

وزارة التعليم العالي و البحث العلمي

BADJI MOKHTAR-ANNABA UNIVERSITY

UNIVERSITE BADJI MOKHTAR-ANNABA



جامعة باجي مختار - عنابة

Année : 2015

Faculté des sciences de l'ingénieur  
Département d'informatique

## THESE

Présenté en vue d'obtention du diplôme de **DOCTORAT en Informatique**

### Conception d'un système multi-agents adaptatif pour la résolution de problème

Option  
Informatique

Par  
BENHAMZA KARIMA

#### Devant le jury

Président :

**SOUICI-MESLATI LABIBA** Pr. Univ. Badji Mokhtar- Annaba

Directeur de Thèse :

**SERIDI HAMID** Pr. Univ. 08 mai 1945- Guelma

Examinateurs :

**MEROUANI HAYET FARIDA** Pr. Univ. Badji Mokhtar- Annaba

**CHIKHI SALIM** Pr. Univ. Mentouri- Constantine

**BOUKROUCHE ABDELHANI** Pr. Univ. 08 mai 1945- Guelma

Année universitaire : 2014-2015

## Remerciements

*Je tiens à remercier Monsieur **Hamid Seridi**, Professeur à l'université de Guelma, qui à encadrer et diriger cette thèse. Je le remercie particulièrement pour la compétence, la disponibilité, le regard critique et la confiance avec lesquelles il m'a guidé tout au long de ce travail.*

*J'adresse mes vifs remerciements à Madame **Labiba Souici**, Professeur à l'université **Badji Mokhtar d'Annaba**, pour l'honneur qu'elle me fait en acceptant de présider le jury de cette thèse.*

*Je tiens à remercier sincèrement Madame **Hayet-Farida Merouani**, Professeur à l'université d'Annaba, Monsieur **Salim Chikhi**, Professeur à l'université de Constantine, Monsieur **A/Hani Boukrouche**, Professeur à l'université de Guelma, qui m'ont fait l'honneur de participer à ce jury et qui ont accepté de prendre de leur temps pour examiner ce travail*

*Je remercie, tous ceux qui m'ont accompagné au cours de toutes ces années, famille, amis et collègues, et qui m'entourent tous les jours de leur affection attentionnée.*

*A la mémoire de mon père, je dédie ce travail.*

# Résumé

L'environnement des applications existantes est de plus en plus complexe et dynamique, compte tenu de la diversité et du grand nombre d'éléments qui le compose. Les fonctions de tels systèmes deviennent alors de plus en plus difficiles à définir, et leur spécification est souvent incomplète, même si ses parties sont spécifiées.

Les nouvelles techniques de conception des systèmes multi-agents tentent de construire des systèmes artificiels dans lesquels l'activité collective observée n'est décrite dans aucune de leurs parties. Ces approches sont particulièrement utiles dans les systèmes distribués, dynamiques, ouverts et qui sont difficilement formalisables. Cependant, le lien existant entre le comportement collectif et celui des individus qui composent ce système n'est pas aisément établi. Dans le cadre de la résolution de problèmes, obtenir le comportement collectif cohérent, alors que les agents sont simples et ne cherchent qu'à atteindre un objectif individuel, est donc également difficile.

De ce fait, l'objectif de cette recherche vise à répondre à cette problématique essentielle dans les systèmes ouverts et dynamiques : la mise en place de mécanismes auto-organisationnels permettant l'émergence de solutions cohérentes et adaptatives. Ceci permet de garantir des propriétés macroscopiques du système conçu en ne développant que des entités et leurs comportements à un niveau élémentaire microscopique. Pour y parvenir, nous proposons tout d'abord de modéliser la dynamique globale du système multi-agents sous forme de contraintes environnementales et le comportement des agents sous forme d'action de contrôle et d'adaptation. Ainsi, ce processus est construit aux niveaux individuel et collectif. Cette approche de conception a été reconduite sur le problème complexe de contrôle de signalisation dans un réseau routier.

**Mots clés** : Système multi-agents, auto-organisation, émergence, adaptation, résolution de problème, trafic routier, contrôle de signalisation.

# Abstract

Given the diversity and the large number of component parts, existing applications environment is becoming more complex and dynamic. So, the systems functions become increasingly difficult to define, and their specification is often incomplete, even though its elements are detailed.

The new technical design of multi-agent systems attempt to build artificial systems in which the observed collective activity is disclosed in any of their parts.

The technics of self-organized multi-agent systems can build artificial systems in which observed collective activity is described in any of their parts. This approach is particularly useful in distributed, dynamic and open systems which are difficult to formulate. However, the existing relation between collective behavior and individuals composing this system is difficult to find. Getting the desired collective behavior of the system while the agents only seek to achieve an individual objective is also difficult.

Therefore, the objective of this research is to address this main problematic in open and dynamic systems: the development of self-organizational mechanisms that allow the emergence of coherent and adaptive solutions. This ensures the macroscopic properties of the system designed by developing simple behaviors of entities at a microscopic level.

To achieve this objective, we propose to model the global dynamics of SMA as environmental constraints and agent behavior as control and adaptation actions. Thus, this process is built at individual and collective levels. This approach was applied in modeling and solving complex problem of traffic signals control in multiple intersections network.

**Keywords:** Multi-agent system, self-organization, emergence, adaptation, problem solving, traffic, traffic control.

## ملخص

بيئة التطبيقات الموجودة حالياً أصبحت أكثر تعقيداً وديناميكية، نظراً للعدد الكبير من الأجزاء المكونة لها على اختلافهم. وبالتالي فإنه أصبح من الصعب تحديد وظائف هذه النظم و مواصفاتها التي غالباً ما تكون ناقصة على نحو متزايد بالرغم من تحديد لخصائص أجزائها.

التقنيات الجديدة لتصميم النظم متعددة الوكالء تحاول بناء أنظمة اصطناعية التي لا يتم وصف النشاط الجماعي في أي من أجزائها. هذه الطرق مفيدة بشكل خاص في النظم الموزعة وديناميكية والمنفتحة والتي هي صعبة لإضفاء الطابع الرسمي عليها. فليس من السهل إقامة الرابط الموجود بين السلوك الجماعي والأفراد الذين يشكلون هذا النظام. فمن الصعب الحصول على السلوك الجماعي المتماسك، في حين أن الوكيل المشكل له بسيط ويسعى فقط لتحقيق هدفه الفردي.

ولذلك، وفي سياق حل مشكلة، فإن الهدف من هذا البحث هو معالجة هذه الإشكالية الرئيسية في النظم المفتوحة والديناميكية بتطوير آليات التنظيمي الذاتي التي تسمح بظهور حلول متماسكة وقابلة للتكييف. وهذا يضمن الخصائص المايكروسوبية للنظام مصمم من خلال تطوير سلوكيات بسيطة على المستوى الميكروسوبوي.

لتحقيق ذلك، نقترح أولاً تصميم ديناميكية النموذج الجماعي لنظام متعدد الوكالء في شكل قيود بيئية أما سلوك الوكالء فيكون في شكل إجراءات رقابة وتكييف معها. وهكذا، يتم بناء هذه العملية على المستويات الفردية والجماعية. وفي المرحلة الثانية، قد تم تمديد هذا المنهج التصميمي للسيطرة على مشكلة معقدة و هي مراقبة الإشارات الضوئية في شبكة الطرق متعددة النقطاقيعات وهذا لتقييم المنهج المقترن.

**كلمات البحث:** نظام متعدد الوكالء، التنظيم الذاتي، ظهور، التكييف، حل المشكلات، حركة المرور على الطرق، ومراقبة الإشارات.

# Table des matières

Remerciements .....	i
Résumé .....	ii
Abstract .....	iii
ملخص .....	iv
<b>INTRODUCTION .....</b>	<b>1</b>
1. Problématique et cadre de travail .....	1
2. Motivations, proposition et contributions .....	2
3. Plan de la thèse .....	4

## CHAPITRE I- RESOLUTION COLLECTIVE DE PROBLEMES ET SYSTEMES MULTI-AGENTS

Introduction .....	6
1.1 Problématique de résolution de problème .....	6
1.1.1 L'aspect collectif dans la résolution de problèmes .....	8
1.1.2 Nouveau défi : Des sources d'inspiration .....	9
1.2 Entité isolé : L'Agent.....	10
1.2.1 Définitions .....	10
1.2.2 Fonctionnement d'un agent .....	12
1.2.3 Architectures d'agent .....	13
1.2.4 Comportement d'agent .....	15
1.2.5 Propriétés d'agent .....	16
1.3 Systèmes Multi-Agents (SMA) .....	17
1.3.1 Définitions.....	17
1.3.2 Environnement d'un SMA .....	19
1.3.2.1 Types d'environnement .....	19
1.3.2.2 Propriétés d'environnement .....	19
1.3.3 Caractéristiques d'un SMA .....	20
1.3.4 Typologie de SMA .....	21
1.4 Interaction dans un SMA .....	23
1.4.1 Définition .....	23
1.4.2 Situation d'interaction .....	23
1.4.3 Types d'interaction .....	23
1.4.4 Conditions d'interaction .....	25
1.4.5 Formes d'interaction .....	26
1.5 Coopération dans les SMA .....	27

1.6	Coordination dans les SMA .....	28
1.7	Apprentissage dans les SMA .....	29
1.8	Communication comme moyen d'interaction.....	29
1.8.1	Communication directe .....	30
1.8.2	Communication indirecte .....	30
1.9	Propriétés attendues des SMA.....	31
1.9.1	Rationalité .....	31
1.9.2	Autonomie .....	31
1.10	Prise de décision dans les SMA.....	32
1.10.1	Observabilité .....	32
1.10.2	Incertitude.....	32
1.10.3	Délibération et décision.....	33
1.11	Domaines d'application des SMA.....	33
	Conclusion.....	35

## **CHAPITRE II- DE L'AUTO-ORGANISATION AUX SYSTEMES ADAPTATIFS MECANISMES ET PRINCIPES.**

Introduction.....	37
2.1 Organisation dans les SMA .....	37
2.1.1 Introduction.....	37
2.1.2 Définition de l'organisation.....	38
2.1.3 Objectifs de l'organisation .....	39
2.1.4 Structure organisationnelle et organisation concrète.....	39
2.1.5 Relations organisationnelles .....	41
2.2 Modèles d'organisation.....	42
2.2.1 Organisation AGR.....	42
2.2.2 Modèle OperA .....	43
2.2.3 Organisation holonique .....	44
2.2.4 Discussion sur l'organisation des SMA .....	45
2.3 Auto-organisation.....	45
2.3.1 Introduction.....	45
2.3.2 Définitions.....	46
2.3.3 Propriétés de l'auto-organisation.....	46
2.3.4 Mécanismes d'auto-organisation.....	47
2.3.4.1 Mécanismes basés sur l'interaction directe .....	47
2.3.4.2 Mécanismes basés sur la stigmergie.....	47
2.3.4.3 Mécanismes basés sur le renforcement.....	48
2.3.4.4 Mécanismes basés sur des architectures génériques.....	48
2.3.4.5 Mécanismes basés sur la coopération .....	48
2.3.4.6 Autres mécanismes.....	49
2.3.5 Observation de l'auto-organisation .....	49
2.3.6 Analyse des résultats observés .....	50
2.4 Emergence .....	52
2.4.1 Définitions .....	52
2.4.1.1 Nouveauté .....	53

2.4.1.2 Auto-organisation .....	53
2.4.1.3 Irréductibilité .....	53
2.4.1.4 Interdépendance des niveaux .....	53
2.4.2 Principe de l'émergence .....	54
2.4.3 Propriétés de l'émergence.....	55
2.5 Auto-Organisation et émergence : similarités et différences .....	55
2.5.1 Les similarités .....	56
2.5.2 Les différences .....	57
2.5.2.1 Auto-organisation sans présence de l'émergence.....	57
2.5.2.2 Emergence sans présence de l'auto-organisation.....	57
2.6 Approche Prometteuse : Combinaison entre approches.....	57
2.6.1 Lien Auto-organisation- émergence .....	58
2.6.2 Etude des phénomènes auto-organisateurs et émergents .....	58
2.6.3 Évaluation des propriétés émergentes d'un système.....	58
2.6.4 Contraintes et difficultés de contrôle des SMA	59
2.6.5 Problèmes liés au développement des systèmes ouverts et dynamiques	60
2.7 Les systèmes adaptatifs.....	60
2.7.1 Dimensions l'adaptation .....	61
2.7.2 Conception de système adaptatif .....	64
2.8 Adaptation dans les SMA .....	64
2.9 Potentiel de l'auto-organisation dans les SMA adaptatifs.....	65
Conclusion.....	66

### **CHAPITRE III- PARADIGMES DE DEVELOPPEMENT DES SMA**

Introduction.....	68
3.1 Modélisation et développement des SMA .....	69
3.1.1 Formalisme de développement de SMA .....	69
3.1.2 Aspect méthodologique.....	72
3.1.3 Phases de conception .....	73
3.2 Principales méthodes de conception des SMA.....	73
3.3 Comparaison des méthodes présentées.....	78
3.4 Plates-formes orientées Agent .....	79
3.5 Travaux connexes .....	83
Conclusion.....	86

### **CHAPITRE IV. PROPOSITION D'UN MODELE DE CONCEPTION D'UN SMA ADAPTATIF POUR LA RESOLUTION DE PROBLEME**

Introduction.....	87
4.1 Résolution de problèmes et Système Multi-Agents	88
4.2 L'auto-organisation dans le processus de résolution de problème par SMA	88
4.3 Modèle proposé de résolution de problème par les SMA adaptatifs	89
4.3.1 Point de vue adopté .....	89
4.3.2 Principe de conception .....	90

4.4 Conception d'environnement .....	92
4.4.1 Aspects structurels .....	93
4.4.2 Aspects dynamiques .....	93
4.5 Conception de comportement auto-organisé des agents .....	95
4.5.1 Conception d'interaction .....	95
4.5.2 Réorganisation dirigée par la fonction du système .....	95
4.6 Conditions sollicitées dans le processus de réorganisation .....	96
4.7 Dualité de l'autonomie et du contrôle du système .....	96
4.8 Lien entre les niveaux macro et micro .....	97
4.9 Etapes de conception de SMA auto-organisé pour la résolution de problèmes .....	97
4.10 Résolution de problème dirigée par une auto-organisation .....	98
Conclusion .....	99

## **CHAPITRE V- ETUDE DE CAS**

### **CONCEPTION D'UN SMA ADAPTATIF POUR LE CONTROLE INTELLIGENT DE LA SIGNALISATION DANS UN RESEAU ROUTIER**

Introduction .....	100
5.1 Problématique du trafic routier .....	101
5.2 Systèmes de transport intelligents (STI) .....	101
5.2.1 Unité d'acquisition .....	102
5.2.2 Modélisations microscopique et macroscopique .....	103
5.2.1 Grandeur physiques associés .....	103
5.3 Système de gestion des feux de circulation .....	104
5.4 Principaux systèmes adaptatifs existants .....	105
5.5 Travaux connexes .....	106
5.6 Conception d'un système de signalisation adaptative utilisant un SMA auto-organisé .....	107
5.6.1 Formulation du problème à travers l'environnement .....	108
5.6.2 Modèle proposé .....	109
5.6.3 Objectif ciblé du modèle proposé .....	110
5.6.4 Définition des agents .....	110
5.6.5 Définition des mécanismes d'interaction .....	112
5.7 Evaluation du résultat de la structure auto-organisée obtenue .....	114
5.8 Discussion des résultats .....	115
Conclusion .....	117
<b>Conclusion générale et perspectives .....</b>	<b>118</b>
Références bibliographiques .....	122
Annexe1: Simulation of Urban MObility – SUMO .....	133
Annexe2: Contrôleurs à Temps fixe, temps dynamique et vague verte Versus contrôleur adaptatif proposé .....	138

# Liste des figures

<b>Figure 1. 1.</b>	Agent /Environnement [Russel et al. 2006]	12
<b>Figure 1. 2.</b>	Agent réactif	13
<b>Figure 1. 3.</b>	Agent cognitif	14
<b>Figure 1. 4.</b>	Architecture d'agent hybride	15
<b>Figure 1. 5.</b>	Système multi-agents [Ferber, 1995]	18
<b>Figure 2. 1.</b>	Organisations et structures organisationnelles.	40
<b>Figure 2. 2.</b>	Représentation UML du méta-modèle AGR [Ferber, 1995]	43
<b>Figure 2. 3.</b>	Une holarchie à trois niveaux	44
<b>Figure 2. 4.</b>	Auto-organisation versus Emergence	56
<b>Figure 4.1</b>	Modèle générique proposé pour la conception d'un SMA adaptatif	92
<b>Figure 4.2</b>	Cadre général d'interactions dans un SMA adaptatif	94
<b>Figure 5. 1.</b>	Modèle d'Intersection	108
<b>Figure 5. 2.</b>	Emplacement de détecteurs dans une intersection	109
<b>Figure 5. 3.</b>	Mouvements autorisés	110
<b>Figure 5. 4.</b>	Proposition du cadre multi-agents : composants d'Intersection	111
<b>Figure 5. 5.</b>	Réseau de simulation	114
<b>Figure 5. 6.</b>	Courbe de comparaison du débit volumique	116
<b>Figure 5. 7.</b>	Courbe de Comparaison du temps moyen d'attente	116
<b>Figure 5. 8.</b>	Courbe de Comparaison du nombre moyen d'arrêts	116
<b>Figure A.1</b>	Illustration d'une carte	134
<b>Figure A.2.</b>	Exemple d'intersection	135
<b>Figure A.3</b>	Exemple de simulation	137

# Liste des tableaux

<b>Tableau 1.1</b>	Comparaison entre agents cognitifs et agents réactifs.	22
<b>Tableau 1.2.</b>	Classification des situations d'interaction [Ferber, 1995].	26
<b>Tableau 2.1.</b>	Ensemble des relations organisationnelles.	41
<b>Tableau 3.1.</b>	Synthèse de comparaison des différentes méthodes de conception.	79

# **Introduction générale**

# Introduction générale

## 1. Problématique et cadre de travail

La croissance de taille et d'hétérogénéité observée des systèmes informatiques a conduit les recherches en informatique au développement des systèmes composés de plusieurs entités parmi lesquelles sont réparties des tâches complexes à exécuter. Le développement des systèmes s'inscrivant dans ce contexte mobilise différentes technologies. Parmi elle, on peut notamment citer la technologie des systèmes multi-agents (SMA) engendré par le domaine de l'Intelligence Artificielle Distribuée (IAD)

Un SMA est composé de plusieurs sous-systèmes autonomes appelés agents dont chacun a une activité et des informations propres. Le comportement général d'un SMA est lié à l'activité combinée de l'ensemble de ses agents et la réalisation d'une tâche peut alors impliquer plusieurs entités.

