurs applications). Cette distinction esprit/corps correspond à la partition entre les variables utilisées par le système décisionnel de l'agent sur lesquelles l'agent dispose d'un contrôle total, et les variables liées à la modélisation de sa partie physique contrôlées par l'environnement et sur lesquelles l'agent n'a aucun pouvoir. La figure 1.5 explicite cette distinction entre l'esprit et le corps d'un agent.

1.4/ CONCLUSION

La modélisation de la dynamique des piétons, des cyclistes et des conducteurs de véhicules est d'un grand intérêt théorique et pratique. Les systèmes multi-agents sont considérés comme des sociétés composées d'entités autonomes et indépendantes, appelées agents, qui interagissent en vue de résoudre un problème ou de réaliser collectivement une tâche. Grâce à la généricité de ces concepts, les domaines d'application des SMA sont vastes (Jennings et al., 1998). Les SMA offrent aussi un cadre pour la modélisation et la simulation de systèmes complexes (Bergenti et al., 2004; Ferber, 1995; Henderson-Sellers et al., 2005; Jennings, 2001). Notamment, les travaux dans le domaine de la simulation de foules et de trafic illustrent l'usage fréquent des SMA (Bouvier et al., 1997; Brogan et al., 1997; Galland et al., 2009; Hodgins et al., 1994; Musse et al., 2001; Niederberger et al., 2003; Razavi et al., 2011a; Reynolds, 1999, 1987; Tu et al., 1994; Ulicny et al., 2002).

Dans ce chapitre, nous avons présenté les principes des SMA et de la simulation orientée-agent (SOA). Cette dernière se réfère aux modèles individu-centrés et fournit un processus permettant de modéliser et simuler la dynamique de populations composées d'individus en interaction. Ce type de simulation assimile généralement un individu à un agent. Comme nous l'avons expliqué dans ce chapitre, quatre aspects composent un modèle de SOA. Le premier aspect permet de modéliser les processus de délibération des agents. Le second aspect concerne la modélisation de l'écoulement du temps dans la SOA. Le troisième aspect s'intéresse aux interactions entre les agents et entre les agents et l'environnement. Enfin, le dernier aspect permet de définir les différents objets physiques du monde simulé et la dynamique endogène de l'environnement.

Toute comme Michel (2004), nous considérons que l'environnement est un élément de première importance dans un modèle de SOA. Cet avis est également partagé par Thalmann et al. (2009) dans le contexte de simulation de foules en environnement virtuel.

Toutefois, les travaux explicitement dédiés à la modélisation de l'environnement restent récents et peu nombreux en proportion des travaux dédiés aux autres aspects de la SOA.

Dans le chapitre suivant, nous décrivons des travaux issus de l'état de l'art concernant la modélisation de l'environnement. Nous focalisons plus particulièrement nos explications dans le cadre de la classe d'applications visée par ce mémoire de thèse. Les modèles d'environnement existants ne permettent pas de répondre à l'ensemble des problématiques énoncées dans l'introduction générale. Cette constatation nous a poussés à proposer un nouveau modèle d'environnement qui est détaillé dans la partie 3 de ce mémoire.

2

Modèles d'environnement physique

Dans le chapitre précédent, nous avons montré l'importance des différents types d'environnements, en particulier l'environnement physique, dans le cadre de la simulation orientée-agent. Nous présentons dans ce chapitre certains modèles d'environnement physique existants pour la simulation microscopique en milieu urbain. L'objectif est de réaliser une synthèse des caractéristiques communes et de mettre en valeur leurs avantages et inconvénients. En fonction du type d'entité qu'ils permettent de simuler, nous les séparons en deux catégories : d'un côté les modèles de simulation de trafic pour les véhicules routiers et de l'autre les modèles de simulation de foules pour les piétons.

Nous comparons alors les modèles d'environnement physique selon différents critères relatifs à nos objectifs globaux, présentés dans l'introduction générale : (i) la typologie des zones de l'espace, (ii) les types des agents supportés par le modèle, (iii) la dimension de ses structures internes, (iv) la possibilité de distinguer les mécanismes de perception et de navigation, (v) les méthodes de modélisation, (vi) les performances du simulateur supportant le modèle. Ces critères permettent aussi de comparer les modèles en considérant les caractéristiques des applications que nous ciblons : la possibilité de simuler piétons, véhicules, cyclistes, et d'autres types d'entités au sein d'un même contexte de simulation, de simuler les flux aussi bien à l'extérieur qu'à l'intérieur d'un bâtiment, et enfin de permettre l'étude des interactions entre ces deux catégories d'espaces urbains. De plus, le cadre industriel de cette thèse nous impose de considérer la facilité et la rapidité de modélisation de l'environnement, ainsi que les performances du simulateur.

Sommaire

2.1 Modèles de simulation de trafic **30**
2.1.1 Outils commerciaux 31
2.1.2 Approches académiques 32
2.2 Modèles de simulation de foules **37**
2.2.1 Modèles basés sur des grilles régulières 37
2.2.2 Modèles basés sur des graphes 38
2.2.3 Les modèles basés sur des arbres de décomposition spatiale . . 40
2.3 Critères de comparaison **41**
2.4 Comparaison des modèles d'environnement **43**
2.5 Conclusion **43**

Dans la suite de ce chapitre, nous commençons par présenter quelques outils commerciaux ou académiques dédiés à la simulation de trafic. Dans la section 2.2, les modèles d'environnement physique dédiés à la simulation de foules sont décrits. Enfin, avant de conclure ce chapitre, nous présentons nos critères de comparaison permettant de comparer ensuite l'ensemble des modèles d'environnement considérés.

2.1/ MODÈLES DE SIMULATION DE TRAFIC

Les modèles de simulation de trafic sont généralement basés sur une structure de graphe dont les sommets représentent les intersections alors que les arcs, modélisés géométriquement par des splines, correspondent aux tronçons de routes. Ils peuvent être divisés en deux catégories :

— **Les modèles 1D :** ils modélisent chaque voie de route distinctement. La position d'une entité est représentée par un facteur d'avancement le long de la spline modélisant la voie. Les manœuvres de changement de voie, lorsqu'elles sont supportées, s'opèrent de manière discrète : un véhicule passe du centre d'une voie à l'autre instantanément.

— **Les modèles 1.5D :** ils modélisent un ensemble de voies adjacentes par un même arc. La position d'une entité est représentée par un facteur d'avancement le long de la spline d'un arc ainsi qu'une distance latérale permettant de déterminer sur quelle voie l'entité se trouve. Les manœuvres de changement de voie s'opèrent de manière continue et progressive.

Au vu de l'avancée des travaux académiques, nous nous focalisons sur les approches 1.5D.

Dans tous les modèles que nous présentons dans cette section, un véhicule peut se déplacer uniquement le long des arcs du graphe. Par conséquent, une manœuvre physiquement impossible à effectuer peut être la conséquence d'une modélisation incorrecte de l'environnement.

Nous distinguons les approches académiques, généralement très documentées, des approches commerciales dont nous ne pouvons qu'apprécier les fonctionnalités offertes à l'utilisateur.

Le comportement des conducteurs dans la majorité les simulateurs étudiés reposent principalement sur trois modèles :

— **Le modèle longitudinal :** appelé modèle du véhicule-suiveur, il a pour but de décrire la dynamique d'un véhicule le long du tronçon de route. Le modèle longitudinal est une fonction de l'accélération dépendant de nombreux stimuli tels que la distance inter-véhiculaire, la vitesse relative des deux véhicules, etc. (Chandler et al., 1958; Treiber et al., 2000; Wiedemann, 1974). Il utilise des équations différentielles pour calculer l'accélération à appliquer à un instant t et modélise le comportement de chaque véhicule (microscopique) afin d'en déduire ses implications sur l'ensemble du système de trafic (macroscopique). L'utilisation conjointe d'un modèle de véhicule-suiveur, comme celui proposé par Gipps (1981), et d'une méthode numérique, comme celle de Euler-Heun (Süli et al., 2003), permet d'extraire des informations importantes sur les conditions de circulation, telles que les retards et l'identification des zones de congestion. Dans le chapitre 5, nous utilisons un modèle de véhicule-suiveur pour modéliser les conducteurs dans le cadre

d'un projet de simulation du trafic dans la ville de Belfort.

— **Le modèle latéral :** appelé modèle de changement de voie, il a pour but d'évaluer l'intérêt et la faisabilité d'une manœuvre de changement de voie. De nombreux auteurs ont proposé des modèles de ce type (Ahmed, 1999; Choudhury, 2005; Gipps, 1986) et (Ehmanns et al., 2000).

— **Le modèle de recherche d'itinéraire :** il a pour but de permettre à l'agent conducteur de choisir un itinéraire et éventuellement de le modifier dynamiquement pour optimiser son temps de parcours. On parle d'affectation dynamique lorsque le conducteur d'un véhicule peut changer d'itinéraire au cours de la simulation (Arikan et al., 2001; Dechter et al., 1985; Desaraju et al., 2012).

2.1.1/ OUTILS COMMERCIAUX

Dans cette section nous présentons certaines solutions commerciales de simulation. À la différence des solutions académiques, il est difficile d'obtenir des informations détaillées sur les modèles d'environnement et de comportement utilisés par ces outils. Il est cependant intéressant d'étudier les fonctionnalités de ces solutions logicielles pour avoir une vue globale sur leurs avantages et inconvénients. Nous ne nous intéressons qu'à un ensemble de logiciels parmi les plus présents sur le marché.

VIssIM ou ``*Verkehr In Städten – SImulationsModell*´´ est un logiciel commercial de simulation de flux multimodaux (Fellendorf et al., 2010). Le logiciel intègre un éditeur graphique permettant une création manuelle du modèle. Il est également possible de générer un réseau à partir de données SIG via un outil externe.

Vissim intègre un modèle de suivi de véhicules (Wiedemann, 1974) spécialement développé pour cette application ainsi qu'un modèle de changement de voie appelé *Cooperative merging*. Le logiciel intègre également une fonctionnalité d'affectation dynamique et un module d'intégration de piétons pour lequel nous ne disposons que de peu d'informations.

Comme nous le soulignons dans le chapitre 5, nous avons été amenés à utiliser ce logiciel dans le cadre de travaux relatifs à cette thèse. De nos expériences nous pouvons formuler les remarques suivantes :

— La grande flexibilité autorisée pour la connexion des connecteurs aux tronçons est parfois un élément pénalisant. En effet, Vissim autorise la connexion d'un connecteur au milieu d'un tronçon plutôt qu'aux extrémités seulement, rendant difficile la lecture du réseau.

— La création du modèle prend un temps relativement conséquent et nécessite une bonne connaissance du logiciel.

— Le logiciel est conçu pour modéliser une intersection ou plusieurs petites intersections sur une zone géographique restreinte et se prête difficilement à la simulation d'un quartier ou d'une ville entière. La configuration des priorités sur les zones de conflits devient rapidement complexe lorsque le nombre d'intersections augmente. En effet, les priorités doivent être ajustées manuellement sur chaque zone de conflit et chaque tronçon impliqué.

S-Paramics aussi appelé simplement Paramics, est un logiciel commercial de simulation de trafic au niveau microscopique (Sykes, 2010). Il est principalement développé par SIAS, une société de consulting sur la planification du transport basé à Edimbourg. Ce logiciel offre des fonctionnalités similaires à VIssIM. Il inclut des paramètres permettant de simuler des conducteurs avec une connaissance incomplète du réseau qui impact la recherche d'itinéraire.

Aimsun ou ``*Advanced Interactive Microscopic Simulator for Urban and Non-urban network*´´, est un cadriciel commercial de simulation de trafic développé initialement à l'Université de Catalogne (Casas et al., 2010). Il inclut des outils de simulation à trois niveaux d'échelle (macroscopique, mésoscopique et microscopique). Les modèles longitudinaux et latéraux intégrés au comportement des conducteurs de véhicule sont basés respectivement sur les propositions de Gipps (1981) et Gipps (1986). Aimsun fournit un module permettant d'intégrer le simulateur de piétons Legion (Still, 2000) conjointement avec la simulation de véhicules. Les informations sur le module de compatibilité ne sont cependant pas accessibles.

TransModeler est une solution commerciale de simulation de trafic incluant un modèle macroscopique et un modèle microscopique (Caliper, 2014). La simulation microscopique est basée sur un graphe des voies des routes généré à partir d'une base de données SIG.

La simulation utilise une approche à temps discret qui exécute le comportement des conducteurs de véhicules toutes les 100 millisecondes. TransModeller intègre une fonctionnalité d'affectation dynamique ainsi que des fonctionnalités moins courantes telle que la recherche de place de stationnement.

2.1.2/ Approches académiques

La plupart des approches académiques présentées dans cette section ont été développées par des laboratoires universitaires. Une particularité qui caractérise ces approches est leur caractère expérimental (Lützenberger, 2014).

2.1.2.1/ ``*Janus Simulation Environment*´´ ou JaSim

JaSim est un modèle d'environnement physique multi-agents dédié à la simulation dans des environnements virtuels 1D, 2D ou 3D (Galland et al., 2009). Il est principalement conçu pour la simulation de foules et de trafic. Le modèle JaSim est conçu comme une extension de la plate-forme Janus [1] combinant l'approche orientée agent et organisationnelle de Janus avec des structures de données dédiées à la modélisation d'environnement physique. Il est conçu pour fournir des mécanismes de perception et d'action réalistes aux agents simulés.

L'architecture générale de la plate-forme JaSim est illustrée par la figure 2.1. Les différentes fonctionnalités offertes par chaque module sont brièvement décrites ci-dessous :

1. http://www.janusproject.io

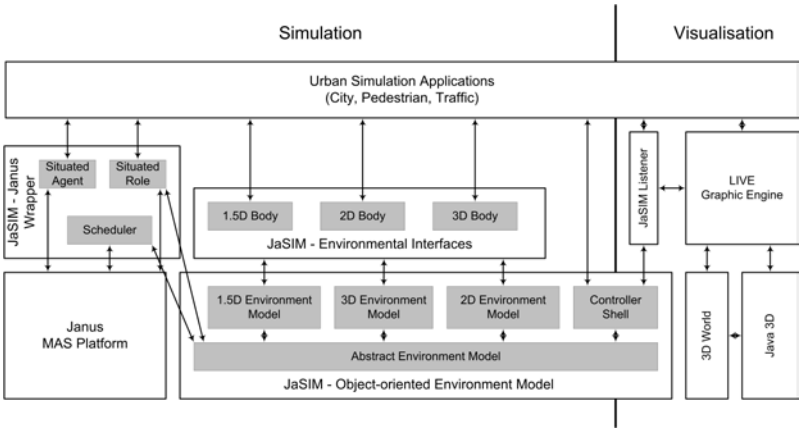

FIGURE 2.1 – Architecture de la plate-forme JaSim

— Le module Janus Wrapper est une extension de la plate-forme Janus. Il fournit un ensemble d'organisations et d'architectures d'agents dédiées à la simulation de foules et de trafic. Il étend également les concepts d'agent et de rôle de Janus afin de faciliter et d'accélérer le processus de développement des simulations.
— Le modèle d'environnement est la partie principale de la plate-forme JaSim. Il fournit tous les outils logiciels nécessaires pour modéliser et gérer la création et la modification des structures de l'environnement. Il contient, en particulier, toutes les structures hiérarchiques utilisées pour calculer les perceptions des agents.
— Les interfaces environnementales permettent de connecter une application externe et la plate-forme JaSim. Elles fournissent un ensemble d'architectures d'agents pour les différents types d'environnements virtuels supportés par la plate-forme. Chaque type de corps fournit aux agents (piétons, conducteurs, etc.) une collection d'outils pour percevoir son environnement et émettre une collection d'actions (appelées influences, voir section 1.3.4.4, page 24).
— Le module Listener est destiné à fournir une interface pour les communications événementielles avec des outils et des interfaces graphiques externes. Une particularité de ce module permet de communiquer avec des applications distantes dans un réseau informatique.

En accord avec la définition des propriétés d'un environnement définie par Russell et al. (1995), le modèle de l'environnement JaSim peut être considéré comme inaccessible, non-déterministe, dynamique et continu. Celui-ci possède ses propres dynamiques endogènes qui peuvent être spécialisées en fonction des besoins de l'application.

Une partie de ce modèle est fortement inspiré du modèle Influence-Réaction (Ferber et al., 1996; Michel, 2007). Un agent ne peut donc pas changer directement l'état de l'environnement physique afin d'éviter les problèmes de causalité dus à des modifications directes et incompatibles avec les règles de l'univers. Le modèle de JaSim considère qu'un agent possède un corps qui est sa représentation dans l'environnement physique. Ce corps fait partie de l'environnement et fournit un ensemble de capteurs (vision, touché, etc.) et d'effecteurs à son agent associé. Ainsi, l'agent ne peut percevoir et agir sur l'environnement physique que par l'intermédiaire de son corps.

Le modèle JaSim contient deux fonctionnalités principales : la première calcule les perceptions des agents à l'aide de structures hiérarchiques ou de graphes représentant l'uni-

vers, et la seconde collecte les influences (ou actions) des agents et calcule les réactions résultant de leur application sur le modèle de l'univers.

JaSim propose des implantations de structures de données dédiées aux univers modélisés en 1D et 1.5D (Galland et al., 2011, 2014d).

2.1.2.2/ *``Multi-Agent Transport Simulation´´* ou MATSim

MATSim est un environnement de simulation de trafic microscopique fondé sur un modèle à événements discrets (Balmer et al., 2004; Illenberger et al., 2007). Cet environnement a évolué à partir de son prédécesseur, le simulateur de trafic TRANSIMS. MATSim est basé sur le paradigme multi-agent et représente donc les véhicules simulés (ou leurs conducteurs) par des agents autonomes.

Il utilise une approche basée sur les activités pour générer des événements qui sont nécessaires pour le moteur de simulation. De façon plus détaillée, chaque agent simulé reçoit une série d'activités qu'il doit accomplir tout au long de la durée de la simulation.

Les deux processus fondamentaux de MATSim constituent la couche stratégique et la couche physique respectivement.

La couche stratégique comprend deux phases distinctes, à savoir la génération de l'activité et le choix du mode de transport et du chemin (Balmer et al., 2004). La génération de l'activité dissocie les données de recensement afin de produire un plan d'activité de 24 heures comprenant les temps et les lieux des activités. Le mécanisme de choix du mode ou de la route relie ces activités. La connexion se fait à l'aide des temps de trajet utilisés en tant que fonction de coût à minimiser et en utilisant l'algorithme de plus court chemin de Dijkstra (1959).

Dans sa couche physique, MATSim utilise des valeurs de flux de trafic pour estimer les coûts des déplacements et refléter ainsi la dynamique des systèmes de signalisation. Cependant, le choix du mode de transport est réalisé avant la phase de simulation. Durant ce processus de choix, une fonction d'utilité est utilisée pour évaluer la qualité des modes de transport disponibles (Balmer et al., 2004).

Le modèle sous-jacent de l'environnement urbain est constitué d'un graphe du réseau routier, où les arcs correspondent à des segments de route, et les nœuds aux carrefours. Les informations disponibles sur chaque arc permettent de construire la perception des agents. D'après Illenberger et al. (2007), cette perception est donnée par l'état actuel de la circulation (taux de congestion de l'arc, etc.). La prise de conscience de l'agent est donc limitée à la circulation environnante. Les agents fournissent un plan d'actions à l'environnement. Ce dernier a la charge de modifier son état en exécutant les actions composant les plans aux moments appropriés.

2.1.2.3/ *``Simulation of Urban Mobility´´* ou SUMO

SUMO est un cadriciel de simulation de trafic basé sur un modèle d'évolution en temps discret (Krajzewicz et al., 2012). Il utilise un modèle de flux de trafic microscopique. SUMO a été conçu comme une plate-forme d'évaluation universelle pour différents modèles et algorithmes liés au trafic ou à sa gestion.

Le modèle longitudinal utilisé par les conducteurs est une extension du modèle de

véhicule-suiveur proposé par Krauß (1998). Comme la plupart des approches microscopiques, le modèle est basé sur l'hypothèse que le conducteur a pour objectif de maintenir une distance à un véhicule de tête qui leur permet d'arrêter sans risquer une collision. Le modèle d'environnement associé est un graphe constitué des routes (Krauß, 1998).

Krajzewicz et al. (2012) ont affiné le travail de Krauß (1998). La possibilité pour le conducteur de maintenir parfaitement une vitesse souhaitée a été améliorée pour les vitesses basses. En outre, des facteurs supplémentaires ont été inclus dans le calcul de la vitesse d'un véhicule. Ces facteurs comprennent : la vitesse autorisée, les règles de droit de passage, et la portée visuelle du conducteur. Ces différents facteurs sont des données intégrées dans le modèle de l'environnement.

Le modèle latéral utilisé par les conducteurs est basé sur la proposition de Krajzewicz et al. (2012). Afin de déterminer si un conducteur doit changer de voie, la route en avant du véhicule est examinée (jusqu'à une distance de vision) et la charge du tronçon est recueillie. Sur la base de cette perception, le conducteur calcule la « vélocité de sécurité » pour toutes les voies voisines. Ce processus de calcul est basé sur les travaux de Ehmanns (2001) de telle sorte que les conducteurs évaluent en permanence l'avantage de changer de voie. Chaque fois que le bénéfice calculé dépasse un seuil spécifique à chaque conducteur, un changement de voie est initié.

2.1.2.4/ *``Advanced & Visual Evaluator for road Networks in Urban arEas´´* OU AVENUE

AVENUE est un simulateur développé dans l'objectif de fournir un outil pour l'analyse des stratégies de gestion du trafic local (Kuwahara et al., 2010).

Il utilise une approche hybride pour représenter les flux de trafic. Le modèle de flux est basé sur l'approche macroscopique de la mécanique des fluides. Néanmoins, AVENUE présente également des représentations visuelles pour certaines intersections. Afin de représenter visuellement les véhicules aux intersections, AVENUE met en œuvre un mécanisme permettant de raffiner les résultats produits par le modèle macroscopique à un niveau microscopique.

Deux modèles de choix d'itinéraire sont mis en œuvre. Le premier modèle est basé sur l'optimum utilisateur dynamique, et le second sur les principes d'optimum utilisateur stochastique (Kuwahara et al., 2010). Ces deux modèles de choix de l'itinéraire peuvent être utilisés pour évaluer les routes en considérant plusieurs facteurs : durée des trajets, longueur du trajet, le nombre de pas de simulation, coûts des péages, etc. Ces facteurs (ou l'évaluation de ces facteurs) représentent la personnalité des conducteurs simulés.

Le deuxième modèle intégré dans AVENUE concerne le changement de voie. Pour chaque pas de simulation, le système détermine si un conducteur reste sur la voie courante ou change de voie. La prise de décision est déterminée par deux facteurs : le débit observé des voies voisines, et la prédisposition ou la préférence du conducteur à changer de voie.

2.1.2.5/ *``Microscopic Traffic Simulation Laboratory´´* OU MITSIMLAB

MITSIMLab est un cadriciel de simulation de trafic en temps discret (Ben-Akiva et al., 2010). Il utilise un modèle microscopique pour représenter les flux de trafic. Le cadriciel a été élaboré dans l'objectif d'analyser les effets des systèmes de gestion du trafic (Ben-Akiva et al., 2010). Le modèle de conducteur tient compte des facteurs humains et a été calibré en utilisant des informations sur le comportement de conducteurs réels.

MITSIMLab modélise globalement un comportement de niveau tactique qui est modélisé comme une série de choix interdépendants et conformes à un plan élaboré à un niveau stratégique (Ben-Akiva et al., 2010). Un modèle de Markov est utilisé pour adapter le comportement aux actions réalisées par les autres conducteurs ou par les systèmes de contrôle de la circulation. Étant un outil de simulation microscopique, MITSIMLab met en œuvre un mécanisme permettant de stimuler les conducteurs en fonction d'événements se produisant dans leur voisinage. Ce modèle permet de reproduire avec réalisme les comportements de changement de voie ou de réalisation d'écarts latéraux.

MITSIMLab supporte également les comportements de niveau stratégique et met en œuvre un modèle de comportement de mobilité sophistiqué (Ben-Akiva et al., 2010). Le modèle spécifique est basé sur l'utilisation de chemins prédéfinis et adapte son choix de trajet dynamiquement. La sélection d'itinéraire est réalisée de deux manières différentes : le choix de l'itinéraire en fonction des routes connues, et la génération dynamique sur la base des liens de connectivité.

2.1.2.6/ *``Dynamic Route Assignment Combining User Learning and Microsimulation´´* OU DRACULA

DRACULA est un cadriciel de simulation microscopique de trafic en temps continu (Liu, 2010).

Il implante le modèle de véhicule-suiveur de Wang et al. (2005). Ce modèle est proche des autres modèles de même type et met essentiellement en œuvre les règles proposées par Gipps (1981). Toutefois, le modèle utilisé intègre des facteurs humains tels que les temps individuels de réaction ou de propension à faire des écarts latéraux. En outre, des éléments plus complexes permettant de modéliser le comportement des conducteurs sont intégrés : les caractéristiques de l'infrastructure routière (par exemple : la répartition du trafic, l'hystérésis, ou la propagation d'ondes de choc) et les comportements de suivi de proximité sont inclus (Liu, 2010).

DRACULA met en œuvre une approche fondée sur les activités pour générer la demande de trafic (Liu, 2010; Liu et al., 2006). Le principe de fonctionnement est basé sur l'hypothèse que les conducteurs cherchent à maximiser leur fonction d'utilité. L'optimisation est réalisée selon un processus itératif. Pour chaque conducteur, les activités pour une journée sont générées. Grâce à ces activités, les conducteurs calculent une stratégie d'amélioration de leur fonction d'utilité, incluant les choix des routes et des heures de départ. Les stratégies calculées sont simulées et leurs efficacités sont évaluées. Sur la base de cette évaluation, les conducteurs adaptent leurs connaissances afin qu'elles soient utilisées au pas de simulation suivant.

2.1.2.7/ ARCHISIM

ARCHISIM est un logiciel de simulation microscopique de trafic basé sur une architecture distribuée (Espié et al., 1995).

Il utilise un serveur central pour remplir les missions de l'environnement. Chaque agent modélisant un conducteur est exécuté par un processus distinct qui communique avec l'environnement au travers d'un réseau Ethernet. L'environnement est basé sur un graphe 1.5D dans lequel les routes constituent les arcs et les intersections correspondent aux nœuds. Un agent est donc repéré dans l'environnement par un identifiant de route, un point kilométrique, une distance par rapport à l'axe central de la route et une information de direction sur l'axe. Les véhicules peuvent ainsi utiliser la totalité de la surface de navigation que représente la route.

ARCHISIM utilise des agents cognitifs pour simuler les conducteurs de véhicules (Ksontini, 2013). Par conséquent, il fait partie des rares logiciels de simulation de trafic qui n'utilisent aucun des modèles longitudinaux présentés au début de cette section.

2.2/ MODÈLES DE SIMULATION DE FOULES

Il existe de nombreux modèles d'environnement physique dédiés à la simulation de foules. Nous pouvons cependant les classer en fonction de la structure de données principale utilisée pour modéliser la topologie de l'environnement. Les grilles régulières, les arbres de partitionnement et les graphes sont les structures de données généralement les plus utilisées. Nous les présentons dans cette section.

2.2.1/ MODÈLES BASÉS SUR DES GRILLES RÉGULIÈRES

Les grilles régulières sont des structures de données classiquement employées pour accélérer les calculs de détection de collision ou de rendu par lancer de rayons. Elles ont l'avantage de permettre d'accéder très rapidement et en temps constant au voisinage d'un point, ce qui constitue un avantage certain pour le calcul des perceptions des agents dans leur environnement.

Des auteurs tels que Treuille et al. (2006) ont proposé d'utiliser ce type de structures pour la simulation de flux de piétons en utilisant des méthodes issues de la mécanique des fluides. La grille est utilisée pour différencier les espaces navigables des espaces inaccessibles. Elle est également utilisée comme champ de potentiel permettant aux piétons de se déplacer en évitant les collisions avec les obstacles tout en se dirigeant vers leurs objectifs. Le champ de potentiel intègre les fonctions d'évitement de collision et de recherche globale de chemin.

Tecchia et al. (2001) propose une modélisation similaire utilisant deux grilles régulières. L'utilisation de deux grilles distinctes permet de séparer les obstacles statiques des obstacles dynamiques. Il réduit ainsi la complexité de la procédure de mise à jour de la structure de données qui ne se concentre que sur la grille contenant les entités mobiles.

Un modèle similaire est proposé par Shao et al. (2005). Celui-ci propose d'utiliser quatre structures de données permettant d'accélérer le calcul des perceptions. Deux sont consacrées à la perception des obstacles, les deux autres à la navigation. Elles

sont générées automatiquement à partir d'un plan de l'emprise au sol des obstacles statiques. Les deux structures de perception sont des grilles régulières. La première contient les obstacles statiques. Elle est générée avant le lancement de la simulation et est globale à l'environnement. La seconde contient les obstacles dynamiques et est mise à jour continuellement durant l'exécution. Elle est locale à une zone de navigation telle qu'une pièce à l'intérieur d'un bâtiment. Les deux structures de navigation sont utilisées pour accélérer la recherche de chemin par les individus. L'une est une grille régulière et utilisée seulement lorsqu'une grande précision est nécessaire, l'autre est un ``*Quadtree*´´ [2] (Samet, 1987).

Les grilles régulières ont cependant de nombreux inconvénients. En premier lieu, les grilles régulières ne permettent pas de représenter précisément la topologie d'un environnement : difficulté à modéliser des courbes, crénelage apparent sur toute surface non-aligné avec les axes de la grille, etc. Une solution à ce problème est d'augmenter la résolution de la grille et donc de réduire la taille des cellules. Ceci a pour conséquence d'augmenter le nombre de cellules à parcourir pour explorer un voisinage d'une taille donnée ce qui diminue les performances.

En second lieu, la consommation mémoire d'une grille régulière est conséquente puisque des cellules de la grille sont allouées également aux espaces vides ainsi qu'aux espaces inaccessibles. Pour cette raison, les grilles ne sont pas des structures de données optimales pour des environnements de grande superficie.

De manière générale, les grilles régulières ne sont pas adaptées à la modélisation des environnements urbains extérieurs qui comprennent de nombreuses zones non accessibles aux piétons, tels que les bâtiments, les routes, etc.

2.2.2/ Modèles basés sur des graphes

Dans la littérature, il est possible de trouver de nombreux modèles d'environnement physique dédiés à la simulation de foules basés sur des graphes modélisant la topologie et la connectivité entre les différents espaces de l'environnement. Il existe cependant différents types de graphes et leur utilisation varie d'un auteur à l'autre. Un graphe a l'avantage de faciliter la recherche de chemin global permettant aux entités de naviguer dans l'environnement en direction de leurs objectifs grâce à l'utilisation d'algorithmes tels que A* (Dechter et al., 1985) ou D* (Stentz, 1995). Parmi les modèles basés sur des graphes nous pouvons distinguer les approches en fonction de la dimension de l'environnement. Ainsi, dans cette section, nous commençons par étudier les approches 1.5D adaptées et héritées, pour la plupart, de la simulation de flux routiers. Puis, nous nous intéressons aux modèles 2D.

2.2.2.1/ Les modèles 1.5D

Certains modèles généralisent l'approche utilisée classiquement pour la simulation de flux routiers. Le graphe modélise un environnement 1.5D, c'est à dire un réseau de couloirs interconnectés sur lesquels les entités peuvent se déplacer latéralement. Cette approche diffère de l'approche 1D dans laquelle les entités se déplace le long de l'axe

2. Chaque nœud d'un Quadtree divise l'espace qu'il recouvre en quatre zones de même taille. Chacune de ces zones est alors associée à un nœud fils, etc. Le nœud racine recouvre la totalité de l'univers.

médian du couloir ; une solution adaptée aux véhicules mais inapplicable aux piétons. C'est le cas de Thomas et al. (2000a) et Willemsen et al. (2003) qui proposent de générer un réseau de splines représentant les trottoirs. Chaque spline possède une largeur permettant aux entités de s'écarter de l'axe médian pour éviter les obstacles. Dans ce type de modèle, la position d'une entité est relative à une spline et représentée par deux distances. L'une représente l'avancement de l'entité par rapport à l'un des points d'entrée du couloir, l'autre représente la distance latérale de l'entité par rapport à l'axe médian. Willemsen et al. (2003) remarquent cependant que l'utilisation d'une spline de degré supérieur à 1 complexifie le calcul de la projection de la position d'une entité sur l'axe médian. Cette opération est nécessaire pour déterminer les coordonnées relatives de l'entité.

L'approche basée sur une modélisation 1.5D est partagée par Karamouzas et al. (2009a) qui utilise des segments de droite, en lieu et place de splines. Les segments sont automatiquement déterminés à partir des arêtes des régions du diagramme de Voronoï généralisé, ou DVG, (Wein et al., 2005) généré à partir d'un plan de l'emprise au sol des obstacles statiques tels que les bâtiments. Le DVG décompose l'environnement en régions dont chaque point est le plus proche d'un certain obstacle. Ainsi, les arêtes de ces régions constituent des zones maintenant une distance maximale à tout obstacle. Les segments ainsi déterminés sont discrétisés en une succession de points rapprochés qui constituent les centres de cercles. Leurs rayons sont obtenus en déterminant la distance minimale entre le point et tous les obstacles environnants. L'utilisation de cercles plutôt que d'une largeur constante permet de varier la forme du couloir dont les extrémités latérales ne sont donc plus rectilignes. Cette approche permet de modéliser avec un seul couloir une zone comportant un rétrécissement là où plusieurs splines auraient été nécessaires avec les approches Thomas et al. (2000a); Willemsen et al. (2003). Cependant, l'utilisation de collections de cercles ne permet pas de représenter finement la topologie d'un environnement urbain. En effet, aucun arrangement de cercles ne permet de représenter correctement les angles entre les obstacles. Le niveau de discrétisation des segments influence beaucoup la qualité de la modélisation en termes de précision. Il impacte nécessairement la consommation mémoire du modèle puisque cela génère un grand nombre de cercles.

Les modèles 1.5D sont adaptés aux environnements urbains extérieurs puisque les couloirs permettent de modéliser efficacement la plupart des zones navigables par les piétons, comme les trottoirs et passages piétons. Cependant, ce type d'environnement physique ne permet que difficilement de modéliser des grands espaces ouverts tels que les places publiques, les parcs, qui doivent être arbitrairement décomposés en plusieurs couloirs. Cette décomposition impacte nécessairement la précision de la simulation puisqu'elle contraint les déplacements des individus. C'est également pour cette raison que les modèles 1.5D ne sont pas utilisés, à notre connaissance, pour modéliser l'intérieur de bâtiment.

2.2.2.2/ LES MODÈLES 2D

Les problèmes posés par les modèles 1.5D sont résolus par une approche 2D au prix d'une complexité calculatoire plus importante.

Les modèles basés sur les graphes 2D sont très utiles pour représenter l'intérieur des bâtiments. C'est le cas du modèle proposé par Braun et al. (2005a) dans lequel les

zones de navigation ou « contextes » représentent des surfaces polygonales convexes. Les contextes sont bornés par des murs ou des entrées/sorties et permettent ainsi de modéliser simplement et précisément les pièces de l'intérieur d'un bâtiment ainsi que les couloirs d'accès. Néanmoins, la complexité calculatoire accrue du modèle 2D impose de faire des choix permettant de conserver des performances satisfaisantes. Ainsi, Braun et al. (2005a) proposent de contraindre la perception des individus aux entités présentes dans le même contexte. Ce choix arbitraire impacte nécessairement la qualité de la simulation. En effet, un piéton doit considérer prioritairement les obstacles dont le temps avant collision est le plus court, quand bien même ceux-ci se trouveraient dans le contexte adjacent.

Un modèle utilisant une approche similaire est proposé par Paris et al. (2005). Dans celui-ci, les zones de navigation sont automatiquement générées en utilisant une triangulation de Delaunay contrainte sur un plan contenant l'emprise au sol des obstacles statiques. Les zones sont classées dans trois catégories : les culs-de-sac possédant une entrée/sortie, les couloirs possédant deux entrées/sorties et les croisements possédant plus de deux entrées/sorties. Cette classification des éléments permet de les regrouper et de générer hiérarchiquement trois niveaux d'abstraction en fusionnant les zones suivant un ensemble de règles. Cette hiérarchisation des espaces de navigation est utilisée pour accélérer la recherche de chemin par les individus en diminuant la complexité des calculs au détriment de la précision en utilisant les niveaux d'abstraction les moins élevés pour les portions de chemin les plus éloignées. Paris et al. (2005) proposent également une optimisation du calcul des perceptions des individus consistant à pré-calculer les informations de visibilité à des endroits représentatifs du graphe. Cette optimisation s'appelle *Potentiel Visibility Sets* (PVS). Elle est calculée uniquement en considérant le niveau d'abstraction le moins précis du graphe pour limiter la consommation mémoire. Le PVS consiste à stocker sur le graphe, à des positions spécifiques, l'ensemble des espaces visibles en tenant compte de l'occlusion des obstacles. Au cours de la simulation, il est ainsi possible d'accéder aux informations stockées les plus proches pour déterminer les zones de l'environnement devant être incluses dans le calcul des perceptions. Cette approche permet de ne pas faire de recherche locale dans le graphe.

2.2.3/ Les modèles basés sur des arbres de décomposition spatiale

Le modèle ``*Janus Simulation Environment*´´ (ou JaSIM) que nous avons déjà présenté dans la section 2.1.2.1 page 32 pour ses modèles dédiés à la simulation de trafic, propose également des modèles 2D, 2.5D et 3D pour la simulation de foules.

Le modèle de l'univers contient toutes les structures de données nécessaires pour décrire et stocker les objets (y compris les corps des agents) situés dans l'environnement. Le modèle de l'univers est divisé en deux structures de données : (i) la description du terrain, et (ii) la localisation des objets dans l'univers.

La description du sol est principalement utilisé pour calculer les perceptions des agents et la hauteur du terrain en chaque point (afin d'appliquer la force de gravité). Cette description du terrain est basée sur une discrétisation appelée carte d'altitudes. Cette carte contient les altitudes des points de référence selon une grille régulière ou irrégulière (De Floriani et al., 2002; Pajarola et al., 2007). Les altitudes des points intermédiaires sont calculées par interpolation tri-linéaire entre les points de référence les plus proches. Le terrain est également utilisé pour déterminer rapidement les zones non franchissables

par le module de modification de l'état de l'environnement.

La seconde structure de données utilisée par JaSim permet de sélectionner des objets situés dans l'environnement en utilisant des critères spatiaux de sélection (champs de perception des agents, etc.). Chaque objet dans l'environnement possède une position et une orientation en 2D ou 3D. La dimension de l'espace est un choix important. Le modèle de l'environnement JaSim propose des implantations de structures de données dédiées aux univers modélisés en 2D, 2.5D et 3D afin de supporter les missions de l'environnement décrites dans la section 1.3.3.2, page 18. Les modèles d'univers 2D, 2.5D et 3D sont basés sur des arbres de partitionnement spatiaux (kD-Tree) (Galland et al., 2014b, 2008).

Bien que JaSim propose des modèles pour les simulations de foules et de trafic, il est important de noter que ces modèles ne peuvent pas cohabiter dans le même contexte d'exécution à l'heure actuelle.

A notre connaissance, il n'existe pas d'autres outils de simulation utilisant un arbre de partitionnement comme structure de données principale de la modélisation de l'environnement.

2.3/ CRITÈRES DE COMPARAISON

Nous proposons de comparer les différents modèles d'environnement présentés selon un certain nombre de critères que nous expliquons dans les paragraphes suivants.

La typologie spatiale Ce critère permet d'établir si un modèle d'environnement permet la modélisation et la simulation de l'espace public d'une ville et/ou l'intérieur de bâtiments. La simulation d'un environnement urbain peut nécessiter l'intégration de ces deux types d'espaces, simultanément ou non, en fonction de ce que l'on cherche à observer, mesurer ou évaluer. Ce besoin de caractériser différents types d'espaces de l'environnement au sein d'un modèle est également au cœur des travaux de Vanbergue (2003b). En effet, elle a proposé une taxonomie des types d'environnements (de la région, aux trottoirs et aux couloirs en passant par les quartiers et les rues) et a défini leurs relations au sein d'une hiérarchie de composition. L'un de ses objectifs était de permettre à des comportements d'agents placés à des niveaux d'abstraction différents d'interagir avec les niveaux associés dans la hiérarchie environnementale.