La décomposition d'un système complexe en sous-systèmes autonomes interagissant permet de se focaliser sur une partie de ces sous-systèmes au lieu d'être obligé de considérer le système dans sa globalité. L'intérêt croissant porté aux SMA vient principalement de ce fait car ils offrent des solutions distribuées aux problèmes auxquels ils sont confrontés.

Toutefois, cette répartition nécessite que chaque agent puisse effectuer localement la tâche qui lui est assignée, mais aussi qu'il puisse coopérer et coordonner avec d'autres agents. Des capacités de raisonnement locales à chaque agent sont alors requises. Elles se basent sur les informations dont il dispose sur ses propres fonctions, sur son environnement et sur sa représentation des autres agents.

D'autre part, cette dernière décennie a vu fleurir un très grand nombre de méthodes destinées à développer les systèmes à plusieurs composantes autonomes. Cependant, peu de modèles existent pour leur application au cadre général de la résolution de problème. Cette difficulté de développement s'explique principalement de par la diversité, la richesse des paradigmes multi-agents, et la complexité des concepts sous-jacents (organisation, interaction, coordination, etc.).

Cette complexité rend l'utilisation de la majorité des outils existants difficilement applicable dans les environnements distribués, ouverts et qui changent dynamiquement de manière imprévue.

En fait, l'ouverture, la distribution et la complexité sont des propriétés inhérentes des systèmes informatiques actuels. Elles ne sont voulues par personne mais c'est un état que l'on ne peut plus se permettre d'ignorer. D'une part, la propriété de distribution impose que le système soit implémenté sous forme d'un ensemble d'entités logicielles avec une distribution des computations et une décentralisation des ressources et des connaissances. La propriété d'ouverture impose de son côté que les applications doivent obéir à des règles d'interopérabilité, de pouvoir intégrer de nouvelles entités ou supprimer d'autres qui la composent tout en assurant une activité et un contrôle adéquats.

Si on peut aisément offrir un système plus flexible et plus évolutif pour faire face aux situations inattendues qui ne peuvent être complètement connues ou décrites à l'avance, il faut toutefois que soient développées des techniques d'adaptation. De ce fait, la propriété d'adaptation s'impose fortement pour que les applications soient capables de réagir aux besoins et être plus conformes avec les environnements d'exécution qui ne cessent de changer constamment.

Le paradigme multi-agents auto-organisés peuvent offrir un cadre favorable aux développements des systèmes adaptatifs mais ne propose pas encore de modèles standards. Ils constituent actuellement un domaine de recherche à part entière.

Dans cette thèse, on s'intéresse à la conception des systèmes multi-agents adaptatifs pour la résolution de problèmes. De ce fait, ce travail de recherche se situe à l'intersection des domaines suivants :

- Tout d'abord, l'ingénierie des systèmes multi-agents et ceci par la mise en place d'un modèle générique de conception de SMA.
- Ensuite, les sciences de complexité, par le fait d'utiliser l'adaptation dans les SMA avec ses deux principes d'auto-organisation et d'émergence comme une approche de résolution de problèmes.

## **2. Motivations, proposition et contributions**

Les systèmes actuels sont formés, le plus souvent, de multitudes de composants, parfois non conçus initialement pour fonctionner ensemble ou ayant leurs propres dynamiques

internes. Ils sont en interaction avec des environnements fortement changeants et doivent réagir et évoluer sans aucune intervention externe.

L'adaptation est ainsi une propriété à fournir à ces systèmes afin de répondre à cette dynamique environnementale. L'approche multi-agents est un aussi excellent paradigme de modélisation et d'implantation de ces systèmes complexes. Les agents sont aptes à faire face aux changements imprévus de leur environnement et Ils peuvent s'organiser de manière autonome pour maintenir leurs performances. La mise en place de ces processus organisationnels permettra l'émergence des comportements cohérents recherchés.

Nous formulons, dans cette thèse, un modèle générique d'adaptation dans les SMA se basant sur l'évolution de l'organisation globale et l'évaluation des interactions locales entre les différents agents. L'approche consiste à maintenir un degré admissible de coopération entre les agents du système par une réorganisation locale. L'effet de ce mécanisme adaptatif permet d'augmenter la performance globale. Les agents du système évoluent, leurs intérêts ainsi que leurs besoins évoluent dans le temps. Face aux perturbations résultant de ces changements, le système se réorganise dans le but de faire émerger sa structure organisationnelle vers une structure coopérative et adaptative.

Cette thèse présente en parallèle au problème de la conception d'un SMA adaptatif, une étude plus générale du cadre et des mécanismes nécessaires à la résolution de problèmes par auto-organisation. La question d'ingénierie soulevée dans ce travail est de savoir comment garantir certaines propriétés macroscopiques ou globales du système conçu en ne développant que des entités et leurs comportements à un niveau microscopique ou local.

Un intérêt particulier est donné aux aspects méthodologiques pour les SMA, tout en illustrant les outils et méthodes proposés dans le cadre de la résolution de problèmes. Ces recherches contribuent donc à répondre à une problématique déclinée en deux axes présentés ci-après.

#### **a. Conception d'un SMA adaptatif**

Cet axe tente de répondre à une question d'ingénierie des SMA. Le but est de fournir un outil pour la modélisation et la conception de SMA adaptatifs. L'approche auto-organisationnelle utilisée se focalise sur les comportements locaux des agents et la coopération à mettre en œuvre pour atteindre un objectif global commun.

Elle permet aussi la mise en place d'un mécanisme d'adaptation qui fait appel à la notion de réorganisation. Cette dernière est manipulable par les agents grâce à leur environnement et par la mise en place d'une boucle micro-macro consolidé par une fonction objective.

### **b. Résolution de problèmes par un SMA adaptatif**

Afin de résoudre les problèmes conceptuellement distribués, les SMA apparaissent comme un paradigme pertinent. Notamment, les SMA auto-organisés qui proposent des mécanismes de résolution de conflits, d'adaptation, de décentralisation, et de stabilisation qui peuvent être vus comme autant de moyens de résolution de problèmes distribués et dynamiques. Cet axe cherche à éclairer donc l'apport de l'adaptation des SMA à la résolution de problèmes complexes. Par résolution de problèmes, on distingue le cadre de modélisation de problèmes réels nécessitant de fournir des solutions solides ou les problèmes d'optimisation.

Enfin, une étude de cas a été menée dans le contexte de l'activité de transport qui est une composante essentielle de nos sociétés. Les exigences de mobilité de personnes toujours croissantes font de cette activité de transport une charnière des systèmes économiques modernes et de leur qualité dépend celle de tout le circuit économique et social. Les spécificités de ces systèmes réels immersés dans des environnements ouverts et dynamiques rendent les techniques classiques difficiles à appliquer.

D'autre part, l'utilisation optimale des ressources disponibles afin de satisfaire ces besoins est une tâche ardue. Il s'agit de garantir à des entités physiques de se déplacer dans un réseau avec un moindre cout financier et temporel. Un système adaptatif pour la résolution de ce problème et la satisfaction de ces contraintes a été dégagé à partir de du modèle multi-agents précédemment proposé Des simulations ont été menées aussi pour essayer d'analyser le niveau global d'adaptation du SMA conçu avec illustration du processus de réorganisation locale.

## **3. Plan de la thèse**

Ce manuscrit est divisé en deux parties:

La première partie qui présente l'état de l'art est composée de trois chapitres. Le premier chapitre aborde l'aspect collectif dans la résolution de problème. Il expose aussi les définitions et les concepts clés des SMA en mettant l'accent sur l'approche réactive.

Dans le second chapitre, les concepts fondamentaux de notre proposition ont été abordés. D'abord, on s'est intéressé aux concepts de l'auto-organisation et l'émergence puis à leur impact dans les systèmes adaptatifs. Dans le chapitre trois, un aperçu détaillé des méthodes de conception des SMA existantes avec une étude comparative et les travaux connexes dans cet axe sont présentés.

La deuxième partie est la partie contribution. Elle comporte deux chapitres : le chapitre quatre où un modèle générique de conception de SMA adaptatif pour la résolution de problème a été dégagé à partir de l’analyse précédemment exposée. Dans le chapitre cinq, l’approche présentée est utilisée pour proposer un modèle de contrôle de signalisation adaptatif dans un réseau routier. Les résultats obtenus et comparés avec d’autres méthodes sont aussi exposés dans ce chapitre.

Enfin, la thèse s’achève par une conclusion générale, qui résume l’apport réalisé et des perspectives pouvant être développées par la suite, ouvrant ainsi de nouvelles voies à envisager pour des recherches futures.

# **Partie I**

## **Etat de l'art**

# **CHAPITRE 1**

## **Résolution collective de problèmes**

## **et Systèmes multi-agents**

# CHAPITRE I

## Résolution collective de problèmes et Systèmes multi-agents

### Introduction

L'émergence des systèmes multi-agents a été l'un des évènements marquants des années quatre-vingt-dix dans la recherche informatique. Apparus tout d'abord comme un sous domaine de recherche de l'Intelligence Artificielle (IA), les systèmes multi-agents (SMA) sont actuellement l'un des domaines les plus actifs de la recherche avec une assise applicative de plus en plus importante.

Cette discipline, joignant plusieurs domaines, allant de l'IA, des systèmes informatiques distribués au génie logiciel, s'intéresse aux comportements collectifs produits par les interactions de plusieurs entités autonomes appelées agents. Elle permet d'offrir une alternative intéressante pour la conception, la mise en œuvre et la simulation de systèmes complexes distribués et ouverts.

D'autre part, la résolution des problèmes distribués se préoccupe de la façon dont un problème donné peut être résolu par plusieurs entités qui coopèrent en divisant et en partageant la connaissance à propos du problème et des solutions développées.

Dans ce chapitre, l'aspect collectif dans la résolution de problème est présenté. Les concepts et principes fondamentaux des SMA sont introduits afin de situer ce travail dans son contexte de recherche. Aussi, les différentes problématiques que soulèvent les SMA sont énoncées telles que l'interaction, la communication et la coopération.

### 1.1 Problématique de résolution de problème

La résolution de problèmes peut être considérée comme le processus permettant, à partir d'un problème donné, de trouver une solution satisfaisante. Cette thématique n'est pas uniquement liée à l'informatique puisqu'on la retrouve dans de nombreuses disciplines telles que les sciences cognitives, la psychologie, l'ingénierie,....

Elle est étroitement liée à la prise de décision. Cette dernière est considérée comme un processus permettant de choisir une action (ou une séquence d'actions) pertinente relativement à des critères parmi un ensemble d'actions possibles dans le but de résoudre un problème donné. Les critères de sélection de l'action la plus adaptée prennent en compte à la fois les contraintes du problème à résoudre et le but que le système doit atteindre.

Dans la plupart des cas, le processus décisionnel peut se décomposer de la manière suivante [Schneider, 1994] :

- (1) Définition du problème,
- (2) Définition des buts,
- (3) Résolution du problème,
- (4) Évaluation des solutions obtenues,
- (5) Sélection d'une solution.

Ainsi, dans ce contexte, la résolution de problème peut être considérée comme une étape particulière du processus de décisionnel.

Toutefois, et depuis sa naissance dans les années cinquante, l'informatique est essentiellement basée sur des théories formelles issues des Mathématiques ou de la Logique. Si le développement de logiciels à partir de ce type d'approche a permis un succès réel, sa pertinence actuelle et son devenir futur auxquels ce développement va devoir faire face dans les prochaines années, n'est pas évident.

En effet, les approches de conception d'applications se heurtent à de nombreuses difficultés. D'une part, l'aspect distribué qui contribue à accroître les problèmes de conception, de stabilité et de maintenance, et d'autre part, l'aspect dynamique et ouvert des applications actuelles qui rend la tâche plus ardue.

On pourrait dire que les systèmes à développer sont caractérisés par les propriétés suivantes:

- Un grand nombre d'artefacts (logiciels, objets intelligents) en interaction;
- Un contrôle centralisé impossible à mettre en œuvre;
- Un environnement évolutif;
- Une tâche globale à réaliser.

La réponse à cette problématique est un challenge scientifique nommé “autonomic computing”, dont l'objectif est de concevoir des entités distribuées capables de s'exécuter seules, d'ajuster leur comportement face à des circonstances variées et de préparer leurs ressources pour gérer au mieux leur charge de travail. Ainsi, la résolution coopérative de problèmes prend une place prépondérante dans ces recherches.

### **1.1.1 L'aspect collectif dans la résolution de problèmes**

Résoudre un problème consiste, en général, à envisager tout type de système informatique chargés de choisir des actions appropriées parmi un ensemble d'actions possibles, à les organiser dans un plan d'actions et à les exécuter. Bien évidemment, cette résolution peut prendre bien différentes formes, de simples programmes de calcul jusqu'à des programmes "intelligents" très complexes.

Néanmoins, elle constitue souvent un processus d'essai-erreur, car l'exécution des actions identifiées comme adéquates peut échouer. Dans ce cas, il faut encore re-choisir, réorganiser et re-exécuter de nouvelles actions car plus le problème est considéré comme "difficile", plus il y a des difficultés à réaliser un système pour le résoudre.

En effet, depuis plusieurs années, les systèmes multi-agents sont devenus une partie importante de la recherche en informatique et plus spécifiquement en intelligence artificielle. Un système multi-agents peut être défini comme étant un ensemble d'entités en interaction entre elles et avec leur lieu d'évolution, appelé communément environnement, et produisant collectivement une organisation spatio-temporelle. L'aspect collectif du système multi-agents est très important car il reflète en général le but global à atteindre par les agents. Cependant, la principale difficulté réside dans la prédiction et la vérification des propriétés globales en ne connaissant que les définitions locales des comportements et des interactions mis en œuvre.

De ce fait, les recherches en intelligence artificielle distribuée (IAD) et les systèmes multi-agents s'orientent vers l'étude de répartition de tâches sur un ensemble d'entités coopérantes. Elles étudient aussi la manière de coordonner le comportement intelligent d'un ensemble d'entités selon les lois sociales pour atteindre une résolution adéquate du problème. C'est ainsi qu'un SMA peut être vu comme un ensemble interconnecté d'agents qui travaillent ensemble pour implémenter des stratégies de résolution basées sur le comportement d'un collectif d'agents autonomes. Néanmoins, les problèmes deviennent plus complexes, et l'aspect collectif de ces agents nécessite en conséquence l'étude de nouveaux concepts et de nouvelles théories qui découlent de la société d'agents.

Le défi est donc de trouver de nouvelles approches adaptatives pour concevoir de nouveaux systèmes informatiques répondant aux difficultés existantes telles que la prise en compte d'une complexité croissante ou la réalisation de systèmes distribués robustes et fiables. Pour découvrir de telles approches, il semble opportun de se pencher sur les systèmes collectifs (biologique, physiques, sociologiques), afin de comprendre les mécanismes et processus qui leur permettent de fonctionner.