La typologie d'agent supporté Ce critère de comparaison permet de déterminer si un modèle d'environnement est adapté pour simuler les comportements de différents types d'agents. En effet, une ville est peuplée par un ensemble d'entités hétérogènes que nous classons dans cinq catégories : les véhicules routiers (voitures, camions, bus, etc.), les cycles, les piétons, les trains ou tramways et enfin les bateaux. Cette classification est dépendante du type de mouvements autorisés. Les véhicules routiers ne se déplacent que sur les routes et sont soumis au code de la route. Les piétons se déplacent librement sur tous types de zones de navigation. Les cycles agissent tantôt comme des véhicules tantôt comme des piétons en fonction de la zone de navigation sur laquelle ils se trouvent. Les trains et tramways se déplacent le long de rails et sont soumis à un code particulier différent du code de la route. Les bateaux se déplacent dans un chenal et sont soumis à

un code de navigation fluviale. Ces différents types d'entités nécessitent que le modèle de l'environnement autorise les déplacements avec les degrés de liberté attendus par les modèles de comportement des entités.

La distinction des aspects de perception et de navigation Ce critère nous semble être une nécessité pour obtenir une simulation précise. En effet, un environnement urbain est composé de zones navigables et d'obstacles à la navigation permettant cependant la perception. C'est le cas des barrières bordant des parcs ou des immeubles, mais aussi des fenêtres des habitations. La possibilité de percevoir une zone à travers un objet est un critère négligé dans les modèles exposés précédemment. Tout comme Zeigler et al. (2000) lorsqu'il définit la relation de modélisation entre le système réel et le modèle de simulation, nous pensons que ce modèle doit être une approximation acceptable du système réel. Par conséquent, il doit permettre de reproduire les mécanismes et les comportements du système. Nous pensons que l'absence de prise en compte de ces mécanismes de perception constitue un biais ayant un impact considérable sur les résultats de la simulation. Par exemple, dans le cas d'une évacuation d'un bâtiment dont les pièces sont munies de fenêtres donnant sur des couloirs intérieurs, un agent peut repérer les issues de secours au travers de celles-ci ou même estimer, lorsqu'il perçoit un danger à proximité de la sortie de secours la plus proche, s'il est préférable de se diriger vers une sortie de secours plus éloignée. Sans la prise en compte de ces fenêtres intérieures dans le cadre de la simulation, le comportement des agents sera différent et peut s'éloigner du comportement réel. Nous considérons quatre niveaux permettant d'évaluer ce critère :

— Niveau 0 : il n'existe qu'une seule structure de données et toute zone navigable est librement perceptible par un agent.
— Niveau 1 : il existe deux structures de données distinctes représentant les zones navigables et les zones perceptibles, cependant, ces zones sont identiques et connectées entre elles de manière similaire.
— Niveau 2 : il existe deux structures de données distinctes représentant les zones navigables et les zones perceptibles. Les connections entres ces zones peuvent être différentes entre les deux structures.

La dimension du modèle Ce critère permet d'évaluer la capacité du modèle à représenter toutes les topologies d'environnements urbains et à permettre de simuler les déplacements d'entités aux contraintes physiques variées. Comme nous l'avons montré précédemment, les environnements 1D et 1.5D permettent de modéliser efficacement les réseaux routiers et leurs contraintes spécifiques. Cependant ceux-ci se révèlent difficilement applicables à la simulation des déplacements piétons à l'intérieur de bâtiments. De même, bien que les environnements 2D permettent de modéliser efficacement les zones de navigation de piétons, ils sont très difficilement adaptables à la simulation des flux de trafic routier et augmentent considérablement la complexité des modèles de comportement des conducteurs de véhicules.

La génération du modèle La caractère industrielle de cette thèse nécessite d'étudier la question du mode de création du modèle d'environnement ainsi que celui d'un scénario de simulation. En effet, l'une des applications envisagées de nos travaux consiste à utiliser la simulation pour peupler rapidement des maquettes virtuelles en lieu et place d'ani-

mations prédéterminées. Pour cela, il est nécessaire que la création d'un environnement de simulation ne prenne pas plus de temps que la création d'animations sur des logiciels de modélisation 3D, et ce même lorsqu'aucune source de données extérieure (données SIG, plan topographique, emprises cadastrales, etc.) n'est disponible. Par conséquent, la création manuelle est préférable et la simplicité de celle-ci fait l'objet d'une attention particulière de notre part dans le cadre de nos travaux.

L'exécution en temps réel de la simulation L'un des objectifs des travaux entrepris dans cette thèse est de permettre la simulation en temps réel d'environnements urbains afin de créer des applications immersives dans le domaine du jeu sérieux, des visites virtuelles, etc. Nous considérons dès lors qu'il est primordial d'identifier les modèles permettant d'y répondre. L'intervalle de temps entre le moment où l'être humain agit physiquement (via ses périphériques d'interaction) et celui où le simulateur est capable d'afficher les changements d'états induits par ces actions est qualifié de latence du simulateur. Selon Fuchs et al. (2006), un simulateur est « temps réel » si la latence du simulateur ne peut être perçue par l'utilisateur. En d'autres termes, l'utilisateur humain doit avoir l'impression d'agir instantanément dans le monde virtuel. Pour minimiser cette latence, le simulateur doit réaliser ses cycles d'exécution (voir la figure 1.3 page 18) à une fréquence supérieure ou égale aux capteurs physiques de l'utilisateur ($26Hz$ si nous considérons le sens de la vision).

2.4/ COMPARAISON DES MODÈLES D'ENVIRONNEMENT

Comme le montre le tableau 2.1, il n'existe que peu de modèles permettant de faire cohabiter efficacement les piétons et les véhicules routiers. Les modèles le permettant limitent les interactions entre ces entités à des zones précises comme les passages piétons. En d'autres termes, chacune des entités d'un certain type évolue dans l'environnement comme si les autres types n'existaient pas. Au niveau des zones d'interactions, les autres entités sont considérées comme des obstacles dynamiques, lorsqu'ils sont effectivement gérés. Nous pouvons remarquer que la dimension des modèles est une des raisons principales qui rend difficile, si ce n'est impossible, la cohabitation d'entités aussi différentes que les véhicules et les piétons. Les modèles 1D et 1.5D, généralement préférés pour la simulation de trafic, sont difficilement transposables aux piétons. L'utilisation de la 2D, plus favorable à ces derniers, implique un surcoût calculatoire important et des difficultés supplémentaires à modéliser le comportement des véhicules.

Il existe peu de modèles permettant de connecter les espaces intérieurs des bâtiments et ceux se trouvant à l'extérieur. Bien souvent, les structures de données et les processus d'optimisation imposent de choisir entre ces deux typologies d'environnement.

Il existe peu de modèles permettant de séparer les aspects liés à la perception de ceux liés à la navigation. Nous estimons que l'absence de distinction constitue un manque important qui impacte sur la précision de la simulation, au sens de la relation de modélisation proposée par Zeigler et al. (2000). Il est très courant dans la réalité de rencontrer des éléments non-traversables mais permettant la perception, comme des vitres ou des barrières. Ces éléments sont très présents dans les milieux urbains et devraient être intégrés dans la modélisation.

Modèle	*Typologie d'entité*	*Typologie d'environnement*	*Temps réel*	*Distinction perception-navigation*	*Génération du modèle*	*Dimension*
VIssIM	véhicules	extérieur	oui	niveau 0	manuel et pré-calcul à partir de données SIG	1.5D
S-Paramics	véhicules	extérieur	oui	niveau 0	manuel	1.5D
Aimsun	véhicules	extérieur	oui	niveau 0	manuel	1.5D
TransModeler	véhicules	extérieur	oui	niveau 0	pré-calcul à partir de données SIG	1.5D
JaSim	véhicules ou piétons	mixte	oui	niveau 0	manuel	1D, 1.5D, 2D, 2.5D ou 3D
MATSim	véhicules	extérieur	oui	niveau 0	manuel	1.5D
SUMO	véhicules	extérieur	oui	niveau 0	NC	1.5D
AVENUE	véhicules	extérieur	oui	niveau 0	NC	NC
MITSIMLab	véhicules	extérieur	NC	niveau 0	NC	NC
DRACULA	véhicules	extérieur	oui	niveau 0	manuel	1.5D
ARCHISIM	véhicules	extérieur	oui	niveau 0	NC	1.5D
(Karamouzas et al., 2009a)	piétons	extérieur	oui	niveau 0	manuel et pré-calcul à partir d'un plan	2D
(Thomas et al., 2000a)	véhicules et piétons	extérieur	oui	niveau 0	pré-calcul à partir d'un modèle procédural	1.5D
(Paris et al., 2005)	piétons	mixte	oui	niveau 0	pré-calcul à partir d'un plan	2D
(Treuille et al., 2006)	piétons	mixte	oui	niveau 0	NC	2D
(Tecchia et al., 2001)	piétons	extérieur	oui	niveau 0	NC	2D
(Braun et al., 2005a)	piétons	intérieur	oui	niveau 0	NC	2D
(Shao et al., 2005)	piétons	intérieur	oui	niveau 1	pré-calcul à partir d'un plan	2D

TABLE 2.1 – Comparaison des propriétés des modèles d'environnement physique issus de la littérature

2.5/ CONCLUSION

Nous avons présenté dans ce chapitre des modèles d'environnement physique pour la simulation microscopique en milieu urbain. Ces modèles se distinguent principalement par le type d'entité qu'ils permettent de simuler.

Aucun des modèles que nous avons étudiés ne permet de répondre à l'ensemble de nos objectifs. Peu de modèles permettent de faire cohabiter les piétons et les véhicules routiers. Ceux qui le permettent sont basés sur des modèles 1.5D. Ils sont difficilement compatibles avec la liberté de mouvements des entités comme les piétons.

Les modèles dédiés aux piétons sont parfois inadaptés à tous les types d'environnements que nous souhaitons pouvoir modéliser : certains sont spécialement conçus pour l'intérieur de bâtiments, d'autres sont exclusivement limités à la modélisation d'environnement urbains extérieurs. Une minorité de ces modèles peut être utilisée pour intégrer conjointement ces différents types d'environnement dans une même simulation.

Afin d'atteindre nos objectifs, nous proposons un nouveau modèle d'environnement physique s'inspirant de certains des aspects des modèles que nous avons présentés dans ce chapitre et répondant aux limitations que nous avons mises en avant. Ce modèle, que nous appelons HEDGE (HEterogenous Dual Graph Environment), permet de simuler des entités de différents types évoluant au sein d'une représentation unique de l'environnement. Il permet de simuler à la fois l'intérieur et l'extérieur des bâtiments par un assemblage de zones de navigation. HEDGE décompose l'environnement selon deux aspects : la navigation et la perception. Cette structuration permet aux agents de percevoir toutes les entités, quel que soit leur type, tout en leur attribuant des zones de déplacement spécifiques. Cette décomposition permet également la cohabitation d'agents hétérogènes dans le même environnement et une gestion simplifiée des zones partagées telles que les passages piétons. HEDGE est basé sur une structure de graphe ne contraignant pas la position d'une entité dans l'environnement. Ce graphe autorise également l'intégration de la simulation des mouvements basée sur les lois de la physique.

3

Structure du modèle de l'environnement HEDGE

Les objectifs de nos travaux comprennent trois exigences quant au modèle d'environnement physique choisit : (i) la possibilité de simuler à la fois l'intérieur et l'extérieur de bâtiments, (ii) la simulation simultanée de piétons et de véhicules (principalement routiers), (iii) l'intégration de modèles simulant la physique des entités.

Comme nous l'avons souligné dans le chapitre 2, plusieurs modèles ont été proposés pour simuler des déplacements de piétons et de véhicules dans un milieu urbain. Cependant, aucun ne permet de répondre pleinement à nos exigences. En effet, la plupart de ces modèles ne sont applicables qu'à un seul type d'entité (piéton ou véhicule), ou ne permettent de simuler qu'une seule topologie d'environnement (intérieur de bâtiment ou extérieur uniquement). Parmi ceux spécialement adaptés aux véhicules routiers, très peu sont susceptibles de permettre l'incorporation d'un modèle physique. Par conséquent, nous proposons un nouveau modèle d'environnement physique, nommé HEDGE (HEterogenous Dual Graph Environment), répondant aux problématiques posées.

Sommaire

3.1 Modélisation de l'environnement par un graphe 48
3.1.1 Contexte 48
3.1.2 Modèle de l'environnement 49
3.2 Composantes structurelles de l'environnement 52
3.2.1 Les nœuds 53
3.2.2 Les connecteurs 58
3.2.3 Les liens 60
3.3 Exemples de modélisation 61
3.3.1 L'intersection 61
3.3.2 Le cas du dépassement et du changement de voie 63
3.3.3 Environnement intérieur et extérieur 65
3.4 Composantes non-structurelles de l'environnement 66
3.4.1 Les corps des agents 66
3.4.2 Les obstacles 67
3.4.3 La signalétique 68
3.5 Vers un modèle informé de l'environnement 68
3.6 Conclusion 69

Nous commencerons par introduire les concepts généraux de notre modèle d'environnement 3.1 avant d'en détailler précisément ses éléments structurels 3.2. Nous présentons quelques exemples de modélisation 3.3 avant de décrire les éléments non-structurels présents dans l'environnement 3.4. Enfin, nous présentons dans la section 3.5 l'intégration d'informations ontologiques dans le modèle HEDGE.

3.1/ MODÉLISATION DE L'ENVIRONNEMENT PAR UN GRAPHE

Dans cette section, nous présentons le contexte amenant à la modélisation de l'environnement physique en utilisant un graphe. Ensuite, nous décrivons les principes de notre modèle.

3.1.1/ CONTEXTE

Comme nous l'avons observé dans le chapitre 2, l'utilisation d'un graphe est une approche courante et intuitive pour modéliser l'environnement physique (Lamarche et al., 2004; Paris et al., 2005, 2007a; Stylianou et al., 2004). Cette assertion est d'autant plus confirmée dans le cas des environnements urbains. En effet, ce type d'environnement est un assemblage complexe de zones possédant des sémantiques diverses et particulières (trottoir, route, passage protégé, etc.).

Considérer l'ensemble de l'environnement comme une seule zone continue comme le propose (Galland et al., 2009) rend difficile la classification des obstacles en fonction de leur position. Par exemple, un conducteur de véhicule n'a besoin de tenir compte des piétons présents sur un trottoir que lorsqu'ils peuvent constituer un danger. Sans découpage explicite préalable des différentes zones de l'univers, un agent conducteur doit déterminer à partir des informations provenant de sa perception si un piéton perçu présente ou non un danger potentiel. Cette opération est délicate et potentiellement coûteuse en temps de calcul. Elle est composée de trois phases exécutées par l'agent : (i) segmenter l'espace, (ii) attribuer une sémantique à chacune des zones, (iii) associer les obstacles aux zones.

Certains travaux ont proposé de décomposer l'environnement pour former une hiérarchie de zones spatiales (Demange, 2012; Galland et al., 2014b; Vanbergue, 2003b). Toutefois, ces modèles font face aux mêmes problématiques que précédemment au niveau le plus bas de la hiérarchie. D'autres travaux ont proposé de décomposer l'environnement sous la forme de zones interconnectées (Thomas, 1999). Cependant, ces travaux ne facilitent pas la modélisation et la simulation de différents types d'entités localisées dans l'environnement (piétons, cyclistes, etc.).

Nous synthétisons les différentes propositions concernant les modèles de l'environnement par le méta-modèle illustré par la figure 3.1. Chaque entité du système devant être simulée possède une vue particulière sur l'environnement. Par exemple, pour reproduire le comportement d'un conducteur de véhicule routier, il est nécessaire de lui fournir l'ensemble des routes, la signalétique routière et l'ensemble des autres usagers (y compris les piétons traversant la route). Chaque vue possède un modèle de l'environnement dédié : (i) contenant les objets de l'environnement associés à la vue, et (ii) permettant de calculer efficacement l'ensemble des informations utilisées par les comportements des agents. Chaque modèle d'environnement propose un aspect dédié à la navigation

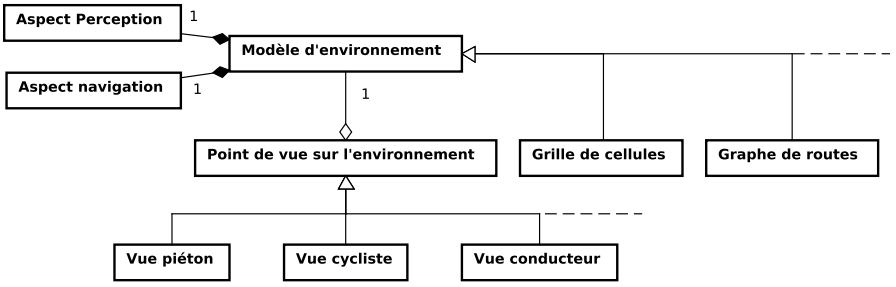

FIGURE 3.1 – Différentes vues sur l'environnement et modèles associés

en son sein, et un autre dédié à la détermination des perceptions des agents. De plus, il permet de structurer l'espace en utilisant une grille de cellules régulières, un graphe de routes, etc. D'après notre connaissance de la littérature, il n'existe pas de modèle de l'environnement supportant plusieurs vues simultanées, garantissant la qualité et la facilité de la modélisation, ainsi que les performances de calcul associées. Dans la suite de ce chapitre, nous proposons un modèle permettant d'intégrer les vues associées aux piétons, aux conducteurs de véhicules et aux cyclistes au sein d'un unique modèle.

Contrairement à l'approche couramment utilisée dans le domaine de la simulation de trafic routier (Gipps, 1981), nous ne considérons pas que les véhicules se déplacent le long du graphe comme le long d'une collection de splines permettant de représenter le réseau routier. Nous considérons le véhicule comme libre de ses déplacements puis nous associons sa position dans l'espace à un emplacement sur le graphe. Cette approche est indispensable au bon fonctionnement du modèle de simulation physique du véhicule. En effet, un véhicule peut être amené à sortir du tracé de la route si celui-ci dérape dans un virage. Cette approche offre également des perspectives intéressantes pour la simulation dans le domaine de la défense puisqu'il s'agit d'un domaine d'application dans lequel les véhicules ne sont pas toujours contraints d'utiliser le réseau routier.

Un graphe offre l'avantage supplémentaire d'associer des informations aux arcs permettant la transition entre les sommets. Il est donc possible d'associer des contraintes sur les transitions entre les zones de navigation. En l'occurrence, un piéton peut emprunter une route pour passer d'un trottoir à un autre, ce qui implique une connexion bidirectionnelle entre la route et le trottoir, mais aucune voiture n'est pour autant autorisée à emprunter ce lien pour se rendre sur le trottoir.

3.1.2/ MODÈLE DE L'ENVIRONNEMENT

Le modèle proposé dans ces travaux de recherche est basé sur une représentation de l'environnement sous la forme d'un graphe défini de la manière suivante :

— Les sommets, ou nœuds, correspondent aux éléments atomiques de l'environnement : les zones de navigation bornées (telles que les voies d'une route, les trottoirs, etc.) sur lesquelles toutes les entités seront localisées.
— Les arcs, ou liens, représentent la connectivité entre les zones et prennent en compte deux aspects : la navigation et la perception.

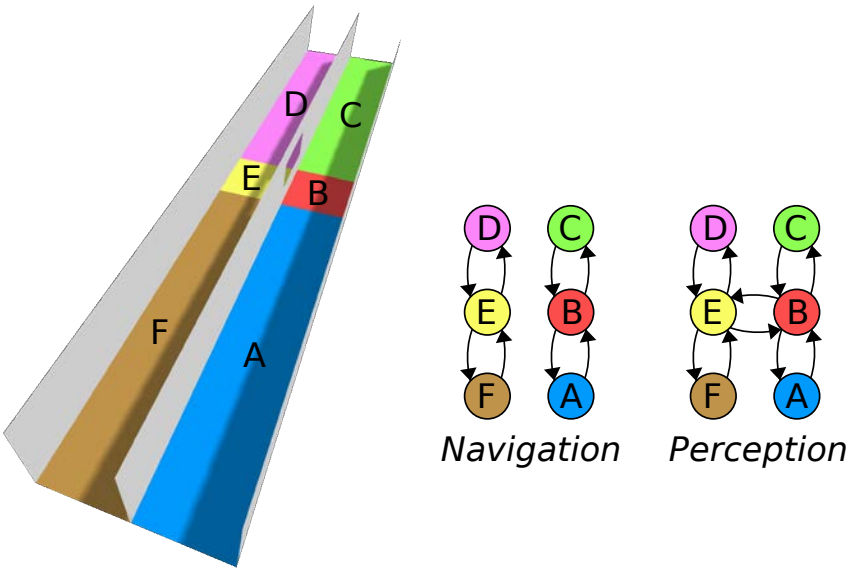

FIGURE 3.2 – Un environnement composé de deux couloirs parallèles connectés par une fenêtre, et la représentation des aspects navigation et perception du graphe associé

La figure 3.2 illustre un exemple que nous utilisons dans la suite de cette section. La partie gauche représente, en trois dimensions, un environnement composé de deux couloirs parallèles. Ces deux couloirs sont respectivement décomposés en trois zones, représentées par les nœuds **A** à **F**. Nous représentons également une fenêtre entre les zones **E** et **B**. Dans cet exemple, il est impossible de naviguer d'un couloir à l'autre. Toutefois, la présence de la fenêtre autorise la perception des objets se trouvant dans le couloir adjacent. La partie droite de la figure 3.2 illustre les deux aspects pris en compte par notre graphe : la navigation et la perception.

Notre proposition utilise une approche de modélisation différente de celles communément utilisées dans la littérature (Galland et al., 2011; Lamarche et al., 2004; Paris et al., 2005, 2007a; Stylianou et al., 2004). Dans ces dernières, les zones de navigation sont souvent considérées comme les liens du graphe, les nœuds constituant de simples points d'accroche ou bien modélisant les intersections. Nous proposons de considérer les zones spatiales comme des nœuds du graphe, et les passages d'une zone à l'autre comme les liens. Ce point de vue est partagé par Bui et al. (2006) et Tungkastan et al. (2008). Notre choix est motivé par le fait que notre graphe est centré sur la connectivité entre les zones de navigation plutôt que sur les zones elles-mêmes, c'est à dire que nous considérons les zones de navigation comme des espaces de l'environnement et les aspects navigation et perception du graphe n'ont pour objectif que de modéliser les relations qui lient ces espaces entres eux. Enfin, les deux aspects de notre graphe de l'environnement nous obligent à modéliser les éléments communs (zones de navigation) par des nœuds.

Nous considérons notre graphe comme statique car nous prenons comme hypothèse que la topologie de l'environnement n'est jamais modifiée au cours d'une simulation. En revanche, il est tout à fait possible et, nécessaire pour certains cas d'études, de modifier dynamiquement les contraintes associées aux liens entre les nœuds. Par exemple, lorsque qu'un accident bloquant complètement la circulation survient au cours d'une simulation, une contrainte interdisant à tout agent de traverser le lien de navigation est

ajouté.

La séparation de notre modèle d'environnement en deux aspects distincts permet de modéliser des connections autorisant uniquement la perception (telles que des fenêtres, des grillages, etc.). Il permet également de contrôler finement l'aspect perception du graphe afin de modéliser des environnements dotés de spécificités telles que des miroirs sans teint. Dans l'exemple illustré par la figure 3.2, il nous suffit de supprimer l'un des liens entre les nœuds **E** et **B** pour obtenir l'effet escompté. À notre connaissance, il n'existe aucun autre modèle 2D permettant de modéliser une fenêtre à partir de laquelle un agent peut voir sans pouvoir y naviguer. Nous détaillons les deux aspects de notre graphe dans les sections suivantes.

3.1.2.1/ Aspect navigation

L'aspect navigation du graphe de l'environnement représente les relations d'adjacence entre zones telles que définies par la définition 7. Ces liens autorisent la navigation entre les zones de l'espace. Cet aspect permet aux agents de déterminer leur potentiel de navigation (définition 8) et ainsi de calculer un chemin pour atteindre leurs objectifs.

Définition 7 : Lien de navigation

Un lien de navigation connectant un nœud **X** à un nœud **Y** implique que les zones de l'environnement associées à ces deux nœuds sont adjacentes et qu'une entité présente sur **X** peut naviguer de **X** vers **Y**.

Définition 8 : Potentiel de navigation

Un nœud **X** est dit **potentiellement navigable depuis Y** s'il existe au moins un chemin dans le graphe de navigation permettant de relier **Y** à **X**.

À l'instar des travaux de Thomas (1999), nous considérons que deux nœuds connectés dans le graphe correspondent à des zones adjacentes. Cette approche facilite la compréhension et la modélisation de la structure de l'environnement en limitant l'explosion du nombre de liens à modéliser. Le graphe de navigation représenté sur la figure 3.2 illustre ce point : il n'existe pas de lien entre A et C. Cette approche possède également d'autres avantages relatifs à son implantation dans un simulateur. Par exemple, il est possible d'utiliser une stratégie de chargement partiel du graphe en mémoire, comme cela a été proposé dans le modèle de rendu 3D « place-portail » (Bui et al., 2006). Seuls les nœuds contenant des agents et les nœuds voisins sont chargés en mémoire.

3.1.2.2/ Aspect perception

L'aspect perception du graphe de l'environnement représente les relations d'adjacence (définition 9) autorisant la perception entre les zones de l'espace. Il a pour rôle de permettre la détermination de ce que les agents peuvent percevoir en réduisant l'espace de recherche aux zones perceptibles présentes dans leurs voisinages.

Définition 9 : Lien de perception

Un lien de perception connectant un nœud **X** à un nœud **Y** implique que les zones de l'environnement associées à ces deux nœuds sont adjacentes et qu'une entité présente sur **X** peut percevoir ce qui se trouve sur **Y**.

Comme il est contre-intuitif qu'il soit possible de naviguer d'un nœud **X** à un nœud **Y** sans qu'il soit possible de percevoir depuis **X** ce qui se trouve sur **Y**, nous faisons l'hypothèse que la présence d'un lien de navigation entre deux nœuds implique nécessairement la présence d'un lien de perception équivalent (reliant les même nœuds dans le même sens). En d'autres termes, le graphe de l'environnement est au moins constitué d'autant de liens de perception (et de même sens) que de liens de navigation entre les zones.

La figure 3.2 illustre notre hypothèse dans une configuration simple. On peut observer en comparant les représentations du graphe dans la partie droite que l'aspect perception enrichie l'aspect navigation. En effet, deux liens entre les nœuds **E** et **B** ont été ajoutés sur l'aspect perception afin de représenter la fenêtre se trouvant entre les deux couloirs.

Définition 10 : Potentiel de perception

Un nœud **X** est dit **potentiellement perceptible depuis Y** s'il existe au moins un chemin dans le graphe de perception permettant de relier **Y** à **X**.

Selon la définition 10, le nœud **A** est potentiellement perceptible depuis **D** ce qui implique que **A** sera inclut dans le domaine de recherche lors du calcul des perceptions d'un agent présent sur **D**. En effet, en observant la représentation 3D sur la gauche de la figure, nous pouvons supposer qu'il soit possible qu'un agent présent sur **D** soit capable de percevoir certains objets sur **A** à travers l'ouverture de la fenêtre entre les deux couloirs.

Selon cette même définition, le nœud **C** est potentiellement perceptible depuis **D** ce qui implique que **C** sera également inclut dans le domaine de recherche lors du calcul des perceptions d'un agent présent sur **D**. Cependant, nous pouvons observer sur la représentation 3D qu'il n'est pas possible qu'un agent présent sur **D** puisse voir un objet de **C** de quelque manière que ce soit à cause du mur séparant ces deux nœuds. Par conséquent, il faut remarquer que l'utilisation de l'aspect perception du graphe de l'environnement ne dispense pas de tenir compte de l'occlusion lors du calcul des perceptions.

Dans la section suivante nous décrivons les composantes structurelles du modèle HEDGE.

3.2/ Composantes structurelles de l'environnement

Le modèle d'environnement physique présenté peut être utilisé pour étudier des environnements urbains en 2D ou 3D en fonction de la définition géométrique des éléments du graphe de l'environnement. Cependant, afin de simplifier la compréhension des différents concepts nous prenons l'hypothèse d'une modélisation 2D dans le reste de ce document.

La structure de l'environnement est composée de quatre éléments principaux (illustrés par le diagramme de classes de la figure 3.3) :

- les **nœuds** (`IPathNode`) représentant les surfaces de navigation,
- les **liens** (`IPathNodeLink`, `INavigationLink`, `IPerceptionLink`) permettant de connecter les surfaces entre-elles,

— les **connecteurs** (`ILinkGroup`, `IPathNodeConnector`) jouant le rôle de collections de liens partageant une même géométrie et certains attributs,
— les **entités** (`IEntity`) représentant les objets mobiles et immobiles dans l'environnement.

Les relations entre ces différents éléments sont décrites dans les sections suivantes.

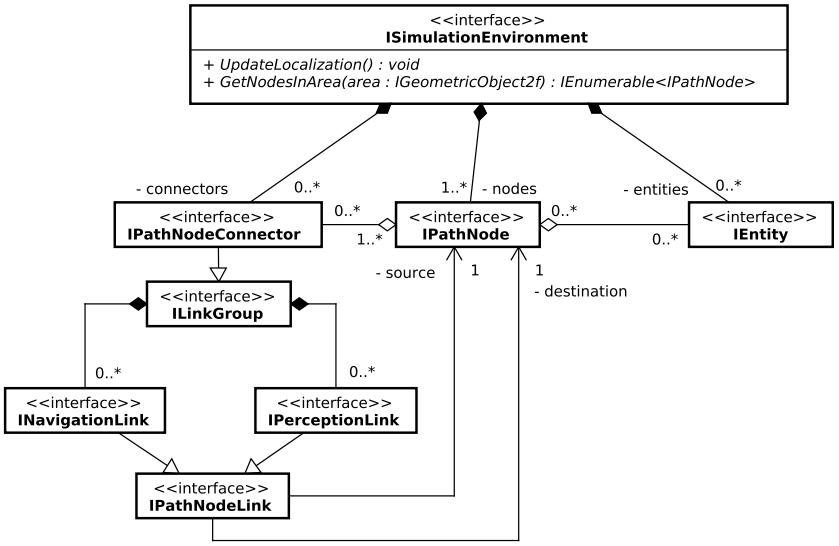

FIGURE 3.3 – Diagramme de classes de l'environnement

3.2.1/ LES NŒUDS

Définition 11 : Nœud

Un nœud est une région continue de l'espace, concave ou convexe, dans laquelle un agent peut se déplacer sans contrainte, bornée par des connecteurs ou des murs.

Les murs sont des obstacles statiques et peuvent être considérés comme des entités (voir section 3.4) implicites. Les connecteurs (voir section 3.2.2) sont des zones de transition entre deux nœuds.

Les modèles de déplacement de piétons sont généralement sensibles à la topologie de l'environnement et peuvent générer des comportements irréalistes voire des blocages dans les concavités (Chalons, 2007; Helbing et al., 2009; Lämmel et al., 2013). De plus, il faut également considérer les contraintes physiques associées au déplacement des véhicules (qui ne peuvent suivre un virage avec une courbure excessive). Pour ces raisons, nous proposons l'utilisation d'au plus trois catégories de nœuds dont la géométrie est suffisamment simple pour faciliter la création manuelle du graphe, mais permettant tout de même de représenter toutes les topologies d'environnement. Ces catégories de nœuds sont : les **couloirs**, les **jonctions** et les **places**.

Un nœud référence l'ensemble des entités telles que les corps des agents et les obstacles statiques en intersection avec sa géométrie, il est donc possible qu'une entité soit

référencée par plusieurs nœuds à un instant donné. Nous utilisons le terme **données de localisation** pour décrire l'ensemble des nœuds avec lesquels une entité est en intersection.

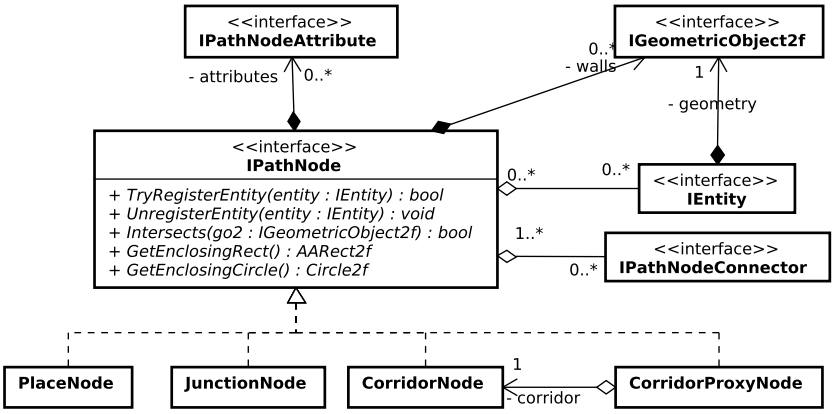

FIGURE 3.4 – Diagramme de classes des nœuds de l'environnement

Comme l'illustre le diagramme de classes 3.4, les nœuds sont représentés par l'interface `IPathNode` implantée par les trois types de nœuds que nous proposons : `PlaceNode` représentant la **place**, `JunctionNode` représentant la **jonction**, `CorridorNode` représentant le **couloir** et `CorridorProxyNode` représentant les **proxies de couloirs** utilisés dans le cas du dépassement de véhicules routiers.

Les nœuds peuvent posséder un certain nombre d'attributs (voir section 3.2.3.1) qui sont utilisés pour stocker des informations spécifiques à l'application permettant ainsi de prendre en compte les aspects liés à la sémantique.

3.2.1.1/ LES COULOIRS

Les couloirs sont des nœuds construits à partir d'une bande, c'est à dire l'extrusion d'une spline le long du vecteur normal. L'axe principal d'un couloir est donc décrit par une fonction S définie par morceau par N polynômes.

$$\begin{aligned} S &: [0,1] \rightarrow \mathbb{R}^2 \\ S(t) &= P_i(t_i) \end{aligned}$$

où P_i est un polynôme de degré quelconque et

$$\begin{aligned} i &= \lfloor tN \rfloor \\ t_i &= tN - i \end{aligned}$$

Nous avons choisi d'utiliser des polynômes de Bernstein (1912) de degré trois faisant des axes centraux de nos couloirs des splines de Bézier cubiques. Ce choix est motivé par la grande flexibilité offerte par ce type de spline. Ils permettent de créer un point d'inflexion par portion de la spline et réduit par là même le nombre de points de contrôle nécessaire pour modéliser un couloir sinueux. Les polynômes que nous utilisons sont donc tous

définis par :

$$P_i : [0,1] \rightarrow \mathbb{R}^2$$
$$P_i(t_i) = \overrightarrow{C}_{i,0}(1-t_i)^3 + 3\overrightarrow{C}_{i,1}t_i(1-t_i)^2 + 3\overrightarrow{C}_{i,2}t_i{}^2(1-t_i)\overrightarrow{C}_{i,3}t^3$$

Avec $\overrightarrow{C}_{i,0}, \overrightarrow{C}_{i,1}, \overrightarrow{C}_{i,2}, \overrightarrow{C}_{i,3}$ les points de contrôle de la portion de spline définie par P_i.

Pour assurer la continuité de la courbure de la spline entre deux portions, il est nécessaire de s'assurer que :

$$\begin{array}{lcl} \overrightarrow{C}_{i,3} & = & \overrightarrow{C}_{i+1,0} \\ \frac{\overrightarrow{C}_{i,3}-\overrightarrow{C}_{i,2}}{|\overrightarrow{C}_{i,3}-\overrightarrow{C}_{i,2}|} & = & \frac{\overrightarrow{C}_{i+1,1}-\overrightarrow{C}_{i+1,0}}{|\overrightarrow{C}_{i+1,1}-\overrightarrow{C}_{i+1,0}|} \end{array}$$

La première contrainte impose qu'une portion commence à la position où se termine la portion précédente. La deuxième contrainte impose que la direction des tangentes au niveau des jonctions entre les portions soit égale en sens et direction. En revanche, il n'est pas nécessaire d'imposer que ces tangentes soient égales en longueur. Ceci permet de contrôler très finement la forme de la spline sans compromettre la continuité de la courbure. La figure 3.5 montre une spline composée de deux portions et met en évidence les relations entre les points de contrôle.

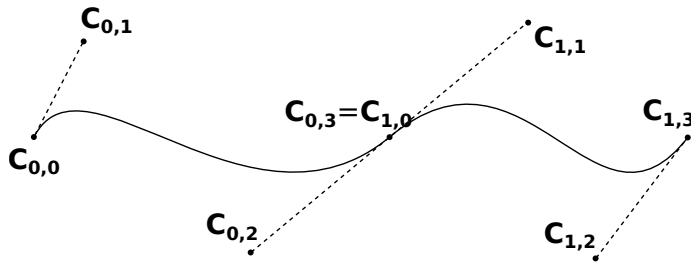

FIGURE 3.5 – Une spline cubique composée de deux portions

Le vecteur normal $\hat{n}$ utilisé pour l'extrusion du couloir est déterminé en tout point à partir du vecteur tangent à la spline $\hat{x}$ en choisissant un des deux vecteurs orthogonaux à ce dernier.

Le vecteur tangent $\hat{x}$ est calculé à partir de la dérivée première des polynômes composant la spline S. La dérivée première d'un polynôme de Bernstein de degré trois est un polynôme de Bernstein de degré deux utilisant les points de contrôle $\overrightarrow{D}_{i,0}, \overrightarrow{D}_{i,1}, \overrightarrow{D}_{i,2}$ définis comme suit :

$$\begin{array}{lcl} \overrightarrow{D}_{i,0} & = & 3(\overrightarrow{C}_{i,1} - \overrightarrow{C}_{i,0}) \\ \overrightarrow{D}_{i,1} & = & 3(\overrightarrow{C}_{i,2} - \overrightarrow{C}_{i,1}) \\ \overrightarrow{D}_{i,2} & = & 3(\overrightarrow{C}_{i,3} - \overrightarrow{C}_{i,2}) \end{array}$$

Finalement la tangente $\overrightarrow{x}$ (non normalisée) et le vecteur tangent $\hat{x}$ sont obtenus pour le polynôme P_i grâce aux formules suivantes :

$$\begin{array}{lcl} \overrightarrow{x} & = & P_i{}'(t_i) = (1-t_i)^2\overrightarrow{D}_{i,0} + 2(1-t)t\overrightarrow{D}_{i,1} + t^2\overrightarrow{D}_{i,2} \\ \hat{x} & = & \frac{\overrightarrow{x}}{|\overrightarrow{x}|} \end{array}$$

Afin de faciliter les tests d'intersection et contourner le problème lié à la complexité du calcul de distance à la spline mis en évidence par Willemsen et al. (2003), la bande et son axe central sont discrétisés. La bande est découpée en sections quadrilatérales alors que son axe central est approximé par une ligne polygonale.

L'axe central discrétisé est conservé et accessible aux agents via leur perception et peut ainsi servir de guide. Par conséquent la spline n'est jamais évaluée au cours de la simulation mais seulement approximée.

La discrétisation peut être basée sur la longueur d'arc pour obtenir des sections de même longueur au niveau de leur axe central. Elle peut aussi être basée sur un découpage uniforme du paramètre de la spline. Nous avons choisi cette deuxième option car elle permet d'approximer plus finement la courbure de la primitive initiale. Il faut cependant ajuster le nombre de sections pour que celles-ci ne soit ni trop petites, ni trop grandes. En effet, des petites sections engendrent un nombre important de tests lors de la détection d'intersections. De même, les grandes sections peuvent engendrer un nombre important de tests lors du calcul des perceptions. En effet, lorsqu'une section est grande, la probabilité que de nombreuses entités s'y trouvent à un instant donné augmente.

La figure 3.6 illustre les différentes étapes permettant la construction d'un nœud de type couloir.