### **1.1.2 Nouveau défi : Des sources d'inspiration**

En Physique et Chimie, les interactions entre des éléments simples donnent naissance à des structures complexes dotées de propriétés totalement nouvelles. Par exemple, l'eau composée d'atomes d'oxygène et d'hydrogène peut être considérée comme un phénomène émergent. La Thermodynamique est aussi un exemple frappant de cet état de fait : alors que les molécules et leurs interactions sont parfaitement connues, les modèles de fluides ou de gaz de plus haut niveau sont limités à des utilisations fortement contraintes. A l'image des ces processus physiques ou chimiques à l'échelle moléculaire ou atomique, plusieurs algorithmes ont été développés permettant le passage d'un micro-niveau connu à un macro-niveau aux fonctionnalités beaucoup plus complexes.

En Biologie aussi, de nombreux systèmes naturels composés d'individus autonomes exhibent des aptitudes à effectuer des tâches qualifiées de complexes sans contrôle global. De plus, ils peuvent s'adapter à leur milieu soit pour y survivre, soit pour améliorer le fonctionnement du collectif. C'est le cas des colonies d'insectes sociaux [Camazine et al., 2001] tels que les termites, les fourmis [Bonabeau et al., 1997], ou les araignées [Bourjot 1999] qui font effectivement preuve de remarquables capacités pour effectuer des tâches telles que : la construction de nids complexes, la construction de pont, la recherche efficace de ressources, la capture de proies. Le fonctionnement du système immunitaire est également représentatif du fonctionnement d'un système complexe composé d'un ensemble d'agents autonomes. Des phénomènes non supervisés et qui résultent de l'activité d'un grand nombre d'individus s'observent aussi dans les activités humaines, tels la gestion des embouteillages, la synchronisation des mouvements d'une foule ou encore leurs applaudissements.

Contrairement aux approches d'inspiration biologique, les méthodes basées sur une inspiration physique s'appuient sur une étude pragmatique s'inspirant de l'ingénierie. Ces approches, proches des méthodes utilisées en automatique pour définir des systèmes à base de boucle de régulation, se concentrent sur la définition, l'élaboration et la mise au point de mécanismes d'interaction permettant d'obtenir un état d'équilibre stable ou méta - stable au niveau du système. Dans ce cadre, la démarche n'est pas induite par le résultat global que l'on veut obtenir mais par les comportements individuels que l'on veut fournir. Les applications d'inspiration physique sont également très répandues et reposent le plus souvent sur des mécanismes de type attraction et répulsion [Zeghal et al., 1994 ; Simonin et al., 2006 ; Moujahed et al., 2009].

Enfin, on peut noter que des travaux sont déjà en cours depuis des années pour l'étude, la compréhension, l'inspiration des phénomènes et son application sur un problème complexe réel. Toutefois, dans la plupart des systèmes étudiés, on peut souligner un facteur commun : la dimension comportementale sociale observée. Ce qui revient à chercher la réalisation d'artefacts simples et distribués capables d'accomplir des tâches complexes dans sa dimension sociale par organisation, coopération et interaction. Ceci pourrait permettre de construire des systèmes robustes, autonomes et adaptatifs. C'est là l'un des objectifs fondamentaux des systèmes multi-agents.

Nous présentons ci-dessous le cadre général des SMA, notamment au travers des concepts clés d'agent ce qui permet de structurer la problématique de ce travail de recherche.

## 1.2 Entité isolé : L'Agent

Il n'y a pas une notion d'« agent » unique, car il est utilisé dans de nombreuses applications par des communautés venant d'horizons divers. En IA, ce terme est pluri défini puisqu'on trouve dans la littérature une multitude d'interprétations et à l'heure actuelle, il n'y a pas de véritable consensus sur sa définition. Nous présentons dans ce qui suit les plus utilisées et s'adaptant à notre problématique.

### 1.2.1 Définitions

Dans [Russell et al., 2006], la notion d'agent est définie comme suit : « Un agent est tout ce qui peut être compris comme percevant son environnement à travers des senseurs et comme agissant sur cet environnement par l'intermédiaire d'effecteurs ».

Selon cette définition, un agent est un système qui décide par lui-même ce qu'il doit faire pour atteindre ses objectifs (déterminés à sa conception). L'agent est situé dans un environnement, doté de capteurs et d'actionnaires, observant l'environnement et cherchant à modifier son évolution.

Dans [Ferber, 1995], la définition du terme « agent » est plus détaillée. L'auteur allie le fonctionnement de l'agent et ses propriétés.

On appelle agent une entité physique ou virtuelle :

1. Qui est capable d'agir au sein d'un environnement E.
2. Qui peut communiquer avec les autres agents de manière directe ou indirecte.
3. Qui est mue par un ensemble de tendances, sous la forme d'objectifs individuels, d'une fonction de satisfaction ou de survie, qu'elle cherche à optimiser.
4. Qui possède des ressources propres.

5. Qui est capable de percevoir, mais de manière limitée, son environnement E.
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. 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.

Dans cette définition, les items (1) à (5) reprennent les principes d'action et de perception, l'auteur intègre d'autres propriétés telles que la capacité de communiquer avec les autres agents (2), la prise de décision en accord avec ses objectifs (3) et en tenant compte de ses ressources propres (4). Enfin la propriété de localité influe sur ses connaissances de l'environnement (6).

Les caractéristiques intéressantes telles que la notion d'objectif et de perception limitée ou locale de l'environnement ont été introduites dans cette définition. Les décisions de l'agent sont dirigées par ses aptitudes en accord avec ses connaissances et les ressources disponibles. L'autonomie de l'agent est aussi une propriété importante mais son évolution est liée à celle de l'environnement.

Jennings, Sycara et Wooldridge [Jennings et al., 1998] ont proposé une autre définition de la notion d'agent qui reste néanmoins proche de celle donnée par [Russell et al. 1995] :

« Un agent est un système informatique, situé dans un environnement qui est capable d'action autonome et flexible dans le but de répondre à ses objectifs de conception ».

Cette définition est fondée sur les trois concepts suivant :

- **Situé** : un agent est dit situé, s'il est capable d'agir sur son environnement à partir des entrées sensorielles qu'il reçoit de ce même environnement.
- **Autonome** : un agent est capable d'agir sans l'intervention directe de l'être humain ou d'autres agents et qu'il est capable de contrôler ses propres actions et son état interne.
- **Flexible** : un agent est dit flexible s'il est capable d'agir de manière réactive, proactive et sociale :
  - **Réactif** : être capable de percevoir son environnement et de réagir dans le temps.

- **Proactif** : prendre l'initiative et être opportuniste au bon moment.
- **Social** : être capable d'interagir avec les autres agents quand la situation l'exige

Dans cette définition, les auteurs présentent des compétences de l'agent pouvant être considérées comme étant émanant du comportement intelligent, évidemment dans le sens de l'IA.

D'après ces définitions d'agent, on peut se permettre de mettre en évidence les caractéristiques importantes d'agent, exprimées par la présentation suivante : « Un agent est défini comme une entité (physique ou abstraite) ayant un objectif individuel avec un comportement autonome et flexible qui résulte de la conséquence de ses observations, de sa connaissance et des interactions avec les autres agents. Il a la possibilité d'agir sur lui-même et sur son environnement, grâce à sa capacité de perception. Enfin, il dispose d'une représentation partielle de ce même environnement ».

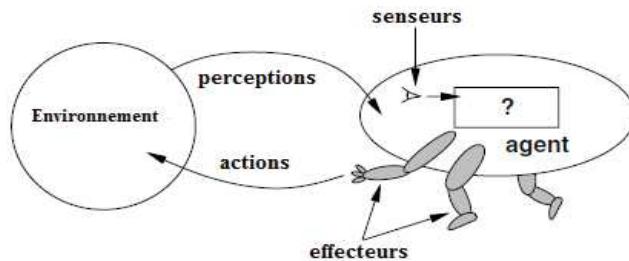


Figure 1. 1. Agent /Environnement [Russell et al. 2006]

De manière générale et quelle que soit sa définition, un agent suit le comportement décrit par une boucle fermée appelée boucle sensori-motrice représentée par la figure 1.1.

### 1.2.2 Fonctionnement d'un agent

Un agent évolue toujours selon un cycle composé de plusieurs étapes (boucle sensori-motrice) :

- Initialement l'agent se trouve dans une certaine configuration initiale ;
- Il perçoit son environnement (ou partie de l'environnement) grâce à ses capteurs ;
- Il choisit une action à entreprendre en fonction de sa configuration et de ses perceptions ;
- Il reçoit de nouvelles perceptions et le processus se répète.

La phase de délibération ou de décision de l'agent dépend de son architecture interne. Cette architecture décrit la manière de concevoir l'agent et dépend de son mécanisme de raisonnement.

### 1.2.3 Architectures d'agent

Nous distinguons deux grandes familles d'agents : les agents réactifs et les agents cognitifs. Cette distinction repose essentiellement sur le processus décisionnel de l'agent et la représentation de l'environnement dont il dispose. Si l'agent est doté d'une représentation symbolique de l'environnement à partir duquel il est capable de formuler des raisonnements, nous disons qu'il est cognitif tandis que s'il ne dispose que d'une représentation limitée à ses perceptions, alors il est réactif.

#### a- Agent réactif

Comme son nom l'indique, un agent réactif ne fait que réagir aux changements qui surviennent dans l'environnement. Autrement dit, un tel agent ne fait ni délibération ni planification, il se contente simplement d'acquérir des perceptions et de réagir à celles-ci (Figure 1.2).

Étant donné qu'il n'y a pratiquement pas de raisonnement, ces agents peuvent agir et réagir très rapidement. Ferber définit ce type d'agent comme suit : « Les agents réactifs, définis par le fait même qu'ils n'ont pas de représentation de leur environnement et des autres agents, sont incapables de prévoir ce qui va se passer et donc d'anticiper en planifiant les actions à accomplir [Ferber, 1995] ».

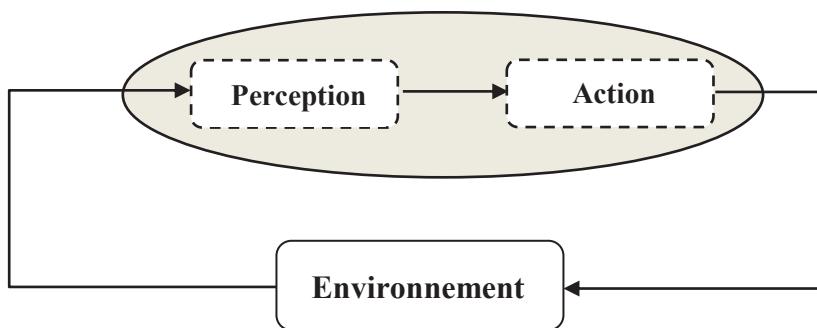


Figure 1.2. Agent réactif

L'agent réactif (figure 1.2.) présente les caractéristiques suivantes:

- Pas de mémoire ;
- Pas de représentation explicite ;
- Prise de décision se basant sur le fait du Stimulus/Réponse ;
- Simple à mettre en œuvre.

**b- Agent cognitif :**

L'agent cognitif est un agent qui dispose d'une base de connaissances comprenant l'ensemble des informations et de savoir-faire nécessaires à la réalisation de sa tâche et la gestion des interactions avec les autres agents et avec son environnement [Ferber ,1995].

Ainsi, ces agents possèdent une représentation explicite de leur environnement, des autres agents et d'eux-mêmes. Ils sont aussi dotés de capacités de raisonnement et de planification ainsi que de communication. Le travail le plus représentatif de cette famille d'agent porte sur le modèle BDI (Believe Desire Intention).

Ce modèle se fonde sur trois attitudes qui définissent la rationalité d'un agent intelligent :

- Croyance (Belief B): ce sont les informations que possède l'agent sur son environnement et sur les autres agents agissant sur le même environnement. Ceci constitue les connaissances supposées vraies de l'agent.
- Désir (Desire D): ce sont les états de l'environnement et parfois de lui-même, qu'un agent aimeraient voir réaliser. Ce sont les objectifs que se fixe un agent.
- Intention (Intention I): ce sont les actions qu'un agent a décidé de faire pour accomplir ses désirs. Ils forment des ensembles de plans qui sont exécutés tant que l'objectif correspondant n'est pas atteint.

Le modèle BDI a inspiré beaucoup d'architectures d'agents cognitifs.

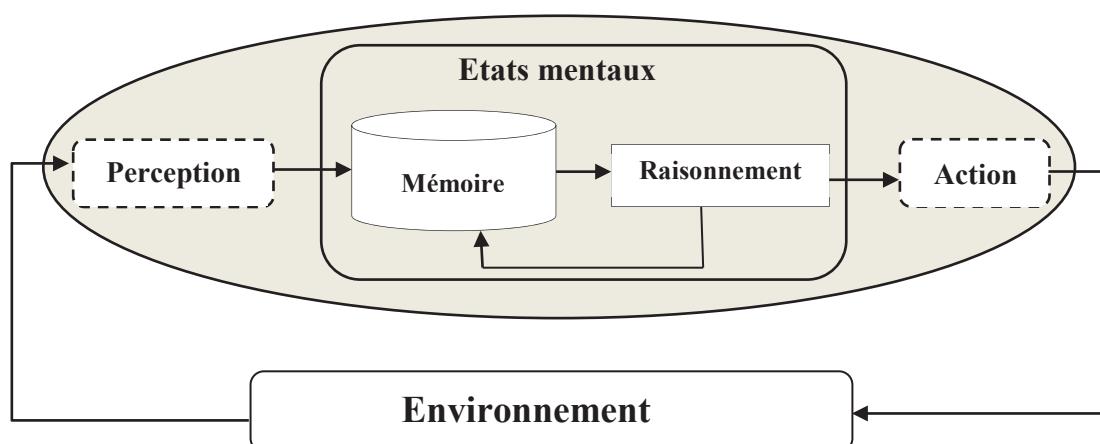


Figure 1.3. Agent cognitif

Ce type d'agent (Figure 1.3) se caractérise par:

- Une représentation explicite de l'environnement et du monde auquel ils appartiennent ;
- Une réaction planifiée ;
- Une base de connaissances comprenant des informations et du savoir-faire ;
- Une mémoire pour mémoriser les anciens états.

Il est possible de combiner ces deux architectures pour obtenir une troisième architecture qui se base sur des agents cognitifs exhibant des capacités de réactions aux événements, on parlera alors d'agents hybrides.

### c- Agent hybride:

L'agent hybride est conçu pour combiner les capacités réactives à des capacités cognitives, ce qui leur permet d'adapter son comportement en temps réel à l'évolution de l'environnement. Dans ce modèle, un agent est composé d'une architecture multicouche qui se base sur la hiérarchie de niveaux (Figure 1.4).

L'agent a une structure modulaire, ce qui est concrètement recommandé dans le développement de processus artificiel. Les capacités de traitement peuvent être améliorées car les différents composants peuvent fonctionner simultanément.

Généralement, on trouve deux présentations pour cette architecture « horizontale et verticale ».

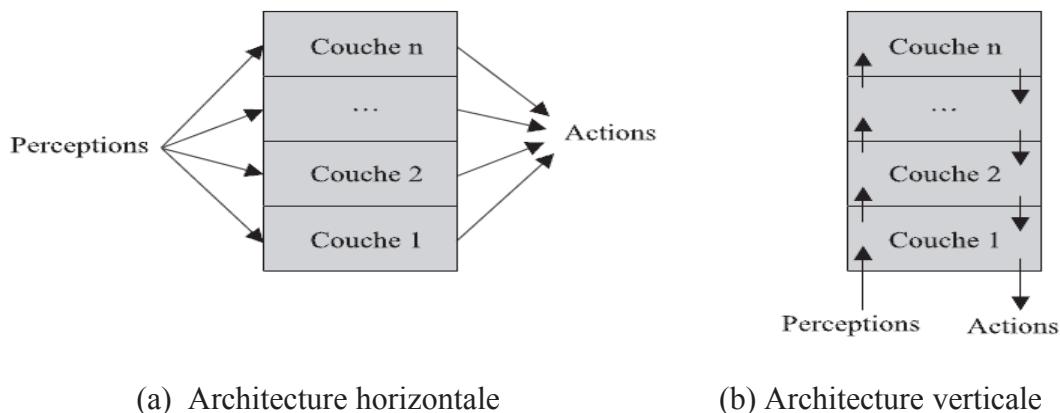


Figure 1.4. Architecture d'agent hybride

#### 1.2.4 Comportement d'agent

Un agent se caractérise essentiellement par la manière dont il est conçu et par ses actions. En d'autres termes par son architecture et son comportement [Ferber, 1995]. Ce dernier terme est fondamental dans la définition et la modélisation d'un agent.