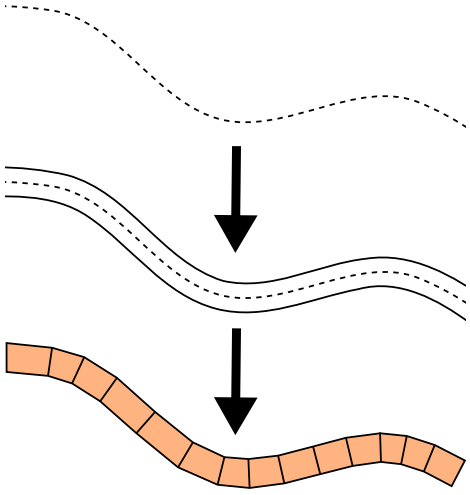

FIGURE 3.6 – Les trois étapes de construction d'un couloir

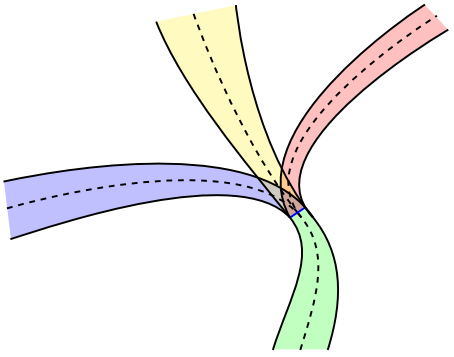

FIGURE 3.7 – Les connections entre couloirs garantissent la continuité dans le tracé par le partage des tangentes et des distances d'extrusion au niveau du point d'accroche

Les tangentes et distances d'extrusion d'un couloir sont fixées à différents points de contrôle dont les deux principaux sont donnés par les deux connecteurs de type *segment* aux extrémités longitudinales. Ces attributs sont ensuite interpolés le long de la spline pour obtenir leur valeur en tout point. Puisque les connecteurs sont partagés entre les nœuds connectés, nous avons la garantie que deux couloirs distincts connectés auront la même tangente et la même distance d'extrusion au point d'accroche, ce qui assure une continuité du tracé comme illustré par la figure 3.7.

Un couloir possède au minimum quatre connecteurs permettant de l'attacher à d'autres nœuds. Ses deux extrémités longitudinales sont rattachées à des connecteurs *segment*,

alors que ses deux extrémités latérales sont quant à elles rattachées à des connecteurs *ligne polygonale*. Comme tout autre type de nœud, le couloir peut également avoir un nombre quelconque de connecteurs *polygone* permettant de gérer les recouvrements de surfaces de navigation. Les différences entre les types de connecteurs sont détaillées dans la section 3.2.2. La figure 3.8 met en évidence les connecteurs de base associés à un couloir.

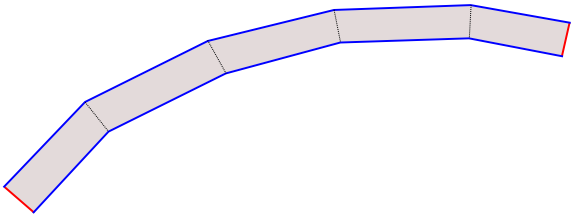

FIGURE 3.8 – Structure générale d'un couloir. Les connecteurs *segment* sont illustrés en rouge, les connecteurs *ligne polygonale* en bleu

Ce type de nœud est parfaitement adapté pour représenter une portion de rails, d'une voie d'une route, d'un trottoir, un couloir à l'intérieur d'un bâtiment ou bien encore une portion d'un couloir de navigation fluvial.

3.2.1.2/ LES PROXIES DE COULOIRS

Les proxies de couloirs sont des nœuds encapsulant un couloir et inversant les liens de navigation. Ils sont utilisés pour représenter les voies de dépassement sur une route tout en permettant la localisation des entités et le calcul des perceptions des agents. Ils représentent une construction équivalente à la superposition de deux couloirs identiques et de sens opposé, connectés entre eux par des liens bidirectionnels de perception. L'utilisation de ces nœuds permet essentiellement de réduire la complexité des opérations de localisation et de perception. Un exemple d'utilisation de ce type de nœuds est présenté dans la section 3.3.2.

3.2.1.3/ LES JONCTIONS

Les couloirs à eux seuls ne sont pas suffisants pour modéliser toutes les topologies que l'on peut trouver dans des environnements urbains réels. En effet, ils sont conçus pour représenter des trajets courbes et ne peuvent donc pas représenter des angles qui sont pourtant indispensables pour modéliser les passages piétons par exemple. Pour ce faire, il nous faut utiliser un second type de nœud que nous appelons **Jonction**. Les jonctions sont de petits nœuds quadrilatéraux dont chaque arête est un **connecteur segment** pouvant être attaché à l'extrémité longitudinale d'un couloir. Ce nœud permet de créer des connexions entre les couloirs en angle droit et donc de modéliser des croisements de type "L", "T" ou "X" comme illustré par la figure 3.9.

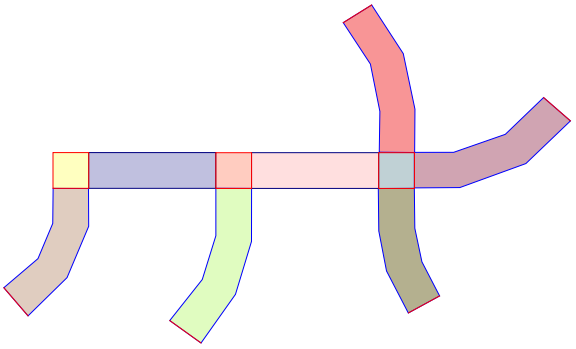

FIGURE 3.9 – Modélisation de croisements de type ”L”, ”T” et ”X” à l’aide de jonctions et de couloirs

3.2.1.4/ LES PLACES

Un assemblage de couloirs et de jonctions permet de modéliser de nombreuses topologies d’environnement. Cependant, ces éléments sont insuffisants pour représenter de larges zones de déplacement libre comme par exemple les pièces à l’intérieur d’un bâtiment et les places publiques. Pour pallier à ce manque, nous introduisons un type de nœud appelé **place**.

Une place représente une zone de forme quelconque approximée par un polygone convexe ou concave mais non croisé et sans trou. Un nombre quelconque de connecteurs *segment* peuvent être placés le long de ses arêtes à condition qu’aucun d’entre eux ne soient en intersection comme illustré par la figure 3.10. Toute arête d’une place qui n’est pas un connecteur est un mur et ne doit pas être traversé par un agent.

Les places pourraient en fait permettre de modéliser n’importe quel type d’environnement piéton comme l’ont montré Lamarche et al. (2004), Stylianou et al. (2004), Paris et al. (2007a), et Pelechano et al. (2007). Ce sont en fait des généralisations de nos couloirs et jonctions. Cependant les opérations de localisation et de calcul des perceptions des agents sur ce type de nœud sont plus complexes et plus coûteuses en temps de calcul. De plus, lorsqu’il n’est pas possible d’obtenir un plan de l’environnement pour générer automatiquement le graphe, la conception du modèle 2D basée uniquement sur des places est une opération délicate, qui prend d’avantage de temps et est une source d’erreur potentielle. Ceci rend la saisie à l’aide de couloirs beaucoup plus avantageuse.

3.2.2/ LES CONNECTEURS

Les connecteurs sont des objets partagés entre les nœuds. Ils représentent les zones de connexion et stockent les liens de navigation et de perception pouvant éventuellement relier les nœuds. Un connecteur modélise un contact entre les surfaces de navigation et a, par conséquent, une représentation géométrique dans l’espace.

La principale fonction du connecteur est de partager certains attributs entre les nœuds qu’il connecte (voir section 3.2.1.1) mais également entre les liens qu’il contient. C’est le cas en particulier de sa géométrie mais aussi de contraintes qui peuvent y être attachées. Un connecteur peut être spécialisé pour lui ajouter d’autres attributs qui peuvent potentiellement influencer l’utilisation des liens qu’il contient. Par exemple, un connecteur

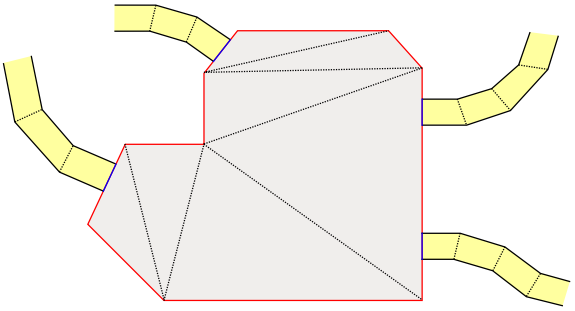

FIGURE 3.10 – Un nœud de type place connecté à des couloirs par l'intermédiaire de connecteurs simples (en bleu)

peut modéliser une porte et n'autoriser un agent à utiliser ses liens de navigation et/ou de perception que lorsque la porte est dans l'état « ouvert ».

Toute entrée ou sortie potentielle d'un nœud est représentée par un connecteur. Cependant, la présence d'un connecteur reliant deux surfaces de navigation n'autorise pas pour autant un agent à le traverser. Lorsqu'un connecteur ne contient aucun lien de navigation ou bien lorsque qu'aucun d'entre eux n'est utilisable par un agent, ce dernier doit considérer le connecteur comme un obstacle infranchissable au même titre qu'un mur. Il existe trois types de connecteurs qui diffèrent par leur géométrie : les **connecteurs *segment***, les **connecteurs *ligne polygonale*** et les **connecteurs *polygone***.

3.2.2.1/ LES CONNECTEURS *segment*

Les connecteurs *segment* sont les plus couramment utilisés : (i) aux extrémités longitudinales des couloirs, (ii) aux extrémités longitudinales des proxies de couloirs, (iii) sur chaque côté des jonctions, (iv) sur les arêtes des places. Ils modélisent une relation d'adjacence directe entre deux surfaces de navigation. Leur représentation géométrique est un simple segment qui correspond à la zone de recouvrement entre les nœuds connectés. La traversée de ce type de connecteur ne nécessite en principe aucun comportement particulier hormis le respect des contraintes de navigation pouvant s'appliquer sur les liens, à la différence des autres types de connecteur.

3.2.2.2/ LES CONNECTEURS *ligne polygonale*

Les connecteurs *ligne polygonale* modélisent une relation d'adjacence latérale entre, au plus, deux nœuds de type couloir. Ces connecteurs sont principalement utilisés pour lier les voies adjacentes d'une route. Lorsque deux voies adjacentes d'une route en sens opposés doivent être connectées par l'intermédiaire de ce type de connecteur, nous utilisons un proxy de couloir pour inverser virtuellement le sens de navigation. Ils permettent aux voitures de dépasser un véhicule à l'arrêt ou changer de voie pour tourner au niveau d'une intersection. Par ailleurs, lorsqu'ils sont utilisés pour connecter une route à un trottoir, ils autorisent des agents piétons à traverser à n'importe quel point. Leur représentation géométrique est une ligne polygonale, c'est à dire une collection de segments contigus qui correspondent aux côtés latéraux en communs des deux nœuds

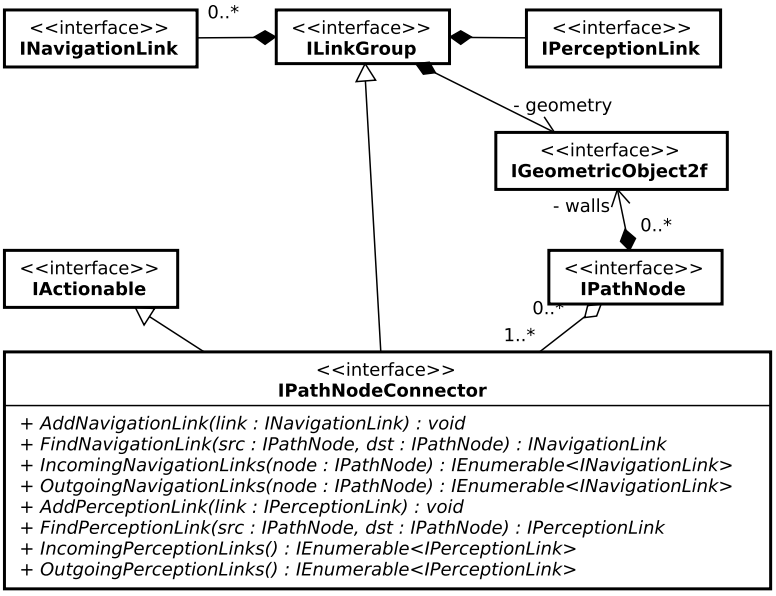

FIGURE 3.11 – Diagramme de classes des connecteurs de l'environnement

connectés. La traversée d'un connecteur de ce type nécessite généralement un comportement adapté :

— pour les véhicules routiers, il est préférable d'utiliser un modèle de dépassement comme ceux proposés par Ahmad et al. (2000); Ghods et al. (2011); Olstam (2005) ou de changement de voie tels que ceux proposés par Ahmed (1999); Choudhury (2005); Ehmanns et al. (2000); Errampalli et al. (2008),
— pour les piétons, il est nécessaire d'utiliser un modèle de traversée libre, c'est à dire en dehors d'un passage piéton modélisé à l'aide de jonctions, de couloirs et donc de connecteurs directs (Godara et al., 2007; Papadimitriou et al., 2009; Waizman et al., 2003).

3.2.2.3/ LES CONNECTEURS *polygone*

Les connecteurs *polygone* modélisent une relation entre deux nœuds se superposant géométriquement sans avoir de connexions directes, c'est à dire sans autre connexion utilisant l'un des deux connecteurs précédents. Ils sont utilisés pour ajouter des liens (de perception généralement) permettant aux agents situés sur un nœud de percevoir les entités se trouvant sur l'autre. La géométrie du connecteur représente la zone de superposition que nous approximons par un rectangle aligné sur les axes pour accélérer les calculs de perception. Ces connecteurs sont essentiels pour modéliser les intersections de plusieurs routes en particulier lorsqu'une voie tournante croise les voies dans le sens opposé comme illustré par la figure 3.12.

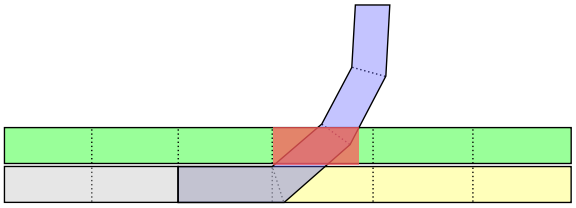

FIGURE 3.12 – Intersection de deux couloirs et un connecteur de conflit illustré par son rectangle englobant (en rouge). Il n'est pas nécessaire d'ajouter un connecteur de conflit entre le couloir jaune et le couloir bleu car ils sont reliés au même connecteur *segment*

3.2.3/ LES LIENS

Les liaisons entre les nœuds du graphe sont modélisées par des objets **liens**. Ces liens sont unidirectionnels et connectent exactement deux nœuds représentant respectivement la source et la destination de la liaison. Afin de créer une liaison bidirectionnelle il suffit de rajouter un lien de sens opposé connectant les mêmes nœuds et rattaché au même connecteur.

Notre modélisation étant basée sur deux aspects d'un même graphe, nous spécialisons les liens en deux catégories : les **liens de navigation** permettant d'informer les agents de la possibilité de naviguer d'un nœud vers un autre et les **liens de perception** utilisés par les capteurs présents sur les corps des agents permettant de détecter les entités à proximité. Chaque lien peut contenir un certain nombre de contraintes limitant son utilisation et hérite des contraintes spécifiées globalement au niveau du connecteur auquel il appartient. Enfin comme tout autre élément de l'environnement, un lien peut posséder un certain nombre d'attributs.

3.2.3.1/ LES CONTRAINTES ET ATTRIBUTS

Les contraintes permettent de limiter l'utilisation d'un lien par les agents. Elles sont souvent utilisées sur les liens de navigation pour forcer certains types d'entités à suivre un chemin spécifique. Ces contraintes peuvent s'appliquer à tous les types de liens. Il est également possible de limiter la perception des agents en fonction du type de capteur dont leur corps est muni ou en fonction de leur morphologie. Par exemple un agent représentant un enfant pourrait ne pas être en mesure de percevoir à travers une fenêtre si celle-ci est trop haute.

Les attributs sont des informations pouvant être attachées sur les nœuds, les connecteurs, les liens et les entités de l'environnement. Ils ont pour principal objectif de permettre d'ajouter à l'environnement des informations spécifiques à l'application. Ils peuvent également être utilisés pour implanter un mécanisme de communication par *stigmergy* (Beckers et al., 1994).

Dans la section suivante, nous présentons trois exemples significatifs de topologies de l'environnement afin d'illustrer l'utilisation des éléments structurels dont nous avons parlé dans les sections précédentes.

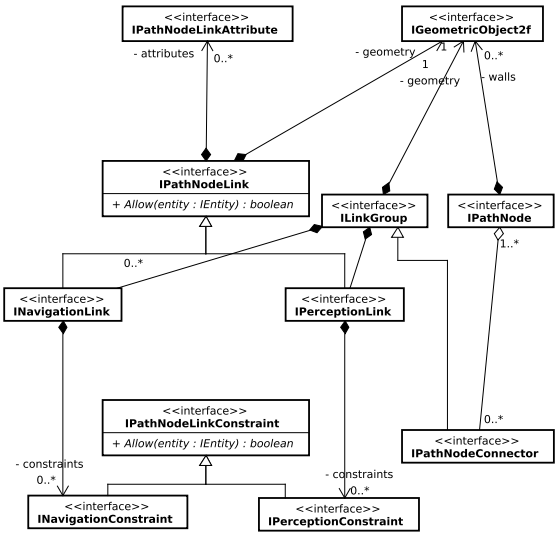

FIGURE 3.13 – Diagramme de classes des liens de l'environnement

3.3/ EXEMPLES DE MODÉLISATION

Dans cette section, nous illustrons la capacité du modèle HEDGE à représenter trois exemples significatifs de topologies de l'environnement.

3.3.1/ L'INTERSECTION

Considérons le cas d'une intersection en X dans une ville illustrée par la figure 3.14. L'intersection est bordée de trottoirs reliés entre eux par des passages piétons. Dans cet exemple, chaque véhicule entrant au niveau de l'intersection n'a que deux choix : aller tout droit ou tourner à droite. L'option tourner à gauche est laissée de côté pour ne pas complexifier inutilement le graphe. On considère que les piétons ne peuvent traverser la route qu'au niveau des passages piétons. Enfin, l'exemple se situant en ville, le dépassement de véhicule est prohibé.

Le zonage de l'environnement en nœuds que nous préconisons dans ce cas est illustré par les figures 3.15 et 3.16.

Les deux aspects du graphe correspondant sont représentés dans la figure 3.17

Dans ce graphe, les cercles représentent les nœuds de type *couloir*, les carrés représentent les nœuds de type *jonction*. Les flèches noires représentent un couple de liens de navigation et de perception entre deux nœuds, les flèches rouges représentent des liens de perceptions (uniquement). Les différentes couleurs des nœuds ont pour but de différencier la vue piéton de la vue véhicule routier.

On voit clairement que le graphe de perception enrichit le graphe de navigation, en particulier au niveau des routes. Il existe un lien de perception mais aucun lien de navigation permettant de parcourir le réseau de route en sens inverse. Par conséquent, tout véhicule sera capable de percevoir ce qui se trouve derrière lui sans avoir la possibilité de faire marche arrière ou bien de rouler à contre-sens.

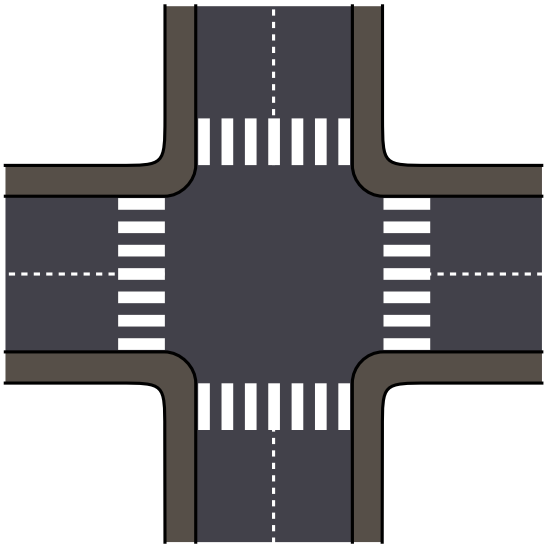

FIGURE 3.14 – Une intersection en X dans une ville. Chaque véhicule entrant au niveau de l'intersection ne peut aller que tout droit ou tourner à droite.

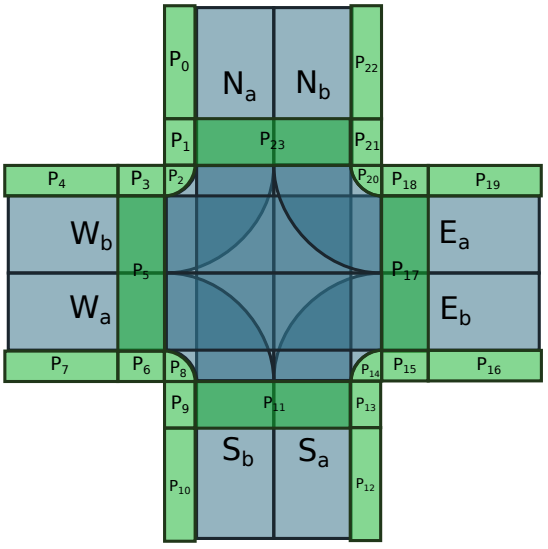

FIGURE 3.15 – Découpage de l'environnement en une collection de nœuds de type *couloir* et *jonction*.

La distinction entre les deux aspects (navigation et perception) du graphe permet également de lier les zones de navigation réservées aux piétons aux zones de navigation réservées aux véhicules. Les passages piétons, représentés par les nœuds P_5, P_{11}, P_{17} et P_{23} sont connectés par des liens de perception aux nœuds N_a, N_b, W_a, W_b, S_a, S_b, E_a et E_b représentant les portions de routes sur lesquelles les passages piéton sont situés. Puisqu'il s'agit ici d'une superposition de nœuds, les liens de perception sont associés à des connecteurs *polygone* représentant chacun la totalité de la surface d'un passage piéton.

On notera un cas de figure similaire entre les nœuds C_{NS}, C_{WE}, C_{EW} et C_{SW}. En effet, ces nœuds se coupent mais ne disposent d'aucune connexion par l'intermédiaire de connecteurs *segment* ou *ligne polygonale* ni directement, ni par l'intermédiaire de l'un de leur voisins. Il est donc nécessaire d'ajouter un connecteur *polygone* permettant aux agents conducteurs de percevoir d'autres véhicules pouvant couper leurs trajectoires. Ce point est très important lorsque la charge sur le réseau routier au niveau de l'intersection devient trop importante pour que le trafic puisse se résorber rapidement. Dans la réalité, il n'est pas rare que des bouchons se créent et que certains véhicules se retrouvent bloqués temporairement au centre de l'intersection. Grâce à l'utilisation des connecteurs *polygone*, nous sommes assurés que les véhicules bloqués seront perçus par tous les autres.

Toutes les superpositions de nœuds ne nécessitent pas systématiquement la création d'un connecteur *polygone*. Par exemple, les nœuds C_{EW} et C_{EN} se superposent mais il existe déjà un lien de perception indirect entre ceux-ci par l'intermédiaire du nœud E_a.

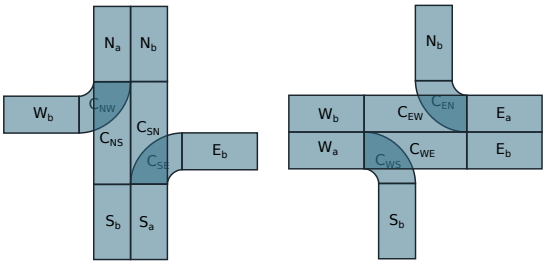

FIGURE 3.16 – Vue détaillée des nœuds routiers correspondant au découpage 3.14.

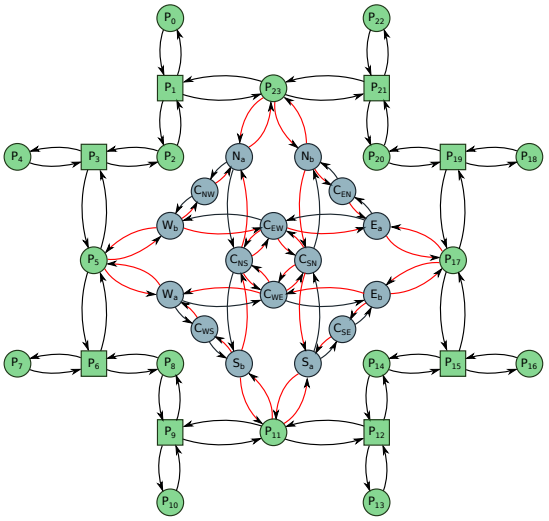

FIGURE 3.17 – Le graphe correspondant au découpage 3.15. Les cercles représentent les *couloirs*, les carrés les *jonctions*. Les flèches noires représentent les couples de liens de navigation et de perception, les flèches rouges représentent les liens de perception uniquement.

3.3.2/ LE CAS DU DÉPASSEMENT ET DU CHANGEMENT DE VOIE

Dans cet exemple, nous prendrons le cas d'une route départementale où le dépassement de véhicules est localement autorisé en empruntant la voie de sens opposé. Nous prendrons également le cas d'une portion de route possédant deux voies adjacentes et de même sens pour illustrer le cas du changement de voie. La situation est représentée par la figure 3.18. La navigation des piétons est ignorée pour simplifier la compréhension des concepts spécifiques au trafic routier.

Nous proposons le zonage illustré par la figure 3.19 reprenant la méthodologie utilisée dans les systèmes d'information géographique (SIG), à savoir, qu'un tronçon de route est modélisé par un élément atomique tant qu'aucun attribut ne vient à changer. Dans notre cas, un tronçon est un composé de plusieurs voies ce qui explique la distinction entre les nœuds C_0 et C_1. Plus précisément, il existe trois tronçons dans cet exemple. Le premier est composé des nœuds A_2, B_2 et C_0, le deuxième est composé des nœuds A_1, B_1 et C_1, enfin le troisième est, lui, composé des nœuds A_0 et C_2.

Le graphe de ce scénario est représenté sur la figure 3.20. Sur cette figure, les nœuds en pointillés représentent les proxies de couloirs tels que définis dans la section 3.2.1.2. Le nœud A'_0 est un proxy du couloir A_0 ce qui signifie que lorsqu'un véhicule emprunte le proxy, il est localisé sur le couloir A_0 sans pour autant en avoir connaissance. Ce point est important car du point de vue du conducteur, il n'est pas nécessaire de savoir que la voie réelle sur laquelle celui-ci roule est dirigée en sens inverse de son mouvement et il est même préférable qu'il l'ignore afin de lui interdire de changer d'itinéraire et faire demi-tour en pleine voie. Dans notre exemple, nous devons empêcher qu'un conducteur empruntant la voie A_0 pour dépasser un véhicule se trouvant sur la voie C_2 ne décide de

FIGURE 3.18 – Une route départementale au niveau d'un agrandissement de une à deux voies.

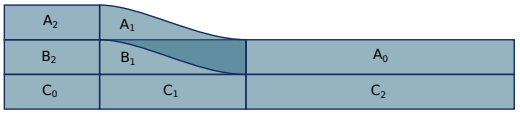

FIGURE 3.19 – Découpage de l'environnement en une collection de nœuds de type *couloir*.

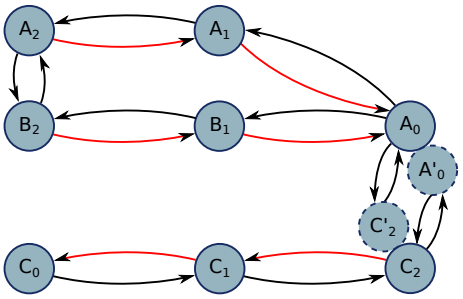

FIGURE 3.20 – Le graphe correspondant au découpage 3.19. Les cercles en pointillés représentent les proxies de couloirs. Les flèches noires représentent les couples de liens de navigation et de perception, les flèches rouges représentent les liens de perception uniquement.

se diriger vers les voies A_1 et B_1. Or, comme nous pouvons le constater sur la figure, il n'existe aucun lien de navigation direct ou indirect entre A_0 et C_2 grâce à l'utilisation du proxy A'_0. Ce qui importe en revanche, est qu'un agent puisse identifier cette voie comme une voie spécifique ne pouvant être utilisée que dans le cas du dépassement, grâce à la sémantique. Il doit aussi être capable de percevoir les véhicules pouvant représenter des obstacles à sa manœuvre ce qui est garanti par le proxy lui-même : percevoir les entités présentes sur A'_0 est équivalent à percevoir directement les entités présentes sur A_0 et réciproquement.

Le cas du changement de voie est illustré sur le premier tronçon de route, celui comprenant les voies A_2, B_2 et C_0. Il est simplement autorisé par l'ajout d'un couple de liens de navigation (et d'un couple de liens de perception) entre les nœuds A_2 et B_2. Ces liens se font par l'intermédiaire d'un connecteur *ligne polygonale* puisque la connexion se fait sur le côté latéral des deux couloirs. Finalement ce type de manœuvre est identifiable par la sémantique associée aux liens qui permet aux agents de différencier ceux-ci des autres liens de navigation comme par exemple entre A_1 et A_2.

3.3.3/ ENVIRONNEMENT INTÉRIEUR ET EXTÉRIEUR

Dans cet exemple, nous prendrons l'exemple d'un bâtiment composé de trois pièces connectées via des couloirs et dont la pièce centrale est ouverte sur une rue. La rue ne comporte aucun passage piéton et la traversée libre est autorisée. La figure 3.21 représente la situation décrite.

Nous proposons le zonage donné en figure 3.22. Celui-ci est composé de *places* pour modéliser les pièces du bâtiment, de *couloirs* pour modéliser les couloirs reliant les

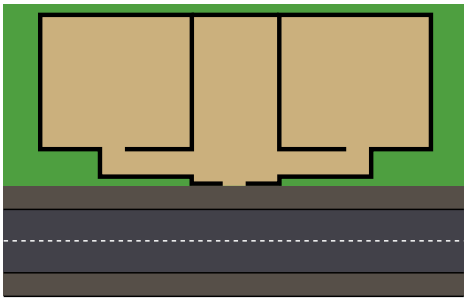

FIGURE 3.21 – Un bâtiment composé de trois pièces ouvert sur une rue.

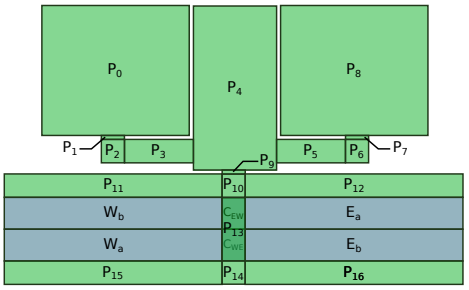

FIGURE 3.22 – Découpage de l'environnement en une collection de nœuds de type *couloir*, *jonction* et *place*.

pièces, les trottoirs et les routes, de *jonctions* pour relier les couloirs entre eux.

Le graphe correspondant à notre découpage est représenté sur la figure 3.23. L'utilisation de *places*, *couloirs* et *jonctions* pour modéliser l'intérieur du bâtiment est relativement simple à appréhender. On notera cependant la présence de minuscules couloirs P_1, P_7 et P_9. Leur existence est liée à notre choix de ne pas autoriser les connexions directes entre les nœuds jonctions et places. Nous rappelons que ce choix est motivé par la volonté de simplifier des procédures de génération manuelle du graphe compte tenu des contraintes industrielles imposées à nos travaux. La connexion du bâtiment au trottoir se fait par l'intermédiaire d'une *jonction* puisque les nœuds de type *couloir* représentant l'intérieur du bâtiment sont du même type que ceux représentant les trottoirs.

Pour permettre la traversée libre de la rue par les piétons, il est nécessaire de connecter latéralement les trottoirs aux voies de la route et ces dernières entre elles par l'ajout de liens de navigation et de perception. Sur ces liens sont ajoutées des contraintes interdisant aux véhicules de les emprunter. Une information sémantique est également attachée à ces liens pour permettre aux agents d'adapter leur comportement (vérifier qu'il n'y a aucun véhicule en approche avant de s'engager). Sur la figure, ces liens sont modélisés par des flèches bleues. Ils permettent à un piéton, par exemple, de naviguer de P_{11} à W_b puis à W_a et enfin P_{15}. Ces liens connectent latéralement les nœuds via des connecteurs *ligne polygonale*. En raison de l'utilisation d'une *jonction* pour connecter l'entrée du bâtiment au trottoir (le nœud P_{10}), il est impossible de créer une connexion avec la route à cet endroit précis. Pour rappel, un nœud *jonction* ne peut être connecté qu'au connecteur longitudinal d'un couloir. Afin d'autoriser malgré tout la traversée des piétons en ce lieu, nous modélisons un passage piétons par l'ajout du nœud P_{13} dont les liens possèdent la même sémantique que ceux connectant latéralement les trottoirs aux voies des routes.

En fonction de l'implantation du comportement piéton, il peut être également utile d'ajouter une contrainte aux liens de navigation connectant les sections de voies de route pour empêcher les piétons de continuer leur chemin sur la route et ainsi les forcer à rejoindre un trottoir.

Dans la section suivante, nous présentons les composantes non-structurelles de l'environnement et en particulier les corps des agents.

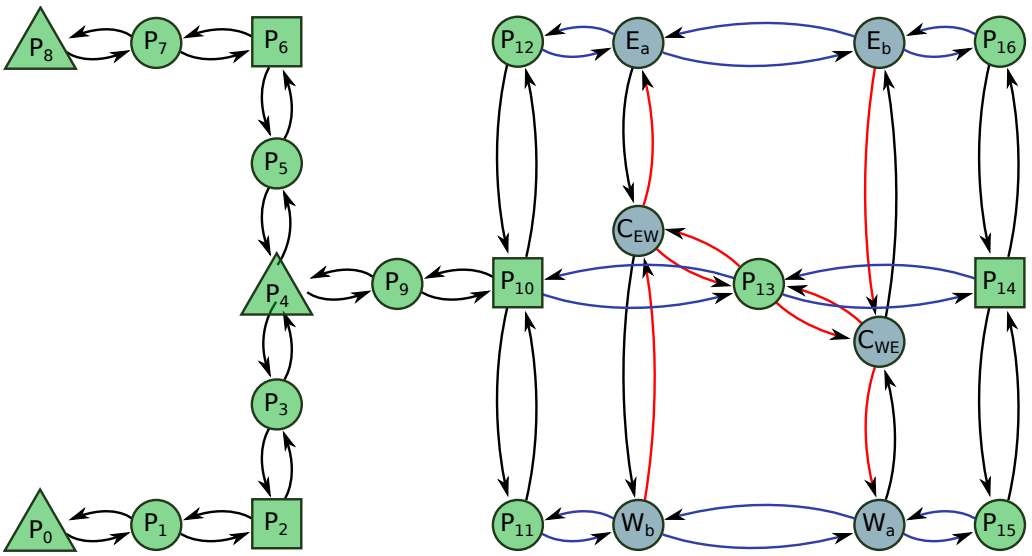

FIGURE 3.23 – Le graphe correspondant au découpage 3.22. Les cercles représentent les *couloirs*, les carrés les *jonctions* et les triangles représentent les *places*. Les flèches noires représentent les couples de liens de navigation et de perception, les flèches rouges représentent les liens de perception uniquement et les flèches bleus sont les couples de liens de navigation et de perception réservés aux piétons pour la traversée de routes.

3.4/ COMPOSANTES NON-STRUCTURELLES DE L'ENVIRONNEMENT

Les composantes non-structurelles de HEDGE sont les entités. Une entité est un objet mobile ou immobile de l'environnement ne faisant pas partie de sa topologie et pouvant être perçu par un agent à l'aide d'un capteur adéquat. Nous distinguons trois types d'entité dans notre modèle : les corps des agents, les obstacles et la signalétique. Une entité possède une géométrie puisqu'elle représente un objet localisé. Cependant une entité n'est pas nécessairement un objet physique devant être considéré comme un obstacle. Les corps des agents et les obstacles constituent des objets physiques de l'environnement alors que la signalétique permet d'insérer des informations dans l'environnement pour guider la navigation des agents.

3.4.1/ LES CORPS DES AGENTS

Les corps des agents sont les principales entités présentes dans l'environnement. Nous utilisons la même définition du corps d'un agent que celle présente dans la plateforme JASIM (Galland et al., 2009). En effet, nous considérons le corps comme une interface entre l'agent et l'environnement agissant comme un filtre. Il permet d'une part de convertir les données extraites depuis l'environnement vers un format compréhensible et exploitable par l'agent et d'autre part de convertir les intentions de l'agent sous la forme d'influences sur l'environnement. Le premier élément est une partie du mécanisme de perception et est géré par un ensemble de capteurs, le deuxième élément est une partie du mécanisme d'action et est géré par un ensemble d'effecteurs. Cette séparation explicite du corps de

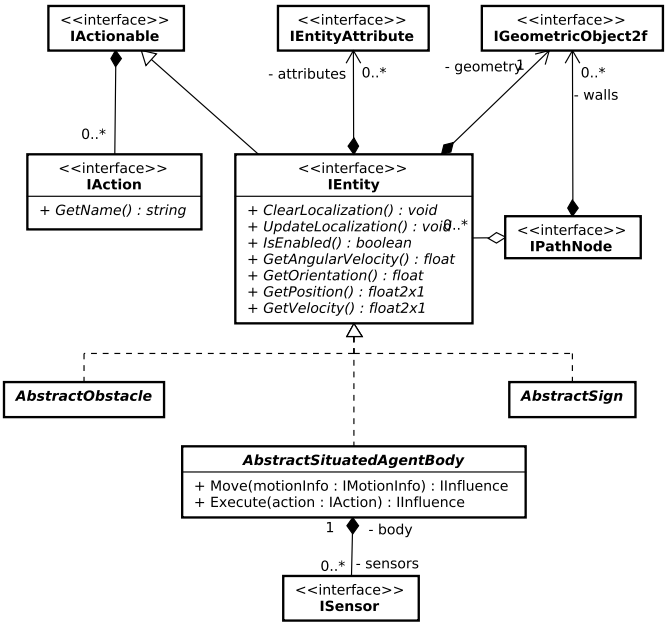

FIGURE 3.24 – Diagramme de classes des entités de l'environnement

l'agent et de son esprit permet une grande flexibilité durant l'exécution d'une simulation. Il est possible de changer dynamiquement le corps d'un agent pour modéliser un changement modal sans pour autant modifier le comportement.

Un corps étant un objet physique, il est représenté par une géométrie, possède une dynamique propre soumise aux lois physiques de l'environnement et met à disposition des agents ses attributs cinématiques tels que sa vitesse et son accélération instantanée, la direction de son mouvement, etc. Ces informations sont accessibles en lecture seule via la perception.

3.4.2/ LES OBSTACLES

Les obstacles sont des entités représentant des objets physiques de l'environnement qui ne sont pas soumis à un contrôle explicite d'un agent. Les agents peuvent potentiellement interagir avec les obstacles, les déplacer par exemple, par l'intermédiaire de l'émission d'influences depuis leurs corps.

Tout comme le corps d'un agent, un obstacle est représenté par une géométrie et expose ses attributs cinématiques tels que sa vitesse et son accélération instantanée, la direction de son mouvement, etc. A la différence d'un corps, un obstacle n'a pas de dynamique propre et ne fait que subir les effets des actions des agents et la dynamique endogène de l'environnement.

3.4.3/ LA SIGNALÉTIQUE

La signalétique est un type d'entité particulier ne représentant pas d'objet physique mais une information ponctuelle ou associée à une zone restreinte de l'environnement. La

distinction de la signalétique de sa représentation physique permet de placer chacun des deux éléments à l'endroit le plus approprié. Dans notre modèle, un feu de signalisation est représenté à la fois par un objet physique placé sur le trottoir pouvant interférer avec les déplacements des agents piétons mais ne véhiculant aucune information particulière. La signalétique ou l'objet informatif, quant à lui placé sur la route, permet aux agents conducteurs de véhicules de s'arrêter lorsque le feu est rouge. Par conséquent, un agent conducteur n'a pas l'obligation de percevoir les entités présentes sur le trottoir pour choisir les actions qu'il doit effectuer sur la route ce qui simplifie le calcul de ses perceptions.

Dans la section suivante nous présentons une méthode permettant d'intégrer des informations ontologiques dans le modèle HEDGE en vue d'en faire un modèle d'environnement urbain (UEM, ``*Urban Environment Model*´´).

3.5/ VERS UN MODÈLE INFORMÉ DE L'ENVIRONNEMENT

Les agents peuvent exhiber des comportements différents en fonction de leurs connaissances sur eux-mêmes (état émotionnel, personnalité, etc.) et sur l'environnement (lieux particuliers, lois de l'univers, etc.). Les environnements informés sont des « modèles de l'environnement intégrant tout ou parties de la connaissance sur le domaine de l'application simulée » (Farenc et al., 1999; Paiva et al., 2005). Thalmann et al. (2013) considère que les environnements informés basés sur les ontologies permettent de créer des comportements plus complexes et plus réalistes dans les systèmes multi-agents. Il est nécessaire de préserver l'individualité et l'autonomie de chaque agent, afin d'augmenter le réalisme des simulations. Toutefois, dans le même temps, des scénarios complexes et composés d'un grand nombre d'agents doivent être pris en compte. L'utilisation d'ontologies et d'outils de raisonnement automatiques permet de réconcilier ces deux points de vue en permettant la génération et le contrôle de comportements complexes à la fois individuels et collectifs.

L'intégration de la connaissance, et plus particulièrement des ontologies, dans les modèles de simulation de foules fait l'objet de nombreux travaux dans la littérature dédiée aux environnements virtuels. Dans (Barros et al., 2004; Braun et al., 2003; Musse et al., 2001), les auteurs se focalisent sur la simulation de foules durant des situations de crise. D'autres ont étudié les comportements « normaux » des individus avant que ces situations de crise ne se produisent. Dans (Paiva et al., 2005), les auteurs qualifient ces types d'environnement informé de « modèles d'environnement urbain » (UEM, ``*Urban Environment Model*´´). Enfin, Durig (2014) propose un modèle d'UEM et un mécanisme d'inférence pouvant être utilisés par des agents pour déterminer un plan d'actions à réaliser dans l'environnement.

Les UEM sont utiles pour répondre à trois problématiques (Béhé et al., 2014; Thalmann et al., 2013) :

1. l'enrichissement des connaissances sur l'environnement afin de permettre la création de comportements d'individus plus complexes et plus réalistes ;
2. permettre de sélectionner les entités qui doivent être des agents dans la simulation, puis de les créer à partir de leurs profils dans l'instance de l'ontologie ;
3. autoriser la création semi-automatique d'objets géométriques constituant l'univers simulé.

Afin de faire du modèle HEDGE un UEM, nous proposons d'associer ses composantes structurelles et non-structurelles à une ontologie de l'environnement urbain en utilisant les attributs présentés dans la section 3.2.3.1. L'approche adoptée s'inspire du marquage consistant à placer des balises dans l'environnement afin qu'elles soient perçues par les agents (Lugrin et al., 2007; Yersin et al., 2005). Fondamentalement, ces balises sont des propriétés des composantes de l'environnement faisant référence à des éléments de l'ontologie de l'environnement (concept ou action). Elles peuvent décrire soit la sémantique d'une zone ou d'un objet, soit l'utilisation pouvant être faite d'un objet. En réalisant la liaison entre une description ontologie et une description spatiale de l'environnement, nous pensons que le modèle HEDGE contribue à l'enrichissement des connaissances sur cet environnement. La définition de cette ontologie n'est pas abordée dans cette thèse. Le lecteur intéressé peut se référer à (Durig, 2014) pour plus de détails.

3.6/ CONCLUSION

Ce chapitre présente un modèle d'environnement physique supportant les vues dédiées aux piétons et aux conducteurs de véhicules. Ce chapitre se focalise sur l'architecture, la structure, ainsi que les éléments non-structurels de notre modèle d'environnement physique.

Notre modèle HEDGE (HEterogenous Dual Graph Environment) est basé sur un découpage de l'espace sous la forme de deux aspects distincts d'un graphe représentant les zones interconnectées constituant l'environnement. Le premier aspect permet la navigation. Le second est utilisé pour déterminer les perceptions des agents.

Il permet de simuler l'intérieur de bâtiments, ainsi que des espaces urbains extérieurs par un assemblage de zones de navigation. Les éléments utilisés pour modéliser tous ces types d'environnement sont les mêmes et peuvent par conséquent être connectés sans difficultés.

Ce modèle permet de simuler des entités de différents types évoluant au sein d'une représentation unique de l'environnement. La décomposition de l'environnement selon deux aspects (navigation et perception) permet aux agents de percevoir toutes les entités, quel que soit leur type, tout en leur attribuant des zones de déplacement spécifiques. Cette décomposition permet la cohabitation d'agents de types différents dans le même environnement et simplifie la gestion des zones partagées telles que les passages piétons. Nous avons illustré notre modèle HEDGE par son utilisation pour représenter les parties de l'environnement permettant la navigation des piétons et des véhicules routiers. En considérant les cycles comme des entités hybrides, évoluant telles des piétons sur les trottoirs et telles des voitures sur les routes, leur intégration dans le même contexte de simulation est rendue possible par la structure de notre modèle. En utilisant les nœuds de type couloir, nous pensons qu'il est également aisé de simuler des trains, des tramways et des bateaux. En effet, ces entités se déplacent de manière analogue aux véhicules routiers. Et leur signalétique, bien que différente, peut être gérée de manière similaire.

Le modèle de graphe utilisé par HEDGE ne contraint pas la position d'une entité dans l'environnement. Ainsi, la position spatiale d'une entité n'est pas déterminée à partir de la description topologique d'un nœud du graphe. Cette position est un attribut et l'association de l'entité avec un nœud est déduite de la projection de la position spatiale sur le graphe de l'environnement. Cette approche est opposée à celle utilisée dans nombres

de modèles dans la littérature dans lesquelles les entités suivent la courbe d'une route (Fellendorf et al., 2010; Galland et al., 2014a; Gipps, 1986; Sykes, 2010; Wiedemann, 1974). Notre approche de modélisation permet d'appliquer des calculs issus de modèles physiques des éléments de l'environnement. Ces derniers permettent de calculer les mouvements des objets dans le plan ou dans l'espace en fonction de modèles issus de la physique Newtonienne par exemple.

Dans le chapitre suivant, nous expliquons le comportement dynamique du modèle HEDGE. Nous présentons une méthode pour la détermination de la réaction de l'environnement face aux actions initiées par les agents. Nous proposons un mécanisme de localisation permettant de garantir une cohérence entre la position d'un objet dans l'univers physique et sa position dans le graphe. Enfin, nous proposons une architecture facilitant l'extension de HEDGE afin de réutiliser des modèles existants de processus endogènes.

4

Comportement dynamique du modèle de l'environnement HEDGE

Nous avons présenté les principes et la structure d'un modèle d'environnement ayant pour objectif de permettre de simuler des déplacements de piétons et de véhicules dans un milieu urbain en vue d'analyser les divers scénarii d'aménagement et leurs impacts sur le trafic.

Toutefois, l'environnement possède d'autres caractéristiques liées à sa dynamique et à ses missions dans le contexte du système étudié. Dans le cadre de cette thèse, nous ne proposons pas de définir un modèle général de la dynamique de l'environnement. Nos contributions se concentrent sur : (i) la détermination des réactions de l'environnement aux actions initiées par les agents, (ii) la localisation des entités et (iii) les processus endogènes de l'environnement.

L'ensemble des influences émises par les agents et par les processus endogènes est traité par l'environnement afin d'en déduire l'évolution de son état global (Michel, 2007). Pour gérer les conflits entre ces influences et garantir le respect des lois de l'univers, nous proposons d'utiliser une approche inspirée de la Physique. Elle facilite également la reproduction des mouvements réalistes des objets.

Comme nous l'avons montré dans le chapitre 3, le graphe de HEDGE est une représentation efficace et pertinente de l'environnement. Afin de garantir une cohérence entre la position d'un objet dans l'univers physique et sa position dans le graphe, nous proposons un mécanisme de localisation. Cette localisation s'inscrit dans le cycle de

Sommaire

4.1 Réaction de l'environnement 72
4.1.1 Principe de la réaction inspirée des lois de la Physique 73
4.1.2 Modèles physiques des corps des agents 74
4.2 Localisation des entités 75
4.2.1 Principe de localisation 76
4.2.2 Adaptation du cycle de simulation 78
4.3 Processus endogènes dans le modèle HEDGE 80
4.4 Conclusion 81

simulation entre la modification de l'état global de l'environnement et l'exécution des agents.

La troisième contribution présentée dans ce chapitre concerne les processus endogènes de l'environnement. L'environnement peut avoir ses propres processus indépendant des actions des agents. Nous proposons une architecture facilitant l'extension de HEDGE afin de réutiliser des modèles existants de processus endogènes.

Ce chapitre est organisé de la façon suivante. La section 4.1 présente les principes de réaction inspirée par les lois de la Physique. La section 4.2 contient une description du mécanisme de localisation des entités sur le graphe HEDGE. La section 4.3 décrit l'architecture permettant d'associer des modèles de processus endogènes à HEDGE. Enfin, nous concluons ce chapitre et réalisons un bilan concernant nos contributions dédiées à la dynamique de l'environnement.

4.1/ Réaction de l'environnement : application des lois de la Physique

La réaction est la dernière étape du cycle caractérisant la dynamique de l'environnement (voir section 1.3.3.2). L'ensemble des influences émises par les agents et par les processus endogènes est traité par l'environnement afin d'en déduire l'évolution de son état global. Cependant, certaines influences peuvent être en conflit avec d'autres. Par exemple, deux agents peuvent vouloir traverser une porte au même instant. Un autre problème concerne le respect des lois et des règles régissant l'univers. À titre d'exemple, un piéton ne peut traverser un mur.

Pour gérer les conflits et garantir le respect des lois de l'univers, nous proposons d'utiliser une approche inspirée de la Physique. L'objectif est de modéliser les propriétés et les comportements physiques des objets en utilisant les lois du mouvement de la mécanique newtonienne. Comme l'illustre Bender et al. (2014); Chen (2013); Hoogendoorn et al. (2001); Thiebaux et al. (2008), cette approche de modélisation permet de reproduire de manière réaliste les mouvements des mobiles dans l'espace, notamment des véhicules.

Par exemple, Lamotte et al. (2010) considèrent qu'il est nécessaire de calculer la position et l'orientation d'un véhicule afin de reproduire fidèlement les images captées par une caméra virtuelle embarquée. En effet, ces images sont traitées à l'aide d'algorithmes d'analyse d'image afin de déterminer les obstacles se trouvant devant le véhicule. La qualité de la reconnaissance des obstacles dépend de leurs projections sur le plan de la caméra, et donc indirectement de la position et de l'orientation du véhicule dans l'univers. Il est alors naturel de vouloir reproduire le plus fidèlement possible le comportement dynamique du véhicule et de ses composants pour obtenir le résultat escompté. L'approche inspirée de la Physique présentée dans cette thèse participe à répondre à cette problématique.

Dans la suite de cette section, nous décrivons les principes de l'application des lois de Newton dans le cadre de la réaction de l'environnement. Nous terminons cette section dédiée à la réaction de l'environnement par une brève description de la modélisation physique d'un corps de piéton et d'un véhicule (plus de détails pourront être trouvés dans le chapitre 5).

4.1.1/ Principe de la réaction inspirée des lois de la Physique

La réaction de l'environnement aux actions initiées par les agents est basée sur l'utilisation des lois de la mécanique newtonienne pour la dynamique d'un corps solide.

Ainsi, la quasi-totalité des influences expriment des forces devant être appliquées sur des objets de l'environnement. Les conflits entre ces influences correspondent à des collisions entre les objets qui résulteraient de l'application des forces qu'elles expriment.

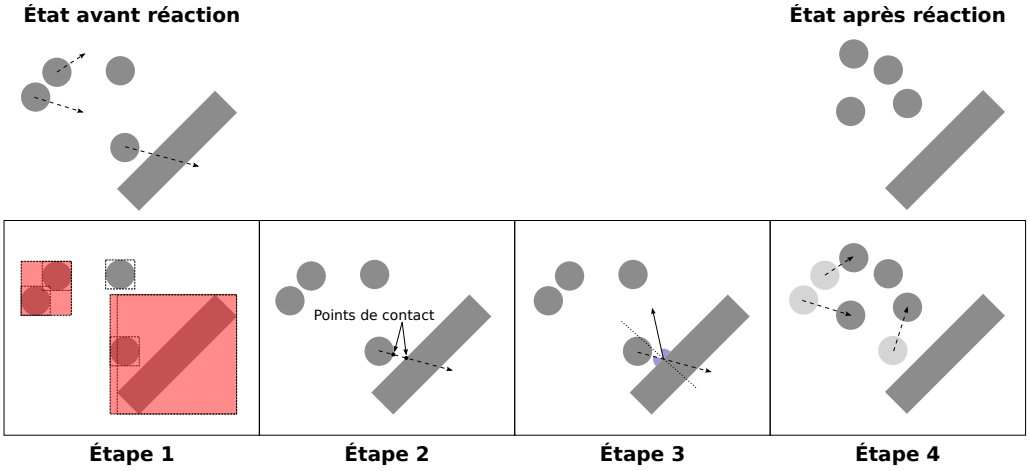

FIGURE 4.1 – Les étapes de la détermination de la réaction de l'environnement basée sur les lois de la Physique.

La détermination de la réaction est basée sur l'utilisation d'algorithmes du domaine de la géométrie algorithmique. Nous nous sommes particulièrement intéressé aux algorithmes permettant de détecter les collisions entre objets en mouvement, puis de calculer les réactions physiques à celles-ci (Bender et al., 2014; Chen, 2013; Van Overveld et al., 1995; Yeh et al., 2007; Yeh, 2007). Le principe de ces algorithmes est basé sur quatre étapes illustrées par la figure 4.1 et décrite ci-dessous :

- **Étape 1 : Détection des objets potentiellement en collision :** consiste à utiliser des volumes englobants approximant les objets pour déterminer les paires de solides possiblement en intersection.
- **Étape 2 : Détermination des points de contact :** permet de déterminer les points de contact entre les paires d'objets en collision.
- **Étape 3 : Calcul des joints de contact :** permet de générer des joints de contacts entre les objets en collision.
- **Étape 4 : Application des forces :** chaque ensemble d'objets en collision est ensuite traité en appliquant les forces sur les différents solides tout en tenant compte des contraintes entre eux. Le moteur calcule les accélérations, les intègre linéairement pour déterminer les nouvelles positions et vitesses des solides. Pour cela, les moteurs de simulation utilisent un solveur de contraintes fonctionnant par relaxation, c'est à dire une méthode itérative d'approximation (Southwell, 1952).

Afin de permettre l'exécution des algorithmes, la définition des objets de l'univers, initialement proposée dans le chapitre précédent, est étendue en ajoutant les attributs suivants :

- P : ensemble des points constituant la forme géométrique de l'objet.
- C : ensemble des équations algébriques représentant les contraintes

géométriques entre les points, telles que la nécessité que la distance entre deux points du même solide soient conservées.

Cette approche de calcul de la réaction de l'environnement aux actions initiées par les agents nécessite la définition d'un modèle physique pour chaque type d'objet. Dans la section suivante, nous décrivons brièvement la définition d'un corps de piéton et d'un véhicule.

4.1.2/ MODÈLES PHYSIQUES DES CORPS DES AGENTS

Une étape importante dans notre approche est la définition d'un modèle physique pour chaque objet dans l'environnement. Afin d'illustrer le potentiel et la facilité (relative) de création de ces modèles physiques, nous décrivons la modélisation d'un corps de piéton et d'un véhicule sans entrer dans les détails de la modélisation mathématique. Le lecteur intéressé peut se référer au chapitre 5 dans lequel nous présentons les modèles de corps de piéton et de véhicule qui ont été utilisés dans des projets de simulation d'aménagements dans la ville de Belfort et sa communauté d'agglomérations.

Modèle de corps de piéton : Le modèle pour le corps d'un piéton est souvent basique. Le corps est considéré comme un solide non-déformable sur lequel les forces peuvent s'appliquer. La figure 4.2 illustre la modélisation d'un corps à l'aide d'un cylindre. Cette modélisation permet de réaliser un lien avec les comportements d'agents utilisant les forces (Gaud, 2007; Helbing et al., 2005; Karamouzas et al., 2009b; Paris et al., 2007b; Reynolds, 1999, 1987; Shao et al., 2007; Van den Berg et al., 2008). En effet, les influences produites par les agents sont des forces qui peuvent être directement intégrées par le modèle de réaction.

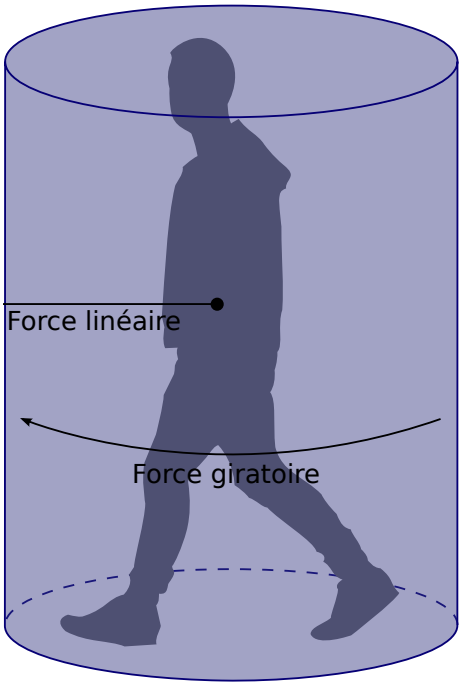

FIGURE 4.2 – Modélisation d'un corps de piéton à l'aide d'un cylindre indéformable.

D'autres modélisations peuvent être utilisées pour répondre aux objectifs de la simulation comme l'animation squelettale pour le rendu cinématographique (Braun et al., 2003; Goto et al., 2001; Seo et al., 2002), ou pour les études médicales ou d'ergonomie (Badler, 1997; Boulic et al., 1995; Honglun et al., 2007; Maurel et al., 1998).

Modèle de véhicule : Le modèle des véhicules est en général plus complexe. De nombreux auteurs le définisse par un ensemble de solides représentant les roues et la carrosserie, reliés par des ressorts modélisant les suspensions [1] (Bender et al., 2014; Chen, 2013; Gechter et al., 2012; Thiebaux et al., 2008). L'influence associée à ce type de modèle est alors constituée par une ou plusieurs forces linéaires permettant de contrôler l'accélération et une ou plusieurs forces angulaires permettant de contrôler la giration du véhicule. Un modèle simple et peu réaliste est présenté dans l'annexe B. La figure 4.3 illustre la modélisation d'un véhicule à l'aide d'un polyèdre. Nous raffinons ce modèle en considérant les composants de la transmission du véhicule afin de reproduire les comportements de roulis du véhicule. Ces comportements ont été nécessaires dans le cadre de nos projets de simulation pour l'aménagement de la ville de Belfort et de sa communauté d'agglomérations présentés dans le chapitre 5.

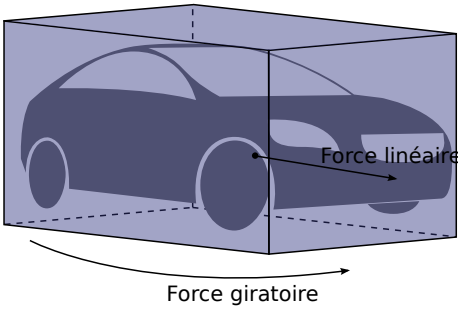

FIGURE 4.3 – Modélisation d'un véhicule à l'aide d'un polyèdre indéformable.

Une fois le mécanisme de réaction défini, et les modèles physiques des objets spécifiés, se pose le problème de la gestion de la cohérence entre la position de l'objet en 2D/3D et sa position dans le graphe HEDGE. Dans la section suivante, nous détaillons notre principe de localisation afin de répondre à ce problème.

4.2/ LOCALISATION DES ENTITÉS

La réaction basée sur la mécanique newtonienne se place dans un espace euclidien 2D ou 3D. Si nous considérons le modèle de graphe présenté dans le chapitre précédent, les informations de position correspondent aux nœuds. Afin de garantir une cohérence entre la position d'un objet dans l'univers physique et sa position dans le graphe, il est nécessaire de définir un mécanisme permettant de localiser les objets dans le graphe à partir de leurs positions 2D/3D.

Dans la suite de cette section, nous présentons le principe de ce mécanisme de localisation. Nous détaillons les choix de modélisation qui nous ont guidés pour réaliser une

1. Les cadriciels de simulation physique, comme NVidia PhysX, intègrent nativement ces modèles de solides et de ressorts.

localisation efficiente. En d'autres termes, le nombre de nœuds explorés dans le graphe pour la détermination de la localisation doit être minimisé, et ainsi permettre une convergence la plus rapide possible vers le ou les nœuds correspondant à la solution.

4.2.1/ Principe de localisation

La localisation des entités sur les nœuds du graphe de l'environnement est une étape essentielle et un prérequis au calcul des perceptions des agents. Cette opération considère deux situations. La première, ou localisation initiale, est utilisée à la création de l'entité ou de manière générale lorsque celle-ci ne possède aucune donnée de localisation. La seconde, ou relocalisation, est utilisée lorsque l'entité est déjà enregistrée sur certains nœuds du graphe.

Le principe de la localisation est donné par l'algorithme suivant :

1: $Localise_{t+\Delta} \leftarrow \emptyset$
2: $Nlocalise_{t+\Delta} \leftarrow \emptyset$
3: **for** $o \in Localise_t \cup Nlocalise_t$ **do**
4: $\quad p \leftarrow Position_{physic}(t, o)$
5: $\quad nodes \leftarrow \{n | n \in Graphe \wedge p \in Shape(n)\}$
6: $\quad$ **if** $nodes \neq \emptyset$ **then**
7: $\quad\quad Position_{graphe}(t, o) \leftarrow Position_{graphe}(t, o) \cup \{nodes\}$
8: $\quad\quad Localise_{t+\Delta} \leftarrow Localise_{t+\Delta} \cup \{o\}$
9: $\quad$ **else**
10: $\quad\quad Position_{graphe}(t, o) \leftarrow \emptyset$
11: $\quad\quad Nlocalise_{t+\Delta} \leftarrow Nlocalise_{t+\Delta} \cup \{o\}$
12: $\quad$ **end if**
13: **end for**
14: **for** $o \in Nlocalise_t$ **do**
15: $\quad$ **if** $o \in \bigcap_{1 \leq \alpha \leq n} Nlocalise_{t-\alpha\Delta}$ **then**
16: $\quad\quad Localise_{t+\Delta} \leftarrow Localise_{t+\Delta} - \{o\}$
17: $\quad\quad Nlocalise_{t+\Delta} \leftarrow Nlocalise_{t+\Delta} - \{o\}$
18: $\quad$ **end if**
19: **end for**

avec :

- $Localise_t$: ensemble des objets situés dans l'environnement à l'instant t.
- $Nlocalise_t$: ensemble des objets ne pouvant pas être localisés dans l'environnement à l'instant t.
- $Graphe$: ensemble des nœuds du graphe de l'environnement décrit dans le chapitre précédent.
- $Shape(n)$: ensemble des positions dans l'espace appartenant à la forme géométrique du nœud n.
- $Position_{graphe}(t, o)$: fonction associant un objet o à un ensemble des nœuds du graphe sur lesquels l'entité o est localisée à l'instant t.
- $Position_{physic}(t, o)$: fonction permettant d'obtenir la position dans l'espace de l'objet o à l'instant t.
- Δ : temps d'un pas de simulation.
- n : nombre maximal de pas de simulation durant lesquels un objet est autorisé à

être relocalisé.

Une entité ne pouvant être localisée dans le graphe de l'environnement ne participe plus au processus de simulation. Lorsqu'il s'agit du corps d'un agent, ce dernier ne peut alors ni percevoir ni être perçu. Il ne peut donc pas agir dans l'environnement et devient un agent inactif jusqu'à ce que son corps puisse être à nouveau localisé ou jusqu'à ce qu'il demande sa propre suppression de la simulation. Ce cas de figure particulier peut se produire lorsqu'un véhicule réalise une sortie de route. Son inertie peut parfois l'amener à retrouver le réseau de nœuds pour y être localisé, dans le cas contraire l'agent et son corps cessent d'exister dans la simulation.

Le mécanisme de localisation doit être capable de déterminer l'ensemble $nodes$ (ligne 5 de l'algorithme). Dans chacun des deux scénarii, l'entité doit s'enregistrer sur tous les nœuds en intersection avec sa représentation géométrique. Par conséquent, il est nécessaire de déterminer efficacement ces nœuds. Cette efficacité dépend de la situation dans laquelle se trouve l'objet considéré. Soit, il s'agit de sa localisation initiale, soit l'objet doit être relocalisé sur le graphe. Nous détaillons ces deux scénarii ci-dessous.

4.2.1.1/ LOCALISATION INITIALE

Dans le cas de la localisation initiale, seule la position 2D/3D de l'entité est connue. Il est donc nécessaire d'explorer les nœuds du graphe à la recherche de ceux possédant une intersection géométrique avec l'entité.

Plutôt que de parcourir de manière linéaire l'ensemble des nœuds, il est préférable d'adopter un algorithme d'exploration de l'espace permettant de converger rapidement vers le bon sous-ensemble de nœuds. En d'autres termes, il est nécessaire de déterminer le plus rapidement possible la zone de l'espace dans laquelle se trouvent le ou les nœuds en intersection avec l'entité.

Ce problème peut être résolu par l'utilisation d'arbres spatiaux de recherche. Ceux-ci ont été proposés pour fournir un compromis entre la complexité de recherche et l'empreinte mémoire de ces structures. Plusieurs approches ont été proposées : ``*Binary Space Partition*´´ (Paterson et al., 1990), ``*Bounding Volume Hierarchy*´´ (Klosowski et al., 1998), ``*kD-tree*´´ (Bentley, 1975), ``*Quadtree*´´ [2] (Samet, 1987), ``*Octree*´´ (Shagam, 2003; Shagam et al., 2003), R-tree (Guttman, 1984), etc. Elles se distinguent principalement par leurs heuristiques de décomposition de l'espace.

À l'instar de Kothuri et al. (2002), nous préconisons l'utilisation d'un Quadtree. Cet arbre spatial fournit de bonnes performances de recherche, et reste simple à implanter. Ce constat a été obtenu en comparant différents types d'arbres généralement utilisés dans le domaine des systèmes d'information géographique : R-tree, Quadtree, kD-Tree (Kothuri et al., 2002).

4.2.1.2/ RELOCALISATION

Lorsque l'entité possède des données de localisation dans le graphe, celles-ci peuvent être utilisées pour rechercher dans leur voisinage les nœuds avec lesquels la géométrie

2. Chaque nœud d'un Quadtree divise l'espace qu'il recouvre en quatre zones de même taille. Chacune de ces zones est alors associée à un nœud fils, etc. Le nœud racine recouvre la totalité de l'univers.

de l'entité pourrait être en intersection après un déplacement.

Pour cela, un parcours local du graphe est nécessaire. Ce parcours utilise une approche similaire à celle employée dans les algorithmes de recherche de plus court chemin (Dechter et al., 1985; Koenig et al., 2005; Stentz, 1995).

En ayant pour hypothèse qu'une entité ne se déplace d'un nœud vers un autre qu'en traversant un lien de navigation, il semble suffisant de ne parcourir que l'aspect navigation du graphe. Cependant cette hypothèse n'est pas toujours vérifiée. Considérons un nœud **A** non borné par un mur physique, c'est à dire dont les murs forment des barrières artificielles comme c'est le cas entre un trottoir et une route. La figure 4.4 illustre la situation.

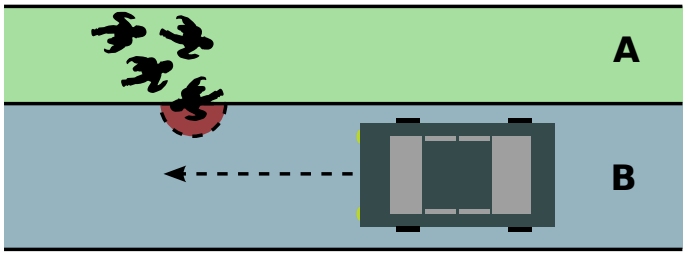

FIGURE 4.4 – Une modélisation incorrecte peut entraîner une localisation incomplète et un risque d'accident.

Si la densité sur **A** est grande, il est possible que les entités s'y trouvant se poussent mutuellement pour essayer de se frayer un chemin [3]. Certaines entités pourraient donc être poussées hors des limites du nœud et entrer en intersection avec un nœud **B** adjacent à **A**. S'il n'existe aucun lien de navigation entre **A** et **B**, la relocalisation telle que décrite ci-dessus sera incomplète. Le nœud **B** ne sera jamais présent dans les données de localisation des entités. Par conséquent, les entités présentes sur **B** ne seront pas capables de percevoir les entités de **A** et inversement. Cependant celle-ci peuvent représenter des obstacles à la navigation des entités de **B**.

On peut donc voir que pour régler cette situation, il convient de parcourir le graphe de perception. L'aspect perception enrichie l'aspect navigation, comme nous l'avons montré dans le chapitre 3. Par conséquent, en parcourant le graphe de perception nous avons l'assurance de parcourir au minimum le graphe de navigation. Il faut également s'assurer que tous les nœuds géométriquement adjacents sont liés au minimum par un lien de perception bidirectionnel.

La localisation doit être intégrée au cycle d'exécution d'une simulation. À cette fin, nous proposons d'adapter le cycle de simulation.

4.2.2/ ADAPTATION DU CYCLE DE SIMULATION

La dynamique de l'environnement s'inscrit dans un cycle de simulation décrit dans l'état de l'art, section 1.3.3.2. Les phases constituant ce cycle sont :

i) le calcul de la perception de chaque agent ;

3. Ce problème est reconnu pour les algorithmes de calcul de déplacements basés sur des forces sociales Helbing et al. (2005); Van den Berg et al. (2008).

ii) l'exécution des agents, générant des influences [4] à partir de leurs perceptions ;

iii) l'exécution des processus endogènes de l'environnement ;

iv) la collecte des influences générées par les agents et par les processus endogènes ;

v) la détection et la résolution de conflits entre les influences ;

vi) la réaction [5] de l'environnement produisant un changement de son état global.

Les cycles de simulation proposés par Ferber et al. (1996), Weyns et al. (2004a) et Galland et al. (2009) ne considèrent pas la localisation comme une étape à part entière du cycle. En effet, ils considèrent que la structure de l'environnement (représentée par un graphe) permet de déterminer naturellement la position dans le plan ou l'espace des entités. Comme nous l'avons indiqué ci-dessus, nous pensons qu'il s'agit d'une contrainte rendant difficile l'intégration de la modélisation et de la simulation des mouvements des objets selon des lois de la Physique.

Pour répondre à cette problématique particulière, nous avons adapté le cycle de simulation pour le modèle HEDGE (illustré par la figure 4.5) en introduisant une phase de localisation. Il s'agit d'un mécanisme permettant :

— de déterminer la position d'un corps relativement au graphe de navigation afin que l'agent puissent se repérer et déterminer son chemin ;
— d'enregistrer une entité sur chacun des nœuds avec lesquels elle est localisée afin qu'elle puisse être perçue par les agents environnants.

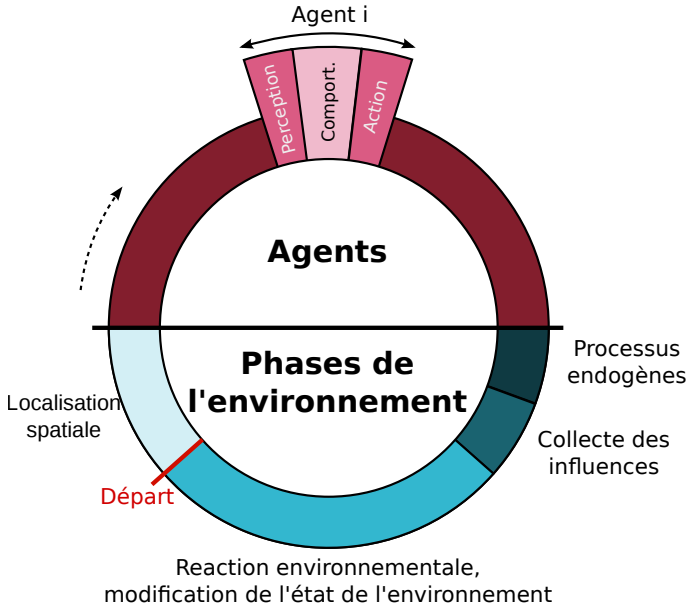

FIGURE 4.5 – Cycle d'exécution de notre modèle durant une simulation

Comme illustré par la figure 4.5, le point de départ de la simulation est l'étape de localisation. Cette étape est globale et doit nécessairement être contrôlée par l'environnement plutôt que par un processus interne aux agents, qui, rappelons-le, ne peuvent à aucun

4. Influence : description d'un changement de l'état de l'environnement désiré par l'agent, voir la section 1.3.4.4, page 24, pour plus de détails sur ce concept.
5. Réaction : description d'un changement de l'état de l'environnement en réponse à des influences, voir la section 1.3.4.4, page 24, pour plus de détails sur ce concept.

moment modifier directement l'état de l'environnement (voir la contrainte d'intégrité environnementale dans la section 1.3.1).

Dans la section suivante, nous présentons notre contribution liée aux processus endogènes de l'environnement. Nous proposons une architecture permettant d'associer des modèles de processus endogènes à HEDGE. Cette proposition permet l'extension de notre modèle à d'autres applications que celles traitées dans ce mémoire.

4.3/ Processus endogènes dans le modèle HEDGE

En dehors des activités des agents, l'environnement peut également produire une activité (Weyns et al., 2005b). Un exemple typique est l'évaporation des phéromones artificielles dans les algorithmes à base de colonies de fourmis, ou bien une balle roulant sur un plan incliné sous l'effet de la gravité.

Selon Helleboogh et al. (2007) une activité endogène est une évolution d'un constituent particulier de l'environnement. Celle-ci dépend de trois éléments : (i) le constituent impliqué, (ii) la stratégie d'évolution, (iii) le laps de temps accordé pour faire évoluer le constituent.

Dans le cadre des applications visées par nos travaux, il existe des dynamiques de l'environnement qui ne doivent pas être contrôlées par des agents. Par exemple, la stratégie de contrôle de feux tricolores peut être modélisée par un processus endogène. Un autre exemple est l'utilisation d'un modèle de simulation hydrologique afin de simuler l'impact d'inondations sur un réseau routier.

Dans le cadre de cette thèse, nous ne proposons ni un nouveau modèle, ni un modèle générique de processus endogène. Un état de l'art concernant les modèles de dynamique endogène de l'environnement montre une très grande diversité de propositions, incluant les modèles de phénomènes météorologiques ou climatiques (Berger et al., 2007; Landsberg, 1981; Oprea, 2012; Tomás et al., 2005), et hydrologiques (O'Brien et al., 1993). D'autres modèles utilisant la mécanique classique pour simuler les mouvements des objets dans l'espace ont également été présentés (Bender et al., 2014; Chen, 2013; Van Overveld et al., 1995; Yeh et al., 2007; Yeh, 2007). De plus, il existe des bibliothèques logicielles « clé en main » proposant un solveur des équations de Newton et dédiées aux environnements virtuels, comme par exemple NVidia PhysX® [6] ou ODE [7].

Au vue de la diversité des propositions, il nous parait préférable de privilégier l'association de HEDGE avec ces modèles, plutôt que d'en proposer un nouveau. Ce point de vue est partagé par Robinson et al. (2004) qui indique que « la réutilisation de modèles de simulation est une approche intéressante. Intuitivement, cette réutilisation permet de réduire les coûts de conception de modèles de simulation et de développement de simulateurs ».

Afin de faciliter cette association d'un modèle de processus endogène avec HEDGE, nous faisons les deux hypothèses suivantes :

i) Un processus endogène est capable d'extraire de l'information à partir de l'état de l'environnement. Cette information est représentée par un ensemble de percepts.

ii) Un processus endogène génère un ensemble d'influences qui sont collectées par

6. http://www.nvidia.com/object/physx_new.html
7. http://www.ode.org

le modèle d'environnement et traitées durant la phase de réaction, au même titre que les influences générées par les agents.

La figure 4.6 illustre les relations entre les processus endogènes et les autres éléments du modèle HEDGE.

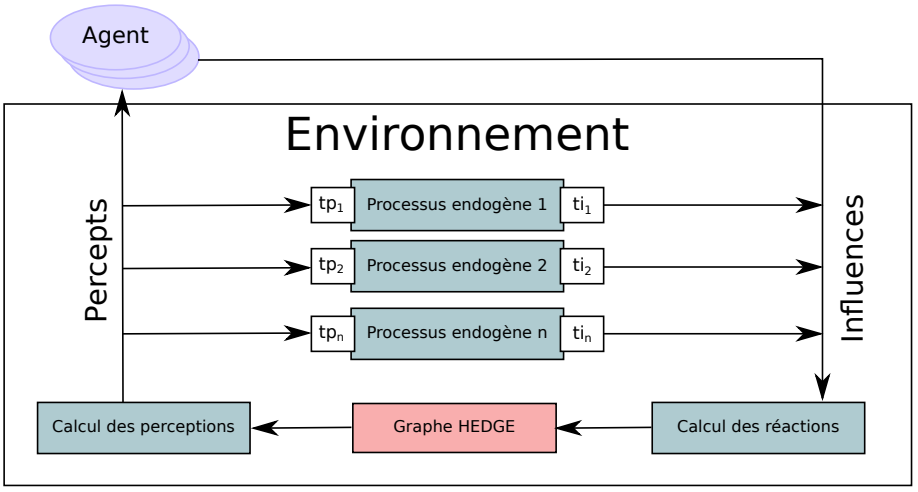

FIGURE 4.6 – Les processus endogènes et leurs relations avec les autres éléments de HEDGE

L'utilisation des percepts et des influences entre le modèle HEDGE et un processus endogène est un moyen efficace de permettre leur association. Pour chaque modèle de processus endogène, il est nécessaire de définir les transformations d'informations qui permettront de respecter ces deux hypothèses. Ces transformations sont illustrées sur la figure 4.6 par les blocs tp_n pour les transformations liées à la perception, et ti_n pour les transformations liées aux influences.

Dans les applications que nous présentons dans le chapitre 5, nous utilisons un processus endogène pour modéliser les contrôleurs de feux de signalisation dans les zones urbaines simulées. Ce modèle endogène perçoit l'environnement à l'aide des capteurs modélisant des boucles magnétiques. Il émet des influences pour changer les états des feux associés au contrôleur de trafic.

4.4/ CONCLUSION

L'environnement est un élément actif dans une simulation multi-agent (Weyns et al., 2005a). Il peut avoir ses propres processus indépendamment des activités des agents. Dans ce chapitre, nous avons présenté nos contributions relatives à la dynamique l'environnement.

Notre première contribution concerne la gestion des actions simultanées et conjointes des agents. Afin de préserver l'intégrité de l'environnement, nous proposons d'appliquer le modèle influence-réaction (Michel, 2007) qui fournit un cadre d'interaction entre les agents et l'environnement. Collecter et gérer les influences générées par les agents impose la définition d'un modèle de réaction de l'environnement. Nous proposons un modèle inspiré par les lois du mouvement de Newton dans lequel les influences expriment

des forces à appliquer aux objets de l'environnement et la réaction est le résultat de la résolution des équations physiques.

Pour cela, nous proposons d'introduire une phase de localisation des entités dans le cycle de simulation. Cette phase permet de localiser les objets dans le graphe à partir de leurs positions 2D/3D afin de garantir la cohérence entre les positions des objets dans le modèle physique et leurs positions dans le graphe HEDGE.

Enfin, nous décrivons une architecture permettant l'association de modèles de processus endogènes existants avec le modèle HEDGE.

Afin d'illustrer nos travaux sur une application concrète, nous présentons deux projets d'aménagement de la ville de Belfort et de sa communauté d'agglomérations dans le chapitre suivant. Les qualités et les défauts de notre approche ont été mis en évidence dans le cadre de cette application.

5

Simulations pour l'aménagement de la ville de Belfort

Dans ce chapitre, nous illustrons l'utilisation du modèle HEDGE dans le cadre de deux projets d'aménagement de la ville de Belfort et de sa communauté d'agglomérations. Ils nous permettent de valider le modèle HEDGE et d'illustrer son applicabilité sur des projets industriels réels.

Le premier projet concerne la mise en place d'un réseau de bus à haut niveau de service. Ce projet participe à une réflexion globale menée sur la planification des transports publics pour inciter plus de personnes à utiliser les transports en commun. HEDGE est utilisé pour modéliser les zones névralgiques à un niveau microscopique. Il permet de modéliser efficacement et rapidement l'infrastructure routière. Les simulations réalisées avec ce modèle ont mis en évidence des problèmes structurels et des problèmes fonctionnels.