Le comportement caractérise l'ensemble des propriétés que l'agent manifeste dans son environnement. On peut le comprendre par les réponses de l'agent aux sollicitations de son environnement ou en regardant sa manière d'évoluer. Il est analysable sans connaître les détails d'implémentation. Il s'agit d'un phénomène qui peut être appréhendé par un observateur extérieur qui au regard des actions qu'entreprend l'agent, peut induire ou spécifier ce qu'une architecture est censé produire.

Ainsi, le comportement apparaît comme une spécification externe de l'agent, et l'architecture définit les relations internes permettant d'aboutir à cette spécification.

### 1.2.5 Propriétés d'agent

A partir des définitions, qui apparaissent dans la littérature, plusieurs propriétés d'agents peuvent être identifiées :

- Autonomie : les agents contrôlent leurs actions et leurs états internes. Le système dans son ensemble est capable de réagir sans l'intervention d'un humain ou d'un autre agent. Il n'y a pas de définition unique du terme agent, par contre, il y a un consensus général pour considérer l'autonomie comme notion centrale de l'agent.
- Réactivité : ils perçoivent leur environnement et réagissent aux changements qui s'y produisent dans le temps requis.
- Initiative : le comportement des agents est déterminé par les buts qu'ils poursuivent et par conséquent ils peuvent produire des actions qui ne sont pas seulement des réponses à leur environnement.
- Habilité sociale : pour satisfaire ses buts un agent peut demander l'aide d'autres agents avec lesquels il partage la réalisation de tâches.
- Raisonnement : un agent peut décider quel but poursuivre ou à quel événement réagir, comment agir pour accomplir un but, ou suspendre ou abandonner un but pour se dédier à un autre.
- Apprentissage : l'agent peut s'accommoder progressivement à des changements dans des environnements dynamiques grâce à des techniques d'apprentissage.
- Mobilité: dans des applications déterminées il peut être intéressant de permettre aux agents de migrer d'un nœud à un autre dans un réseau tout en préservant leur état lors de sauts entre nœuds.

En considérant les définitions existantes, nous pouvons identifier deux tendances principales dans la présentation des agents. Quelques chercheurs considèrent l'agent en isolation, alors que certains d'autres le présentent principalement comme agissant dans un environnement incluant d'autres agents.

Les deux tendances ont déjà porté des résultats mais le paradigme des agents s'impose car il est plutôt difficile de dire qu'un agent existe seulement comme une entité unique et qu'il ne rencontrera pas d'autres agents dans son environnement.

### 1.3 Systèmes Multi-Agents (SMA)

Au premier abord, un SMA peut être considéré comme un ensemble d'agents partageant un environnement commun et capable d'interagir entre eux et sur l'environnement. Néanmoins, et pareille au concept « agent », il n'existe pas de définition acceptée en unanimité dans la littérature.

#### 1.3.1 Définitions

Plusieurs définitions ont été proposées, nous allons présenter la définition donnée par [Ferber, 1995].

« On appelle système multi-agents (ou SMA), 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 ( $A \subset O$ ), 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**.
- Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, que l'on appellera les lois de l'univers ».

La définition proposée est basée sur certaines caractéristiques, et plus précisément sur la notion d'interaction entre les agents et l'environnement, et/ou les agents eux-mêmes.

En effet, l'auteur présente un formalisme de description du SMA permettant de dériver un modèle formel pour décrire cette relation entre agents et environnement. La Figure 1.5 illustre cette définition.

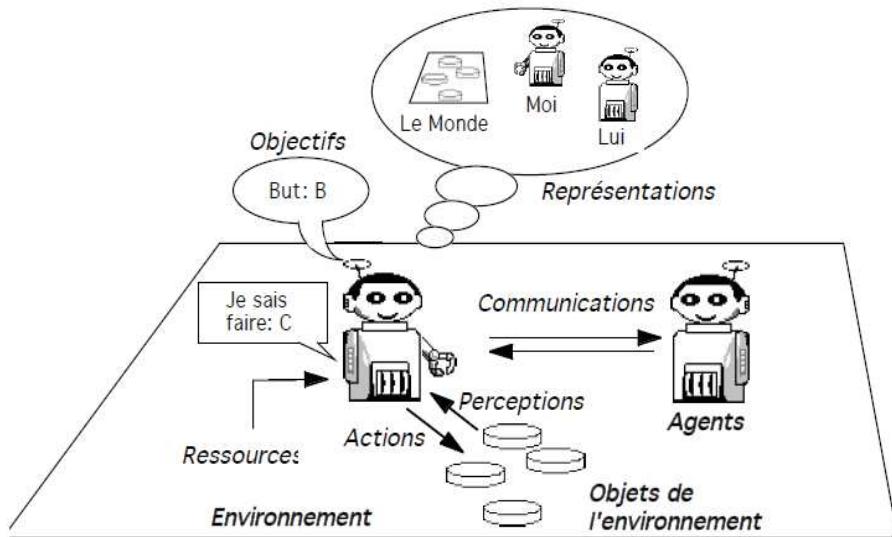


Figure 1.5. Système multi-agents [Ferber, 1995]

Une autre définition, donnée par Wooldridge et Jennings, présente un SMA comme étant : « Un SMA est un ensemble d'agents ayant des buts ou des tâches, et qui interagissent pour les accomplir » [Wooldridge et al., 2000].

Un SMA est donc un système composé d'entités informatiques, appelées des agents, qui évoluent et interagissent dans un environnement commun. La notion d'interaction entre agents est essentielle car chacun d'eux est impliqué dans une dynamique commune. Elle représente aussi ce qui permet de construire la réponse collective à partir des réponses individuelles.

[Demazeau, 1995] propose une décomposition d'un SMA en quatre dimensions qui correspondent aux quatre voyelles A, E, I et O, et qui est développée dans [Demazeau, 2001] :

- Agent (A): définition des modèles ou des architectures des composants du système.
- Environnement (E): milieu dans lequel sont plongés les agents, composé d'objets qui sont perçus et manipulés par les agents, et qui obéissent à des lois physiques.
- Interactions (I): ensemble des infrastructures, langages et protocoles d'interaction entre agents.
- Organisation (O): structure des agents en groupes, hiérarchies, relations, etc.

On y trouve souvent associée une cinquième dimension : « *Utilisateur* » qui représente un agent (opérateur) extérieur au système mais qui possède une influence sur lui et peut l'observer.

Ces notions sont à la base de notre proposition, elles sont utilisées plus en détails certains concepts propres aux SMA.

### 1.3.2 Environnement d'un SMA

L'environnement est un élément important dans le système multi-agents. C'est grâce à lui que les agents peuvent coexister et interagir. L'environnement doit pouvoir être perçu par les agents et ces derniers doivent pouvoir agir dessus et interagir au travers.

Avec celle des comportements individuels, la spécification de l'environnement permet de définir la dynamique d'un SMA. Lorsque les agents sont réactifs, l'environnement détient une importance capitale. En effet, comme ces agents ne peuvent communiquer directement entre eux, il est le médiateur de leurs interactions. Ils s'influencent mutuellement soit par leur position s'ils sont situés, soit par l'intermédiaire d'objets qu'ils perçoivent et modifient.

#### 1.3.2.1 Types d'environnement

Selon le point de vue que l'on adopte, différents types d'environnements peuvent être identifiés :

- **Point de vue du système multi-agents** : l'environnement correspond à l'ensemble des entités extérieures au système.
- **Point de vue de l'agent** : l'environnement est tout ce qui est extérieur à lui-même.
- **Point de vue du concepteur** : il peut correspondre à l'état du système, ou représenter l'ensemble des outils permettant de simuler, de visualiser et d'évaluer le SMA.

#### 1.3.2.2 Propriétés d'environnement

L'environnement possède certaines propriétés

- **Accessible ou inaccessible** : un agent a accès à l'état complet de l'environnement ou non.
- **Déterministe ou indéterministe** : le changement de l'état de l'environnement est uniquement déterminé par l'état courant et les actions des agents ou non.

- **Statique ou dynamique** : l'environnement peut changer quand l'agent est en action (réflexion) ou non.
- **Discret ou continu** : le nombre de perceptions et d'actions est limité ou pas.

En ce qui suit, nous présentons aussi quelques caractéristiques importantes des SMA.

### 1.3.3 Caractéristiques d'un SMA :

Un SMA possède la plupart des caractéristiques suivantes [Boissier, 2004] :

- **Distribution** : le système est décomposable, l'élément de base étant l'agent.
- **Décentralisation** : les agents sont indépendants, il n'y a pas de décisions centrales valables pour tout le système.
- **Autonomie** : un agent est en activité permanente et prend ses propres décisions en fonction de ses objectifs et de ses connaissances.
- **Echange de connaissances** : les agents sont capables de communiquer entre eux, selon des langages plus ou moins élaborés.
- **Interaction** : les agents ont une influence localement sur le comportement des autres agents.
- **Organisation** : les interactions créent des relations entre les agents, et le réseau de ces relations forme une organisation qui peut évoluer au cours du temps.

D'autre part, l'ouverture d'un système, qu'il soit physique ou informatique, représente la possibilité qu'il échange de l'information ou de la matière avec l'extérieur, et que son environnement possède une dynamique propre avec des évolutions imprévisibles. Pour un SMA, l'ouverture désigne la capacité d'ajouter ou de retirer dynamiquement dans le système des agents, ou des fonctionnalités et des services de ces agents [Vercouter, 2004]. Un SMA ouvert est donc un système extensible et évolutif.

La dynamique d'un SMA est définie au niveau des comportements individuels des agents, et son évolution découle de leurs interactions. Cette décentralisation impose également au SMA de fortes contraintes. Elle peut être vue comme le fait qu'un intervenant extérieur au SMA est soumis à des contraintes de localité similaires à celles des agents : il ne peut percevoir et agir que sur une partie du système.

L'état courant du SMA n'est pas nécessairement connu entièrement, seuls des indicateurs limités sur son état sont disponibles, et permettent son observation partielle. De la même manière, une action destinée à influer sur le SMA pourra ne pas s'appliquer à l'ensemble du système, ou pas de manière synchronisée.

Aussi, on distingue deux typologies des SMA selon le type des agents qui les composent, à savoir les systèmes multi-agents cognitifs et les systèmes multi-agents réactifs.

#### **1.3.4 Typologies de SMA**

##### ***a) SMA réactif***

Un système multi-agents réactif est composé d'agents réactifs. L'étude des SMA réactifs cherche à comprendre le fonctionnement du système comme un tout, en se focalisant sur les aspects collectifs du système c'est-à-dire sur les interactions et la dynamique qui en résulte.

Ce système présente plusieurs avantages tels que la fiabilité (assurée par le grand nombre d'agents du système et leur simplicité) et la grande flexibilité vis à vis du changement dynamique de son environnement. Cependant, la création de tels systèmes doit faire face aux difficultés de prédiction du comportement global non représenté explicitement dans le système, et aussi on peut avoir des difficultés à contrôler les comportements individuels par rapport à un objectif non représenté explicitement.

##### ***b) SMA cognitif***

Un système multi-agents cognitif est composé d'agents cognitifs. L'étude de ce système cherche à améliorer les comportements individuels des agents en s'intéressant à leur intelligence individuelle, leur modèle cognitif, et leurs communications. Ce type de système met l'accent sur l'agent et ses capacités.

Ce système est facile à concevoir pour les applications dont l'évolution est prévisible, aussi il tire profits des mécanismes de représentation complexe, et permet l'échange d'information. Cependant, il présente quelques inconvénients tels que : la difficulté de représenter les connaissances dans les problèmes complexes, la complexité de communications entre agents, la faible performance pour des actions en temps réel, temps important de réalisation des tâches et l'impossibilité d'adaptation pour un environnement dynamique.

En général, la différence entre les agents réactifs et les agents cognitifs peut être expliquée par le compromis efficacité/complexité. Nous partageons l'opinion d'un nombre important de chercheurs dans le domaine, qui affirment qu'aucune des architectures (réactive et cognitive) n'offre une solution unique.

Le tableau suivant résume les différences entre les modèles cognitifs et les modèles réactifs.

Système d'agents cognitifs	Système d'agents réactifs
Représentation explicite de l'environnement	Pas de représentation
Tiennent compte de leur passé	Pas de mémoire de leur passé
Agents complexes	Agent simple (fonctionnement stimulus/réaction)
Petit nombre d'agents	Grand nombre d'agents

**Tableau 1.1** Comparaison entre agents cognitifs et agents réactifs

En fonction du contexte problème, l'une des architectures est plus adaptée que l'autre. La distinction réside avant tout dans l'objectif lié à l'étude du système, comme indiqué dans [Parunak, 1999] :

- L'étude des SMA cognitifs cherche à améliorer les comportements individuels des agents en s'intéressant à leur intelligence individuelle, leur modèle cognitif, et aux communications. Elle met l'accent sur l'agent et ses capacités.
- L'étude des SMA réactifs cherche à comprendre le fonctionnement du système comme un tout, en se focalisant sur les interactions et la dynamique qui en résulte, donc sur les aspects collectifs du système.

L'une des caractéristiques importantes qui constitue l'essence d'un SMA est l'interaction. Grâce à elle, les agents vont pouvoir produire des comportements collectifs complexes dont les conséquences exercent en retour une influence sur le comportement futur du SMA. Qu'ils soient réactifs ou cognitifs, le problème principal à aborder est « comment faire interagir les agents afin qu'ils accomplissent leurs tâches, et atteignent les buts pour lesquels ils ont été conçus.

## 1.4 Interaction dans un SMA

### 1.4.1 Définition

Jacques Ferber [Ferber, 1995] définit l'interaction comme «Un ensemble de comportements résultant du regroupement d'agents qui doivent agir pour satisfaire leurs objectifs en tenant compte des contraintes provenant des ressources plus ou moins limitées dont ils disposent et de leurs compétences individuelles».

Une interaction est donc une mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques. Elle représente l'élément essentiel et fondateur des formes d'intelligence.

Les agents interagissent le long d'une suite d'événements pendant lesquels ils sont d'une certaine manière en contact les uns avec les autres, que ce contact soit direct ou qu'il s'effectue par l'intermédiaire d'un autre agent ou de l'environnement.

### 1.4.2 Situations d'interaction

Les situations d'interactions sont diverses et nombreuses. Leurs classification se fait selon plusieurs critères :

- La présence d'objectifs communs ou compatibles.
- L'accès à des ressources communes.
- La répartition des compétences au sein des agents.

### 1.4.3 Types d'interaction

Les situations d'interactions s'expriment comme des attitudes adoptées entre les agents. Elles sont classées en trois catégories de la manière suivante :

- *Indifférence des agents* : Dans ce cas les agents prônent l'indépendance. Cette situation ne pose aucun problème d'interaction de points de vue multi-agents et se résume à la simple juxtaposition des actions des agents pris indépendamment sans qu'il y ait effectivement d'interaction du fait qu'il n'existe aucun point de rencontre entre eux. Aucun agent n'a besoin des autres agents pour atteindre son objectif, les ressources sont suffisantes et leurs buts sont indépendants, dans telle situation chaque agent poursuit son propre planning indépendamment des autres. Il peut aussi interagir avec les autres pour atteindre ces propres objectifs. Dans une situation d'indépendance, les agents peuvent même avoir les mêmes buts mais chacun d'entre eux travaille indépendamment des autres.

Dans ces situations la satisfaction de l'un des buts des agents n'est pas affectée ni négativement ni positivement par celle de l'autre.

• *Coopération des agents* : Contrairement à la situation d'indifférence, dans les situations coopératives la satisfaction de l'un des buts des agents est renforcée par la satisfaction de celle de l'autre. On distingue trois situations possibles vis-à-vis les compétences des agents et les ressources utilisées : Collaboration simple, encombrement et collaboration coordonnée.

○ *Collaboration simple (ressources suffisantes, compétences insuffisantes)*: La collaboration simple consiste en une simple addition des compétences pour réaliser les tâches jugées irréalisables individuellement. Puisque les ressources sont suffisantes, cette situation ne nécessite pas d'actions supplémentaires de coordination entre les intervenants.

○ *Encombrement (ressources insuffisantes, compétences suffisantes)*: Les agents se gênent mutuellement dans l'accomplissement de leurs tâches, du fait de l'insuffisance des ressources. Puisque leurs buts sont compatibles, ils essayent de coordonner leurs actions pour vaincre l'encombrement.

○ *Collaboration coordonnée (ressources insuffisantes, compétences insuffisantes)*: Représente une situation où les agents doivent collaborer pour combler leur manque de compétences. En plus de la collaboration, les agents doivent coordonner leurs actions à cause de l'insuffisance des ressources.

• *Antagonisme entre les agents*: Dans les situations d'antagonisme, la satisfaction de l'un des buts des agents est affectée négativement par celle de l'autre. Quatre situations sont possibles : compétition individuelle pure, conflit individuelle sur les ressources, compétition collective pure et conflit collective sur les ressources.