Le second projet concerne l'aménagement de la Place d'Armes de Belfort afin de renfor-

Sommaire

5.1 Logiciel Voxelia Simulate . . . 84
5.1.1 Architecture du logiciel Voxelia Simulate . . . 84
5.1.2 Éditeur et Générateur de graphe HEDGE . . . 85
5.2 Contexte et objectifs des études . . . 86
5.2.1 Contexte général . . . 87
5.2.2 Contexte des projets de simulation . . . 87
5.3 Définition du comportement des agents . . . 88
5.3.1 Comportement des piétons et son extension aux cyclistes . . . 88
5.3.2 Comportement des conducteurs de véhicules routiers . . . 91
5.4 Définition du modèle physique des véhicules . . . 94
5.4.1 Architecture du modèle de véhicule . . . 94
5.4.2 Modèles sélectionnés pour les composants d'un véhicule . . . 95
5.5 Validation de HEDGE . . . 98
5.5.1 La place Rabin . . . 98
5.5.2 Le quartier Denfert-Thiers . . . 102
5.5.3 La Place d'Armes . . . 104
5.6 Conclusion . . . 107

cer l'attractivité commerciale et touristique du centre historique de la ville. La valorisation des déplacements des piétons et la mise en valeurs des grands équipements historiques sont au centre de ce projet. HEDGE a permis de donner vie à un projet tout en respectant son essence architecturale. Le produit de l'application de ce modèle est utilisé à des fins d'études techniques (études préliminaires, avant-projet), de communication (concertations et débats publics) et de promotion (expositions, valorisation) sous la forme de logiciels interactifs en 3D.

Ce chapitre est structuré comme suit. Dans la section suivante, nous présentons le logiciel Voxelia Simulate qui intègre une implantation du modèle HEDGE. Il a été utilisé pour réaliser les projets d'applications mentionnés ci-dessus. Dans la section 5.2, nous présentons le contexte et les objectifs de ces deux études. Les modèles de comportement des agents sont détaillés dans la section 5.3. Le modèle physique des véhicules est détaillé dans la section 5.4. La validation du modèle HEDGE est décrite à travers la modélisation de trois zones névralgiques de la ville de Belfort dans la section 5.5. Enfin, nous concluons ce chapitre en réalisant un bilan de l'application du modèle HEDGE dans le cadre des deux projets décrits dans ce chapitre.

5.1/ Logiciel Voxelia Simulate

Le logiciel **Voxelia Simulate** [1] permet de reproduire le trafic dans un environnement 3D aussi fidèle à la réalité que techniquement possible. Les aspects de visualisation et de simulation physique constituent les principaux apports de ce nouveau simulateur. Par exemple, un arbre qui obstrue la vue, une ligne de stop placée trop loin de l'intersection, une voie d'insertion qui mérite une signalisation, une signalisation horizontale inadéquate ou accidentogène sont autant de problèmes pouvant être mis en lumière à l'aide de Simulate. En effet, il est admis que les caractéristiques spatiales et géométriques des éléments d'une ville jouent un rôle essentiel dans la sécurité, l'efficacité et la régularité du trafic ; et ceci quel que soit la sophistication des dispositifs de signalisation. Le résultat final d'une simulation avec Simulate est la visualisation en 3D temps réel des trajectoires réalistes pour chacune des entités simulées dans des applications interactives.

Dans le cadre de cette thèse, nous avons enrichie les fonctionnalités de Voxelia Simulate en intégrant une implantation du modèle HEDGE. Nous proposons également un éditeur de graphe HEDGE afin de faciliter la construction du modèle de simulation.

5.1.1/ Architecture du logiciel Voxelia Simulate

Le logiciel Voxelia Simulate est structuré en modules illustrés par la figure 5.1 et décrits comme suit :

- **Maquette 3D :** contient la représentation géométrique et graphique 3D de l'univers simulé.
- **Graphe HEDGE :** est l'implantation du graphe HEDGE présenté dans le chapitre 3.
- **Comportements d'agents :** contient la définition des comportements des agents.
- **Corps d'agents :** fournit la définition des modules logiciels représentant les corps des agents.

1. http://www.voxelia.com

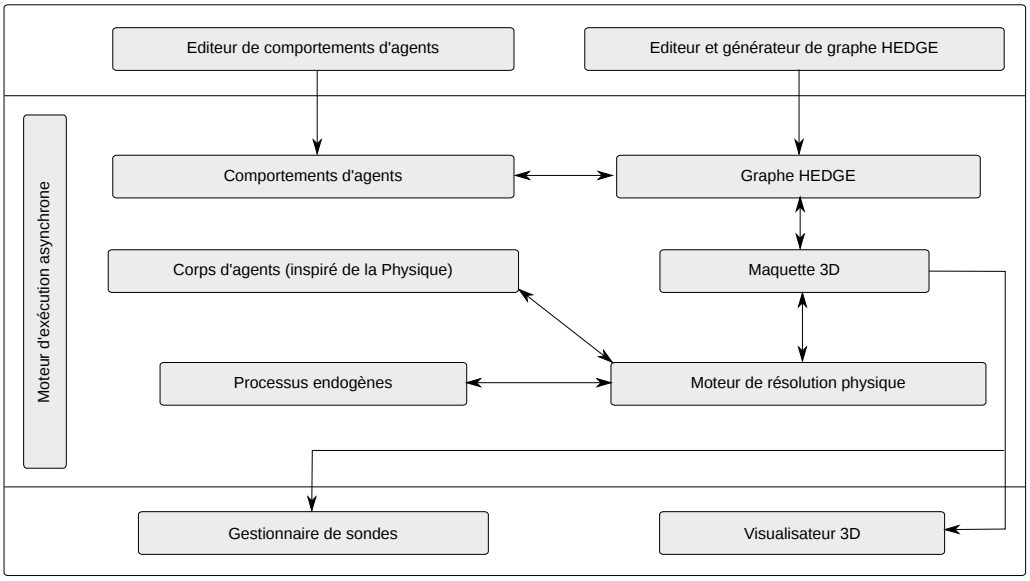

FIGURE 5.1 – Architecture des modules composant le logiciel Voxelia Simulate

— **Processus endogènes :** contiennent les implantations des processus endogènes de l'environnement.
— **Moteur de résolution physique :** fournit une implantation du moteur de calcul de la réaction de l'environnement. Dans Voxelia Simulate, ce module utilise la bibliothèque de simulation physique NVidia PhysX.
— **Moteur d'exécution asynchrone :** fournit les outils d'exécution asynchrone (basés sur des « threads ») utilisés par les autres modules de Voxelia Simulate.
— **Gestionnaire de sondes :** fournit les outils logiciels pour extraire de l'information de l'état de la maquette 3D. Ce gestionnaire peut également être utilisé pour collecter les valeurs associées aux états des agents.
— **Visualisateur 3D :** génère en temps réel des images de haute qualité graphique à partir de la maquette 3D. Ce modèle permet également d'intégrer l'utilisateur humain en lui fournissant les outils d'interaction nécessaire pour modifier son point de vue sur l'univers.
— **Éditeur de comportements d'agents :** permet de spécifier et décrire les comportements des agents à l'aide d'un langage de programmation (C#).
— **Éditeur et générateur de graphe HEDGE :** permet de créer une instance du graphe HEDGE à partir de la maquette 3D. Cet éditeur est décrit dans la section suivante.

Dans la section suivante, nous présentons le modèle dédié à l'édition et la génération du graphe HEDGE utilisé dans le cadre des cas d'application que nous présentons dans ce chapitre.

5.1.2/ ÉDITEUR ET GÉNÉRATEUR DE GRAPHE HEDGE

L'outil de génération a été développé de manière à profiter des données mises à disposition par les utilisateurs. En effet, la plupart de ceux-ci disposent de maquette 3D des villes

qui peuvent donc être utilisées comme base à la génération du graphe. Ces outils permettent de réaliser des études en utilisant la simulation sur des maquettes 3D existantes tout en minimisant les coûts d'intégration. Ces données ne permettent pas d'automatiser totalement la création du modèle qui doit donc faire l'objet d'interventions manuelles afin d'obtenir un graphe HEDGE.

FIGURE 5.2 – Processus simplifié d'édition d'un graphe HEDGE

La génération du graphe HEDGE s'appuie sur un nombre limité de données numériques externes comme l'illustre la figure 5.2. La saisie est réalisée sur une maquette 3D correspondant au terrain et aux bâtiments de l'environnement étudié. La construction de cette maquette 3D est réalisée par un ensemble d'outils externes (3DS-Max, Unity3D, etc.) à partir de sources diverses : plan, vue aérienne, etc.

La création du modèle dans Voxelia Simulate s'appuie sur des données hétérogènes et pouvant être difficiles d'accès comme des plans d'aménagement, des vues aériennes de la zone avant travaux, des modèles numériques de terrain, et d'autres données issues des systèmes d'information géographique.

À partir de ces données, l'utilisateur trace le graphe de l'environnement depuis une interface graphique que nous avons développé et intégré à Voxelia Simulate.

Cet outil permettant de tracer les zones de navigation est inspiré d'une part des logiciels de dessin vectoriels (pour l'édition de splines, la modification des tangentes, le tracé d'un polygone, etc.), d'autre part des logiciels de simulation de trafic existants pour la connexion des tronçons de routes et le placement de la signalétique. L'outil développé permet de générer un graphe dans un temps très court et de s'assurer que l'utilisateur n'introduit aucune erreur de modélisation.

Voxelia Simulate est un logiciel utilisé pour la simulation des déplacements de piétons et de véhicules dans un milieu urbain en vue d'analyser les divers scénarii d'aménagement et leurs impacts sur le trafic. Dans la section suivante, nous présentons le contexte de deux études de l'aménagement de la ville de Belfort qui ont permis de valider le modèle HEDGE et son implantation dans Voxelia Simulate.

5.2/ CONTEXTE ET OBJECTIFS DES ÉTUDES

Dans cette section, nous présentons le contexte et les objectifs du projet de simulation du système de transport en commun de la ville de Belfort : Optymo [2].

2. http://www.optymo.fr

5.2.1/ CONTEXTE GÉNÉRAL

Optymo est un réseau de bus à haut niveau de service, lancé en 2008 suivant deux phases, Optymo 1 et 2. Ce lancement intervient dans le cadre d'une réflexion globale menée en France sur la planification des transports publics. Le déploiement de ce réseau s'est accompagné d'aménagements et de mesures visant à métamorphoser la ville. Un effort a été porté sur l'intégration du système de transport du point de vue architectural pour son mobilier et son équipement, ainsi que du point de vue socio-économique par rapport aux habitudes des citoyens et de l'activité. Ces mutations ont également contribué à optimiser le transfert modal, et ainsi inciter plus de personnes à délaisser l'automobile au profit du bus.

La Phase Une (2008) concerne la mise en place du post-paiement, l'équipement en bus en GPL et la rénovation du réseau qui n'avait pas évolué depuis 1979, malgré les différentes évolutions connues par la Ville. Le concept s'articule autour du cadencement de lignes à 10 minutes, ainsi que la consolidation des services autour des axes à forte fréquentation.

La Phase Deux (2013) implique la mise en place de services à la demande, la commande de billet par SMS, la mise en place de vélos en libre-service et le renforcement du cadencement des lignes structurantes à 5 minutes. Des perspectives concernant la Phase Trois sont annoncées, et dont l'objectif est de généraliser le cadencement des lignes à 5 minutes. Le réseau est décomposé en deux parties totalement distinctes sur les aspects exploitation et fonctionnement : le réseau urbain de 5 lignes, cadencé de 5 à 10 minutes, et le réseau suburbain, composé de 10 lignes, aux horaires fixés et renforcé par un service à la demande.

L'accès aux personnes à mobilité réduite a également été un des aspects repensés sur la Phase Deux, en aménageant des quais hauts sur le centre-ville et en développant le service de transport à la demande.

5.2.2/ CONTEXTE DES PROJETS DE SIMULATION

Ces réflexions sur la planification ont nécessité le déploiement de nouveaux outils d'aide à la décision basés sur des données numériques recueillies par la régie du réseau. Une des missions essentielles des exploitants de réseau est de garantir une vitesse commerciale qui soit la plus régulière possible, quel que soit les aléas que rencontreront les matériels roulants.

Une des premières pistes de réflexion envisagée a été de réduire ces aléas, en s'assurant par exemple que le bus circule sur un couloir de site propre ou en installant des priorités aux feux tricolores. Cette approche a des limites, les kilomètres de sites propres dédiés au bus sont autant de kilomètres retirés à la voiture.

La seconde piste de réflexion s'est située dans l'étude et la prévision des temps de parcours de bus, en cherchant des motifs dans les services et la régulation effectuée. À ce titre, la simulation de trafic, à l'échelle sub-microscopique, microscopique, macroscopique, fédère les vues métier en partageant la connaissance et permet un débat autour de points précis.

Trois modèles de simulation ont été construits à l'échelle de la ville de Belfort, à titre d'étude. Il s'agissait sur le plan décisionnel de disposer des meilleurs indicateurs pos-

sibles sans se soucier du type de données et sans arbitrage d'échelle, et sur le plan scientifique de bien appréhender les apports de chaque modèle à la problématique courante.

Les modèles utilisés dans le cadre de ce projet sont les suivants :

— **MOSDAU (modèle macroscopique stratégique de déplacement de l'aire urbaine) :** développé dans un premier temps pour étudier la desserte de la nouvelle gare TGV de Meroux Moval, il a été précisé et recentré sur l'agglomération belfortaine pour le projet Optymo et réalisé avec PTV Davisum. Il s'agit du modèle classique de prévision de transport en quatre étapes (O'Flaherty, 1986) : génération du trafic, distribution, choix modal et affectation.

— **MISDAU (modèle microscopique stratégique de déplacement de l'aire urbaine) :** développé sur le centre-ville de Belfort, il a été réalisé avec PTV Vissim. Son objectif est de valider les nouveaux aménagements prévus dans le cadre du projet. Ce modèle intègre 50 plans de feux adaptatifs et demeure plus précis dans la géométrie que le modèle précédent.

— **Optymo Simulate :** conçu sur la plate-forme Voxelia Simulate incluant le modèle HEDGE, une dizaine de carrefours ont été modélisés finement en prenant en compte l'influence d'aménagements en dehors du cadre Optymo. Il a pour objectif de servir de support de communication et de valider les aménagements en considérant les contraintes topologiques et géométriques.

Le deux premiers modèles sont détaillés dans l'annexe C, le troisième est détaillé dans la suite de ce chapitre.

Dans les sections suivantes nous détaillons les comportements des agents que nous utilisons dans Voxelia Simulate. Les scénarii mettent en jeu des piétons, parfois des cyclistes, et des véhicules routiers divisés en trois catégories : voitures particulières, bus et poids lourds. Nous détaillons le modèle physique des véhicules utilisé dans le cadre de ces projets. Nous présentons ensuite quelques scénarii de simulation représentatifs du projet Optymo : la place Rabin et le quartier Denfert-Thiers. Enfin, nous terminons par une présentation d'un scénario de simulation sur la Place d'Armes.

5.3/ Définition du comportement des agents

La réalisation des études présentées précédemment avec Voxelia Simulate nécessite la définition du comportement des agents peuplant l'univers simulé. Nous considérons trois types de comportements de déplacement : les piétons, les cyclistes, les conducteurs de véhicules.

5.3.1/ Comportement des piétons et son extension aux cyclistes

Dans notre cadre d'application, les piétons doivent se déplacer « naturellement » dans l'environnement urbain. L'impact de leurs déplacements sur le comportement des conducteurs de véhicules, et donc sur le trafic, est au cœur de nos expérimentations.

Nous utilisons le même comportement pour les piétons et les cyclistes. Nos expérimentations dans (Buisson et al., 2013) montre que la principale différence entre ces deux types d'agents est leur vitesse de croisière, leur accélération et leur décélération

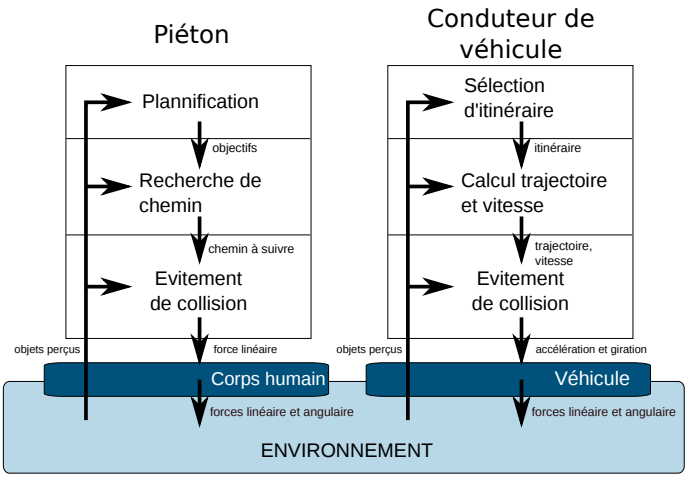

FIGURE 5.3 – Architecture en couches des agents piétons et conducteurs de véhicule

préférées. Ainsi, il est possible d'utiliser le même algorithme (basé sur des forces de contournement) pour reproduire les déplacements de ces deux types de comportements.

Le comportement d'un agent piéton est basé sur une architecture en couche présentée par la figure 5.3. Les trois couches fonctionnelles sont décrites dans les sections suivantes.

5.3.1.1/ PLANIFICATION ET DÉTERMINATION DES OBJECTIFS

La première couche d'un agent piéton a pour fonction de déterminer les tâches à effectuer à court terme en se basant sur ses objectifs. Ces tâches sont représentées par une position dans la scène où le piéton doit effectuer une action ou une série d'actions.

Dans le cadre de notre projet, les activités (au sens des modèles d'activité (Bellemans et al., 2010)) des piétons sont prédéterminées et structurées au sein d'une séquence. Les piétons devant avoir une activité de déplacement durant la période de simulation sont créés dans l'univers simulé. La couche de planification que nous avons implémenté est très simple et consiste à construire dynamiquement un chemin, définit par une collection de nœuds consécutifs que le piéton doit traverser. Cette approche de construction dynamique peut être rapprochée des algorithmes de recherche dynamique de plus court chemin (Dechter et al., 1985; Stentz, 1994, 1995), ou de calcul d'itinéraire dans un réseau routier (Knapen et al., 2014).

Le principe général de ce calcul de chemin peut être décrit comme suit : le chemin est calculé pour avoir une longueur minimale prédéfinie. A l'initialisation, nous ajoutons l'un des nœuds sur lequel l'agent est situé comme point de départ du chemin puis nous ajoutons aléatoirement les nœuds suivants en parcourant le graphe de navigation jusqu'à ce que le chemin atteigne la longueur minimale choisie. Lorsque le piéton a traversé le premier nœud du chemin, celui-ci est supprimé et d'autres sont ajoutés aléatoirement à l'extrémité jusqu'à ce que la longueur du chemin soit supérieure ou égale à la limite que nous avons fixé.

5.3.1.2/ Recherche d'un chemin dans l'environnement

La seconde couche d'un agent piéton a pour fonction de déterminer la cible virtuelle (une position dans l'espace) vers laquelle le piéton doit se déplacer. Cette cible est choisie de manière à ce que le piéton puisse traverser un nœud, s'arrêter aux feux de signalisation lorsqu'ils sont rouges ou bien lorsque le piéton doit traverser une route mais qu'un véhicule en approche l'en empêche.

Par exemple, la couche de recherche de chemin est le composant du comportement qui prend la décision de traverser une route en fonction du chemin construit par la couche planification.

Pour traverser un nœud, la cible P_c est obtenue en projetant la position du piéton à une distance donnée à l'avant de la trajectoire sur l'axe central du nœud s'il s'agit d'un couloir ou sur le segment reliant le connecteur d'entrée au connecteur de sortie s'il s'agit d'un nœud de type place ou jonction. Les figures 5.4 et 5.5 illustrent respectivement ces deux cas. Nous décalons latéralement la cible d'une faible distance aléatoire d pour que chaque piéton ait une trajectoire différente en l'absence d'obstacles. Cette approche est similaire au comportement de suivit de chemin décrit par Reynolds (1999).

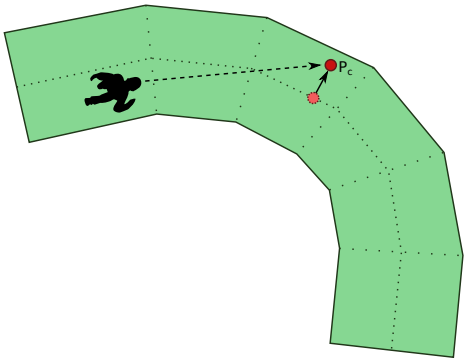

FIGURE 5.4 – Gestion de la trajectoire d'un piéton dans un couloir.

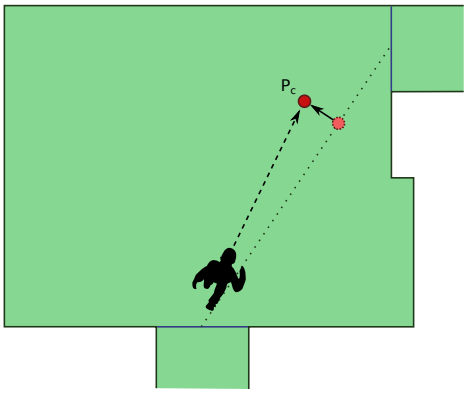

FIGURE 5.5 – Gestion de la trajectoire d'un piéton dans une place.

Lorsque le piéton doit s'arrêter à un feu de signalisation, nous plaçons la cible sur le point le plus proche du piéton appartenant à la géométrie de l'objet signalétique. La procédure est similaire dans le cas où le piéton doit patienter sur le bord de la route avant de traverser.

5.3.1.3/ Évitement de collision

La couche d'évitement de collision a pour objectif de déterminer la direction et l'accélération avec laquelle le piéton doit se déplacer.

Celle que nous utilisons pour les piétons et cyclistes dans le cadre de ce projet implante un modèle basé sur les champs de forces (Buisson et al., 2013).

Le modèle d'évitement de collision est basé sur l'hypothèse qu'un individu essayera toujours d'éviter un obstacle en minimisant l'énergie requise pour résoudre le conflit (principe du moindre effort). Minimiser l'énergie requiert un changement de direction aussi

progressif que possible et doit par conséquent être initié le plus tôt possible tout en continuant à avancer vers l'objectif. Le modèle d'évitement de collision utilise un type de force particulier appelé force de glissement ou ``*sliding force*´´. Contrairement aux forces de répulsion classiquement employées par les modèles basés sur les champs de forces les plus répandus, la force de glissement n'a pas seulement pour objectif de dévier l'individu de la direction de l'obstacle mais aussi de le guider vers son objectif. Le modèle d'évitement utilise une fonction monotone décroissante du temps pour ajuster la magnitude de la force d'évitement. Le temps estimé avant impact avec un obstacle est utilisé comme paramètre d'une fonction représentant la réponse de l'individu face à l'urgence d'une collision. Plus le temps avant impact est grand, moins la force d'évitement sera importante. Un ajustement des forces basé sur le temps permet de donner une priorité plus grande aux forces qui permettent d'éviter les collisions les plus immédiates.

Les détails de ce modèle sont donnés dans l'annexe C.3.

Dans la section suivante, nous présentons le comportement des conducteurs de véhicules routiers utilisé dans le cadre des cas d'application que nous présentons dans ce chapitre.

5.3.2/ COMPORTEMENT DES CONDUCTEURS DE VÉHICULES ROUTIERS

Le comportement d'un conducteur de véhicule routier est basé sur l'architecture en couche présentée dans la figure 5.3 page 89. L'articulation entre les composants de ces couches est illustrée par la figure 5.6 et nous les décrivons dans les sections suivantes.

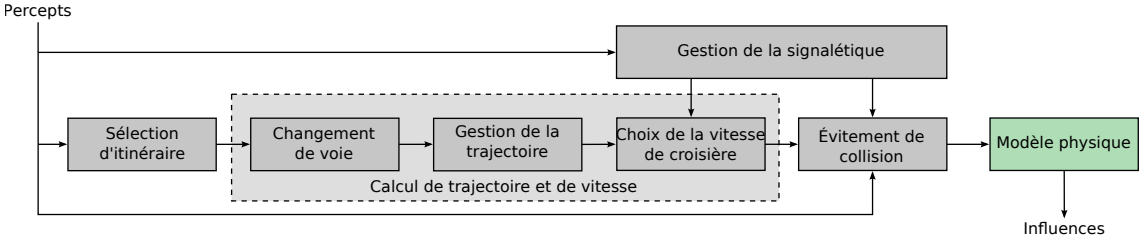

FIGURE 5.6 – Architecture du comportement des conducteurs de véhicules.

5.3.2.1/ SÉLECTION D'ITINÉRAIRE

Dans cette couche de son comportement, l'agent choisit l'itinéraire à emprunter pour atteindre ses objectifs. A l'instar des comportements de piétons, nous utilisons une séquence d'activités dans laquelle les activités de déplacement seront considérées par notre simulateur. Les itinéraires sont pré-calculés et fournis aux agents suivant une distribution stochastique découlant directement des résultats de l'exécution des simulations macroscopique (MOSDAU) et microscopique (MISDAU).

5.3.2.2/ CALCUL DE TRAJECTOIRE ET DE VITESSE

Le choix de la vitesse de croisière, de la direction à court terme du véhicule (permettant d'orienter les roues) et de l'obstacle le plus proche à prendre en compte (qu'il s'agisse

d'un véhicule ou d'un panneau de signalisation) doit être réalisé le long de l'itinéraire sélectionné.

Choix de la vitesse de croisière : Le choix de la vitesse de croisière effective est obtenu en modulant la vitesse maximale autorisée imposée par la signalétique. La couche de calcul de trajectoire et de vitesse doit donc parcourir les percepts pour détecter le moment où l'agent passe un panneau de signalisation informant un changement de vitesse maximale autorisée. Lorsque le changement de vitesse maximale est effectif, un nombre dans un intervalle paramétrable autour de cette limite est généré aléatoirement et devient la vitesse de croisière préférée de l'agent. L'intervalle est définit en pourcentage autour de la vitesse maximale. Par exemple, nous pouvons définir qu'un conducteur normal choisira une vitesse de croisière entre 90% et 100% de la vitesse maximale autorisée. Pour modéliser des conducteurs à comportement à risque (dépassant systématiquement les limitations de vitesse) nous choisissons un intervalle entre 100% et 110% de la vitesse maximale.

Gestion de la trajectoire : La couche de calcul de trajectoire et de vitesse doit également choisir la direction à court terme du véhicule. Pour ce faire, le point P_c projeté sur l'axe central des nœuds *couloirs* le long de la route à l'avant du véhicule à une distance donnée est utilisé comme cible (Reynolds, 1999). Cette méthode permet à un véhicule d'anticiper une courbe et de ne pas déraper vers l'extérieur d'un virage tant que la vitesse du véhicule est raisonnable. Nous utilisons une distance comprise entre 5 à 10 mètres à l'avant du véhicule dans toutes nos simulations.

Afin que le véhicule ne dérape pas lors d'un virage très serré, il est nécessaire de réduire la vitesse de croisière choisit par le conducteur. Pour cela, nous projetons un point P_t sur l'axe central des couloirs, de la même manière que pour la détermination de la direction à une distance beaucoup plus grande. Nous réduisons ensuite la vitesse en fonction du cosinus de l'angle α entre la direction courante du véhicule $\hat{z}$ et la direction vers le point P_t. Plus le cosinus est petit, plus la vitesse de croisière doit être réduite. La figure 5.7 illustre notre méthode.

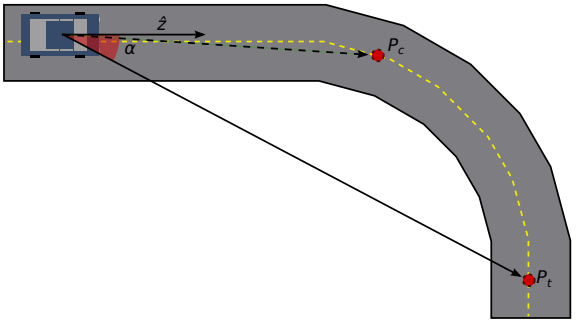

FIGURE 5.7 – Estimation de la courbure d'un virage pour ajuster la vitesse d'un véhicule.

Dans nos simulations nous utilisons une distance de 50 mètres pour la détermination du point P_t. Cette valeur est adaptée à une simulation en milieu urbain, c'est à dire à une vitesse inférieure à 50 km/h. Dans le cas d'une simulation en milieu rural ou sur une voie rapide il est raisonnable de doubler voire tripler cette valeur pour que le conducteur

anticipe d'avantage le virage.

Changement de voie : Lorsque le véhicule doit changer de voie pour suivre l'itinéraire qui lui est imposé, la couche de calcul de trajectoire et de vitesse doit à la fois modifier la direction du véhicule et choisir le véhicule le plus proche à suivre. Le véhicule suivit, s'il en existe un, est celui qui se trouve le plus proche sur la voie courante ou celle de destination.

Avant de changer de voie, le conducteur doit vérifier que la voie de destination est libre. De nombreux auteurs ont proposé des modèles permettant au conducteur de prendre la décision de changer de voie en estimant la dangerosité de la manœuvre (Ahmed, 1999; Choudhury, 2005; Ehmanns et al., 2000). Dans notre application nous utilisons le modèle proposé par (Gipps, 1986) car celui-ci est utilisé dans plusieurs logiciels de références présentés dans l'état de l'art.

Pour adapter la direction du véhicule, nous utilisons la même méthode de projection que nous avons précédemment décrite, à la différence que le point cible P_c se trouve sur la voie de destination au lieu de la voie courante.

Gestion de la signalétique : Afin de tenir compte de la signalétique, en particulier les feux de signalisation, les cédez-le-passages et les stops, le module de gestion de la signalétique doit déterminer si le véhicule a l'autorisation de passer ou s'il doit s'arrêter.

Dans le cas d'un feu de signalisation, l'agent doit vérifier son état. Lorsque le feu est vert, celui-ci est ignoré. Dans tous les autres cas, le feu est considéré comme un obstacle à la circulation du véhicule.

Dans le cas d'un cédez-le-passage ou un panneau de stop, l'agent doit vérifier qu'il n'existe aucun véhicule sur la ou les voies prioritaires. Afin de simplifier cette vérification et éliminer les ambiguïtés pouvant survenir à certains carrefours complexes, chaque panneau de signalisation contient les références vers les voies que l'agent doit surveiller. Ces voies sont définies par un point de départ et une distance à parcourir en sens inverse de la direction du trafic.

Dans la section suivante, nous présentons le modèle de la couche inférieure du comportement responsable de l'évitement de collision.

5.3.2.3/ ÉVITEMENT DE COLLISION

Cette couche est responsable de l'émission des influences de l'agent conducteur. Elle fait appel au véhicule (son corps) pour émettre les influences et doit pour cela fournir un certain nombre d'informations telles que l'accélération désirée et l'orientation du volant. Ces informations sont dépendantes des paramètres provenant des couches supérieures du comportement (choix de direction, de vitesse de croisière et obstacle le plus proche).

Cette couche est principalement basée sur un modèle de suivi de véhicules comme il en existe de nombreux dans la littérature (Chandler et al., 1958; Gipps, 1981). Nous avons choisi d'utiliser le modèle IDM (*Intelligent Driver Model*) (Treiber et al., 2000) car c'est un modèle qui garantit l'absence d'accidents grâce à la dépendance à la vitesse relative entre les deux véhicules, suiveur et suivi (Bevrani et al., 2012). De plus, les paramètres

du modèle sont facilement quantifiables ; et ont des valeurs réalistes. Enfin, la simulation numérique du modèle est rapide.

Le modèle IDM est régi par plusieurs équations. Lorsqu'aucun autre véhicule ni panneau de signalisation forçant l'arrêt n'est présent dans les percepts à l'avant du conducteur, l'accélération du véhicule désirée par l'agent est déterminée par l'équation :

$$\frac{dv}{dt} = a\left[1 - (\frac{v}{v_0})^{\delta}\right] \tag{5.1}$$

Avec v_0 la vitesse de croisière préférée du conducteur en mètre par seconde, v la vitesse courante du véhicule en mètre par seconde et δ l'exposant d'accélération qui permet d'ajuster le comportement pour obtenir des accélérations douces où brutales.

Lorsqu'il existe un obstacle sur la trajectoire du véhicule, nous utilisons l'équation :

$$\frac{dv}{dt} = a\left[1 - (\frac{v}{v_0})^{\delta} - (\frac{s^*(v, \Delta v)}{s})^2\right] \tag{5.2}$$

avec

$$s^*(v, \Delta v) = s_0 + max\left[0, (vT + \frac{v\Delta v}{2\sqrt{ab}})\right] \tag{5.3}$$

et Δv est la vitesse de l'obstacle le plus proche relativement au véhicule en mètre par seconde, T la distance de sécurité exprimée en seconde, a l'accélération préférée de l'agent en mètre par seconde, b la décélération préférée de l'agent en mètre par seconde et s_0 la distance minimale entre deux véhicules à l'arrêt en mètre.

Dans la section suivante, nous présentons le modèle physique de véhicule utilisé pour convertir l'accélération et la direction désirée par l'agent en influences exprimant des forces.

5.4/ Définition du modèle physique des véhicules

Dans cette section, nous détaillons le modèle permettant de simuler le mouvement des véhicules basé sur les lois de la Physique que nous avons utilisé dans le cadre des cas d'application que nous présentons dans ce chapitre.

Le modèle de corps utilisé pour les piétons et les cycles a été présenté dans la section 4.1.2. Il s'agit d'un cylindre indéformable sur lequel peuvent s'appliquer des forces.

Dans les sections suivantes nous présentons l'architecture de notre modèle de véhicule. Nous détaillons ensuite chacun de ses composants.

5.4.1/ Architecture du modèle de véhicule

Dans cette section, nous proposons un modèle de véhicule routier prenant en compte une partie des composants mécaniques internes du véhicule (moteur, transmission, roues, etc.)

Nous avons étendu les travaux de Gechter et al. (2012); Lamotte et al. (2010) en proposant un modèle composé de six sous-modèles pour chacun des composants physiques

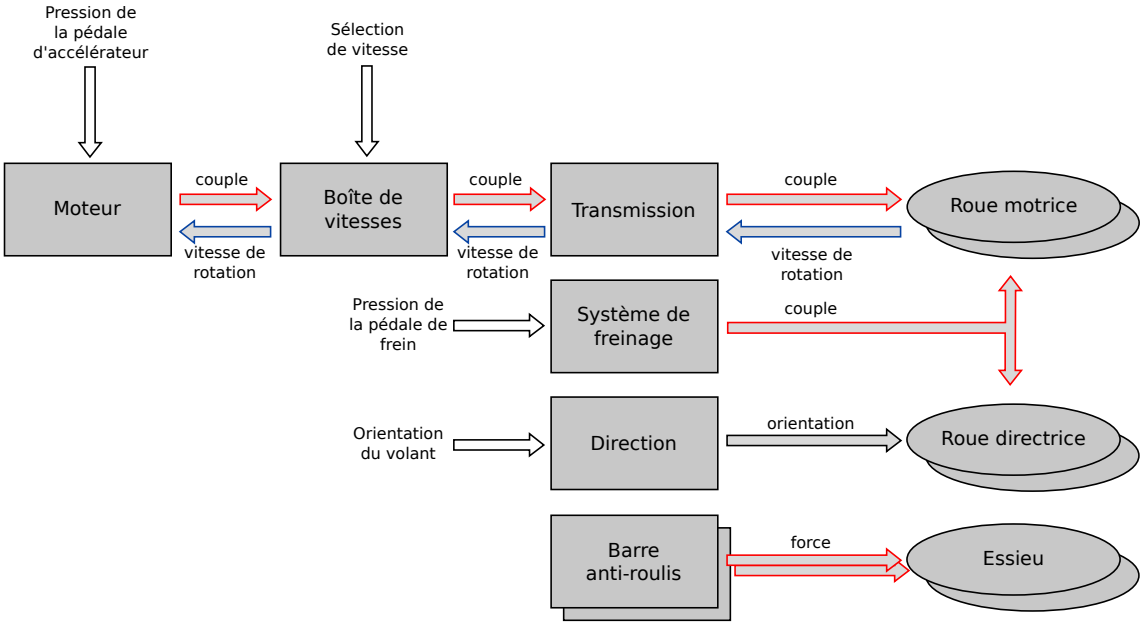

FIGURE 5.8 – Représentation schématique du modèle physique de véhicule

considérés du véhicule. La figure 5.8 fournit un aperçu de cette décomposition et des relations entre les sous-modèles.

Un avantage d'une telle décomposition est qu'il est possible de remplacer un sous-modèle par une de ses variantes indépendamment des autres sous-modèles. Par conséquent, il est très simple de changer le moteur ou la boîte de vitesses d'une voiture dans le cadre d'une simulation. Il est également possible de prévoir plusieurs modélisations du même composant.

En nous inspirant des travaux issus des domaines de la simulation mécanique (Dassen, 2003; Liebemann et al., 2005; Liu et al., 2012; Lu et al., 2007; Wellstead et al., 1997; Zhang et al., 2008) et de la simulation sub-microscopique de véhicules en environnement virtuel (Gechter et al., 2012; Hoogendoorn et al., 2001; S.A., 2010), nous avons sélectionné et adapté un ensemble de sous-modèles pour construire le modèle physique du véhicule. La recherche constante d'un compromis entre la qualité des résultats de simulation et le temps de calcul est au centre de notre sélection.

5.4.2/ MODÈLES SÉLECTIONNÉS POUR LES COMPOSANTS D'UN VÉHICULE

Le modèle physique de véhicule que nous avons choisi d'implanter dans le cadre des projets présentés dans ce chapitre est décomposé en six sous-modèles que nous décrivons dans les sections suivantes. Nous proposons des variantes simples de ces modèles et donnons quelques pistes de recherche concernant des alternatives plus complexes pouvant être utilisées dans des cas d'application différents de ceux que nous présentons.

5.4.2.1/ MOTEUR

Ce sous-modèle permet de déterminer la force générée par le moteur en fonction de la vitesse de rotation du vilebrequin et de l'enfoncement de la pédale de l'accélérateur. Il est possible de simuler finement le fonctionnement interne d'un moteur à combustion, comme nous le montre (Liu et al., 2012). Cependant, si nous souhaitons simuler un grand nombre de véhicules, il nous faut considérer une approche nécessitant très peu de temps de calcul. Notons r_c la vitesse de rotation du vilebrequin, le couple C_e délivré par le moteur est alors obtenu simplement par :

$$C_e = \tau_A f(r_c) \tag{5.4}$$

Où $f(x)$ est la courbe représentant le couple maximal produit par le moteur en fonction de la vitesse de rotation du vilebrequin et τ_A est le coefficient de pression sur la pédale d'accélérateur (entre 0 et 1). Cette courbe peut être dérivée des courbes de puissance motrice assez facilement accessibles dans les spécifications produites par les motoristes.

5.4.2.2/ BOÎTE DE VITESSES

Le sous-modèle de la boîte de vitesses a deux fonctions. D'une part, il permet de modifier le couple fournit par le moteur C_e et transféré à l'arbre de transmission C_g en fonction de la vitesse i sélectionnée et du ratio g_i associé à celle-ci :

$$C_g = g_i C_e \tag{5.5}$$

Les ratios sont généralement fournis par les constructeurs dans les spécifications des boîtes de vitesses.

D'autre part, il permet de calculer la vitesse de rotation du vilebrequin r_c à partir de la vitesse de rotation de l'arbre de transmission r_d

$$r_c = g_i r_d \tag{5.6}$$

Ce module, tout comme pour le sous-modèle du moteur, peut être implanté de façon plus complexe et intégrer l'embrayage (Dassen, 2003). Dans le cadre de notre problématique, ce niveau de précision est inutile et augmente considérablement le temps de calcul par véhicule.

5.4.2.3/ TRANSMISSION

Le sous-modèle de la transmission est très similaire au précédent. Sa fonction première est de modifier le couple fournit par la boîte de vitesse en le multipliant par le ratio fixe g_d de la transmission. Il atténue également le couple d'un ratio $w \in [0, 1]$ pour tenir compte de la perte d'énergie due aux frottements et jeux entre les nombreuses pièces mécaniques.

$$C_d = w g_d C_g \tag{5.7}$$

Le couple C_d est celui s'appliquant directement sur l'essieu moteur et, par conséquent, sur les roues motrices. Sa deuxième fonction est de déterminer la vitesse de rotation de l'arbre de transmission r_d en fonction de la vitesse moyenne de rotation des roues motrices r_w.

$$r_d = g_d r_w \tag{5.8}$$

5.4.2.4/ SYSTÈME DE FREINAGE

Le module du système de freinage est un module très important car son implantation va déterminer la distance d'arrêt d'un véhicule. La manière la plus simple de l'implanter (et celle que nous avons utilisé) est de considérer que le système de freinage est capable de produire un couple $C_{b_{max}}$ s'exerçant sur toutes les roues du véhicule. Ce couple est de sens opposé au couple C_d appliqué sur l'essieu moteur, il peut être calculé simplement :

$$C_b = \tau_B C_{b_{max}} \tag{5.9}$$

Où τ_B est le coefficient de pression sur la pédale de frein (entre 0 et 1). Cette approche simpliste implique que l'agent devrait contrôler son freinage pour éviter de déraper. Il est préférable de simuler les systèmes d'assistance au freinage d'urgence tels que l'ABS (Anti-lock Braking System) (Wellstead et al., 1997; Zhang et al., 2008) ou l'ESP (Electronic Stability Program) (Liebemann et al., 2005; Lu et al., 2007) puisque ceux-ci sont équipés sur une grande proportion des véhicules en circulation à l'heure actuelle.

5.4.2.5/ DIRECTION

Nous avons vu comment déterminer les couples (d'accélération et freinage) s'appliquant aux roues. Il nous reste cependant à traiter de la question de l'orientation des roues directrices pour modifier la trajectoire du véhicule.

Lorsque l'orientation du volant σ est nulle, l'orientation du véhicule ne change pas et ce dernier se déplace alors en ligne droite. Dans le cas contraire, le véhicule se déplace d'un arc de cercle dont le rayon n'est autre que le rayon de courbure ρ_C que l'on peut calculer grâce à l'équation suivante :

$$\rho_C = \frac{\rho_{min}}{\sigma} \tag{5.10}$$

Où ρ_{min} est le rayon du cercle minimal que le véhicule peut former en tournant. Cette constante dépend, bien évidemment, des propriétés physiques du véhicule et fait partie des caractéristiques techniques très facilement accessibles.

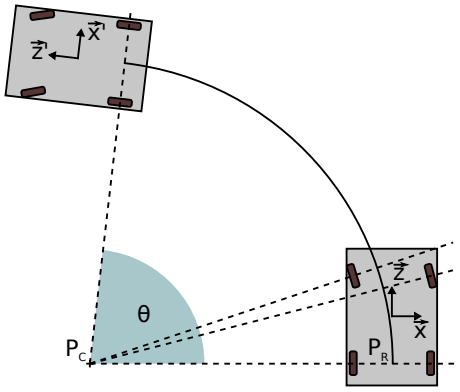

FIGURE 5.9 – Modèle pour le calcul de la trajectoire d'un véhicule dans un virage.

Le problème est représenté par la figure 5.9. Il suffit de déterminer la position P_C du centre du cercle C formé par le véhicule lors d'un virage. Celui-ci est très facilement déterminé

une fois le rayon de courbure calculé grâce à l'équation 5.10 en translatant le point P_R, le milieu de l'essieu arrière dans la direction de l'axe $\overrightarrow{x}$ local au véhicule d'une distance ρ_C (qui est négative lorsque le véhicule tourne à gauche).

5.4.2.6/ Les barres antiroulis

Les véhicules réels sont équipés d'un dispositif permettant de les empêcher de se renverser latéralement et il est indispensable d'intégrer celui-ci dans une simulation physique de véhicules.

Ce dispositif est la barre antiroulis qui permet de synchroniser le mouvement des roues appartenant à un même essieu. La synchronisation s'effectue en transférant une partie de la force de compression s'appliquant sur une roue à la seconde par un effet de levier. La quantité de force transférée dépend de la rigidité de la barre. Notons $S_i \in [0,1]$ le taux d'élongation de l'amortisseur associé à la roue i et $S_j \in [0,1]$ le taux d'élongation de l'amortisseur associé à la roue j. Les roues i et j sont liées par le même essieu et par conséquent la même barre antiroulis. Nous pouvons calculer les forces F_{ij} et F_{ji} transférée par la barre, respectivement de i à j et de j à i par l'équation suivante :

$$F_{ij} = -Fji = \omega\overrightarrow{y}(S_i - S_j) \tag{5.11}$$

Ou ω est la force maximale que peut transférer la barre et correspond à une fraction de la raideur de la suspension et $\overrightarrow{y}$ est l'axe pointant vers le haut du véhicule.

Dans la section suivante nous présentons les applications dans lesquelles le modèle de véhicule physique que nous avons décrit est utilisé.

5.5/ Validation de HEDGE : modélisation de trois zones névralgiques de Belfort

Dans cette section, nous présentons trois applications du modèle HEDGE dans le cadre de deux projets d'aménagement de la ville de Belfort. Elles permettent de mettre en exergue les points forts et les points faibles du modèle HEDGE. Le premier cas concerne la place Rabin et permet d'illustrer la capacité de HEDGE à modéliser une structure routière complexe. L'application dédiée au quartier Denfert-Thiers met en avant la définition de topologies de routes inhabituelles et leurs impacts sur les flux de véhicules. Le dernier cas, concernant la Place d'Armes, illustre la modélisation d'un environnement unique pour simuler des piétons, des cyclistes et des véhicules.

5.5.1/ La place Rabin

La place Rabin est un lieu très significatif du projet Optymo et connu localement pour sa complexité routière et son importance sur le plan stratégique (illustré par la figure 5.10). Il s'agit d'un lieu de passage obligatoire pour traverser la ville. La décision de simuler ce lieu s'appuie sur trois points principaux :

— **La demande :** La demande est très importante en heure de pointe du soir (environ 5000 v/h sur le nœud). Le système de sas contraint les possibilités d'écoulement car un sas doit d'abord se vider avant de laisser d'autres branches au vert.

— **La synchronisation :** La place contient trois cellules à coordonner avec certaines programmations en heure creuse à priorité bus. La régulation à phase escamotable adopté à Belfort ne peut pas être appliquée ici en heure de pointe pour des raisons fonctionnelles : la demande est telle que les véhicules particuliers ne doivent pas être perturbés.
— **La géométrie :** L'emprise étant limitée sur la zone, des compromis ont été trouvé pour répondre à toutes les fonctions du carrefour.

Le réaménagement de cette place est drastique : les sens de circulation sont modifiés, des sites propres sont ajoutés, certaines voies sont élargies, la végétation est remplacée, la régulation des carrefours est adaptée pour permettre le transit de nouvelles lignes de bus avec des fréquences deux fois plus importantes qu'auparavant.

5.5.1.1/ Objectifs de l'étude de la place Rabin

La simulation de cette place a des objectifs doubles :
— d'une part, la visualisation spatiotemporelle du site couplée à la simulation des véhicules et piétons permettent au maître d'ouvrage d'appréhender la portée de ses décisions et le bon fonctionnement de ses choix d'aménagement. Celle-ci lui permet également de communiquer sur le projet en fournissant une illustration dynamique d'un futur aux riverains et aux usagers lors de réunions de concertation publique.
— D'autre part, la simulation physique des véhicules permet de vérifier la conformité des plans d'aménagement en particulier la largeur et la courbure des voies de bus.

La figure 5.11 montre une vue aérienne d'une partie de la place Rabin avant les travaux d'aménagement illustrés par le plan 5.12. Bien que ces deux figures représentent exactement la même position géographique, on peut remarquer que la différence est majeure.

La complexité de ce carrefour en fait un parfait exemple de la flexibilité du modèle HEDGE ainsi que de la facilité à créer un modèle lorsque la topologie de l'environnement réel est complexe.

FIGURE 5.10 – Aperçu du rendu 3D de la simulation au niveau du carrefour de la place Rabin.

Dans la section suivante, nous détaillons la modélisation de ce carrefour réalisée avec HEDGE.

5.5.1.2/ MODÉLISATION AVEC HEDGE

La création du modèle de ce carrefour dans Voxelia Simulate est relativement directe 5.13. Les voies entrantes et sortantes des carrefours sont d'abord créées puis reliées ensuite par d'autres voies pour former l'intersection et permettre tous les mouvements autorisés par le plan de circulation.

Les lignes de feux sont placées conformément au plan et les phases implantées sont celles, longuement évaluées à l'aide de MISDAU et finalement approuvées par le maître d'ouvrage. On remarquera que la modélisation utilisant HEDGE est très similaire à celle réalisée dans le logiciel PTV Vissim et le temps consacré à la création du modèle est sensiblement équivalent.

La figure 5.14 montre la modélisation de ce même carrefour dans le logiciel PTV Vissim. Les tracés bleus représentent les tronçons et les tracés violets représentent les connecteurs.

Dans le cas de cette place, nous n'avons pas souhaité permettre aux piétons de traverser en dehors des passages piétons car en pratique ces comportements seraient extrêmement dangereux et il est fort probable que les débits importants de véhicules dans ce secteur dissuadent effectivement les piétons de le faire dans la réalité. Avec cette hypothèse de travail, il n'est donc pas nécessaire de créer les chemins piétons en même temps que le réseau routier pour permettre les connexions avec des liens de navigation comme nous l'avons illustré par l'exemple 3.3.3 page 65.

Nous faisons remarquer que notre choix de n'utiliser que des nœuds de type couloir et jonction pour représenter les chemins piétons donnent lieu à la présence d'une configuration atypique dans le coin supérieur droit de la figure 5.13 (la zone entourée d'un cercle rouge). Ce choix induit un biais puisqu'une partie du trottoir n'est pas couverte par un nœud et devient donc effectivement impraticable. Cependant, la simulation de ce secteur porte principalement sur les flux de véhicule et la présence de piétons dans la simulation est secondaire. Nous avons choisis d'accepter ce biais qui nous permet de bénéficier de meilleurs performances puisque la localisation et la gestion des places par le comportement de piétons est plus coûteuse en temps de calcul.

FIGURE 5.11 – Vue aérienne d'un carrefour de la place Rabin avant travaux.

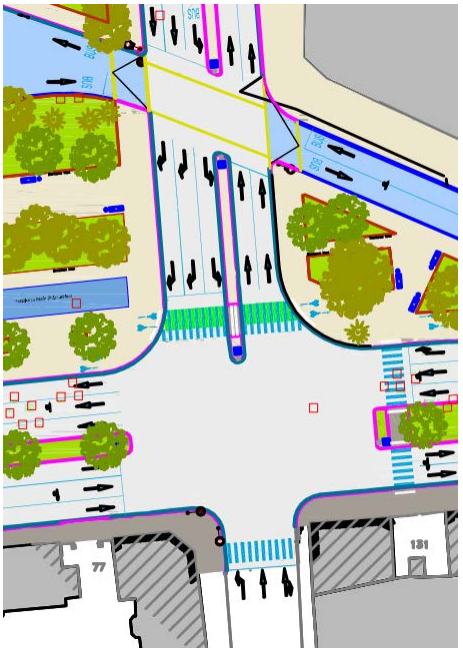

FIGURE 5.12 – Plan d'aménagement Optymo2.

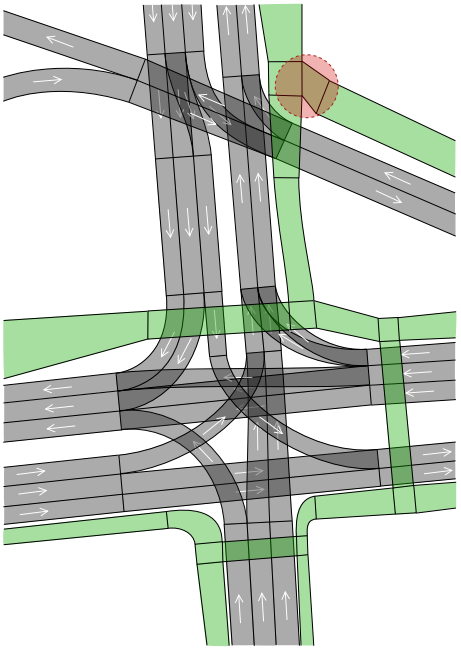

FIGURE 5.13 – Le modèle du carrefour correspondant au plan 5.12. Les nœuds réservés aux piétons sont représentés en vert, les routes en gris.

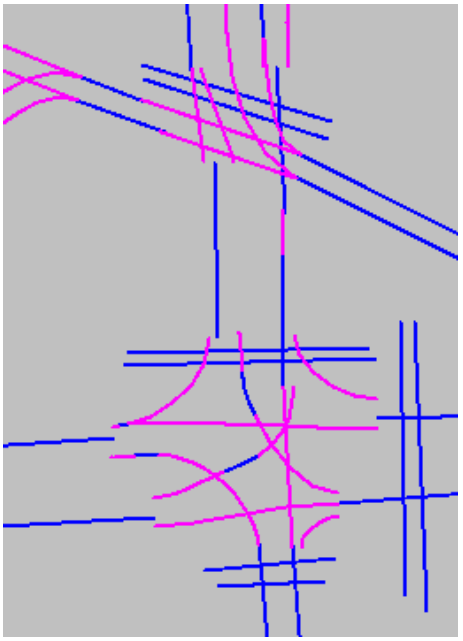

FIGURE 5.14 – Le modèle Vissim du carrefour correspondant au plan 5.12.

5.5.2/ Le quartier Denfert-Thiers

La simulation du quartier Denfert-Thiers se concentre sur un axe est-ouest permettant de relier la gare au cœur historique de la ville. Elle concerne principalement les rues Thiers et Denfert Rochereau et comprend une intersection avec le Faubourg de Montbéliard qui est l'une des artères les plus empruntées puisqu'elle permet les déplacements nord/sud à travers la ville.

La figure 5.15 montre une vue aérienne d'une intersection du quartier avant les travaux d'aménagement illustrés par le plan 5.16.

5.5.2.1/ Objectifs de l'étude du quartier Denfert-Thiers

La simulation du quartier Denfert-Thiers avait pour principal objectif d'être utilisé lors de consultations publiques pour expliquer aux riverains que les modifications entraînées par le projet Optymo n'impacteraient pas négativement leur quotidien. En effet, certaines rues de ce quartier sont passées de simple rue de transit occasionnel à un axe avec deux lignes de bus à haute fréquence. De plus, afin de faciliter le passage des bus, les concepteurs du projet ont choisi de supprimer un grand nombre de places de stationnement le long des rues impactées. La simulation n'avait pas, par conséquent, pour but de valider les aménagements mais simplement d'illustrer le fonctionnement normal.

Dans la section suivante, nous détaillons la modélisation de ce cas d'application réalisée avec HEDGE.

5.5.2.2/ Modélisation avec HEDGE

La figure 5.17 représente notre modélisation conformément au plan d'aménagement 5.16.

Sur la partie gauche on peut voir que deux voies de sens opposés se superposent créant un emplacement vide. Cette configuration correspond à l'une des hypothèses du scénario : l'emplacement vide contourné par les nœuds environnants est occupé par des véhicules stationnés. Afin de simplifier le comportement des agents conducteurs, nous avons choisi de ne pas nous appuyer sur le modèle de dépassement pour simuler l'écart que doit faire un véhicule passant d'est en ouest pour contourner l'obstacle. À la place, nous avons simplement placé un cédez-le-passage avant le début de la manœuvre permettant d'obliger les véhicules à patienter avant de s'engager le temps que la voie en sens opposée se libère. Cette solution a cependant un inconvénient. À cause de la superposition des nœuds, tant qu'un véhicule est en cours de contournement, un autre véhicule situé à l'arrière de celui-ci ne pourra pas s'engager car il le percevra comme présent sur la voie opposée. Nous pouvons cependant négliger ce phénomène puisque les débits de véhicules sur ces voies sont faibles et la manœuvre d'évitement est relativement brève.

Sur la partie droite de l'intersection, nous pouvons remarquer que deux nœuds en sens opposés sont en exacte superposition. Ceci modélise une voie unique partagée par les deux sens de circulation. En pratique, cela ne pose aucun problème car les phases des feux de circulation régulant le trafic de l'intersection sont conçues pour qu'il n'y ait aucun conflit possible.

FIGURE 5.15 – Vue aérienne d'un carrefour du quartier Denfert-Thiers avant travaux.

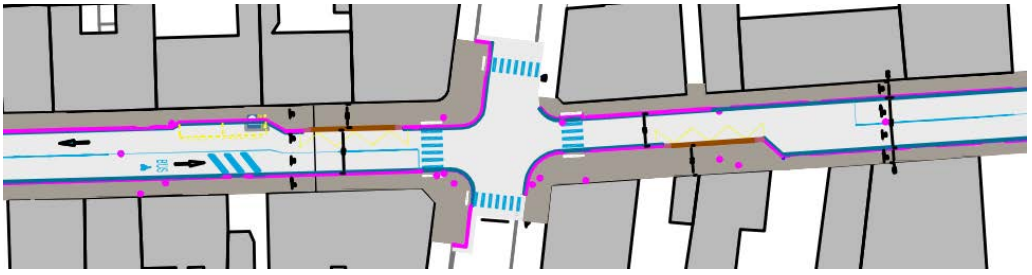

FIGURE 5.16 – Plan d'aménagement Optymo2.

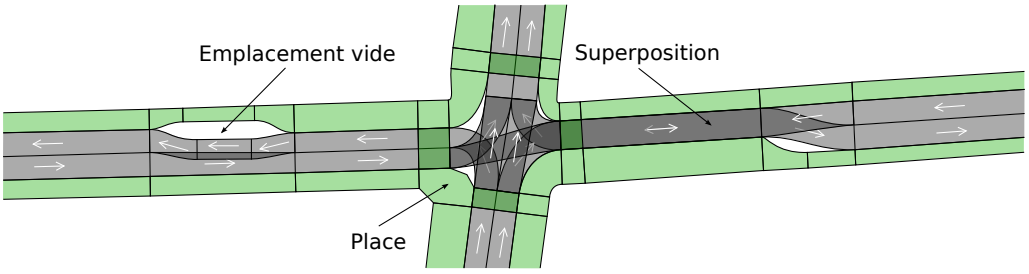

FIGURE 5.17 – Le modèle du carrefour correspondant au plan 5.16. Les nœuds réservés aux piétons sont représentés en vert, les routes en gris.

Concernant les chemins piétons, tout le réseau est conçu à partir de nœuds couloirs et jonctions à l'exception d'une place qui se trouve sur la partie sud-ouest de l'intersection. En effet, la topologie particulière du carrefour nous oblige à utiliser ce type de nœud. Cependant nous pouvons remarquer que grâce à nos trois types de nœud nous pouvons représenter des environnements très atypiques de façon très simple et ce même lorsque la création du modèle est manuelle. Nous avons laissé à nos agents piétons la possibilité de traverser les rues est-ouest en dehors des passages piétons mais pas l'axe nord-sud à cause du débit important de véhicules sur ce dernier.

5.5.3/ La Place d'Armes

Le projet d'embellissement de la Place d'Armes (illustré par la figure 5.18) est un projet décidé par la Ville dans la continuité d'une volonté de renforcer l'attractivité du patrimoine belfortain. Cette volonté s'est traduite par l'embellissement de nombreux espaces publics de la vielle ville. Ce projet comportait quatre principaux objectifs :

- organiser un espace central pour donner une image commercial, dynamique et attractive,
- valoriser le piéton,
- mettre en scène les grands équipements historiques,
- contribuer au parcours urbains Gare-Citadelle.

Les moyens retenus par le maître d'ouvrage pour remplir ces objectifs sont principalement : (i) de donner une place prioritaire au piéton en supprimant le trafic routier sur certaines rues adjacentes, (ii) de limiter la circulation de transit tout en permettant un accès au riverains et aux livraisons, (iii) de renouveler les arbres, (iv) organiser des terrasses pour les différents restaurants et bars aux abords de la place.

La figure 5.19 montre une vue aérienne de la place avant les travaux d'aménagement illustrés par le plan 5.20.

FIGURE 5.18 – Aperçu du rendu 3D de la simulation de la Place d'Armes.

5.5.3.1/ Objectifs de l'étude de la Place d'Armes

Dans le cadre de ce projet, la simulation avait pour principal objectif de préparer des captures vidéo de la place pour les présenter au public. Pour réaliser cet objectif, la modélisation en 3D des lieux dans un logiciel dédié et l'animation de personnages auraient pût être suffisant. Cependant, la simulation permet de donner vie à une scène 3D beaucoup plus rapidement puisqu'il n'est pas nécessaire de saisir les trajectoires individuelles de toutes les entités mobiles. Il est seulement nécessaire de créer un modèle de l'environnement, tâche extrêmement simple et rapide pour un scénario de la taille de celui que nous présentons ici.

Dans la section suivante, nous détaillons la modélisation de ce cas d'application réalisée avec HEDGE.

5.5.3.2/ Modélisation avec HEDGE

La figure 5.21 représente notre modélisation conformément au plan d'aménagement 5.16.

La place principale est modélisée par un nœud de type place. Les terrasses aux abords de la place sont retirées du tracé du nœud pour simplifier les déplacements des piétons et cyclistes. Des nœuds de type couloir sont ensuite connectés à la place pour modéliser les rues piétonnes adjacentes et les trottoirs. Enfin, le réseau routier est simplement généré conformément au tracé du plan d'aménagement.

Le scénario de simulation inclut un nombre important d'entités statiques peuplant la place, telles que des groupes de piétons en train de discuter. Ces entités sont modélisées comme de simples obstacles au même titre que la statue et le kiosque se trouvant sur la place. Ainsi, ils sont perçus et évités par les agents piétons et cyclistes.

La simulation de cette place a permis de mettre en évidence une difficulté pour les conducteurs due au tracé de la route. Le virage dans le coin inférieur gauche de la place était difficile à négocier par nos agents. A cause de l'intégration du comportement physique du véhicule, ceux-ci avaient tendance à prendre le virage soit trop à l'intérieur, ce qui obligeait le véhicule à rouler sur le trottoir longeant la place, soit trop à l'extérieur ce qui forçait les véhicules à rouler sur le trottoir se situant sur le côté de la rue opposée à la place. Il s'avère que cette difficulté se retrouve dans la réalité maintenant que les travaux d'aménagement sont réalisés. Les véhicules tournant à ce virage doivent le faire à une vitesse très réduite pour ne pas risquer des dommages.

FIGURE 5.19 – Vue aérienne de la Place d'Armes avant travaux.

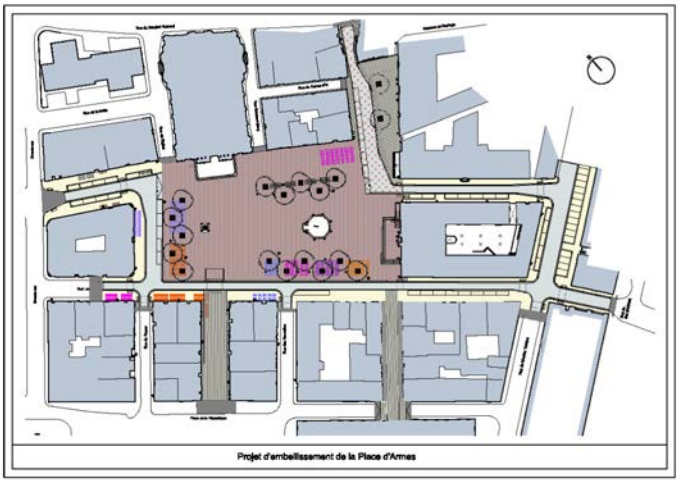

FIGURE 5.20 – Plan d'aménagement de la place.

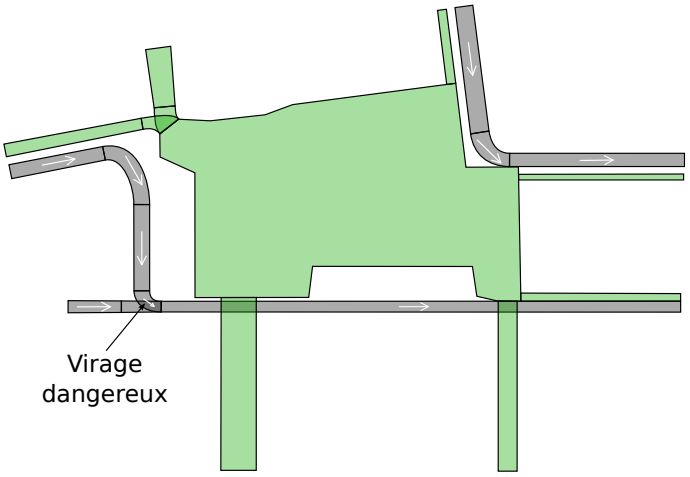

FIGURE 5.21 – Le modèle du carrefour correspondant au plan 5.20. Les nœuds réservés aux piétons sont représentés en vert, les routes en gris.

5.6/ CONCLUSION

Dans ce chapitre, nous décrivons l'application du modèle de l'environnement HEDGE, proposé dans les chapitres précédents, sur deux projets d'aménagement de la ville de Belfort et de sa communauté d'agglomérations. Ces applications ont été réalisées dans le logiciel Voxelia Simulate dans lequel nous avons implanté le modèle HEDGE.

Le premier projet concerne la mise en place d'un réseau de bus à haut niveau de service. HEDGE est utilisé pour modéliser les zones névralgiques à un niveau microscopique. Il permet de modéliser efficacement et rapidement l'infrastructure routière. Les simulations réalisées avec ce modèle ont mis en évidence des problèmes structurels (géométrie de voie inappropriée, etc.) et des problèmes fonctionnels (congestion, synchronisation des feux, etc.). Notre modèle est utilisé dans le but de valider les aménagements en considérant les contraintes topologiques et géométriques, et servir de support de communication.

Le second projet concerne l'aménagement de la Place d'Armes de Belfort afin de renforcer l'attractivité commerciale et touristique du centre historique de la ville. La valorisation des déplacements des piétons et la mise en valeurs des grands équipements historiques sont au centre de ce projet. HEDGE a permis de donner vie à un projet tout en respectant son essence architecturale. Le produit de l'application de ce modèle est utilisé à des fins d'études techniques (études préliminaires, avant-projet), de communication (concertations et débats publics) et de promotion (expositions, valorisation) sous la forme de logiciels interactifs en 3D.

Le modèle HEDGE est une solution adaptée au niveau le plus précis dans l'architecture multi-modèle proposée. Pour conclure ce chapitre, nous pensons que le modèle HEDGE est adapté aux cas d'applications et à la résolution des problématiques décrits dans l'introduction générale. Notre modèle permet de reproduire la simulation d'un système complexe, comme un réseau routier urbain, tout en gérant efficacement les ressources de calcul disponibles. Nous avons montré dans ce chapitre que ce modèle est utilisable et utilisé dans un contexte non académique.

Conclusion générale

1/ Bilan

Dans cette thèse, nous avons proposé un modèle d'environnement physique permettant de simuler le déplacement de véhicules et de piétons en milieu urbain. La simulation orientée-agent se prête idéalement à la simulation des systèmes composés d'entités autonomes en interaction. L'environnement est considéré comme un élément essentiel dans ce type de simulation. Il s'agit d'une abstraction de premier ordre permettant la prise en compte des interactions entre les agents.

Le modèle HEDGE proposé est basé sur un découpage de l'espace sous la forme d'un graphe représentant les zones interconnectées de l'environnement. Nous avons séparé les aspects liés à la perception et à la navigation des agents. En effet, cette distinction enrichit la description de l'environnement, et contribue à faciliter les processus de délibération des agents. Cette décomposition permet la cohabitation d'agents de types différents dans le même environnement et simplifie la gestion des zones partagées. Il en résulte une représentation de l'environnement utilisant différents types de nœuds (couloirs, places, jonctions) et de liens (navigation, perception).

L'environnement possède d'autres caractéristiques liées à sa dynamique et à ses missions dans le contexte du système étudié. Dans ce cadre, nos contributions concernent la définition de la réaction de l'environnement aux actions initiées par les agents en s'appuyant sur un modèle inspiré de la physique. Cette approche permet de faciliter la détermination de la réaction de l'environnement. De plus, elle permet de simuler avec précision les mouvements des entités tout en simplifiant la gestion de la simultanéité des actions des agents ainsi que la résolution d'éventuels conflits. Par ailleurs, un mécanisme de localisation a été proposé pour renseigner sur le graphe de l'environnement la position des différentes entités. La localisation fait le lien entre la position dans l'espace d'un objet (propriété associée au modèle physique) et sa position dans la représentation abstraite de l'environnement (utilisée par les agents).

Afin de valider nos travaux sur des applications réelles, nous avons étudié la simulation des déplacements dans le cadre de deux projets d'aménagement de la ville de Belfort et de sa communauté d'agglomérations. HEDGE a été utilisé pour modéliser les zones névralgiques du milieu urbain à un niveau microscopique. Cette application a permis de mettre en avant la précision et la rapidité de modélisation de l'infrastructure routière par le modèle HEDGE. Les simulations réalisées ont mis en évidence des problèmes structurels et des problèmes fonctionnels. Le modèle HEDGE permet d'obtenir des simulations avec un niveau de réalisme satisfaisant, tout en respectant la contrainte temps réel imposée par le contexte industriel dans lequel s'inscrivent nos travaux.

Les principaux avantages des abstractions et des outils présentés dans cette thèse peuvent être résumés comme suit :

— Modélisation des différentes topologies d'environnements à l'aide d'un graphe.

— Distinction des aspects liés à la perception et à la navigation des agents en proposant deux types de liens entre zones spatiales dans le graphe.
— Intégration de modèles inspirés de la physique permettant d'obtenir des comportements plus précis et de simplifier la gestion de la simultanéité des actions des agents.
— Simplification du processus de modélisation en permettant de créer un modèle de l'environnement unique pour supporter différents types d'agents (piétons, conducteurs).

Les travaux présentés dans cette thèse nécessitent plusieurs approfondissements. La section suivante recense quelques-unes des améliorations possibles pouvant faire l'objet de futures recherches.

2/ Perspectives

Nous définissons deux axes d'approfondissement pour la suite de ce travail. Le premier axe s'intéresse à la définition d'un processus méthodologique dédié à la modélisation de l'environnement. Le second axe concerne l'étude et l'adaptation de notre modèle pour permettre sa mise à l'échelle dans le cadre d'applications faisant intervenir un grand nombre d'entités.

2.1/ Méthodologie de conception

Il s'agit de définir précisément la méthodologie à employer pour modéliser les environnements urbains en utilisant les structures fournies par notre modèle. En effet, le modèle HEDGE, de par la flexibilité qu'il offre, permet de modéliser une même topologie de différentes manières. Certains choix de modélisation peuvent introduire un biais dans la simulation en contraignant arbitrairement les mouvements des entités. Il nous parait alors nécessaire d'aider le concepteur du modèle de simulation en lui proposant un guide méthodologique adapté. L'extension du processus méthodologique ASPECS (Cossentino et al., 2010) pour la modélisation d'environnements virtuels semble être prometteuse. Ce processus propose déjà un cadre pour la modélisation d'agents et de leurs interactions. Ce cadre pourrait être étendu pour prendre en compte les spécificités de la modélisation d'environnements physiques (virtuels ou réels).

2.2/ Passage à l'échelle du modèle de l'environnement

Il s'agit de permettre la prise en compte d'un grand nombre d'entités au sein d'une même simulation. Il devient alors nécessaire d'évaluer précisément les opérations associées à l'environnement afin de déterminer celles pouvant être optimisées. Par exemple, nos expérimentations tendent à montrer que la localisation des entités et le calcul des perceptions sont des opérations critiques. Cette optimisation peut être envisagée selon différentes approches. La première piste concerne l'utilisation de modèles spécialisés tels que les modèles multi-niveaux de l'environnement. La seconde piste s'intéresse à la parallélisation des opérations, et notamment à l'élaboration de modèles pouvant s'exécuter sur un processeur graphique.

BIBLIOGRAPHIE

Ahmad, O., et Papelis, Y. E. (2000). **An autonomous driver model for the overtaking maneuver for use in microscopic traffic simulation**. Dans *Driving Simulator Conference, DSC'00, Paris, France*.

Ahmed, K. I. (1999). **Modeling drivers' acceleration and lane changing behavior**. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA.

Allemand, S., Ascher, F., et Levy, J. (2004). **Les Sens du mouvement. Modernité et mobilités dans les sociétés urbaines contemporaines**. Belin.

Amblard, F. (2003). **Comprendre le fonctionnement de simulations sociales individus-centrées, Application à des modèles de dynamiques d'opinions**. PhD thesis, Cemagref, Clermont-Ferrand, France.

Arikan, O., Chenney, S., et Forsyth, D. A. (2001). **Efficient multi-agent path planning**. Dans *In Proceedings of the 2001 Eurographics Workshop on Animation and Simulation*, pages 151–162.

Badler, N. I. (1997). **Virtual humans for animation, ergonomics, and simulation**. Dans *IEEE Workshop on Non-Rigid and Articulated Motion*, pages 28–36.

Balmer, M., Cetin, N., Nagel, K., et Raney, B. (2004). **Towards truly agent-based traffic and mobility simulations**. Dans *the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS04)*, pages 60–67, Washington, DC, USA. IEEE Computer Society.

Barbuceanu, M., et Fox, M. S. (1995). **COOL : A language for describing coordination in multi agent systems**, volume pages, pages 1–15. Citeseer.

Barceló, J. (2010). **Fundamentals of traffic simulation**. Dans Barceló, J., éditeur, *Fundamentals of Traffic Simulation*, volume 145 de *International Series in Operations Research & Management Science*. Springer New York.

Barella, A., Ricci, A., Boissier, O., et Carrascosa, C. (2012). **MAM5 : Multi-agent model for intelligent virtual environments**. Dans *10th European Workshop on Multi-Agent Systems (EUMAS 2012)*, pages 16–30.

Barros, L., Da Silva, A., et Musse, S. (2004). **Petrosim : An architecture to manage virtual crowds in panic situations**. Dans *Computer Animation and Social Agents*, pages 111–120.

Beckers, R., Holland, O. E., et Deneubourg, J. L. (1994). **From local actions to global tasks : Stigmergy and collective robotics**. pages 181–189. MIT Press.

Begum, M., et Karray, F. (2011). **Visual attention for robotic cognition : A survey**. *IEEE T. Autonomous Mental Development*, 3(1) :92–105.

Bellemans, T., Kochan, B., Janssens, D., Wets, G., Arentze, T., et Timmermans, H. (2010). **Implementation framework and development trajectory of feathers activity-based simulation platform**. pages 111–119. Transportation Research Record : Journal of the Transportation Research Board.

Ben-Akiva, M., Koutsopoulos, H., Toledo, T., Yang, Q., Choudhury, C., Antoniou, C., et Balakrishna, R. (2010). **Traffic simulation with MITSIMLab**. Dans Barceló (2010), pages 233–268.

Bender, J., Erleben, K., et Trinkle, J. (2014). **Interactive simulation of rigid body dynamics in computer graphics**. *Computer Graphics Forum*, 33(1) :246–270.

Bentley, J. L. (1975). **Multidimensional binary search trees used for associative searching**. *Commun. ACM*, 18(9) :509–517.

Bergen, von den, G. (2003). **Collision Detection in Interactive 3D Environments**. Morgan Kaufmann, San Francisco.

Bergenti, F., Gleizes, M.-P., et Zambonelli, F., éditeurs (2004). **Methodologies and Software Engineering for Agent Sytems**. Kluwer Academic Press.

Berger, T., Birner, R., Mccarthy, N., DíAz, J., et Wittmer, H. (2007). **Capturing the complexity of water uses and water users within a multi-agent framework**. *Water Resources Management*, 21(1) :129–148.

Bernstein, S. N. (1912). **Démonstration du théorème de weierstrass, fondée sur le calcul des probabilités**. *Comm. Soc. Math. Kharkov Ser.*, 13.

Bevrani, K., et Chung, E. (2012). **Comparison and modification of car following models toward a more accurate microscopic simulation parameters reproduction**. Dans Stevens, A., et Chung, E., éditeurs, *19th ITS World Congress*, 19th ITS world congress, Messezentrum Wien Exhibition and Congress Center, Vienna, Austria. AustriaTech.

Boissier, O., et Demazeau, Y. (1994). **ASIC : An architecture for social and individual control and its application to computer vision**. Dans *Preproceedings of the 6th European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, pages 107–118.

Bottaci, L. (1995). **A direct manipulation interface for a user enhanceable crowd simulator.** *Journal of Intelligent Systems*, 5(2-4) :249–272.

Boulic, R., Capin, T., Huang, Z., Kalra, P., Lintermann, B., Magnenat-Thalmann, N., Moccozet, L., Molet, T., Pandzic, I., Saar, K., Schmitt, A., Shen, J., et Thalmann, D. (1995). **The HUMANOID environment for interactive animation of multiple deformable human characters**. *Computer Graphics Forum*, 14 :337–348.

Bouvier, E., Cohen, E., et Najman, L. (1997). **From crowd simulation to airbag deployment : particle systems, a new paradigm of simulation**. *Journal of Electrical Imaging*, 6(1) :94–107.

Bouvier, E., et Guilloteau, P. (1996). **Crowd simulation in immersive space management**. Dans *Eurographics Workshop on Virtual Environments and Scientific Visualization'96*, pages 104–110. Springer.

Braun, A., B.E.J., B., Oliveira, L., et Musse, S. (2003). **Modeling individual behaviors in crowd simulation**. Dans *Computer Animation and Social Agents*, pages 143–148, New Brunswick, NJ, USA.

Braun, A., Bodmann, B. E. J., et Musse, S. R. (2005a). **Simulating virtual crowds in emergency situations**. Dans *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '05, pages 244–252, New York, NY, USA. ACM.

Braun, A., Bodmann, B. E. J., et Musse, S. R. (2005b). **Simulating virtual crowds in emergency situations**. Dans *Proceedings of the ACM symposium on Virtual reality software and technology*, VRST '05, pages 244–252, New York, NY, USA. ACM.

Bretagnolle, A., Daudé, E., et Pumain, D. (2003). **From theory to modelling : urban systems as complex systems**. *Cybergeo, 13th European Colloquium on Quantitative and Theoretical Geography*, 335.

Brogan, D., et Hodgins, J. (1997). **Group behaviors for systems with significant dynamics**. *Autonomous Robots*, 4 :137–153.

Brooks, R. A. (1991). **Intelligence without representation**. *Artificial intelligence*, 47(1) :139–159.

Bui, M., Lowe, N., et Takatsuka, M. (2006). **Visualization of a closed three-dimensional surface using portal-based rendering**. Dans *Asia-Pacific Symposium on Information Visualization*, Tokyo, Japan.

Buisson, J., Galland, S., Gaud, N., Gonçalves, M., et Koukam, A. (2013). **Real-time collision avoidance for pedestrian and bicyclist simulation : a smooth and predictive approach**. Dans Yasar, A.-u.-h., et Knapen, L., éditeurs, *2nd International Workshop on Agent-based Mobility, Traffic and Transportation Models, Methodologies and Applications (ABMTRANS13)*, Halifax, Nova Scotia, Canada. Elsevier.

Buisson, J., Galland, S., Gonçalves, M., et Koukam, A. (2012). **Agent-based simulation of pedestrian and vehicles in 3D virtual cities**. Dans *2èmes Journées des Jeunes Chercheurs de l'UTBM (IngéDoc 2012)*. Doceo, UTBM, UTBM Press.

Béhé, F., Galland, S., Gaud, N., Nicolle, C., et Koukam, A. (2014). **An ontology-based metamodel for multiagent-based simulations**. *Simulation Modelling Practice and Theory*, 40(0) :64 – 85.

Cabri, G., Leonardi, L., et Zambonelli, F. (2000). **MARS : A programmable coordination architecturefor mobile agents**. *IEEE Internet Computing*, 4(4) :26–35.

Caliper (2014). **Transmodeler overview**. http://caliper.com/transmodeler.

Casas, J., Ferrer, J. L., García, D., Perarnau, J., et Torday, A. (2010). **Traffic simulation with aimsun**. Dans Barceló (2010), pages 173–232.

Chalons, C. (2007). **Transport-equilibrium schemes for pedestrian flows with non-classical shocks**. Dans Schadschneider, A., Pöschel, T., Kühne, R., Schreckenberg, M., et Wolf, D., éditeurs, *Traffic and Granular Flow'05*, pages 347–356. Springer Berlin Heidelberg.