○ *Compétition individuelle pure (ressources suffisantes, compétences suffisantes)*: Les ressources ne sont pas limitées. L'accès aux ressources ne constitue pas l'enjeu du conflit. Les agents luttent pour atteindre leurs objectifs individuels.

- *Conflit individuel pour les ressources (ressources insuffisantes, compétences suffisantes)*: Dans ce cas les agents luttent pour atteindre leurs objectifs individuels et luttent aussi sur les ressources limitées. Chaque agent essaye d'acquérir les ressources pour pouvoir atteindre ces buts. Cette situation pousse les agents à négocier l'accès aux ressources.
- *Compétition collective pure (ressources suffisantes, compétences insuffisantes)*: Les agents n'ont pas des compétences suffisantes. Ils vont se regrouper au sein de coalitions ou d'associations pour parvenir à atteindre leurs objectifs, l'interaction entre les agents d'un même groupe est collaboration simple. Les collections d'agents (les groupes) entrent en compétitions pour atteindre leurs objectifs.
- *Conflits collectifs pour des ressources (ressources insuffisantes, compétences insuffisantes)*: Dans ce cas les groupes d'agents formés à cause de l'insuffisance de compétences, en plus de leurs compétitions dues à l'incompatibilité de leurs buts, ils entrent en compétition sur les ressources vues qu'elles sont limitées

#### 1.4.4 Conditions d'interaction

La notion d'interaction suppose les conditions suivantes [Ferber, 1995]:

1. La présence d'agents capables d'agir et/ou de communiquer.
2. Des situations susceptibles de servir de point de rencontre entre agents.
3. Des éléments dynamiques permettant des relations locales et temporaires entre agents.
4. Un certain “jeu” dans les relations entre les agents leur émettant à la fois d'être en relation, mais aussi de pouvoir se séparer de cette relation. Si des agents sont totalement liés par un couplage fixe, leur interaction devient rigide et ils n'interagissent plus au sens plein du terme.

Ferber [Ferber, 1995] donne une classification des situations d'interaction abordée selon le point de vue d'un observateur extérieur. Cette classification présente les différentes situations d'interaction que l'on retrouve en fonction des objectifs des agents (compatibles ou incompatibles), des ressources dont ils disposent (suffisantes ou insuffisantes) et de leurs compétences pour la résolution d'un problème.

Buts	Ressources	Compétences	Situation	Remarques
Compatibles	Suffisantes	Suffisantes	Indépendance	Situation d'indifférence
Compatibles	Suffisantes	Insuffisantes	Collaboration simple	Situations de coopération
Compatibles	Insuffisantes	Suffisantes	Encombrement	
Compatibles	Insuffisantes	Insuffisantes	Collaboration coordonnée	
Incompatibles	Suffisantes	Suffisantes	Compétition individuelle pure	Situations d'antagonismes
Incompatibles	Suffisantes	Insuffisantes	Compétition collective pure	
Incompatibles	Insuffisantes	Suffisantes	Conflits individuels pour des ressources	
Incompatibles	Insuffisantes	Insuffisantes	Conflits collectifs pour des ressources	

**Tableau 1.2.** Classification des situations d'interaction [Ferber, 1995]

Il aboutit ainsi à une première typologie des situations d'interactions : indépendance, collaboration simple, encombrement, collaboration coordonnée, compétition individuelle pure, compétition collective pure, conflit individuel, conflits collectifs. Le tableau met en évidence l'influence de trois critères (objectifs, compétences, ressources) sur la situation d'interaction induite. En l'occurrence, la compatibilité des objectifs induira des situations de coopération dès que les ressources ou les compétences sont insuffisantes.

Cette classification permet de distinguer des situations d'interaction. En revanche, elle ne propose pas de moyens aux agents pour détecter le type de situation dans laquelle ils se trouvent ainsi que leur besoin de coopérer.

Un SMA se distingue, donc, par une collection d'agents indépendants qui interagissent en vue de réaliser conjointement une tâche ou pour atteindre conjointement un but particulier. Les interactions sont non seulement la conséquence d'actions effectuées par plusieurs agents en même temps, mais aussi l'élément nécessaire à la constitution d'organisations.

#### 1.4.5 Formes d'interaction

Les interactions peuvent être mises en œuvre pour différents objectifs tels :

- La coopération entre les agents, lorsqu'ils ont des buts communs,
- La coordination, c'est à dire d'organisation pour éviter les conflits et tirer le maximum de profit de leurs interactions afin de réaliser leurs buts,
- La compétition, lorsque les agents ont des buts antagonistes.

La coopération et la coordination sont les formes générales d'interaction les plus étudiées dans les SMA. Elles représentent l'attitude sociale qui permet l'augmentation des performances de groupe, [Doran et al., 1997; Panait et al., 2005 ; Picard et al., 2006]. Dans ce qui suit, un intérêt particulier est adressé à ces deux formes d'interaction.

## 1.5 Coopération dans les SMA

La coopération est une caractéristique essentielle en univers multi-agents pour l'exécution d'une tâche. En effet, chaque agent ne possède qu'une vue partielle de l'environnement auquel il appartient. Grâce à la coopération, il peut accomplir cette tâche avec beaucoup plus de performance. Très tôt, les chercheurs se sont intéressés à la formulation de ce concept utilisé pour la résolution du problème et aussi pour que le système ait un fonctionnement optimal [Wooldridge et al., 1995]. L'aspect collectif lors de la résolution d'un problème et plus exactement la coopération des agents est un point important dans le domaine des SMA.

Pour [Ferber, 1995], la coopération revêt deux dimensions :

- La coopération peut être une attitude des agents qui décident de travailler en commun
- Un comportement est qualifié de coopératif à partir des caractéristiques sociaux tels que le nombre de communications effectuées ou l'interdépendance des actions.

Ferber considère que plusieurs agents coopèrent pour atteindre un objectif commun si l'une des deux conditions est vérifiée :

- a. L'ajout d'un nouvel agent accroît différemment les performances du groupe.
- b. Il existe des conflits potentiels d'accession à des ressources et l'action des agents sert à éviter ou à sortir de tels conflits.

Dans [Demazeau et al., 1990], les auteurs parlent de coopération pour une tache locale, lorsqu'un agent a besoin de coopérer avec autrui parce qu'il n'est pas capable de l'accomplir par lui-même ou parce que les autres peuvent l'accomplir de manière plus efficace que lui. Cependant, le terme de « coopération n'est pas adéquat dans le cas des agents réactifs ». Il est substitué au terme de « co-action ». Pour garantir la cohérence globale d'un SMA, il est nécessaire que les agents coordonnent leurs différentes actions.

## 1.6 Coordination dans les SMA

La coordination d'actions est l'ensemble des activités supplémentaires qu'il est nécessaire d'accomplir dans un environnement multi-agents dans la création d'un ordre commun au sens physique. Ainsi, dans une société composée d'agents autonomes qui poursuivent des objectifs individuels, la coordination est une méthode indispensable pour agencer de manière cohérente les interventions de chacun. Elle définit une relation d'ordonnancement et de dépendance entre les actions.

La coordination est nécessaire :

- Lorsque les agents ont besoin d'informations et de résultats que d'autres agents peuvent fournir ;
  - Lorsque les ressources communes sont limitées ;
  - Pour éviter des actions inutiles ;
  - Pour permettre aux agents de satisfaire des objectifs dépendants ;

De ces faits, le but de la coordination est de trouver parmi un ensemble de comportement d'agents qui interagissent, une collection de comportement qui réalise d'une façon satisfaisante les objectifs les plus importants des agents.

La coordination de multiples agents est essentielle pour la viabilité des systèmes dans lesquels ces agents partagent des ressources. La pluparts des recherches en intelligence artificielle distribuée se concentrent sur le développement de stratégie de coordination en différé.

Ces stratégies préfabriquées peuvent devenir rapidement inadéquates si le modèle du monde du concepteur du système est incomplet ou incorrect ou si l'environnement change dynamiquement. L'apprentissage est un mécanisme inestimable qui permet aux agents de développer des stratégies de coordination qui satisfont les demandes des environnements et les exigences des agents individuels.

## 1.7 Apprentissage dans les SMA

L'apprentissage est un axe de recherche de plus en plus important dans le domaine des SMA car il permet d'avoir des systèmes adéquats, performants et évolutifs. Il devient une nécessité lorsque l'environnement est changeant.

L'apprentissage des agents peut porter sur quatre axes différents [Russell et al., 2006]:

- **L'apprentissage centre agent** concerne ce qu'un agent peut apprendre sur lui-même ou sur les autres agents. Cet apprentissage porte sur le comportement de l'agent, ses stratégies et ses décisions.
- **L'apprentissage centre environnement** se focalise sur ce que l'environnement peut apprendre à l'agent. Cela peut porter sur de nombreux objets tant la diversité des environnements peut être grande. En effet, dans un environnement à forte dynamique, un agent peut apprendre sur des parties lui étant nouvellement apparues. Ce type d'apprentissage se retrouve naturellement dans des systèmes de vie artificielle.
- **L'apprentissage centre interaction** porte sur les moyens mis en œuvre par les agents pour communiquer ou interagir. Cela peut être l'apprentissage de nouveaux langages d'interactions ou bien de nouveaux thèmes de communication.
- Enfin, **l'apprentissage centre organisation** s'occupe de faire évoluer les rôles des agents au sein de leur société. Lorsque le système est supervisé, les agents sont guidés dans leur démarche, alors que pour des systèmes d'agents autonomes, les agents doivent apprendre à se réorganiser.

Pour effectuer ces différents types d'apprentissage, plusieurs moyens ont été développés. L'apprentissage peut être la conséquence d'attribution directe ou indirecte de récompenses aux agents, on parle alors d'algorithmes par renforcement.

Il peut aussi être induit par l'interaction et la coopération au sein d'un groupe. Ce dernier consiste en la capacité d'un agent à apprendre grâce aux interactions coopératives qu'il a avec ses congénères. Les agents doivent alors avoir à leur disposition des moyens de communication et des protocoles associés à l'apprentissage.

## 1.8 Communication comme moyen d'interaction

La communication caractérise tout échange d'information entre des agents. Sans communication, les agents se trouvent isolés les uns des autres et aucune coopération n'est possible. La communication constitue donc un moyen fondamental de mise en œuvre des méthodes de coopération.

Différentes catégories ont été répertoriées au cours des nombreuses études sur la théorie de la communication. Les types de communication ont été classés suivant leur fonction, les éléments de la communication ou le type de message, etc..

Deux types de communication entre agents, suivant le mode de transmission du message, ont été répertoriés [Goldman et Zilberstein, 2004]: la communication directe, la communication indirecte.

### **1.8.1 Communication directe**

Elle consiste à envoyer directement un message à un ou plusieurs agents. Lorsque le message est envoyé à un seul agent, on parle de communication point à point. Si l'information est envoyée à tous les autres agents, le message est envoyé en mode diffusion (ou broad-casting).

La communication directe peut être synchrone ou asynchrone. Une communication synchrone est une communication dans laquelle le message est reçu « en temps réel » par le destinataire, c'est le cas d'une conversation. Dans une communication asynchrone, l'envoi et la réception du message sont réalisés en temps distincts et séparés par un laps de temps plus ou moins important. Notons que l'envoi de message par communication directe est généralement intentionnel : l'agent décide d'entrer en interaction avec un ou plusieurs agents et d'envoyer une information précise.

### **1.8.2 Communication indirecte**

Elle est réalisée par modification de l'environnement ou par manipulation de connaissances communes. Les agents peuvent laisser des traces dans l'environnement afin d'indiquer aux autres quelle action a été exécutée, quel est leur état, etc. Lorsque les agents ont tous accès à une base de données commune, la communication peut s'effectuer par modification des connaissances stockées dans cette base.

Dans le cas de la communication indirecte, il n'est pas certain que tous les agents observeront ces modifications. Par conséquent, l'information peut ne pas être reçue. Notons que la communication indirecte est asynchrone. Elle peut être intentionnelle ou non. En effet, l'agent peut modifier l'environnement sans pour autant vouloir communiquer avec les autres agents.

En ce qui concerne les modes de communication, de nombreuses études se sont intéressées aux communications chez les animaux et les insectes afin de les retranscrire dans un contexte d'agents informatiques [Bonabeau et al., 1999].

## 1.9 Propriétés attendues des SMA

Les SMA proposent une approche originale de modélisation de systèmes intelligents et coopératifs. Ils se caractérisent par la présence de plusieurs agents. Ainsi, les propriétés en termes de rationalité et d'autonomie doivent être exprimées.

### 1.9.1 Rationalité

Pour [Ferber, 1995] "Un agent est dit rationnel quand il ajuste ses moyens aux fins qu'il se propose d'atteindre". C'est un agent qui agit d'une manière lui permettant d'obtenir le plus de succès possible dans la réalisation des tâches assignées. Pour mesurer, le succès d'un agent "rationnel", une mesure de performance objective est associée à une certaine tâche que l'agent doit exécuter.

Tout d'abord, la rationalité peut s'exprimer de manière différente selon les types de systèmes multi-agents:

Pour certains systèmes, chaque agent dispose d'une rationalité propre et mesure de manière individuelle sa performance. Cette vision des choses est principalement compétitive, chaque agent cherche alors à maximiser ses gains (critères de performance) éventuellement au détriment des autres agents du système.

D'autres systèmes s'intéressent à une mesure de performance globale : le système est caractérisé par une fonction globale de performance (éventuellement calculée à partir d'une combinaison de mesures de performance locales). Nous soulignons la présence de capacité sociale permettant d'estimer les comportements des autres agents pour mieux agir par la suite.

Les systèmes de cette catégorie seront intelligents, collectifs et coopératifs du fait qu'une mesure de performance globale caractérise les performances du système.

### 1.9.2 Autonomie

La notion d'autonomie d'un agent s'exprime de manière différente dans un système multi-agents. En effet, dans un SMA, l'autonomie inclut la capacité à s'adapter à l'environnement global du système mais aussi aux autres agents présents dont le comportement est inconnu et peut évoluer au cours du temps.

Généralement, l'autonomie dans un système multi-agents coopératif doit permettre à l'ensemble du système d'évoluer et de s'adapter à son utilisation.

## 1.10 Prise de décision dans les SMA

La prise de décision dans les SMA fait intervenir les deux notions importantes d'observabilité et d'incertitude qui seront développées ci-dessous.

### 1.10.1 Observabilité

L'observabilité d'un environnement caractérise l'ensemble des informations qui sont accessibles à un agent. Suivant ces informations, nous pouvons distinguer différents types d'observabilité. Un agent qui exhibe un comportement de localité et d'imprécision des perceptions, introduit la notion d'observabilité partielle. Dans ce cas, les agents seront contraints de faire face à ce manque de connaissances et devront décider au mieux comment agir en fonction des informations dont ils disposent.

Nous différencierons tout d'abord l'observabilité de l'état global du système et l'observabilité de l'état local de l'agent. Le terme état global désignera l'état du système multi-agents (agents et environnement). L'état local d'un agent sera défini selon le degré de délibération de l'agent. Il pourra correspondre aux perceptions courantes de l'agent ou bien à une représentation interne de ses connaissances.

Le degré d'observabilité influence la complexité de la prise de décision. Intuitivement, si les états de l'environnement sont connus alors il est assez facile de décider comment agir. Au contraire, si nous ne disposons que d'informations partielles sur les autres agents, nous sommes contraints d'envisager leurs comportements possibles, ce qui rend la décision très complexe.

### 1.10.2 Incertitude

Il existe plusieurs causes d'incertitude dans un système multi-agents [Russell et al., 2006]. Parmi ces causes, on peut citer :

- l'imprécision et les limitations des capteurs,
- la modélisation incomplète de l'environnement,
- la dynamique des interactions entre agents.

L'imprécision des capteurs entraîne des incertitudes sur l'information perçue par l'agent et par conséquent sur l'environnement. En fait, tout ce qui est hors de portée des capteurs se trouve alors non-observable et par conséquent incertain.

L'incomplétude de la modélisation de l'environnement, entraînent par ailleurs des incertitudes sur le résultat des actions des agents. L'incomplétude de la modélisation est essentiellement due au concepteur. Ce dernier peut, en effet, avoir une connaissance incomplète de tous les phénomènes et lois de l'environnement.

Par ailleurs, les agents évoluent dans un même environnement. Leurs actions peuvent donc entrer en interaction, conduisant alors à des effets non souhaités ou non envisagés. Le fait qu'un agent ne connaisse pas les actions entreprises par les autres agents ajoute donc un degré d'incertitude sur le résultat de ses actions.

Pour qu'un agent évolue dans un environnement déterministe, il serait nécessaire que le système soit parfaitement modélisé, que les capteurs et effecteurs des agents soient d'une précision irréprochable et que l'agent connaisse exactement les états et comportements des autres agents. Il paraît évident qu'un tel système est difficilement possible.