Chandler, R. E., Herman, R., et Montroll, E. W. (1958). **Traffic dynamics : Studies in car following**. *Operations Research*, 6(2) :165–184.

Chelhorn, T., O'Sullivan, D., Haklay, M., et Thurstain-Goodwin, M. (1999). **Streets : An agent-based pedestrian model**. Dans *Computers in Urban Planning and Urban Management*.

Chen, X. (2013). **Real-time physics based simulation for 3d computer graphics**.

Choudhury, C. F. (2005). **Modeling lane-changing behavior in presence of exclusive lanes**. Master's thesis, Massachusetts Institute of Technology.

Ciancarini, P. (1996). **Coordination models and languages as software integrators**. *ACM Computing Surveys*, 28(2) :300–302.

Conte, R., Gilbert, N., et Simao Sichman, J. (1998). **MAS and social simulation : A suitable commitment**. Dans *First International Workshop on Multi-Agent Systems and Agent-Based Simulation*, volume 1534 de *LNCS*, pages 1–9. Springer-Verlag.

Coquillard, P., et Hill, D. R. (1997). **Modélisation et Simulation des Ecosystèmes**. Masson.

Cossentino, M., Gaud, N., Hilaire, V., Galland, S., et Koukam, A. (2010). **ASPECS : an agent-oriented software process for engineering complex systems - how to design agent societies under a holonic perspective**. *Autonomous Agents and Multi-Agent Systems*, 2(2) :260–304.

Daamen, W. (2004). **Modelling Passenger Flows in Public Transport Facilities**. PhD thesis, Technical University of Delft.

Dassen, M. (2003). **Modelling and control of automotive clutch systems**. Rapport technique 2003.73, Department of mechanical engineering TU/e Eindhoven.

De Floriani, L., et Magillo, P. (2002). **Regular and irregular multi-resolution terrain models : a comparison**. Dans *Proceedings of the 10th ACM international symposium on Advances in geographic information systems*, GIS '02, pages 143–148, New York, NY, USA. ACM.

Dechter, R., et Pearl, J. (1985). **Generalized best-first search strategies and the optimality of a***. *J. ACM*, 32(3) :505–536.

Demange, J. (2012). **Un modèle d'environnement pour la simulation multiniveau - Application à la simulation de foules**. PhD thesis, Université de Technologie de Belfort-Montbéliard, Belfort, France. In French.

Desaraju, V. R., et How, J. P. (2012). **Decentralized path planning for multi-agent teams with complex constraints**. *Auton. Robots*, 32(4) :385–403.

Dey, P., et Roberts, D. (2007). **A conceptual framework for modelling crowd behaviour**. Dans *DS-RT '07 : Proceedings of the 11th IEEE International Symposium on Distributed Simulation and Real-Time Applications*, pages 193–200, Washington, DC, USA. IEEE Computer Society.

Dijkstra, E. W. (1959). **A note on two problems in connexion with graphs**. *Numerische mathematik*, 1 :269–101.

Drogoul, A. (1993). **De la simulation multi-agents à la résolution collective de problèmes**. PhD thesis, University Paris 6, Paris, France.

Drogoul, A., et Picault, S. (1999). **Microbes : Vers des collectivités de robots socialement situés**. Dans Gleizes, M.-P., et Marcenac, P., éditeurs, *VIIèmes Journées Francophones pour l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents (JFIADSMA)*, pages 265–278. Hermès.

Du, Y., Geng, Y., et Sun, L. (2009). **Simulation model based on monte carlo method for traffic assignment in local area road network**. *Frontiers of Architecture and Civil Engineering in China*, 3(2) :195–203.

Durig, T. (2014). **Environnement informé sémantiquement enrichi pour la simultion multi-agents - Application à la simulation en environnement virtuel 3D**. PhD thesis, Université de Bourgogne, Dijon, France.

Eberly, D. H. (2001). **3D Game Engine Design**. Morgan Kaufmann, San Francisco.

Ehmanns, D. (2001). **Simulation model of human lane change behaviour**. Dans *VDI/SAE/JSAE Gemeinschaftstagung*, pages 60–67, Berlin, Germany. VDI-Verlag.

Ehmanns, D., et Hochstädter, A. (2000). **Driver-model of lane change maneuvers**. Dans *7th World Congress on Intelligent Transportation Systems*.

Elfes, A. (1989). **Using occupancy grids for mobile robot perception and navigation**. *Computer*, 22 :46–57.

Errampalli, M., Okushima, M., et Akiyama, T. (2008). **Fuzzy logic based lane change model for microscopic traffic flow simulation**. *Advanced Computational Intelligence and Intelligent Informatics*, 12 :172–173.

Espié, S., et others (1995). **Microscopic traffic simulation and driver behaviour modelling : the archisim project**.

Fagin, R. (1995). **Reasoning about Knowledge**. MIT Press.

Farenc, N., Boulic, R., et Thalmann, D. (1999). **An informed environment dedicated to the simulation of virtual humans in urban context**. Dans *Proceedings of EUROGRAPHICS 99*, pages 309–318.

Farkas, I., Helbing, D., et Vicsek, T. (2002). **Mexican waves in an excitable medium.** *Nature*, 419(6903) :131–2.

Fellendorf, M., et Vortisch, P. (2010). **Microscopic traffic flow simulator Vissim**. Dans Barceló (2010), pages 63–94.

Ferber, J. (1995). **Les systèmes multi-agents. Vers une intelligence collective.**

Ferber, J., et Müller, J.-P. (1996). **Influences and reactions : a model of situated multiagent systems**. Dans Press, A., éditeur, *ICMAS'96 (International Conference on Multi-Agent Systems)*.

Finin, T., Fritzson, R., Mckay, D., et Mcentire, R. (1994). **Kqml as an agent communication language**. MIT Press.

FIPA (1998). **Agent communication language**. Rapport technique.

Fishwick, P. A. (1997). **Computer simulation : growth through extension**. *Trans. Soc. Comput. Simul. Int.*, 14(1) :13–23.

Franck, R. (2002). **Microeconomics and Behavior**. McGraw-Hill, New-York, 5th edition édition.

Fuchs, P., et Moreau, G., éditeurs (2006). **Le traité de la réalité virtuelle**. Presses de l'École des Mines, 3ème édition édition.

Galland, S., Buisson, J., Gaud, N., Gonçalves, M., Koukam, A., Guiot, F., et Henry, L. (2014a). **Agent-based simulation of drivers with the janus platform**. Dans *3nd International Workshop on Agent-based Mobility, Traffic and Transportation Models, Methodologies and Applications (ABMTRANS14)*, pages 738–743, Hasselt, Belgium. Elsevier. Procedia Computer Science, vol. 32.

Galland, S., et Gaud, N. (2014b). **Holonic model of a virtual 3D indoor environment for crowd simulation**. Dans *International Workshop on Environments for Multiagent Systems (E4MAS14)*. Springer.

Galland, S., Gaud, N., Demange, J., et Koukam, A. (2009). **Environment model for multiagent-based simulation of 3D urban systems**. Dans *the 7th European Conference on Multiagent Systems (EUMAS09)*, Ayia Napa, Cyprus. Paper 36.

Galland, S., Gaud, N., et Koukam, A. (2008). **Towards a multilevel simulation approach based on holonic multiagent systems**. Dans *the 10th International Conference on Computer Modeling and Simulation (EUROSIM-UKSIM 2008)*, pages 180–185. IEEE Computer Society.

Galland, S., Gaud, N., Rodriguez, S., Balbo, F., Picard, G., et Boissier, O. (2014c). **Contextualiser l'interaction entre agents en combinant dimensions sociale et physique au sein de l'environnement**. Dans *22èmes Journées Francophones sur les Systèmes Multi-Agents*, pages 65–74, Loriol-sur-Drôme, France. Cépaduès Éditions. Best Paper Award.

Galland, S., Lamotte, O., et Gaud, N. (2011). **MetroB : Evaluation and simulation of public transport system**. Dans *the Insitution of Engineering and Technology Conference on Smart and Sustainable City 2011 (ICSSC11)*, Shanghaï, China. Institute Engineering and Technology and Shanghai University, Shanghaï University Press.

Galland, S., Yasar, A.-U.-H., Knapen, L., Gaud, N., Janssens, D., Lamotte, O., Wets, G., et Koukam, A. (2014d). **Multi-agent simulation of individual mobility behavior in carpooling using the janus and jasim platforms**. *International Journal on Transport Research Part C*.

Gaud, N. (2007). **Holonic Multi-Agent Systems : From the analysis to the implementation. Metamodel, Methodology and Multilevel simulation**. PhD thesis, Université de Technologie de Belfort-Montbéliard, Belfort, France. In French.

Gaud, N., Galland, S., Gechter, F., Hilaire, V., et Koukam, A. (2008). **Holonic multilevel simulation of complex systems : Application to real-time pedestrians simulation in virtual urban environment**. *Simulation Modelling Practice and Theory*, 16(10) :1659–1676.

Gechter, F., Contet, J.-m., Lamotte, O., Galland, S., et Koukam, A. (2012). **Virtual intelligent vehicle urban simulator : Application to vehicle platoon evaluation**. *Simulation Modelling Practice and Theory (SIMPAT)*, 24 :103–114.

Genesereth, M. R., et Nilsson, N. J. (1987). **Logical foundations of artificial intelligence**. Morgan Kaufmann Publishers Inc.

Geramifard, A., Nayeri, P., Zamani-Nasab, R., et Habibi, J. (2005). **A hybrid three layer architecture for fire agent management in rescue simulation environment**. *International Journal of Advanced Robotic Systems*, 2(2) :111–116.

Ghods, A. H. G., et Saccomanno, F. (2011). **A micro-simulation overtaking gap acceptance model for rural two-lane highways**. *Canadian Institute of Transportation Engineers (CITE) Conference*.

Gibson, J. (1977). **Perceiving, Acting and Knowing**, chapitre The Theory of Affordances. Hillsdale, Erlbaum.

Gilbert, N., et Troitzsch, K. G. (2005). **Simulation for the Social Scientist**. Open University Press, Maidenhead and New York, 2 édition.

Gipps, P. (1981). **A behavioural car-following model for computer simulation**. *Transportation Research Part B : Methodological*, 15(2) :105 – 111.

Gipps, P. (1986). **A model for the structure of lane-changing decisions**. *Transportation Research Part B : Methodological*, 20(5) :403 – 414.

Godara, A., Lassarre, S., et Banos, A. (2007). **Simulating pedestrian-vehicle interaction in an urban network using cellular automata and multi-agent models**. Dans Schadschneider, A., Pöschel, T., Kühne, R., Schreckenberg, M., et Wolf, D., éditeurs, *Traffic and Granular Flow'05*, pages 411–418. Springer Berlin Heidelberg.

Goto, T., Kshirsagar, S., et Magnetat-Thalmann, N. (2001). **Automatic face cloning and animation**. *IEEE Signal Processing Magazine*, 18(3) :17–25.

Gottschalk, S., Lin, M., et Manocha, D. (1996). **OBBTree : A hierarchical structure for rapide interference detection**. Dans *Computer Graphics (SIGGRAPH96)*, pages 171–180.

Gouaïch, A., Michel, F., et Guiraud, Y. (2005). **MIC* : A deployment environment for autonomous agents**. Dans Weyns, D., van Dyke Parunak, H., et Michel, F., éditeurs, *First International Workshop on Environments for Multiagent Systems (E4MAS)*, numéro 3374 in Lecture Notes In Artificial Intelligence, pages 109–126, Berlin, Germany. Springer-Verlag.

Guttman, A. (1984). **R-trees : A dynamic index structure for spatial searching**. *SIGMOD Rec.*, 14(2) :47–57.

Hareesh, P. (2000). **Evacuation simulation : Visualisation using virtual humans in a distributed multi-user immersive vr system**. Dans *VSMM'00*.

Helbing, D., Buzna, L., Johansson, A., et Werner, T. (2005). **Self-organized pedestrian crowd dynamics : Experiments, simulations, and design solutions**. *Transportation Science*, 39(1) :1–24.

Helbing, D., Farkas, I., et Vicsek, T. (2000). **Simulating dynamical features of escape panic**. *Nature*, 407 :487–490.

Helbing, D., et Johansson, A. (2009). **Pedestrian, crowd and evacuation dynamics**. Dans Meyers, R. A., éditeur, *Encyclopedia of Complexity and Systems Science*, pages 6476–6495. Springer New York.

Helbing, D., et Molnar, P. (1995). **Social force model for pedestrian dynamics**. *Physical Review Part E*, 51 :4282–4286.

Helbing, D., et Treiber, M. (1998). **Gas-kinetic-based traffic model explaining observed hysteretic phase transition**. *Phys. Rev. Lett.*, 81(14) :3042–3045.

Helleboogh, A., Vizzari, G., Uhrmacher, A., et Michel, F. (2007). **Modeling dynamic environments in multi-agent simulation**. *Autonomous Agents and Multi-Agent Systems*, 14 :87–116.

Henderson-Sellers, B., et Giorgini, P., éditeurs (2005). **Agent-Oriented Methodologies**. Idea Group publishing.

Hodgins, J., et Brogan, D. (1994). **Robot herds : group behaviors for systems with significant dynamics**. Dans *Artificial Live IV*, pages 319–324.

Honglun, H., Shouqian, S., et Yunhe, P. (2007). **Research on virtual human in ergonomic simulation**. *Computers & Industrial Engineering*, 53(1) :350–356.

Hoogendoorn, S., et Body, P. (2002). **Normative pedestrian behaviour theory and modelling**. Dans *15th International Symposium on Transportation and Traffic Theory*, pages 219–245, Adelaide, Australia.

Hoogendoorn, S., et Bovy, P. (2001). **State-of-the-art of vehicular traffic flow modelling**. *Journal of Systems and Control Engineering, Special Issue on Road Traffic Modelling and Control*, 215(4) :283–304.

Illenberger, J., Flötteröd, G., et Nagel, K. (2007). **Enhancing MATSim with capabilities of within-day re-planning**. Dans *IEEE Intelligent Transportation Systems Conference*, pages 94–99, Seattle, WA, USA.

Jager, W., Popping, R., et van de Sande, H. (2001). **Clustering and fighting in two-party crowds : Simulating the approach-avoidance conflict**. *Journal of Artificial Societies and Social Simulation*, 4(3).

Jarras, I., et Chaib-draa, B. (2002). **Aperçu sur les systèmes multiagents**.

Jennings, N. (2001). **An agent-based approach for building complex software systems**. *Communications of the ACM*, 44(4) :35–41.

Jennings, N. R. (2000). **On agent-based software engineering**. *Artif. Intell.*, 117 :277–296.

Jennings, N. R., et Wooldridge, M. J. (1998). **Applications of intelligent agents**. *Agent Technology : Foundations, Applications and Markets*.

Jian, L., Lizhong, Y., et Daoliang, Z. (2005). **Simulation of bi-direction pedestrian movement in corridor**. *Physica A : Statistical Mechanics and its Applications*, 354(0) :619 – 628.

Jorissen, P., Wijnants, M., et Lamotte, W. (2005). **Dynamic interactions in physically realistic collaborative virtual environments**. *IEEE Trans Vis Comput Graph*, 11(6) :649–660.

Kallman, M., et Thalmann, D. (1998). **Modeling objects for interaction tasks**. Dans *Eurographics Workshop on Animation and Simulation*, pages 73–86. Springer.

Karafyllidis, I., et Thanailakis, A. (1997). **A model for predicting forest fire spreading using cellular automata**. *Ecological Modelling*, 99(1) :87 – 97.

Karamouzas, I., Geraerts, R., et Overmars, M. (2009a). **Indicative routes for path planning and crowd simulation**. Dans *Proceedings of the 4th International Conference on Foundations of Digital Games*, FDG '09, pages 113–120, New York, NY, USA. ACM.

Karamouzas, I., Heil, P., van Beek, P., et Overmars, M. (2009b). **A predictive collision avoidance model for pedestrian simulation**. Dans Egges, A., Geraerts, R., et Overmars, M., éditeurs, *Motion in Games*, volume 5884 de *Lecture Notes in Computer Science*, pages 41–52. Springer Berlin / Heidelberg.

Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., et Osawa, E. (1997). **RoboCup : The robot world cup initiative**. Dans Johnson, W. L., et Hayes-Roth, B., éditeurs, *Proceedings of the First International Conference on Autonomous Agents (Agents)*, pages 340–347, New York. ACM Press.

Klosowski, J., Held, M., Mitchell, J., Sowizral, H., et Zikan, K. (1998). **Efficient collision detection using bounding volume hierarchies of k-dops**. *Visualization and Computer Graphics, IEEE Transactions on*, 4(1) :21–36.

Klüpfel, H., Meyer-König, M., Wahle, J., et Schreckenberg, M. (2000). **Microscopic simulation of evacuation processes on passenger ships**. *Theoretical and Practical Issues on Cellular Automata*, pages 63–71.

Knapen, L., Bellemans, T., Janssens, D., et Wets, G. (2014). **Canonic route splitting**. *Procedia Computer Science*, 32(0) :309 – 316. The 5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014), the 4th International Conference on Sustainable Energy Information Technology (SEIT-2014).

Koenig, S., et Likhachev, M. (2005). **Fast replanning for navigation in unknown terrain**. *Robotics, IEEE Transactions on*, 21(3) :354–363.

Kothuri, R. K. V., Ravada, S., et Abugov, D. (2002). **Quadtree and r-tree indexes in oracle spatial : A comparison using gis data**. Dans *Proceedings of the 2002 ACM SIGMOD International Conference on Management of Data*, SIGMOD '02, pages 546–557, New York, NY, USA. ACM.

Krajzewicz, D., Erdmann, J., Behrisch, M., et Bieker, L. (2012). **Recent development and applications of SUMO - Simulation of Urban MObility**. *International Journal On Advances in Systems and Measurements*, 5(3–4) :128–138.

Krauß, S. (1998). **Microscopic Modeling of Traffic Flow : Investigation of Collision Free Vehicle Dynamics**. PhD thesis, Mathematisch–Naturwissenschaftlichen Fakultät, Universit‘”at zu Köln, Köln, Germany.

Ksontini, F. (2013). **Affordance based agent model : application to the traffic simulation**. Theses, Université de Valenciennes et du Hainaut-Cambresis.

Kuwahara, M., Horiguchi, R., et Hanabusa, H. (2010). **Traffic simulation with AVENUE**. Dans Barceló (2010), pages 95–129.

Lamarche, F., et Donikian, S. (2004). **Crowds of Virtual Humans : a New Approach for Real Time Navigation in Complex and Structured Environments**. *Computer Graphics Forum (Proc. of Eurographics 2004)*, 23(3) :509–518.

Lamotte, O., Galland, S., Contet, J.-m., et Gechter, F. (2010). **Submicroscopic and physics simulation of autonomous and intelligent vehicles in virtual reality**. Dans *2nd International Conference on Advances in System Simulation (SIMUL10)*, Nice, France. IEEE CPS.

Landsberg, H. (1981). **The Urban Climate**. International Geophysics. Elsevier Science.

Liebemann, E., Meder, K., Schuh, J., et Nenninger, G. (2005). **Safety and performance enhancement : The bosch electronic stability control (esp)**. *Proceedings : International Technical Conference on the Enhanced Safety of Vehicles*, 2005 :9p–9p.

Liu, F., Gehan, A. J. A., Nick, C., et Ahmed, S. (2012). **An experimental study on engine dynamics model based in-cylinder pressure estimation**. Rapport technique, SAE International.

Liu, R. (2010). **Traffic simulation with DRACULA**. Dans Barceló (2010), pages 295–322.

Liu, R., Vliet, van, D., et Watling, D. (2006). **Microsimulation models incorporating both demand and supply dynamics**. *Transportation Research Part A : Policy and Practice*, 40 :125–150.

Lu, J., Messih, D., et Salib, A. (2007). **Roll rate based stability control - the roll stability control system**.

Lugrin, J., et Cavazza, M. (2007). **Making sense of virtual environments : action representation, grounding and common sense**. Dans *International Conference on Intelligent User Interfaces*, pages 225–234, Honolulu, Hawaii, USA.

Lützenberger, M. (2014). **Cause And Effect - An Agent Based Approach to Simulate Strategic Level Driver Behavior**. PhD thesis, Elektrotechnik und Informatik der Technischen Universitë, Berlin, Germany.

Lämmel, G., Seyfried, A., et Streffen, B. (2013). **Large-scale and microscopic : a fast simulation approach for urban areas**. Dans *Transportation Research Board*. Washington, D.C.

Macal, C. M., et North, M. J. (2005). **Tutorial on agent-based modeling and simulation**. Dans *Proceedings of the 37th conference on Winter simulation*, WSC '05, pages 2–15. Winter Simulation Conference.

Manovich, L. (2005). **The poetics of urban media surfaces**. *First Monday, Special Issue #4 : Urban Screens : Discovering the potential of outdoor screens for urban society*.

Mathieu, P., Picault, S., et Seq, Y. (2014). **Les environnements : en avoir ou pas ? formalisation du concept et patterns d'implémentation**. Dans *22èmes Journées Francophones sur les Systèmes Multi-Agents*, pages 55–64, Loriol-sur-Drôme, France. Cépaduès Éditions.

Maurel, W., et Thalmann, D. (1998). **Human shoulder modeling including scapulo-thoracic constraint and joint sinus cones**. *Computers and Graphics*, 24 :203–218.

Michel, F. (2001). **Le modèle influence/réaction pour la simulation multi-agents**. Dans *Premières Journées Francophones Modèles Formels de l'Interaction (MFI)*, Toulouse, France.

Michel, F. (2004). **Formalism, tools and methodological elements for the modeling and simulation of multi-agents systems**. PhD thesis, LIRMM, Montpellier, France.

Michel, F. (2006). **Le modèle irm4s : le principe influence/réaction pour la simulation de systèmes multi-agents**. Dans *Journées Francophones sur les Systèmes Multi-Agents (JFSMA)*.

Michel, F. (2007). **The IRM4S model : the influence/reaction principle for multiagent based simulation**. ACM.

Musse, S., et Thalmann, D. (2001). **A hierarchical model for real time simulation of virtual human crowds**. *IEEE Transactions on Visualization and Computer Graphics*, 7(2) :152–164.

Niazi, M., et Hussain, A. (2011). **Agent-based computing from multi-agent systems to agent-based models : a visual survey**. *Scientometrics*, 89(2) :479–499.

Niederberger, C., et Gross, M. (2003). **Hierarchical and heterogenous reactive agents for real-time simulation**. *Computer Graphics Forum*, 22(3).

O'Brien, J., Julien, P., et Fullerton, W. (1993). **Two-dimensional water flood and mud-flow simulation**. *Journal of Hydraulic Engineering*, 119(2) :244–261.

Odell, J. J., Van Dyke Parunak, H., Fleischer, M., et Brueckner, S. (2002). **Modeling agents and their environment**. Dans *Proceedings of the 3rd international conference on Agent-oriented software engineering III*, AOSE'02, pages 16–31, Berlin, Heidelberg. Springer-Verlag.

O'Flaherty, C. A. (1986). **Highways. Volume 1. Traffic planning and engineering**.

Okazaki, S., et Matshushita, S. (1993). **A study of simulation model for pedestrian movement with evacuation and queuing**. Dans *International Conference on Engineering for Crowd Safety*.

Okuyama, F., Bordini, R., et Rocha Costa, A. C. (2005). **ELMS : An environment description language for multi-agent simulations**. Dans Weyns, D., van Dyke Parunak, H., et Michel, F., éditeurs, *First International Workshop on Environments for Multiagent Systems (E4MAS)*, numéro 3374 in Lecture Notes In Artificial Intelligence, pages 67–83, Berlin, Germany. Springer-Verlag.

Olstam, J. (2005). **Simulation of rural road traffic for driving simulators**. *Proceedings of the 84th Annual meeting of the Transportation Research Board, Washington D.C., USA*.

Omicini, A., et Zambonelli, F. (1999). **Coordination for internet application development**. *Autonomous Agents and Multi-Agent Systems*, 2(3) :251–269.

Omicini, A., Zambonelli, F., Klusch, M., et Tolksdorf, R., éditeurs (2001). **Coordination of Internet Agents : Models, Technologies, and Applications**. Springer-Verlag.

Oprea, M. (2012). **Agent-based modeling of an air quality monitoring and analysis system for urban regions**. Dans Iliadis, L., Maglogiannis, I., Papadopoulos, H., Karatzas, K., et Sioutas, S., éditeurs, *Artificial Intelligence Applications and Innovations*, volume 382 de *IFIP Advances in Information and Communication Technology*, pages 371–379. Springer Berlin Heidelberg.

Orkin, J. (2003). **Applying goal-oriented action planning to games**. Dans Rabin, S., éditeur, *AI Game Programming Wisdom 2*, pages 217–228. Charles River Media.

Owen, M., Galea, E., P., L., et Filippidis, L. (1998). **The numerical simulation of aircraft evacuation and its application to aircraft design and certification**. *Aeronautical Journal*, 102 :301–312.

P., M., et Starke, J. (2001). **Control of distributed autonomous robotic systems using principles of pattern formation in nature and pedestrian behavior**. *IEEE Transactions on Systems, Man and Cybernetics B*, 31(3) :433–436.

Paiva, D., Vieira, R., et S.R., M. (2005). **Ontology-based crowd simulation for normal life situations**. Dans *International Conference on Computer Graphics 2005*.

Pajarola, R., et Gobbetti, E. (2007). **Survey of semi-regular multiresolution models for interactive terrain rendering**. *The Visual Computer*, 23 :583–605.

Papadimitriou, E., Yannis, G., et Golias, J. (2009). **A critical assessment of pedestrian behaviour models**. *Transportation Research Part F : Traffic Psychology and Behaviour*, 12(3) :242 – 255.

Paris, S., Donikian, S., et Bonvalet, N. (2005). **Towards more realistic and efficient virtual environment description and usage**. Dans *First International Workshop on Crowd Simulation (V-Crowds' 05)*.

Paris, S., Donikian, S., et Bonvalet, N. (2007a). **Vers une architecture pour la simulation microscopique de foule**. *Revue Electronique Francophone d'Informatique Graphique*, 1(1) :33–43.

Paris, S., Pettré, J., et Donikian, S. (2007b). **Pedestrian reactive navigation for crowd simulation : a predictive approach**. *Computer Graphics Foru*, 26 :665–674.

Parunak, H. (2003). **Making swarming happen**. Dans *Conf. on Swarming and Network Enabled Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance (C4ISR)*, McLean, Virginia, USA.

Paterson, M., et Yao, F. (1990). **Efficient binary space partitions for hidden-surface removal and solid modeling**. *Discrete & Computational Geometry*, 5 :485–503. 10.1007/BF02187806.

Pelechano, N., Allbeck, J. M., et Badler, N. I. (2007). **Controlling individual agents in high-density crowd simulation**. Dans *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, SCA '07, pages 99–108, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

Penn, A., et Turner, A. (2001). **Space syntax based agent simulation**. Dans Schreckenberg, M., et S., S., éditeurs, *Pedestrian and Evacuation Dynamics*. Springer.

Peters, C., Dobbyn, S., Mac Namee, B., et O'Sullivan, C. (2003). **Smart objects for attentive agents**. Dans *WSCG'2003*, Plzen, Czech Republic.

Platon, E., Sabouret, N., et Honiden, S. (2006). **Environmental support for tag interactions**. Dans Weyns, D., van Dyke Parunak, H., et Michel, F., éditeurs, *Third International Workshop on Environments for Multiagent Systems (E4MAS)*, numéro 4389 in Lecture Notes in Computer Science, pages 106–123, Hakodate, Japan. Springer-Verlag.

Razavi, S. N., Gaud, N., Koukam, A., et Mozayani, N. (2011a). **Using motion levels of detail in the fast multipole method for simulation of large particle systems**. Dans *the 15th World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI)*, Orlando, Florida, USA. International Institute of Informatics and Cybernetics.

Razavi, S. N., Gaud, N., Mozayani, N., et Koukam, A. (2011b). **Multi-agent based simulations using fast multipole method : Application to large scale simulations of flocking dynamical systems**. *Artificial Intelligence Review*, 35(1) :53–72.

Reynolds, C. (1999). **Steering behaviors for autonomous characters**. Dans *Proceedings of the Game Developers Conference*, page 763–782.

Reynolds, C. W. (1987). **Flocks, herds and schools : A distributed behavioral model**. Dans *SIGGRAPH '87 : Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 25–34, New York, NY, USA. ACM.

Ricci, A., Omicini, A., et Denti, E. (2003). **Activity theory as a framework for mas coordination**. *LNCS on Engineering Societies in the Agents World III*, 2577 :96–110.

Ricci, A., Viroli, M., et Omicini, A. (2005). **Environment-based coordination through coordination artifacts**. Dans Weyns, D., Parunak, H. V. D., et Michel, F., éditeurs, *Environments for Multi-Agent Systems*, volume 3374 de *Lecture Notes in Computer Science*, pages 190–214. Springer Berlin / Heidelberg.

Ricci, A., Viroli, M., et Omicini, A. (2007). **Carta go : A framework for prototyping artifact-based environments in mas**. Dans Weyns, D., Parunak, H., et Michel, F., éditeurs, *Environments for Multi-Agent Systems III*, volume 4389 de *Lecture Notes in Computer Science*, pages 67–86. Springer Berlin Heidelberg.

Rincon, J., Garcia, E., Julian, V., et Carrascosa, C. (2014). **Developing Adaptive Agents Situated in Intelligent Virtual Environments**. Dans *International Conference on Hybrid Artificial Intelligence Systems*, pages 98–109. Springer.

Robbins, C. (1999). **Computer simulation of crowd behaviour and evacuation**. *ECMI Newsletter*, 25.

Robinson, S., Nance, R. E., Paul, R. J., Pidd, M., et Taylor, S. J. (2004). **Simulation model reuse : definitions, benefits and obstacles**. *Simulation Modelling Practice and Theory*, 12 :479–494.

Rubinstein, R. Y., et Kroese, D. P. (2011). **Simulation and the Monte Carlo Method**. Wiley & Sons, 2nd edition édition.

Russell, S. J., et Norvig, P. (1995). **Artificial Intelligence : A Modern Approach**. Prentice Hall, (second edition 2003), 1st édition.

S.A., O. (2010). **CALLAS & PROSPER, professionnal car dynamics simulation**. http://callasprosper.com/.

Samet, H. (1987). **Recent developments in linear quadtree-based geographic information systems**. *Image and Vision Computing*, 5(3) :187–197.

Saunier, J. (2014). **Bridging the gap between agent and environment : the missing body**. Dans *International Workshop on Environments for Multiagent Systems (E4MAS14)*, Paris, France. IFAAMAS, Springer.

Saunier, J., et Jones, H. (2014). **Mixed agent/social dynamics for emotion computation**. Dans *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 645–652. International Foundation for Autonomous Agents and Multiagent Systems.

Sauter, J., Matthews, R., Parunak, V., et Brueckner, S. (2002). **Evolving adaptive pheromone path planning mechanisms**. Dans *First International Conference on Autonomous Agents and Multi-agent Systems*, pages 434–441, Bologna, Italy.

Schaefer, L. A., Mackulak, G. T., Cochran, J., et Cherilla, J. L. (1998). **Application of a general particle system model to movement of pedestrians and vehicles**. Dans *WSC '98 : Proceedings of the 30th conference on Winter simulation*, pages 1155–1160, Los Alamitos, CA, USA. IEEE Computer Society Press.

Seo, H., Yahia-Cherif, L., Goto, T., et Magnetat-Thalmann, N. (2002). **Genesis : Generation of e-population based on statistical information**. Dans *Computer Animation '02*. IEEE Press.

Shagam, J. (2003). **Dynamic spatial partitioning for real-time visibility determination**. Master's thesis, New Mexico State University, Department of Computer Science.

Shagam, J., Pfeiffer, J., et Jr. (2003). **Dynamic irregular octrees**.

Shannon, R. E. (1977). **Simulation modeling and methodology**. *SIGSIM Simul. Dig.*, 8(3) :33–38.

Shao, W., et et al. (2005). **Environmental modeling for autonomous virtual pedestrians**.

Shao, W., et Terzopoulos, D. (2007). **Autonomous pedestrians**. *Graphical Models*, 69(5-6) :246–274. Special Issue on SCA 2005.

Shiller, Z. (2001). **Motion planning in dynamic environments : obstacles moving along arbitrary trajectories**. Dans *IEEE International Conference on Robotics and Automation*, volume 4, pages 3716–3721.

Shoham, Y. (1994). **Agent oriented programming : An overview of the framework and summary of recent research**. Dans Masuch, M., et Pólos, L., éditeurs, *Knowledge Representation and Reasoning Under Uncertainty*, volume 808 de *Lecture Notes in Computer Science*, pages 123–129. Springer Berlin Heidelberg.

Simonin, O., et Ferber, J. (2000). **Modeling self satisfaction and altruism to handle action selection and reactive cooperation**. Dans *6th International Conference On the Simulation Of Adaptive Behavior (SAB 2000 volume 2)*, pages 314–323.

Southwell, R. (1952). **Relaxation Methods in Theoretical Physics : A Continuation of the Treatise Relaxation Methods in Engineering Science**. Numéro v. 1 dans The Oxford engineering science series. Clarendon Press.

Stentz, A. (1994). **Optimal and efficient path planning for partially-known environments**. Dans *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pages 3310–3317 vol.4.

Stentz, A. (1995). **The focussed D* algorithm for real-time replanning**.

Still, G. (2000). **Crowd Dynamics**. PhD thesis, Warwick University.

Stylianou, S., Fyrillas, M. M., et Chrysanthou, Y. (2004). **Scalable pedestrian simulation for virtual cities**. Dans *Proceedings of the ACM symposium on Virtual reality software and technology*, VRST '04, pages 65–72, New York, NY, USA. ACM.

Süli, E., et Mayers, D. (2003). **An Introduction to Numerical Analysis**. Cambridge University Press.

Sykes, P. (2010). **Traffic simulation with paramics**. Dans Barceló (2010), pages 131–172.

Szymanezyk, O., Dickinson, P., et Duckett, T. (2011). **Towards agent-based crowd simulation in airports using games technology**. Dans O'Shea, J., Nguyen, N. T., Crockett, K. A., Howlett, R. J., et Jain, L. C., éditeurs, *KES-AMSTA*, volume 6682 de *Lecture Notes in Computer Science*, pages 524–533. Springer.

Tecchia, F., Loscos, C., et Chrysanthou, Y. (2002). **Visualizing crowds in real-time**. *Computer Graphics Forum*, 21(4) :753–765.

Tecchia, F., Loscos, C., Conroy, R., et Chrysanthou, Y. (2001). **Agent behaviour simulator (abs) : a platform for urban behaviour development**. Dans *In GTEC'2001*, pages 17–21.

Thalmann, D., Grillon, H., Maim, J., et Yersin, B. (2009). **Challenges in crowd simulation**. Dans *CyberWorlds, 2009. CW '09. International Conference on*, pages 1–12.

Thalmann, D., et Musse, S. R. (2013). **Crowd simulation**. Springer, London, 2nd edition édition.

Thiebaux, M., Marsella, S., Marshall, A., et Kallmann, M. (2008). **Smartbody : Behavior realization for embodied conversational agents**. Dans *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 1*, pages 151–158.

Thomas, G. (1999). **Environnements virtuels urbains : modélisation des informations nécessaires à la simulation de piétons**. PhD thesis, Informatique, Université de Rennes 1.

Thomas, G., et Donikian, S. (2000a). **Modelling virtual cities dedicated to behavioural animation**. *Computer Graphics Forum*, 19(3) :71–80.

Thomas, G., et Donikian, S. (2000b). **Virtual humans animation in informed urban environments**. Dans *CA '00 : Proceedings of the Computer Animation*, page 112, Washington, DC, USA. IEEE Computer Society.

Thompson, P., et E., M. (1995a). **A computer-model for the evacuation of large building population**. *Fire Safety Journal*, 24(2) :131–148.

Thompson, P., et Marchant, E. (1995b). **A computer-model for the evacuation of large building population**. *Fire Safety Journal*, 24(2) :131–148.

Tomás, V. R., et Garcia, L. A. (2005). **A cooperative multiagent system for traffic management and control**. Dans *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '05, pages 52–59, New York, NY, USA. ACM.

Treiber, M., Hennecke, A., et Helbing, D. (2000). **Congested traffic states in empirical observations and microscopic simulations**. *Phys. Rev. E*, 62 :1805–1824.

Treuille, A., Cooper, S., et Popović, Z. (2006). **Continuum crowds**. *ACM Trans. Graph.*, 25(3) :1160–1168.

Tu, X., et Terzopoulos, D. (1994). **Artificial fishes : Physics, locomotion, perception, behavior**. Dans *ACM SIGGRAPH Computer Graphics*, volume 28, pages 43–50, Orlando, FL, USA. ACM.

Tucker, C., Schweingruber, D., et McPhail, C. (1999). **Simulating arcs and rings in temporary gatherings**. *Int. Journal of Human-Computer Systems*, 50 :581–588.

Tungkastan, A., Jongsawat, N., et Premchaiswadi, W. (2008). **Using web portal for 3D grid-based rendering**. Dans *Sixt International Conference on ICT and Knowledge Management (ICT&KM2008)*.

Turner, A., et Penn, A. (2002). **Encoding natural movement as an agent-based system : An investigation into human pedestrian behaviour in the built environment**. *Environment and Planning B : Planning and Design*, 29 :473–490.

Ulicny, B., Heras Ciechomski, de, P., Musse, S., et Thalmann, D. (2006). **Populating virtual environments with crowd**. Dans *Eurographics 2006*.

Ulicny, B., et Thalmann, D. (2002). **Towards interactive real-time crowd behavior simulation**. *Computer Graphics Forum*, 21(4) :767–775.

Van den Berg, J., Lin, M., et Manocha, D. (2008). **Reciprocal velocity obstacles for real-time multi-agent navigation**. Dans *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 1928–1935.

Van Overveld, K., et Barenbrug, B. (1995). **All you need is force : a constraint-based approach for rigid body dynamics in computer animation**. Dans Terzopoulos, D., et Thalmann, D., éditeurs, *Computer Animation and Simulation '95*, Eurographics, pages 80–94. Springer Vienna.

Vanbergue, D. (2003a). **Conception de simulation multi-agents : application à la simulation des migrations intra-urbaines de la ville de Bogota**. PhD thesis, Université Paris VI.

Vanbergue, D. (2003b). **Conception de simulation multi-agents : application à la simulation des migrations intra-urbaines de la ville de Bogota**. PhD thesis, Université Paris VI.

Varner, D., Scott, D., Micheletti, J., et Aicella, G. (1998). **UMSC small unit leader non-lethal trainer**. Dans *ITEC'98*.

Viroli, M., et Omicini, A. (2003). **Coordination as a service : Ontological and formal foundation**. *FOCLASA 2002 – Foundations of Coordination Languages and Software Architecture, Electronic Notes in Theoretical Computer Science*, 68(3).

Wainer, G. A. (2009). **Discrete-Event Modeling and Simulation : A Practitioner's Approach**. Wiley & Sons.

Waizman, G., Shoval, S., et Benenson, I. (2003). **Micro-simulation model for assessing the risk of vehicle-pedestrian road accidents**. *Journal of Intelligent Transportation Systems*.

Wang, J., Liu, R., et Montgomery, F. (2005). **Car-following model for motorway traffic**. *Transportation Research Record : Journal of the Transportation Research Board*, 1934 :33–42.

Wein, R., Berg, J., et Halperin, D. (2005). **The visibility-voronoi complex and its applications**. Dans *Annual Symposium on Computational Geometry*, pages 63–72.

Weiss, G. (1999). **Multiagent Systems : A Modern Approach to Distributed Artificial Intelligence**, volume 3. MIT Press.

Wellstead, P. E., et Pettit, N. B. O. L. (1997). **Analysis and redesign of an antilock brake system controller**. *Control Theory and Applications, IEE Proceedings* -, 144(5) :413–426.

Werner, T., et Helbing, D. (2003). **The social force pedestrian model applied to real life scenarios**. Dans *Pedestrian and Evacuation Dynamics'03*.

Weyns, D., et Holvoet, T. (2004a). **Formal model for situated multi-agent systems**. *Formal Approaches for Multi-agent Systems, Special Issue of Fundamenta Informaticae*, 63(2-3). Eds. B. Dunin-Keplicz, R. Verbrugge.