### **1.10.3 Délibération et décision**

L'observabilité et l'incertitude sont à l'origine de nombreuses difficultés posées lors de la prise de décision dans les systèmes multi-agents. Lorsque l'observabilité est partielle et le résultat des actions incertain, il est d'autant plus difficile pour les agents de coopérer ou de se coordonner.

Il est possible de contourner le problème en faisant appel à une entité centrale prenant toutes les décisions et coordonnant les actions de chacun. Dans certaines applications, une telle solution est cependant impraticable. Chaque agent devra donc décider de manière autonome comment agir dans un environnement incertain tout en faisant face au manque d'observabilité.

Enfin, l'utilisation des SMA se retrouve dans nombreux domaine d'application, selon leur utilité et leur intérêt.

## **1.11 Domaines d'application des SMA**

Les applications des SMA couvrent de plus en plus de domaines. Toutefois ces derniers partagent les caractéristiques suivantes [Parunak, 2000] :

- Modulaires : les applications modulaires, dont les différentes entités les composant forment un découpage précis, facilitent l'application d'un SMA. En effet, la proximité des agents avec les objets au sens informatique permet de s'adapter à une décomposition existante.
- Décentralisées : point proche du côté modulaire, l'aspect décentralisé comporte en plus une notion d'autonomie, qui peut être reliée facilement aux capacités de prises de décisions locales d'un agent appartenant à un SMA.

- **Dynamiques** : l'intérêt de l'utilisation d'un SMA pour traiter une problématique fortement dynamique dérive des deux points précédents. En effet, une application connaissant des modifications fréquentes et rapides bénéficiera d'autant plus des avantages de prises de décisions locales et de la robustesse liée à une conception modulaire.
- **Non-entièrement observables** : cette caractéristique concerne les applications pour lesquelles des résultats doivent être produits même lorsque toutes les données ne sont pas accessibles. Dans un SMA, un agent ne possédant pas toutes les informations attendues ne bloque pas le fonctionnement global du système.
- **Complexes** : enfin, les SMA s'adaptent particulièrement aux applications complexes, faisant intervenir une grande quantité de comportements différents et un nombre élevé d'interactions.

Néanmoins, les SMA développés actuellement peuvent être classés en trois catégories:

1. ***La simulation*** dont l'objectif est la modélisation de phénomènes du monde réel, afin d'observer, de comprendre et d'expliquer leur comportement et leur évolution. Ce sont par exemple, des applications de simulation de phénomènes sociaux, environnementaux, éthologiques, etc.

2. ***Les applications*** dans lesquelles les agents dits assistants jouent le rôle d'êtres humains. La notion d'agent simplifie la conception de ces systèmes et amène de nouvelles problématiques centrées utilisateur telles que la communication, la sécurité, etc.

3. Et enfin, ***la résolution de problèmes***. Il s'agit de concevoir des systèmes artificiels capables de résoudre des problèmes distribués, par la mise en œuvre au sein d'un ensemble d'agents permettant l'accomplissement collectif d'une tâche.

Ainsi, lorsqu'un problème à résoudre exhibe des propriétés de distribution, de décentralisation, ou de dynamique, les SMA peuvent se révéler être une alternative efficace [Müller, 1998 ; M.-P. Gleizes, 2004 ; Yokoo, 2001 ; Shoham et al., 2009].

Dans ce cas, la modélisation va souvent consister à considérer certaines variables, données, critères ou contraintes du problème comme des agents, en les dotant d'autonomie et de capacités d'interaction.

De manière complémentaire, certaines techniques de résolution de problèmes ont été conçues à l'origine pour répondre à des problèmes spécifiques à la communauté multiagent tels que les problèmes de formation de coalitions [Shehory et al., 1998], d'affectation de ressources [Chevaleyre et al., 2005] ou bien à des problèmes de planification multi-agents [de Weerdt et al., 2009].

Les contributions dans ce domaine vont également se baser sur des travaux provenant de la recherche opérationnelle ou de l'intelligence artificielle classique, et vont la plupart du temps reposer sur des cadres formels éprouvés comme l'optimisation sous contraintes, ou la théorie des jeux [Shoham et al., 2009].

## **Conclusion**

Pour situer le contexte dans lequel se déroule ce travail de thèse, ce chapitre a d'abord abordé la problématique de résolution de problème et les nombreuses difficultés dont se heurtent les approches de conception. Les travaux en cours pour l'étude, la compréhension, l'inspiration des phénomènes naturels et leurs applications sur des problèmes réels ont été aussi soulevés.

Ensuite, les notions d'agent et de système multi-agents ont été introduites. Ces systèmes sont composés d'entités autonomes qui interagissent et dont les compétences, les connaissances (sur les autres agents et sur leur environnement), et les ressources peuvent être limitées.

En étudiant les caractéristiques principales de ces systèmes, on peut conclure que les SMA constituent une technologie potentiellement novatrice et adéquate pour résoudre des problèmes complexes et dynamiques. Ces caractéristiques sont dérivées des interactions entre agents autonomes permettant de retracer le comportement adaptatif d'agent dans un système complexe.

La complexité des agents et des interactions peut varier selon l'objectif de la modélisation. L'interaction est, en effet, l'un des points clés dans les SMA, et mobilise un effort de recherche considérable.

Dans ce travail nous nous intéressons aux agents ayant des comportements simples, qui sont capables d'accomplir des tâches complexes dans leur dimension sociale. Ceci permettra de construire des systèmes robustes, autonomes et adaptatifs et simple à mettre en œuvre. Nous prenons donc en considération des systèmes réactifs, tout en se basant sur leur organisation, leur coopération et leur interaction fera apparaître un comportement global qualifié d'intelligent.

Nous souhaitons étudier et maîtriser l'aspect collectif de ces SMA sans augmenter les capacités des agents et par la même modifier leurs comportements individuels

La résolution de problèmes par des SMA propose une approche centrée agent. Ceci permet de fournir au système des capacités de robustesse à la dynamique du problème et la compréhension des phénomènes locaux. Néanmoins, les SMA réactifs sont conçus suivant une structure distribuée sans avoir un contrôle central du système. Aussi, le mécanisme d'adaptation du comportement complexe en fonction des interactions des entités simples du système est difficile à déduire à l'aide de méthodes analytiques.

La théorie de réorganisation qui consiste à décrire les caractéristiques d'un tel mécanisme est présentée dans le chapitre suivant.

## **CHAPITRE 2**

**De l'auto-organisation aux systèmes adaptatifs  
Mécanismes et Principes.**

## CHAPITRE II

# De l'auto-organisation aux systèmes adaptatifs Mécanismes et Principes.

### Introduction

Les recherches menées actuellement dans le domaine des SMA s'intéressent particulièrement aux systèmes dotés d'une très forte dynamique et constitués de nombreux agents. Chacun de ces agents n'a qu'une vue partielle du système dans lequel il est situé et évolue de manière autonome. Ceci crée des situations qui peuvent dégrader ou perturber la performance du système. Elles se produisent fréquemment car les agents n'ont qu'une connaissance incomplète de leur environnement.

L'objectif de ce chapitre est de fournir un cadre pour les travaux présentés dans la thèse. L'accent est mis sur les caractéristiques principales du système complexe, et plus particulièrement l'auto-organisation et l'émergence. Les deux concepts ont été longtemps considérés comme un seul, et parfois en confusion. Le fait de combiner les deux phénomènes est une approche prometteuse pour les SMA adaptatifs.

### 2.1 Organisation dans les SMA

#### 2.1.1 Introduction

Présenter la diversité des approches multi-agents existantes passe par la description de leurs différentes organisations. Celles-ci offrent une autonomie variable aux agents du système et contraint plus ou moins leurs comportements. En contrepartie, l'utilisation d'une organisation particulière permet de s'assurer que le système développé respectera une certaine cohérence vis-à-vis de son comportement global.

Etant donné que les SMA reposent sur l'idée qu'un ensemble de petites entités relativement simples peut accomplir des tâches complexes. L'autonomie des agents, leurs comportements, la richesse de leurs interactions, et l'environnement dans lequel ils sont plongés poussent le concept d'organisation encore plus loin.

Les sections suivantes détaillent les organisations des SMA les plus couramment utilisées et précisent leur impact sur le comportement des agents ainsi que sur l'évolution du SMA dans sa recherche de la solution au problème qu'il doit traiter.

### 2.1.2 Définition de l'organisation

Actuellement, il manque une définition claire et consensuelle de ce qu'on nomme «organisation». Mais avant de présenter les plus utilisées, il est important de lever une ambiguïté inhérente à ce terme. Celui-ci peut désigner aussi bien un processus qui est l'action d'organiser ou un état qui est le résultat de l'action d'organiser. Dans notre cas, c'est le deuxième sens qui est utilisé.

D'un point de vue sociologique, Morin a donné une définition riche en concepts clés : “L'organisation peut être définie comme un agencement de relations entre composants ou individus qui produit une unité ou système doté de qualités inconnues au niveau des composants ou individus. L'organisation lie de façon inter-relationnelle des éléments ou événements ou divers individus qui dès lors deviennent les composants d'un tout. Elle assure solidarité et solidité relative, donc assure au système une certaine possibilité de durée en dépit des perturbations aléatoires.” [Morin, 1977].

D'autres définitions existent dans la littérature, elles diffèrent selon le point de vue considéré par l'auteur. Nous pouvons encore citer les suivantes:

Une organisation est caractérisée par une décomposition de tâches, une distribution de rôles, un système d'autorité, un système de communication et un système de récompense et de pénalisation [Bernoux, 1987].

Une organisation est un schéma de décision et de communication appliqué à un ensemble d'acteurs qui accomplissent un ensemble de tâches pour satisfaire des buts tout en gardant un état cohérent du système [Malone, 1999].

Apparemment différentes, ces définitions de la notion d'organisation présentent quelques similitudes car, on retrouve dans chacune, les caractéristiques de structuration, de répartition des tâches et de régulation et ceci avec des formulations différentes.

Dans le contexte des SMA, sa signification varie souvent entre deux visions basiques [Castelfranchi, 1998; Scott, 1998]:

- (i) une entité collective ayant sa propre identité qui est représentée par un groupe d'agents exhibant des structures sociales hautement formalisées.
- (ii) un schéma ou structure stable d'activités conjointes qui peut contraindre ou affecter les actions et les interactions des agents pour un but donné.

Donc, une organisation fournit un moyen de partitionner le système. Chaque partition (ou groupe) définit un contexte d'interaction pour les agents. Les agents membres d'un groupe peuvent interagir librement, mais un groupe est opaque pour tous les agents qui sont externes à ce groupe. Les organisations permettent ainsi de structurer les entités du système [Ferber et al., 2003].

### 2.1.3 Objectifs de l'organisation

L'organisation d'un SMA désigne un niveau de description intermédiaire, entre l'agent et le système qui rend compte d'une forme de structuration des comportements des agents et de leurs interactions. Lorsqu'un agent est amené à interagir, l'organisation lui indique généralement avec quel autre agent le faire et comment le faire, lui évitant de faire cette recherche par lui-même [Gutknecht, 2001]. De ce point de vue, elle permet de réduire la complexité de l'espace de recherche qu'un agent doit envisager pour atteindre son but.

D'un point de vue global, l'organisation permet de limiter le nombre de conflits et donc d'augmenter l'efficacité globale de la société. Aussi, une organisation peut se référer à un schéma de coopération pouvant être plus ou moins formalisé.

C'est ce que [Parunak and al., 2004] expriment en des termes différents lorsqu'ils proposent une définition d'organisation à trois niveaux :

- (i) un ordre (ou mesure) sur les organisations, c.à.d. une application de l'ensemble des organisations vers l'ensemble des nombres réels ;
- (ii) un processus dans un système dans lequel la mesure précédente croît avec le temps (de moins organisé à plus organisé) ;
- (iii) la structure résultant de cette précédente évolution.

La représentation d'aspects organisationnels dans un SMA prend diverses formes et dénominations selon que l'organisation du SMA est implicite ou explicite, statique ou dynamique, produite ou imposée par le concepteur.

### 2.1.4. Structure organisationnelle et organisation concrète

Deux niveaux d'organisation peuvent être distingués [Ferber, 1995] et la figure 2.1 illustre cette différence.

- *La structure organisationnelle*: La structure organisationnelle est un invariant d'une classe d'organisations. C'est ce qui persiste quand les composants ou les individus rejoignent ou quittent une organisation, c'est-à-dire les relations qui font d'un agrégat d'éléments un tout.

Elle est caractérisée par :

- Des rôles affectés aux agents.
- Des relations abstraites existant entre ces rôles.

- *Organisation concrète* : *L'organisation concrète (ou simplement organisation) est une instantiation possible d'une structure organisationnelle.*

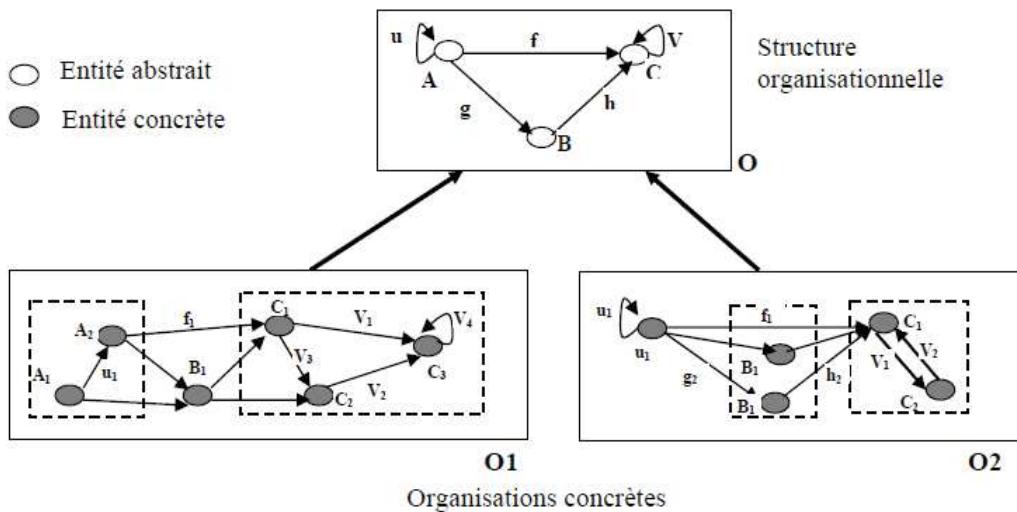


Figure 2. 1. Organisation et structure organisationnelle [Ferber, 1995].

La structure organisationnelle  $O$  est décrite par l'ensemble des rôles qui peuvent y être tenus ( $A, B, C$ ) alors que les structures concrètes comprennent un grand nombre d'entités concrètes ( $A_1, A_2, B_1, \dots$ ), les  $A_i$  étant associées au rôle  $A$ . Les liens entre les entités concrètes ( $g_1, g_2, h_1, h_2, \dots$ ) correspondent aux liens abstraits qui existent entre les entités abstraites ( $g, h, \dots$ ). Par exemple, le lien abstrait  $g$  qui lie l'entité abstraite  $A$  à  $B$  s'exprime concrètement dans l'organisation concrète  $O_1$  sous la forme de deux liens  $g_1$  et  $g_2$  qui relient respectivement  $A_2$  à  $B_1$  et  $A_1$  à  $B_1$ , alors que dans  $O_2$  des liens  $g_1$  et  $g_2$  semblables relient respectivement  $A_1$  à  $B_1$  et  $A_1$  à  $B_2$ .

Une organisation comporte un aspect structurel et un aspect fonctionnel. L'aspect structurel est lié à la structure de partition et à la structure de rôle, c'est-à-dire les relations qui sont établies entre les groupes et entre les rôles. L'aspect fonctionnel exprime comment l'activité des individus est reliée à l'activité de l'organisation.

### 2.1.5. Relations organisationnelles

En termes d'organisation nous devons modéliser les relations qu'entretiennent les agents entre eux. Les agents devant disposer d'une représentation de ces relations pour pouvoir communiquer.

On distingue deux types de relations organisationnelles :

- *les relations statiques*: qui forment le squelette d'une organisation.
- *les relations dynamiques*: qui constituent la chair des organisations et décrivent les relations qui sont modifiées lors des interactions.

Le tableau 2.1 décrit les principales relations que l'on rencontre dans les organisations multi-agents:

Relations	Statique	Dynamique
Accointance	-	-
Communicationnelle	-	-
Subordination	Maitre / esclave	Demande de service
Opérative	Dépendance entre tâches	Engagement de faire
Informationnelle	Dépendance entre savoirs	Engagement de validité
Conflictuelles	-	-
Compétitive	-	-

**Tableau 2.1.** Ensemble des relations organisationnelles [Ferber, 1995].

Ainsi, on peut distinguer sept types principaux de relations :

- *Une relation d'accointance* entre deux agents A et B indiquant que A connaît B, qu'il en a une représentation et qu'il peut adresser ses communications directement à cet agent. Il s'agit de la relation minimale entre deux agents. Elle sert généralement de support à toutes les autres relations.