Weyns, D., Parunak, H. V. D., Michel, F., Holvoet, T., et Ferber, J. (2005a). **Environment for multiagent systems state-of-the-art and research challenges**. Dans / Heidelberg, S. B., éditeur, *Environments for Multi-Agent Systems (E4MAS)*, pages 1–47.

Weyns, D., Schumacher, M., Ricci, A., Viroli, M., et Holvoet, T. (2005b). **Environments in multiagent systems**. *The Knowledge Engineering Review*, 20(2) :127–141.

Weyns, D., Steegmans, E., et Holvoet, T. (2004b). **Towards active perception in situated multi-agent systems**. *EUMAS, Special Issue of Journal on Applied Artificial Intelligence (AAI)*, 18(9-10) :867–883.

Wiedemann, R. (1974). **Simulation des strassenverkehrsflusses.**

Willemsen, P., Kearney, J., et Wang, H. (2003). **Ribbon networks for modeling navigable paths of autonomous agents in virtual urban environments**. Dans *Virtual Reality, 2003. Proceedings. IEEE*, pages 79–86.

Williams, J. (1995). **A Simulation Environment to Support Training for Large Scale Command and Control Tasks**. PhD thesis, University of Leeds.

Yeh, T. Y., Faloutsos, P., Patel, S. J., et Reinman, G. (2007). **Parallax : An architecture for real-time physics**. Dans *Proceedings of the 34th Annual International Symposium on Computer Architecture*, ISCA '07, pages 232–243, New York, NY, USA. ACM.

Yeh, T. Y.-H. (2007). **Architectural and algorithmic acceleration of real-time physics simulation in interactive entertainment**. PhD thesis, University OF California Los Angeles.

Yersin, B., Maim, J., Heras Ciechomski, de, P., Schertenleib, S., et Thalmann, D. (2005). **Steering a virtual crowd based on a semantically augmented navigation graph**. Dans *First International Workshop on Crowd Simulation (V-CROWDS'05)*, pages 169–178, Lausanne, Switzerland.

Zeigler, B. P., Kim, T. G., et Praehofer, H. (2000). **Theory of Modeling and Simulation**. Academic Press, 2nd edition édition.

Zhang, J.-M., Song, B.-Y., et Sun, G. (2008). **An advanced control method for abs fuzzy control system**. Dans *Intelligent Computation Technology and Automation (ICICTA), 2008 International Conference on*, volume 1, pages 845–849.

TABLE DES FIGURES

1 Plan de la thèse 5

1.1 Les quatre aspects d'un modèle de simulation multi-agent selon Michel (2004) 15

1.2 Architecture en couches d'un agent 17

1.3 Cycle d'exécution d'un système multi-agent durant une simulation (Galland et al., 2009) 18

1.4 Problème de simultanéité dans l'application des robots footballeurs (Michel, 2006) 25

1.5 Distinction entre l'esprit et le corps d'un agent dans le contexte du principe Influence/Réaction (Galland et al., 2014b) 26

2.1 Architecture de la plate-forme JaSim 33

3.1 Différentes vues sur l'environnement et modèles associés 48

3.2 Un environnement composé de deux couloirs parallèles connectés par une fenêtre, et la représentation des aspects navigation et perception du graphe associé 50

3.3 Diagramme de classes de l'environnement 53

3.4 Diagramme de classes des nœuds de l'environnement 54

3.5 Une spline cubique composée de deux portions 55

3.6 Les trois étapes de construction d'un couloir 56

3.7 Les connections entre couloirs garantissent la continuité dans le tracé par le partage des tangentes et des distances d'extrusion au niveau du point d'accroche 56

3.8 Structure générale d'un couloir. Les connecteurs *segment* sont illustrés en rouge, les connecteurs *ligne polygonale* en bleu 57

3.9 Modélisation de croisements de type "L", "T" et "X" à l'aide de jonctions et de couloirs 58

3.10 Un nœud de type place connecté à des couloirs par l'intermédiaire de connecteurs simples (en bleu) 59

3.11 Diagramme de classes des connecteurs de l'environnement 60

3.12 Intersection de deux couloirs et un connecteur de conflit illustré par son rectangle englobant (en rouge). Il n'est pas nécessaire d'ajouter un connecteur de conflit entre le couloir jaune et le couloir bleu car ils sont reliés au même connecteur *segment* 61

3.13 Diagramme de classes des liens de l'environnement 62

3.14 Une intersection en X dans une ville. Chaque véhicule entrant au niveau de l'intersection ne peut aller que tout droit ou tourner à droite. 63

3.15 Découpage de l'environnement en une collection de nœuds de type *couloir* et *jonction*. 63

3.16 Vue détaillée des nœuds routiers correspondant au découpage 3.14. 64

3.17 Le graphe correspondant au découpage 3.15. Les cercles représentent les *couloirs*, les carrés les *jonctions*. Les flèches noires représentent les couples de liens de navigation et de perception, les flèches rouges représentent les liens de perception uniquement. 64

3.18 Une route départementale au niveau d'un agrandissement de une à deux voies. 65

3.19 Découpage de l'environnement en une collection de nœuds de type *couloir*. 65

3.20 Le graphe correspondant au découpage 3.19. Les cercles en pointillés représentent les proxies de couloirs. Les flèches noires représentent les couples de liens de navigation et de perception, les flèches rouges représentent les liens de perception uniquement. 65

3.21 Un bâtiment composé de trois pièces ouvert sur une rue. 66

3.22 Découpage de l'environnement en une collection de nœuds de type *couloir*, *jonction* et *place*. 66

3.23 Le graphe correspondant au découpage 3.22. Les cercles représentent les *couloirs*, les carrés les *jonctions* et les triangles représentent les *places*. Les flèches noires représentent les couples de liens de navigation et de perception, les flèches rouges représentent les liens de perception uniquement et les flèches bleus sont les couples de liens de navigation et de perception réservés aux piétons pour la traversée de routes. 67

3.24 Diagramme de classes des entités de l'environnement 68

4.1 Les étapes de la détermination de la réaction de l'environnement basée sur les lois de la Physique. 73

4.2 Modélisation d'un corps de piéton à l'aide d'un cylindre indéformable. . . . 74

4.3 Modélisation d'un véhicule à l'aide d'un polyèdre indéformable. 75

4.4 Une modélisation incorrecte peut entraîner une localisation incomplète et un risque d'accident. 78

4.5 Cycle d'exécution de notre modèle durant une simulation 79

4.6 Les processus endogènes et leurs relations avec les autres éléments de HEDGE 81

5.1 Architecture des modules composant le logiciel Voxelia Simulate 85
5.2 Processus simplifié d'édition d'un graphe HEDGE 86
5.3 Architecture en couches des agents piétons et conducteurs de véhicule . . 89
5.4 Gestion de la trajectoire d'un piéton dans un couloir. 90
5.5 Gestion de la trajectoire d'un piéton dans une place. 90
5.6 Architecture du comportement des conducteurs de véhicules. 91
5.7 Estimation de la courbure d'un virage pour ajuster la vitesse d'un véhicule. 92
5.8 Représentation schématique du modèle physique de véhicule 95
5.9 Modèle pour le calcul de la trajectoire d'un véhicule dans un virage. 97
5.10 Aperçu du rendu 3D de la simulation au niveau du carrefour de la place Rabin. 99
5.11 Vue aérienne d'un carrefour de la place Rabin avant travaux. 101
5.12 Plan d'aménagement Optymo2. 101
5.13 Le modèle du carrefour correspondant au plan 5.12. Les nœuds réservés aux piétons sont représentés en vert, les routes en gris. 101
5.14 Le modèle Vissim du carrefour correspondant au plan 5.12. 101
5.15 Vue aérienne d'un carrefour du quartier Denfert-Thiers avant travaux. . . . 103
5.16 Plan d'aménagement Optymo2. 103
5.17 Le modèle du carrefour correspondant au plan 5.16. Les nœuds réservés aux piétons sont représentés en vert, les routes en gris. 103
5.18 Aperçu du rendu 3D de la simulation de la Place d'Armes. 104
5.19 Vue aérienne de la Place d'Armes avant travaux. 106
5.20 Plan d'aménagement de la place. 106
5.21 Le modèle du carrefour correspondant au plan 5.20. Les nœuds réservés aux piétons sont représentés en vert, les routes en gris. 106

B.1 Modèle pour le calcul de la trajectoire d'un véhicule dans un virage. 139

C.1 Exemple de résultats issus du modèle macroscopique. 142
C.2 Exemple de simulation microscopique. 143
C.3 Description générale 145
C.4 Détermination d'un obstacle 145

LISTE DES TABLES

2.1 Comparaison des propriétés des modèles d'environnement physique issus de la littérature 44

Liste des définitions

1 Définition : Agent (Ferber, 1995) 13
2 Définition : Système multi-agent (Ferber, 1995) 14
3 Définition : Environnement (Weyns et al., 2005a) 17
4 Définition : Environnement de communication (Odell et al., 2002) 20
5 Définition : Environnement social (Odell et al., 2002) 20
6 Définition : Environnement physique (Odell et al., 2002) 20

7 Définition : Lien de navigation 51
8 Définition : Potentiel de navigation 51
9 Définition : Lien de perception 51
10 Définition : Potentiel de perception 52
11 Définition : Nœud 53

A

Publications de l'auteur

Cette annexe présente les publications de l'auteur de cette thèse, classées par catégorie de publications.

A.1/ Journal international avec comité de lecture

Ahmane, M., Abbas turki, A., Perronnet, F., Wu, J., El moudni, A., Buisson, J., et Zéo, R. (2013). **Modeling and controlling an isolated urban intersection based on cooperative vehicles**. *Journal on Transport Research, Part C*. In press TRC-D-10-00018R3.

A.2/ Conférences internationales avec comité de lecture

Abbas turki, A., Perronnet, F., Buisson, J., El moudni, A., Ahmane, M., et Zéo, R. (2012). **Cooperative intersections for emerging mobility systems**. Dans *15th European Working Group on Transportation*, Paris, France.

Buisson, J., Galland, S., Gaud, N., Gonçalves, M., et Koukam, A. (2013). **Real-time collision avoidance for pedestrian and bicyclist simulation : a smooth and predictive approach**. Dans Yasar, A.-u.-h., et Knapen, L., éditeurs, *2nd International Workshop on Agent-based Mobility, Traffic and Transportation Models, Methodologies and Applications (ABMTRANS13)*, Halifax, Nova Scotia, Canada. Elsevier.

Buisson, J., Gaud, N., Galland, S., Gonçalves, M., et Koukam, A. (2014). **Toward an environment for the simulation of heterogeneous entities in virtual cities**. Dans *International Workshop on Environments for Multiagent Systems (E4MAS14)*, Paris, France. IFAAMAS, Springer.

Galland, S., Buisson, J., Gaud, N., Gonçalves, M., Koukam, A., Guiot, F., et Henry, L. (2014). **Agent-based simulation of drivers with the janus platform**. Dans *3nd International Workshop on Agent-based Mobility, Traffic and Transportation Models, Methodologies and Applications (ABMTRANS14)*, pages 738–743, Hasselt, Belgium. Elsevier. Procedia Computer Science, vol. 32.

Perronnet, F., Abbas turki, A., Buisson, J., El moudni, A., Zéo, R., et Ahmane, M. (2012). **Cooperative intersection management : Real implementation and feasibi-**

lity study of a sequence based protocol for urban applications. Dans *ITSC-2012*, pages 42–47.

A.3/ Conférences nationales sans comité de lecture

Buisson, J., Galland, S., Gonçalves, M., et Koukam, A. (2011). **Parallélisation sur GPU des modèles d'exécution d'un environnement situé 3D et d'agents virtuels. application à la simulation de foule**. Dans *1ères Journées des Jeunes Chercheurs de l'UTBM (IngéDoc 2011)*. UTBM, UTBM Press.

Buisson, J., Galland, S., Gonçalves, M., et Koukam, A. (2012). **Agent-based simulation of pedestrian and vehicles in 3D virtual cities**. Dans *2èmes Journées des Jeunes Chercheurs de l'UTBM (IngéDoc 2012)*. Doceo, UTBM, UTBM Press.

A.4/ Lecture Notes en cours d'évaluation

Buisson, J., Galland, S., Koukam, A., et Gonçalves, M. (2015). **Environment model for the microscopic simulation of mobility behaviors in cities**. *Lecture Notes in Computer Science*. en cours d'évaluation par le comité de lecture.

B

Modèle physique « simple » d'un véhicule

Cette annexe présente un modèle physique de véhicule simple alternatif à celui que nous présentons dans le chapitre 5.

Le modèle physique de véhicule que nous présentons ici consiste à déterminer l'orientation et la vitesse instantanée du véhicule au pas de temps suivant puis de forcer ces valeurs à la place de celles calculées par le moteur physique.

Lorsque l'orientation du volant σ est nulle, l'orientation du véhicule ne change pas et ce dernier se déplace alors en ligne droite. Dans le cas contraire, le véhicule se déplace d'un arc de cercle ont le rayon n'est autre que le rayon de courbure ρ_C que l'on peut calculer grâce à l'équation suivante :

$$\rho_C = \frac{\rho_{min}}{\sigma} \tag{B.1}$$

Où ρ_{min} est le rayon du cercle minimal que le véhicule peut former en tournant. Cette constante dépend, bien évidemment, des propriétés physiques du véhicule et fait partie des caractéristiques techniques très facilement accessibles.

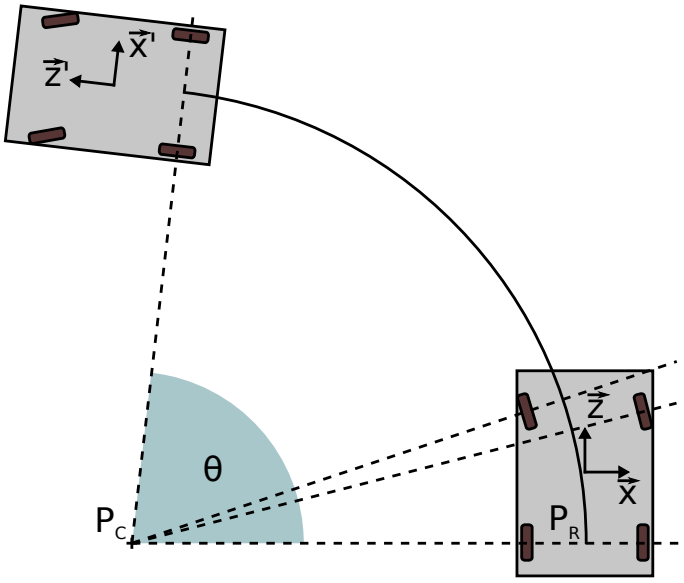

FIGURE B.1 – Modèle pour le calcul de la trajectoire d'un véhicule dans un virage.

Le problème est représenté par la figure B.1, page 139. Il nous faut à présent déterminer la distance d_C parcourue par le véhicule durant un laps de temps δt. Elle est calculée simplement grâce aux équations de la cinématique :

$$v(t) = \overrightarrow{v} \cdot \frac{\overrightarrow{z}}{||\overrightarrow{z}||} \tag{B.2}$$

$$a(t + \delta t) = \tau_A \alpha_{max} - \tau_B \beta_{max} \tag{B.3}$$

$$v(t + \delta t) = max(0, v(t) + \delta t a(t)) \tag{B.4}$$

$$d_C = \delta t \frac{v(t) + v(t + \delta t)}{2} \tag{B.5}$$

La projection du vecteur vitesse $\overrightarrow{v}$ du véhicule sur sa direction $\overrightarrow{z}$ dans l'équation B.2 permet d'obtenir la portion de ce vecteur vitesse qui est directement affectée par le moteur. L'équation B.3 permet de calculer l'accélération $a(t + \delta t)$ s'appliquant au véhicule sur la durée δt. L'équation B.4 permet de calculer la vitesse instantanée le long de la direction du véhicule $v(t + \delta t)$ à la fin du pas de temps, c'est à dire en appliquant l'accélération tout en s'assurant que cette valeur ne soit pas inférieure à 0. Nous ne considérons pas utile à nos simulation qu'un agent conducteur puisse de déplacer en marche arrière, cependant cette fonctionnalité peut facilement être ajoutée en altérant les équations précédentes.

Nous pouvons à présente calculer l'angle θ de l'arc de longueur d_C du cercle formé par le véhicule en tournant grâce à la relation suivante :

$$\theta = \frac{d_C}{\rho_C} \tag{B.6}$$

L'angle de rotation nous permet de déterminer la nouvelle orientation du véhicule et de modifier son vecteur vitesse. L'orientation est simplement modifiée en appliquant au solide une rotation d'angle θ autour de l'axe vertical du véhicule. Quant au vecteur vitesse du solide, il est remplacé par le vecteur :

$$\overrightarrow{v'} = \overrightarrow{v} + (v(t + \delta t) - v(t)) \frac{\overrightarrow{z'}}{||\overrightarrow{z'}||} \tag{B.7}$$

Où $\overrightarrow{z'}$ est la direction du véhicule après avoir subi une rotation d'angle θ.

Finalement, il ne reste qu'à configurer les propriétés du matériau physique du solide pour que celui puisse avancer et tourner, déraper lors d'un virage trop brusque, etc. Pour cela, il est important de configurer finement les coefficients de frottement statiques et dynamiques selon le modèle bidimensionnel de Coulomb lorsqu'il est supporté par le moteur de simulation physique. Il est également nécessaire de modifier le coefficient de résistance de l'air du solide afin que sa vitesse maximale corresponde approximativement à la valeur théorique fournit dans les spécifications du véhicule réel. Puisque le modèle que nous présentons ici pour simuler la dynamique du véhicule n'est pas physiquement correct, la détermination des différents coefficients ne peut se faire que de manière empirique.

C

Les projets d'aménagement de la ville de Belfort

Dans cette annexe, nous détaillons les deux premiers modèles de simulation utilisés dans le cadre des projets d'aménagement de la ville de Belfort et de sa communauté d'agglomérations présentés dans le chapitre 5. Nous détaillons ensuite le modèle d'évitement d'obstacles intégré au comportement des piétons et cyclistes utilisés dans le cadre de ces projets.

C.1/ MOSDAU

Ce modèle fournit l'estimation de la charge du trafic sur chaque tronçon de route. Il est basé sur le modèle classique de prévision de transport en quatre étapes (O'Flaherty, 1986) : génération du trafic, distribution, choix modal et affectation. Il s'agit d'affecter la demande sur les différents segments en fonction de leur capacité, des opportunités d'itinéraires et des temps de trajet en prenant en compte la charge de trafic.

L'avantage principal de cet outil réside dans le fait qu'il permet une vision globale du réseau et l'affectation du trafic en fonction de la charge. En d'autres termes, l'ensemble des itinéraires opportuns sont considérés, ce qui permet d'appréhender sur une échelle élargie les interrogations relatives aux mutations de déplacement induites par un projet. À ce niveau d'échelle, les mutations sont exprimées en termes de report de trafic estimé en comparant les charges du réseau induites par le projet avec celles qui résultent de la situation de référence, à savoir le scénario au fil de l'eau, comme sur la figure C.1.

À ce niveau d'échelle, il s'agit de relever l'impact des reports sur la régularité du trafic. Par exemple, il s'agit pour le trafic routier, de déduire les zones de congestion ou des zones qui connaissent des modifications sensibles de la charge du trafic. Ces zones nécessitent des réponses en termes d'approche de régulation devant être étudiées à des niveaux plus fins. Le tableau suivant met en exergue les avantages et les mises en garde relatifs au modèle macroscopique :

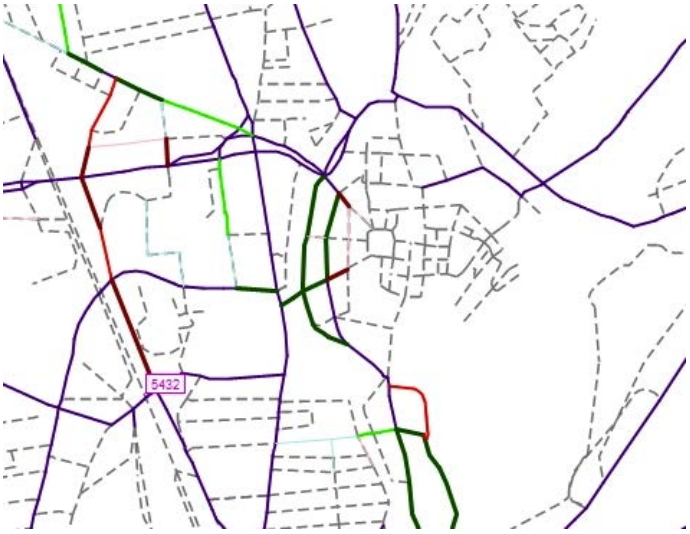

FIGURE C.1 – Exemple de résultats issus du modèle macroscopique.

Avantages	Mise en garde
Approche prouvée de la modélisation du trafic.	Modèle hérite des faiblesses de l'enquête OD et du zonage : plus le zonage est fin (surface couverte) mieux le modèle se comportera.
Vision globale du réseau.	Modèle hérite des faiblesses de la numérisation des routes qui détermine le choix de l'emplacement des connecteurs des zones sur le réseau.
Résultats facilement interprétables.	Modèle hérite des faiblesses des hypothèses des évolutions futures : par exemple, il est souvent difficile de prévoir les évolutions des commerces attractifs.

Ainsi la lecture du modèle doit être faite d'une manière globale sur les axes structurants (présents et futurs) du réseau étudié. Pour les demandes faibles, le problème de représentativité statistique se pose clairement. En outre, les lectures faites des espaces congestionnés doivent être étendues en amont et en aval. En effet, bien que le modèle dispose d'outils mathématiques pour considérer les remontées de file, l'affectation du trafic est réalisée sur un horizon de temps discret ; des segments discrets avec des capacités ajustées selon les états présents des boucles électromagnétiques. Il est donc difficile à travers le modèle à 4 étapes de localiser avec précisions les congestions. Nous parlerons plutôt de zones nodales. Ce sont ces zones qui méritent une attention particulière à travers la simulation microscopique, qui est détaillée dans la section suivante.

C.2/ MISDAU

Le modèle de simulation microscopique MISDAU reproduit les comportements de conduite (ou de déplacement piéton) en fonction des obstacles observés dans un environnement discret utilisant un modèle topologique basé sur un graphe (la définition de ce type d'environnement est détaillée dans le chapitre 2). Contrairement au modèle précédent où l'ensemble des individus (véhicules ou piétons) est modélisé sous la forme d'un flux, l'intérêt particulier de ce modèle est de permettre de simuler le trafic en considérant chaque individu (véhicule ou piéton) comme une entité individuelle qui se déplace en fonction de son environnement (voir figure C.2). Il s'agit donc d'affiner les résultats fournis par le modèle précédent en mettant en lumière les problèmes relatifs à la régulation :

- **Choix de la signalisation :** Nous pouvons simuler plusieurs politiques de régulation, comme les stops, les feux de signalisation avec des détails sur l'algorithme utilisé pour le calcul des temps de cycles et des phases.
- **Succession de carrefours, remontée de file et temps de trajets :** Nous pouvons observer les phénomènes de congestion et leurs origines avec précision. Par rapport au modèle précédent où il est préférable de parler de zone, ce modèle permet de mieux localiser les phénomènes. A titre d'exemple, des intersections en amont peuvent retenir le trafic à cause des ralentissements au niveau de la giration.
- **Changement dynamique d'itinéraire :** Nous pouvons analyser les comportements des conducteurs et modifier dynamiquement l'affectation des flux. En effet, souvent les remontées de files occasionnelles incitent les conducteurs à changer d'itinéraire.

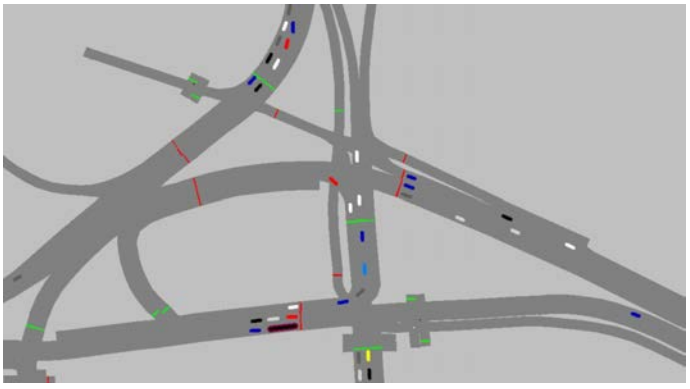

FIGURE C.2 – Exemple de simulation microscopique.

En permettant de tester et de confronter les choix des systèmes de régulation et de la signalisation correspondante, la simulation microscopique soulève clairement les problématiques relatives à ces choix de conception. Dans le cadre de ce projet, nous avons choisi de traiter ce niveau de simulation avec le logiciel **PTV Vissim**. En effet, ce logiciel est l'un des plus importants sur le marché. De plus, les services de la région Franche-Comté disposait déjà de licences pour ce logiciel.

Le tableau suivant met en exergue les avantages et les mises en garde relatifs à la simulation microscopique :

Avantages	Mise en garde
Approche éprouvée de la modélisation du trafic avec prise en compte des détails de la signalisation.	Modèle de simulations 1.5D qui nécessite l'introduction de plusieurs paramètres.
Vision localisée des risques d'accident et de la congestion.	Lecture des résultats parfois trompeuse dans le sens où la simulation microscopique, bien qu'elle semble réaliste, ne permet pas de relever tous les aspects liés à la topologie. Appréciation de la vitesse difficile en raison d'un manque d'informations de référence (mobilier urbain, végétation).

Il s'agit ici de poser clairement des stratégies de régulation et de visualiser le comportement des véhicules face à celles-ci. Néanmoins, il faut souligner que le modèle ne prend pas en compte les aspects topologiques. En outre, bien que les logiciels du commerce offre une option 3D, il convient de remarquer que cette option ne permet pas de prendre en compte des effets liés à la visibilité et aux difficultés d'insertion, d'où la nécessité de simuler le trafic avec un degré plus fin.

C.3/ Modèle d'évitement de collision des agents piétons et cyclistes

Dans cette section, nous détaillons le modèle d'évitement de collision intégré au comportement des agents piétons et cyclistes utilisés dans l'outil Simulate dans le cadre des projets présentés dans le chapitre 5.

Le modèle d'évitement de collision a pour objectif de déterminer la direction et l'accélération avec laquelle le piéton doit se déplacer. Celui que nous utilisons pour les piétons et cyclistes dans le cadre de ce projet implémente un modèle basé sur les champs de forces (Buisson et al., 2013). Dans ce modèle, chaque individu est modélisé par un volume convexe dans l'espace. Dans le cas d'une modélisation dans un espace 2D, comme c'est le cas pour ce projet, chaque individu a est modélisé par un disque de rayon r_a. L'individu a doit naviguer vers une position $\overrightarrow{p_t}$ en évitant les collisions avec tout obstacle environnant, statique ou bien mobile. Le piéton ou cycliste se déplace avec une accélération préférée w_a et une vitesse de préférée v_a^{des}. Dans la suite de ce document, p_a désigne la position du centre du volume englobant l'individu et v_a désigne son vecteur vitesse. La trajectoire instantanée de l'individu, représentée par la bande jaune pâle dans la figure C.3 est bornée par deux lignes $T_{left}(t)$ et $T_{right}(t)$, et le cercle orange représente la distance minimale de sécurité d_m que l'individu tente de conserver en permanence en lui et tout objet environnant ($d_m \geq r_a$).

Le modèle d'évitement de collision est basé sur l'hypothèse qu'un individu essayera toujours d'éviter un obstacle en minimisant l'énergie requise pour résoudre le conflit (principe du moindre effort). Minimiser l'énergie requiert un changement de direction aussi progressif que possible et doit par conséquent être initié le plus tôt possible tout en continuant à avancer vers l'objectif. Bien que ce principe soit simple à mettre en place pour des obstacles statiques, la tâche devient complexe dès que l'on introduit des obstacles

mobiles. Nous utilisons une approche prédictive pour détecter si un obstacle peut devenir un danger potentiel et estimer le temps avant impact. Ce temps est utilisé pour ajuster la magnitude de la force d'évitement correspondante.

Soit M l'ensemble des obstacles potentiels perçus par un individu. Le modèle d'évitement peut être généralement décrit par les équations suivantes :

$$\overrightarrow{F} = \sum_{i \in M} U(t_c^i) \cdot \hat{S}_i \qquad \text{(C.1)}$$

$$\overrightarrow{F_a} = \overrightarrow{F} + w_a.\delta_{\|\overrightarrow{F}\|} \frac{\overrightarrow{p_t} - \overrightarrow{p_a}}{\|\overrightarrow{p_t} - \overrightarrow{p_a}\|} \qquad \text{(C.2)}$$

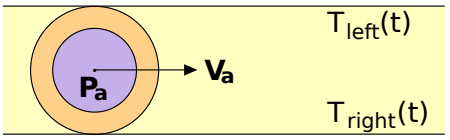

FIGURE C.3 – Description générale

$\overrightarrow{F_a}$ est la force devant être appliquée sur l'individu a pour se déplacer vers son objectif tout en évitant les obstacles de M. Il s'agit de la somme de toutes les forces F d'évitement calculées pour chaque élément de M. $\hat{S}_i$ est la direction de la force d'évitement. t_c^i et $U(t)$ sont respectivement le temps estimé avant collision et la fonction d'ajustement de la magnitude de la force en fonction du temps. Deux cas particuliers peuvent se présenter lors du calcul de $\overrightarrow{F_a}$: (i) l'individu ne perçoit aucun obstacle dans son champs de vision, (ii) les obstacles perçus sont trop loin pour contribuer à F. Afin d'éviter que l'individu reste immobile dans ces deux cas, $\overrightarrow{F_a}$ inclue une force d'attraction vers $\overrightarrow{p_t}$. Cette force n'est appliquée que lorsque F a une magnitude quasi nulle.

Détermination des obstacles Avant de calculer les forces d'évitement il est nécessaire de déterminer les obstacles qui présentent un danger dans le champ de vision du piéton ou du cycliste, c'est à dire ceux qui sont en intersection avec la trajectoire instantanée de l'individu. La détermination des obstacles est inspirée de Shiller (2001) : le volume englobant du mouvement de l'individu et d'abord calculé. Si ce volume est en intersection avec un obstacle potentiel, celui-ci est considéré comme un danger et une force d'évitement correspondante sera calculée. La figure C.4 illustre le processus de détermination d'obstacles en montrant l'individu comme un cercle et l'obstacle potentiel comme un rectangle.

Le volume englobant du mouvement est représenté par le rectangle jaune en pointillés. La détermination de ce volume est basée sur la vitesse relative Δv entre l'individu et l'obstacle potentiel. Elle est obtenue par $\Delta v = \overrightarrow{v_a} - \overrightarrow{v_j}$, où $\overrightarrow{v_a}$ et $\overrightarrow{v_j}$, où $\overrightarrow{v_a}$ et $\overrightarrow{v_j}$ sont respectivement les vecteurs vitesse de l'individu a et de l'obstacle potentiel j. Le volume englobant du mouvement est une boîte orientée définie par son centre c et ses deux directions $d1$ et $d2$, telles que définie par l'équation C.3. Le test d'intersection entre les deux volumes est donné par Bergen (2003); Eberly (2001); Gottschalk et al. (1996).

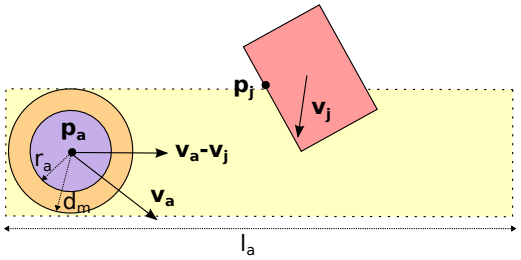

FIGURE C.4 – Détermination d'un obstacle

$$\begin{cases} c = p_a + \left(\frac{l_a}{2} - r_a\right).\widehat{\Delta v} \\ d_1 = r_a \widehat{\Delta v} \times \hat{y} \\ d_2 = \frac{l_a \widehat{\Delta v}}{2} \end{cases} \tag{C.3}$$

Force de glissement Le modèle d'évitement de collision introduit un nouveau type de force appelé force de glissement ou *sliding force*. Contrairement aux forces de répulsion classiquement employées par les modèles basés sur les champs de forces les plus répandus, la force de glissement n'a pas seulement pour objectif de dévier l'individu de la direction de l'obstacle mais aussi de le guider vers son objectif.

Les modèles classiques utilisent deux types de forces : une pour éviter la collision, l'autre pour la navigation. Nous considérons que la détermination des coefficients permettant d'ajuster leurs magnitudes relatives est une tâche très complexe. Elle nécessite un processus empirique long et très précis pour chaque topologie d'environnement que l'individu pourrait avoir à rencontrer. Puisque la force de glissement inclue ces deux types de forces, elle constitue une solution élégante de simplicité à ce problème. Afin de s'assurer que la trajectoire d'évitement d'une collision est aussi souple que possible, seules les forces qui sont absolument nécessaires sont considérées dans la détermination de la force d'évitement. En d'autres termes, seuls les obstacles représentant un danger avéré à court terme, c'est à dire en intersection avec la trajectoire de l'individu, doivent être étudiés. La détermination de la direction et l'ajustement de la magnitude de la force de glissement sont deux étapes distinctes dans ce modèle.

Détermination de la direction des forces La direction de la force de glissement est colinéaire au vecteur $\overrightarrow{s_j}$, lui-même perpendiculaire à la direction vers l'obstacle j et calculé par la relation suivante :

$$\overrightarrow{s_j} = (\overrightarrow{p_j} - \overrightarrow{p_a}) \times \hat{y} \tag{C.4}$$

Où $\overrightarrow{p_j}$ est le point sur l'obstacle le plus proche de l'individu tout en étant contenu dans la trajectoire de ce dernier comme illustré par la figure C.4. $\hat{y}$ est le vecteur vertical normalisé.

Puisque deux directions colinéaire à $\overrightarrow{s_j}$ sont possibles, $\hat{s}_j$ et $-\hat{s}_j$, celle permettant de minimiser le changement de direction et de se diriger vers l'objectif $\overrightarrow{p_t}$ est choisie.

$$\hat{S}_j = \mathrm{sgn}(\overrightarrow{s_j} \cdot (\overrightarrow{p_t} - \overrightarrow{p_a}))\frac{\overrightarrow{s_j}}{||\overrightarrow{s_j}||} \tag{C.5}$$

Ajustement de la magnitude des forces basé sur le temps Certains modèles d'évitement de collision basés sur les champs de forces utilisent une fonction monotone décroissante de la distance à l'obstacle pour ajuster la magnitude de la force d'évitement. Bien que cette approche soit évidente et parfois efficace, celle-ci ne prend pas en compte la vitesse de déplacement de l'individu. De plus, cette approche sous-estime le danger que peut représenter un obstacle mobile se dirigeant vers l'individu à une vitesse élevée.

Nous proposons une alternative se basant sur l'utilisation d'une fonction monotone décroissante du temps. Le temps t_c estimé avant impact avec un obstacle est utilisé comme paramètre d'une fonction U représentant la réponse de l'individu face à l'urgence d'une collision. Plus le temps avant impact est grand, moins la force d'évitement sera importante. Un ajustement des forces basé sur le temps permet de donner une priorité plus grande aux forces qui permettent d'éviter les collisions les plus immédiates. Nous proposons l'utilisation d'une fonction U définit par :

$$U(t_c) = \frac{\sigma}{{t_c}^{\phi}} - \frac{\sigma}{{t_{max}}^{\phi}} \tag{C.6}$$

Avec t_{max} le temps d'anticipation maximum. σ et ϕ sont les paramètres qui permettent d'ajuster la forme de la courbe en fonction des besoins de l'application. Le temps estimé avant impact est définit par la relation suivante :

$$t_c = \frac{||\overrightarrow{p_j} - \overrightarrow{p_a}|| - r_a}{||\Delta v||} \tag{C.7}$$

Document réalisé avec LaTeX et :
le style LaTeX pour Thèse de Doctorat créé par S. Galland — http://www.multiagent.fr/ThesisStyle
la collection de paquets `tex-upmethodology` — http://www.arakhne.org/tex-upmethodology/

Résumé :

La simulation constitue une approche majeure pour la conception, le développement, l'analyse et l'évolution des systèmes urbains. Dans le contexte de cette thèse, nous nous intéressons à la modélisation et à la simulation des déplacements de flux (piéton, véhicule, cycle) au sein d'environnements urbains. Le contexte industriel dans lequel s'inscrit également cette thèse impose de considérer les déplacements de chaque entité dans un univers modélisé en trois dimensions. Nous proposons d'aborder les nombreux défis scientifiques et technologiques en modélisant les flux sous la forme d'entités autonomes appelées agents. La simulation orientée-agent nécessite de définir trois mécanismes principaux : le comportement des agents, leurs interactions avec les autres agents et leurs interactions avec l'environnement dans lequel ils évoluent. Ce dernier modélise la structure du monde ainsi que sa dynamique endogène. Malheureusement, il reste difficile de trouver des modèles d'environnement répondant aux différentes problématiques scientifiques et technologiques abordées dans cette thèse. Par conséquent, nous proposons un modèle d'environnement, nommé HEDGE (HEterogenous Dual Graph Environment). Il est basé sur une décomposition de l'environnement en zones reliées par des liens de navigation et de perception. Cette structure de graphe est au centre de notre modèle. Elle a été conçue pour permettre une modélisation naturelle de l'environnement et une simulation efficace du système constitué par cet environnement et les agents. Les missions et les processus dynamiques de l'environnement sont modélisés à l'aide de mécanismes inspirés des lois de la Physique. Les actions fournies par les agents et par les algorithmes calculant la dynamique endogène de l'environnement sont des forces à appliquer aux objets du monde. Un modèle de détection et de résolution de conflits entre ces forces est utilisé afin de garantir un état cohérent du modèle de l'environnement. Dans la dernière partie de cette thèse, nous illustrons l'utilisation du modèle HEDGE dans le cadre de deux projets d'aménagement de la ville de Belfort et de sa communauté d'agglomérations Le produit de l'application de ce modèle est utilisé à des fins d'études techniques (études préliminaires, avant-projet), de communication (concertations et débats publics) et de promotion (expositions, valorisation) sous la forme de logiciels interactifs en 3D.

Mots-clés : Modèle d'environnement, Systèmes multi-agents, Simulation multi-agent, Environnement physique, Simulation de trafic, Simulation de foule, Aménagement de la ville de Belfort

Abstract:

Simulation constitutes a major approach for the conception, development, analysis and evolution of urban systems. This thesis focusses on the modeling and simulation of movements (pedestrian, vehicle, cycle) within urban environments. The industrial context associated with this thesis requires to consider the movements of each entity in a three dimensional universe. In this thesis, we propose to address the scientific and technological challenges by using autonomous entities, called agents, to model the individuals. In agent-based simulation three elements must be defined : the agent behaviors, their interactions with other agents, and with the environment. The environment models the structure of the world and its endogenous dynamics. Unfortunately, it is still difficult to find an environment model that answers all the different scientific and technological problems addressed in this thesis. Consequently, an environment model named HEterogenous Dual Graph Environment (or HEDGE) is proposed. The HEDGE model is based on a decomposition of the environment using zones that are linked with navigation and perception links. This graph structure is at the center of our model. It is designed to "naturally" model the environment, and it allows an efficient simulation of the system composed of the environment and the agents. The environment's missions and dynamic processes are modeled using mechanisms, which are inspired by the laws of Physics. The actions provided by the agents and the endogenous dynamics algorithms correspond to physical forces to apply to objects in the world. A model of conflict detection and resolution between those forces is used to ensure the coherence of the environment model state. In the last part of this thesis, the HEDGE model is used in the context of two urban planning projects in the city of Belfort, and its metropolitan area. The product of this model's application is used in the context of technical studies (preliminary studies, pre-project), communication (consultation and public debates) and promotion (exposition, valorization) in the form of interactive 3D software.

Keywords: Environment model, Multi-agent systems, Agent-oriented simulation, Physical environment, Traffic simulation, Crowd simulation, Belfort city urban planning

SPIM

■ École doctorale SPIM - Université de Technologie Belfort-Montbéliard
F - 90010 Belfort Cedex ■ tél. +33 (0)3 84 58 31 39
■ ed-spim@univ-fcomte.fr ■ www.ed-spim.univ-fcomte.fr

utbm
université de technologie Belfort-Montbéliard