- *Une relation communicationnelle*, ou plus exactement d'un canal de communication, entre deux agents A et B indiquant que A peut envoyer des messages à B. Les relations d'accointances sont les supports privilégiés de canaux de communication.

- *Les relations de subordination* décrivent un transfert d'exécution entre un agent demandeur A et un agent exécutant B. Lorsque cette relation est statique, cela signifie que B ne peut pas refuser la demande de A et donc que A et B sont dans une relation maître/esclave. Lorsqu'elle est dynamique, cela indique que A ne fait que demander des services à B et que B peut refuser les services de A.

- *Les relations opératives* représentent des dépendances liées aux tâches. Un agent A qui doit accomplir une tâche  $T_x$  peut avoir besoin, pour qu'elle soit réalisée, des résultats d'une tâche  $T_y$  d'un agent B. On dira dans ce cas, que A dépend de B pour ses actions. La forme dynamique de ces relations recouvre alors ce que l'on appelle les dépendances d'engagements. Un agent, en s'engageant à accomplir quelque chose, permet à d'autres agents de tabler sur cette réalisation et donc de planifier leurs propres actions, tout en dépendant de l'accomplissement de cet engagement.

- *Les relations informationnelles* portent sur les dépendances de validité entre les différents savoirs d'un agent. Lorsque le savoir d'un agent A dépend de celui de B, cela signifie que A n'a pas expérimenté par lui-même, mais qu'il s'en remet à B en ce qui concerne certaines croyances.

- *Les relations conflictuelles* indiquent que plusieurs agents sont en conflit pour accéder à certaines ressources. L'existence de telles relations indique qu'il est nécessaire de coordonner les accès aux ressources, éventuellement à partir d'une négociation entre agents.

- *Les relations compétitives* correspondent à une situation de compétition entre agents et signalent que leurs buts sont incompatibles.

## 2.2 Modèles d'organisation

Il existe une grande variété d'organisations et de nombreuses façons de les implémenter. Ces différentes organisations contraint de manière plus ou moins importante le comportement des agents, et par conséquent leur autonomie. Elles apportent en revanche une certaine assurance concernant la cohérence du comportement global du système développé. Les paragraphes suivants présentent quelques exemples de modèles d'organisation les plus utilisés.

### 2.2.1 Organisation AGR

La première organisation d'agents présentée est nommée AGR pour Agent-Groupe-Rôle définie dans [Ferber et al., 2003] (autrefois appelé Aalaadin [Ferber et Gutknecht, 1998]). Cette organisation repose sur les composants suivants :

– **Agent** : un agent est l'entité active et communicante, jouant un rôle au sein du groupe. Un agent peut assumer plusieurs rôles et appartenir à de nombreux groupes différents.

– **Groupe** : un groupe est un ensemble d'agents partageant des caractéristiques communes. Cette entité structure les agents par regroupement contextuel selon les activités à accomplir. Pour communiquer, deux agents doivent appartenir à un même groupe, mais un agent peut appartenir à différents groupes.

– **Rôle** : un rôle est une représentation abstraite de la fonction d'un agent au sein d'un groupe. Un agent doit jouer au moins un rôle dans le groupe, rôles locaux aux groupes. Un rôle peut être joué par plusieurs agents.

L'organisation AGR se trouve adaptée à l'application d'un SMA à un système pour lequel les rôles des différents agents sont spécifiables. De surcroît, cette approche peut être instanciée à l'aide d'une large gamme d'agents de types distincts, aucun présupposé n'étant fait sur leurs comportements ou sur leur niveau de cognition requis.

Ce type d'organisation AGR est l'un des plus répandu de par sa générnicité et sa simplicité. Il complète naturellement la définition d'un SMA.

La représentation UML du méta-modèle AGR est donnée sur la figure 2.1 ci-dessous.

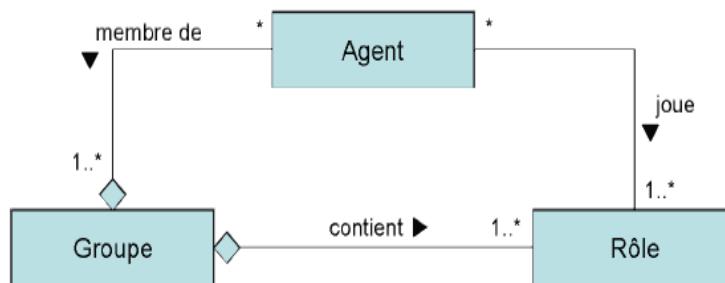


Figure 2.2. Représentation UML du méta-modèle AGR [Ferber et al., 2003]

Des travaux récents ont abouti à des extensions de ce modèle:

- AGRE [Ferber 2005] qui prend en compte l'environnement dans la modélisation ;
- AGREEN [Baez 2005] qui tente d'intégrer à la fois la dimension environnementale et normative des organisations multi-agents.

### 2.2.2 Modèle OperA

OperA est un modèle d'organisation qui spécifie les structures, les exigences et les objectifs tout en laissant aux agents la liberté d'agir selon leurs propres capacités et buts [Aldewereld et al., 2001]. L'objectif est d'apporter suffisamment de garanties sur le fonctionnement global du système tout en laissant de la marge afin d'intégrer des agents hétérogènes et de laisser place à une certaine adaptation.

OperA est en fait constitué de trois modèles interdépendants :

- **Le Modèle Organisationnel (MO)** résulte de l'analyse du domaine d'application et décrit le comportement désiré pour l'organisation. Cela comprend la définition d'objectifs, de normes, de rôles, d'interactions et d'ontologies. Le MO doit

disposer, à chaque instant, des agents adéquats pour chaque rôle. Il ne définit cependant pas de structure de groupe et ne constraint pas le comportement des agents.

– **Le Modèle Social (MS)** définit les règles de l'adoption d'un rôle du MO par un agent. Il décrit les accords que passe un agent lorsqu'il s'engage à remplir un rôle. C'est un système de contrat qui constraint le comportement de l'agent pendant la période durant laquelle il occupe un rôle.

– **Le Modèle d'Interactions (MI)** spécifie les accords entre les agents engagés dans des rôles afin d'assurer leurs interactions futures.

Ce modèle a besoin d'agents capables de décider quand rejoindre ou quitter une organisation et de passer des contrats. Il s'agit donc d'agents plutôt proactifs et bien souvent d'agents cognitifs.

### 2.2.3 Organisation holonique

Le terme « holon » est avant tout un concept philosophique. Il a été employé pour la première fois par [Koestler, 1979] et désigne une entité auto-similaire, stable, cohérente et autonome, composée de plusieurs holons ou elle-même faisant partie d'un tout d'ordre supérieur.

Son utilisation en tant qu'organisation d'un SMA permet de fournir différents niveaux d'abstractions, de nombreux points de vue sur le système ainsi que le maintien d'interactions locales entre un grand nombre d'entités. Une structure composée de holons forme une « holarchie » (figure 2.2). Ce type d'organisation présente une structure équilibrée entre autonomie et ouverture, au prix de la nécessité d'intégrer un comportement coopératif aux agents la composant.

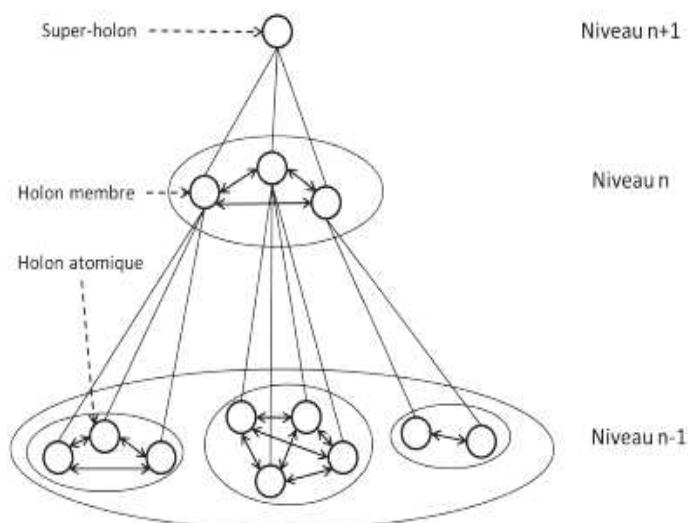


Figure 2. 3. Une holarchie à trois niveaux

Les holons présentent une imbrication de structures leur confère une place particulière. En effet, les actions d'un holon doivent satisfaire ses propres objectifs tout en favorisant l'atteinte des objectifs du holon dont il fait partie. Ainsi, il possède un comportement combinant autonomie et coopération. Les agents peuvent abandonner une partie de leur autonomie pour "fusionner" leur activité et être vus depuis l'extérieur comme un seul agent, un holon.

#### **2.2.4 Discussion sur l'organisation des SMA**

Ces trois organisations sont les plus utilisées, mais il en existe d'autres types. Ainsi, définir une organisation pour un SMA permet d'assurer son activité et son comportement global attendu. Cette assurance est obtenue au prix d'une autonomie réduite pour les agents qui doivent se conformer au rôle qui leur est attribué.

Généralement, la complexité de la tâche à effectuer par le SMA rend très difficile, voire impossible, de définir une organisation préalable capable de la réaliser. Une solution se trouve au croisement de l'apprentissage et de l'autonomie, en laissant aux agents le soin d'apprendre eux-mêmes la meilleure organisation.

En effet, l'activité d'un agent peut être vue comme l'application d'une fonction. Dans ce cas, l'organisation d'un SMA est la composition des fonctions de ses agents et ainsi l'organisation réalise la fonction globale du SMA. Apprendre cette fonction globale revient donc à trouver la bonne organisation. Cela peut se faire en modifiant le comportement d'un ou plusieurs agents, en modifiant les relations entre eux, ou en ajoutant ou supprimant des éléments dans le système. Si ces ajustements relèvent de la responsabilité des agents eux-mêmes, il s'agit d'une auto-organisation.

### **2.3 Auto-organisation**

#### **2.3.1 Introduction**

L'auto-organisation présente un intérêt certain, en déléguant les contraintes d'organisation du SMA aux agents eux-mêmes. La complexité de cette étape se retrouve de manière différente, au sein du comportement de chacun des agents et de ce fait, influe sur la manière de les développer afin d'assurer cette auto-organisation.

Cette approche est particulièrement adaptée lorsqu'il est impossible de définir une organisation appropriée et lorsque les agents sont de conception simple. Le travail du concepteur du système se focalise alors sur les comportements locaux des agents, et l'organisation globale du SMA se produira par l'ensemble de ces comportements.

L'auto-organisation est avant tout une notion physique et biologique, concernant l'état de certains systèmes capables d'exhiber des fonctions ou des formes structurées sans intervention extérieure. Ce phénomène concerne également des populations d'individus et fait également l'objet de nombreux travaux en sciences sociales.

En informatique, cette notion a été initialement utilisée dans le connexionnisme et les algorithmes génétiques. Elle est maintenant employée dans une panoplie de disciplines et d'applications parmi lesquelles s'inscrivent les SMA [Dewolf et al. 2004 ; Georgé 2004].

### 2.3.2 Définitions

Plusieurs définitions existent dans la littérature, on peut citer les plus utilisées:

L'auto-organisation est la faculté d'un système à se réorganiser de manière autonome, elle permet au système de s'adapter de manière autonome aux conditions changeantes du milieu. Un système auto-organisateur est un système qui change sa structure de base en fonction de son expérience et de son environnement [Glizez, 2004].

« L'auto-organisation est un processus dynamique et adaptatif à travers lequel les systèmes peuvent acquérir et maintenir une structure sans contrôle externe. » [DeWolf et al. 04].

Dans [Mostefaoui et al., 2006], les auteurs affirment aussi qu'un système auto-organisateur est un système qui a la faculté de former spontanément des structures bien organisées. Ces définitions permettent de dégager les propriétés communes citées ci-dessous.

### 2.3.3 Propriétés de l'auto-organisation

Quatre propriétés fondamentales caractérisent, l'auto-organisation et un système est considéré comme non auto-organisé par l'absence de l'une de ces propriétés :

- **Contrôle central :** Le système, en interaction continue avec l'environnement, est capable de recevoir des stimuli et de produire des effets, sans aucun contrôle central, ni un planning donnant au système la manière de réaction.
- **Interaction :** Sans interaction, les composantes ne sont qu'un groupe indépendant d'entités, incapables d'exprimer des comportements collectifs et cohérents.
- **Incrémantation dans l'ordre :** Le système peut avoir une incrémantation dans l'ordre comme étant le résultat d'un processus auto-organisé. Cependant l'ordre ne s'incrémentera pas indéfiniment et ne restera pas tout le temps dans une configuration d'ordre supérieur. L'ordre est en alternance entre la croissance et la décroissance.

- **Adaptabilité** : Un système auto-organisé doit être robuste vis-à-vis des perturbations, et capable de s'adapter aux changements survenus dans l'environnement.

Il existe de nombreux mécanismes permettant de reproduire une auto-organisation. Dans le cadre des SMA, ils peuvent être décomposés en plusieurs classes évoquées ci-dessous.

### 2.3.4 Mécanismes d'auto-organisation

Les mécanismes d'auto-organisation permettent aux agents de transformer l'organisation. En un sens ils décrivent les différentes étapes de l'auto-organisation. Tout d'abord, il faut souligner que l'auto-organisation est articulée autour de trois points [Calderoni et al., 1997] qui sont :

- un mécanisme qui permet de détecter les meilleures conditions d'apparition du phénomène d'auto-organisation, de décrire la sensibilité des différents paramètres qui l'influencent. Cette tâche est distribuée sur les différents composants de l'organisation. Elle consiste à trouver les agents qui sont dans un état dit instable (l'agent ne sera stable que si ses aspirations, fixées par le rôle qu'il tient dans l'organisation, sont satisfaites);
- l'apparition du phénomène : ce mécanisme est chargé d'agréger les agents et d'assurer la persistance de la structure;
- l'arrêt du développement du phénomène nécessite de repérer, avec des observations, les agents qui sont stables ; il est donc nécessaire de procéder.

Plusieurs mécanismes d'auto-organisation peuvent être identifiés :

#### 2.3.4.1 Mécanismes basés sur l'interaction directe

Ces mécanismes consistent à utiliser des principes, comme l'échange d'informations et de localisation, couplés avec des interactions locales entre agents. Le but est de produire un état global stable et cohérent. Ces approches diffèrent de l'approche traditionnelle des algorithmes distribués, par le fait qu'ils s'intéressent à la convergence et au maintien d'un état stable, en dépit d'éventuel micro niveaux d'interactions [Zambonelli et al. 2004].

#### 2.3.4.2 Mécanismes basés sur la stigmergie

Ces mécanismes visent à obtenir du système complexe en question des comportements provenant de l'interaction indirecte entre les agents. Ces interactions sont des réponses, qui sont dues aux changements survenus dans l'environnement. Le comportement final mène à la fonction globale désirée.

Plusieurs approches d'auto-organisation basées sur ce mécanisme ont été proposées et leurs efficacités à réaliser des tâches difficiles ont été démontrées. La stigmergie est parfois désignée sous le nom d'essaim intelligent, par analogie avec les insectes et parce que les agents sont homogènes. Elle est particulièrement adaptée aux problèmes d'optimisation ou de contrôle.

#### *2.3.4.3 Mécanismes basés sur le renforcement*

Dans ces approches, l'auto-organisation est basée sur la capacité des agents d'adapter dynamiquement leurs comportements selon certains renforcements. L'agent sélectionne un nouveau comportement (ou action) basé sur la probabilité, qui dépend de son état actuel, de l'état de l'environnement observé et des décisions passées [Buffet et al., 2005 ; Benhamza et al., 2008]. Le principe est que les récompenses favorisent certains comportements d'agent et les punitions les défavorisent.

#### *2.3.4.4 Mécanismes basés sur des architectures génératives*

Une classe particulière de l'auto-organisation est basée sur des architectures de références ou des méta-modèles, où l'instanciation de l'organisation des agents est modifiée d'une manière dynamique. Parmi les exemples d'architectures de références, l'architecture holonique qui a été citée précédemment. Un système centré sur l'utilisation de holons peut donc être considéré comme équivalent à un SMA dont le comportement des agents vérifie un ensemble de contraintes.

Dans de tels systèmes, chaque holon cherche à maximiser sa satisfaction, et doit pour cela se regrouper en super-holon avec d'autres agents. La satisfaction d'un holon comprend la satisfaction issue de ses propres efforts, celle découlant des autres holons avec qui il interagit, et celle relative au rôle qu'il détient.

L'auto-organisation permet donc à un système multi-agents holonique de s'adapter aux changements de son environnement. Toute fois, elle nécessite d'intégrer un comportement spécifique aux agents qui le composent afin qu'ils puissent coopérer.

#### *2.3.4.5 Mécanismes basés sur la coopération*

Les approches par coopération ont été utilisé en premier dans la théorie d'AMAS [Capera et al., 2003], pour lesquelles des comportements collectives désirés apparaissent. Chaque agent possède une capacité de s'organiser, par le réarrangement de ces interactions avec les autres agents et l'environnement.

Ces interactions dépendent des connaissances acquises, à travers la représentation de l'autre agent ou dans la tâche individuelle qui lui est assignée. Aussi, le concept d'auto-organisation semble être rattaché de façon inhérente à ce mécanisme présentant un ensemble de processus basés sur des règles locales, qui conduisent le système à produire des structures ou des comportements spécifiques non dictés par l'extérieur du système [Georgé, 2004].

#### 2.3.4.6 *Autres mécanismes*

D'autres modèles d'auto-organisation dans les SMA existent. Certains se basent sur la confiance et la réputation [Dondio et al. 2006], d'autres sont inspirés des capacités d'adaptation du système immunitaire humain [Ishida, 2004], et d'autres encore imitent la diffusion d'information (sous forme de rumeurs, ou de maladies) au sein d'une population [Jelasity et al. 2007]. Tous mettent à profit la distribution et la décentralisation des SMA pour aborder des problèmes dynamiques auxquels l'auto-organisation permet de s'adapter.

L'auto-organisation permet ainsi à un SMA d'apprendre et de s'ajuster à un environnement dynamique. Or, les SMA soulèvent notamment un besoin crucial au bon déroulement d'un processus d'auto-organisation : les agents doivent coopérer. De ce fait, la coopération est cernée comme la pierre angulaire de l'auto-organisation.

#### 2.3.5 **Observation de l'auto-organisation**

On peut distinguer trois niveaux d'observation d'un système multi-agents auto-organisé:

- un niveau externe au système, si l'on considère le système comme une boîte noire dont on ne peut observer que les entrées et les sorties ;
- un niveau interne au système si l'on prend comme objet d'étude les interactions entre les éléments de la société d'agents. L'observateur est ainsi plongé dans le système et regarde ce qui se passe entre ses différents membres ;
- un niveau interne à l'agent si l'on prend comme objet d'étude les membres de la société et plus précisément leurs architectures.

### 2.3.6 Analyse des résultats observés

Dans un premier temps, l'auto-organisation peut être analysée sans prendre en compte l'application (analyse en termes de processus). Dans un second temps, on peut évaluer la cohérence entre le produit du processus d'auto-organisation et les attentes pour l'application (analyse en termes de résultats).

- **Analyse en termes de processus.** D'une manière générale, pour qu'un processus d'auto-organisation soit intéressant et exploitable, il doit être à la fois stable, sensible et convergent. La stabilité d'un système fait référence à une constance de celle-ci dans le temps. Le système doit donc être stable afin de mettre en évidence des structures persistantes qui ne se modifieraient que sous l'influence de perturbations (internes ou externes).

La sensibilité du processus d'auto-organisation fait référence à sa capacité à faire évoluer une structure vers un état de reconnaissance (partiel ou total) selon que tous les critères retenus aient été satisfaits ou non. Il faut donc que le processus d'auto-organisation soit sensible afin qu'une remise en cause des structures qui émergent soit possible. La convergence d'un système reflète sa capacité à évoluer vers des structures connues. Le système doit donc être convergent afin que l'on aboutisse à de nouvelles structures.

- **Analyse en termes de résultats.** Comme nous l'avons vu précédemment, cette partie de l'analyse consiste à quantifier l'accord entre le processus d'auto-organisation conçu et les résultats espérés. Il faudra donc déterminer si le système est valide et/ou pertinent. Ces deux propriétés seront corrélées avec l'intérêt et la simplicité.

Le système est dit valide si le résultat produit par le processus auto-organisé est conforme avec ce que l'on attendait. L'intérêt est une grandeur qui estime la variation de cette validité en fonction de la complexité d'un raisonnement. Le système est dit pertinent si la structure peut être considérée comme simple pour ce qui est de la mise en œuvre et de la compréhension qu'en aurait un observateur. La simplicité, à l'image de l'intérêt, est une grandeur qui estime la variation de la pertinence en fonction de la complexité du raisonnement.

D'une manière générale il faut s'intéresser, lorsque l'on désire établir une bonne analyse, à différentes classes de perturbations. Injecter dans le système des événements perturbateurs crée des dysfonctionnements permettant de mesurer son degré d'auto-adaptation et d'auto-organisation. Cela permet de déterminer ses facultés à atteindre des états stables et ses performances.

- **Eléments et mesures d'évaluation.** [Serugendo et al., 2005] ont défini les éléments servant de base à l'évaluation en termes de résultats lorsque l'on s'intéresse à la conception de SMA auto-organisés:

- Une structure organisationnelle observée;
- Un processus qui produit et maintient cette structure;
- Une certaine fonction ou objectif global que le système cherche à réaliser en utilisant l'auto-organisation au maximum.

Les mesures concernant la structure du système, son processus ou sa fonction peuvent être prises en compte :

- Les mesures fondées sur la structure se concentrent sur la validation de la structure après sa stabilisation suivant une série de changements auto-organisationnels.
- Les mesures se concentrant sur le processus d'auto-organisation sont relatives à la dynamique du système et à son évolution à travers le temps.
- Les mesures se concentrant sur la fonction du système sont relatives au degré de réalisation du système de son objectif. Par conséquent, les mesures dans ce cas concernent les caractéristiques du problème que le système est dédié à résoudre et sont similaires à celles utilisées pour l'analyse de performances dans les systèmes classiques.

Des exemples de mesures typiques d'auto-organisation incluent :

- La capacité d'atteindre une organisation qui réalise les objectifs du système comme un tout (succès/échec/temps de calcul, convergence) ;
- La capacité d'atteindre une réorganisation après un événement perturbateur (succès/échec/temps nécessaire) ;
- Le degré de contrôle décentralisé (central/totalement décentralisé/hybride);

- La capacité de contenir les perturbations : stabilité/adaptabilité

Eventuellement, on s'intéresse dans un premier temps à l'observabilité du système puis, dans un second temps, à l'analyse de ces observations.

La notion « auto-organisation » s'applique dans un grand nombre de systèmes complexes où une différence est faite entre le niveau local et global. Elle est souvent rapprochée à un autre concept : celui de « l'émergence ».

## 2.4 Emergence

### 2.4.1 Définitions

L'émergence est une notion centrale de l'étude des systèmes complexes, qu'ils soient naturels ou artificiels, mais qui n'a pas de définition formelle adoptée par tous. Plusieurs auteurs avaient essayé de classifier ce terme ou au moins, cherchaient à en donner une meilleure interprétation. Dans ce qui suit, nous citerons les plus utilisées.

Un système exhibe de l'émergence quand il y a des émergents cohérent au niveau macro qui apparaissent dynamiquement des interactions entre les parties au niveau micro. De tels émergents sont nouveaux par rapport aux parties du système prises individuellement [De Wolf et al., 2004].

Cette définition utilise le concept d'« émergent » comme un terme général qui désigne le résultat du processus d'émergence : propriétés, comportements, structures, modèles, etc. Le niveau se rapporte au point de vue considéré. Le niveau macro considère le système dans son intégralité tandis que le niveau micro considère le système du point de vue des différentes entités qui le composent.

L'émergence a été aussi définie comme un phénomène observé dans un système à plusieurs niveaux : Il s'agit de l'identification par un observateur de nouvelles régularités associées à un processus qui ne peuvent être déduites à partir de la connaissance de seules propriétés des éléments constitutifs du système.

La définition plus formelle donnée par [Muller, 2002] souligne la nécessité d'un couplage du processus avec le niveau d'observation du processus. Un phénomène est émergent si :

- Il y a un ensemble d'agents en interaction entre eux et via l'environnement dont la dynamique n'est pas exprimée dans les termes du phénomène émergent à produire mais dans un vocabulaire ou une théorie « D »;

- La dynamique des agents en interaction produit un phénomène global qui peut être une structure stable, une trace d'exécution ou n'importe quel invariant statique ou dynamique;
- Ce phénomène global est observé soit par un observateur extérieur, soit par les agents eux-mêmes en des termes distincts de la dynamique sous-jacente, c'est à dire, avec un autre vocabulaire ou théorie « D ' ».

Plutôt que de prétendre donner une définition exacte et exhaustive de l'émergence, d'autres définitions sont données ci-dessous. Chacune d'elles se focalise sur une particularité plutôt que de donner une description précise.

#### *2.4.1.1 Nouveauté*

"D'une part, l'émergence pré suppose qu'il y a apparition de nouveauté – propriétés, structures, formes ou fonctions –, et d'autre part, elle implique qu'il est impossible de décrire, d'expliquer ou de prédire ces nouveaux phénomènes en termes physiques à partir des conditions de base définies aux niveaux inférieurs" [Van de Vijver, 1997].

#### *2.4.1.2 Auto-organisation*

Goldstein définit l'émergence comme l'apparition de processus d'auto-organisation. "L'émergence fait référence à l'apparition durant le processus d'auto-organisation dans un système complexe de structures ou de schémas ("patterns") ou de propriétés nouvelles et cohérentes" [Goldstein, 1999].

#### *2.4.1.3 Irréductibilité*

"L'émergence est décrite en termes d'irréductibilité des propriétés associées à une théorie de haut niveau à des propriétés associées à des composants dans une théorie de plus bas niveau" [Ali et al., 1997]. D'où, l'irréductibilité des propriétés de macro-niveau aux propriétés de micro-niveau. Ceci implique l'impossibilité de déduire, à partir des propriétés de micro-niveau, les propriétés de macro-niveau.

#### *2.4.1.4 Interdépendance des niveaux*

"Des structures et/ou des fonctions apparaissent à un niveau macroscopique sans que la seule observation des propriétés des constituants permette de les prédire" [Langton, 1990].

Ainsi, l'émergence permet de fournir un outil de contrôle capable de s'adapter aux changements dynamiques du système.

### **2.4.2 Principe de l'émergence**

La notion d'émergence peut être définie d'une manière intuitive comme une propriété macroscopique d'un système qui ne peut pas être inférée à partir de son fonctionnement microscopique [Gleizes, 2004].

Cependant, limiter l'émergence à la séparation entre des interactions de niveau local (micro) et des observations au niveau global (macro) n'est pas suffisant. D'autres critères sont à prendre en compte: le caractère dynamique du phénomène et le maintien de sa cohérence au fil du temps [Goldstein, 1999 ; Heylighen, 2001].

Ainsi, un phénomène émergent apparaît au cours de l'évolution du système, il n'est pas formé au préalable, et il se maintient durant une période suffisamment longue pour avoir une identité propre. À cela s'ajoute un troisième aspect qui est pour beaucoup au cœur de l'émergence : la nouveauté du phénomène émergent vis-à-vis du niveau micro (De Wolf et al., 2005).

Un phénomène émergent au niveau macro n'est pas prévisible depuis le niveau micro. Aussi, les entités du niveau micro n'ont aucune représentation explicite du comportement global et celui-ci ne peut pas être déduit depuis elles. De cela découle un autre critère de l'émergence, la décentralisation du contrôle : le niveau macro n'est pas directement contrôlable et aucune entité centralisée ne le maîtrise, ce contrôle passe toujours par les différentes entités du niveau micro.

C'est pour désigner cette nouveauté que l'expression populaire "le tout est supérieur à la somme des parties" est parfois utilisée [Odell, 2002]. La définition assez peu évidente d'une "somme" appliquée aux parties d'un système complexe, ainsi que l'omission des interactions de micro-niveau (qui peuvent constituer ce "plus" dans le comportement du "tout" sans pour autant parler d'émergence) doivent nous pousser à mettre de côté cette formulation.

Il demeure cependant pertinent d'appuyer sur le fait que l'émergence est caractérisée par l'impossibilité de prévoir ce qu'il advient du macro-niveau en ne se basant que sur l'étude du micro-niveau.

Lorsque l'on ne sait pas dérouler la chaîne de causalité entre les deux niveaux, c'est que l'on observe un phénomène émergent. Ces dernières considérations mettent en lumière un aspect très important (et controversé) de l'émergence : elle dépend des connaissances de celui qui l'observe.

Un observateur qui acquiert suffisamment de connaissances sur le système observé, au point de pouvoir dérouler avec exactitude la chaîne de causalité entre les niveaux micro et macro, ne verra plus de phénomène émergent se former puisque l'aspect nouveauté sera manquant.

Enfin, l'émergence peut-être vue comme la compensation de notre incapacité à suivre les multitudes d'interactions, potentiellement complexes, qui ont lieu au niveau micro. Nous séparons naturellement les phénomènes au niveau macro afin de nous donner la capacité d'étudier un système donné. C'est là toute son utilité.

#### 2.4.3 Propriétés de l'émergence

A partir du principe suscité et pour juger du caractère émergent d'un phénomène, les quatre propriétés communes ci-dessous doivent être identifiées [Georgé, 2003, 2004]:

- l'observation d'un phénomène apparent au niveau global ou macro.
- la nouveauté radicale du phénomène (non observé au niveau micro et non prévisible).
- la cohérence et la corrélation du phénomène (liée aux parties qui le produisent).
- l'observation d'une dynamique particulière.

D'après ces propriétés, nous pouvons souligner l'importance du rôle de l'observateur. Un utilisateur du SMA peut avoir le double rôle d'observer le système pour identifier et classer les formes émergentes, et d'agir sur le SMA pour orienter l'émergence. En réalité, le phénomène global identifié par un observateur possède un sens pour lui, et peut correspondre à des états intéressants ou à des solutions recherchées.

En tant qu'utilisateur, il cherche à assurer un résultat émergent. Pour y parvenir, une analyse est nécessaire : explorer les états en faisant varier la source d'imprévisibilité, et classifier les résultats en observant leurs régularités [Klein, 2009].

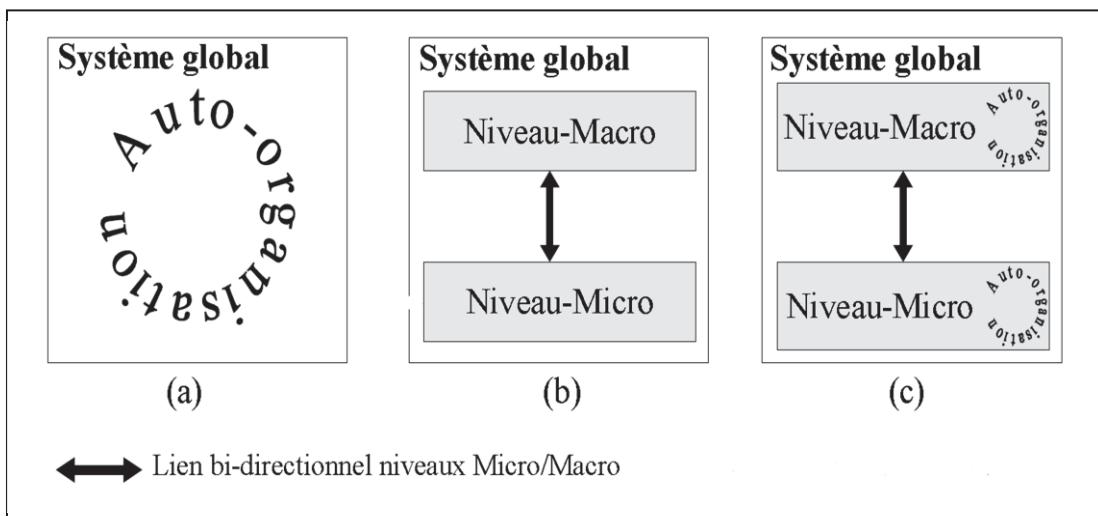
### 2.5 Auto-Organisation et émergence : similarités et différences

Une analyse comparative de l'auto-organisation et de l'émergence existe dans (De Wolf et al. 2004) et une orientation de ces concepts vers les SMA se trouve aussi dans [Serugendo et al. 2006]. La comparaison est importante dans la mesure où l'auto-organisation et l'émergence sont longtemps considérées, comme un seul concept et parfois en confusion.

Dans le plus part des cas, le processus d'auto-organisation est doté d'un comportement émergeant. En effet les composantes individuelles utilisent des règles simples, et à travers un modèle d'interaction représentant la dynamique du système, ils sont capables de produire collectivement des comportements complexes.

Camazine définissait l'auto-organisation comme un processus dans lequel un modèle de niveau global émerge d'un grand nombre de composantes en interactions. De plus, les règles spécifiant les interactions entre composantes de système utilisent uniquement des informations locales, sans référence à aucun modèle global, [Camazine et al., 2001].

En réalité nous sommes confrontés à des systèmes avec ou sans les deux concepts à la fois et la confusion vient généralement des systèmes exhibant de l'auto-organisation et aussi de l'émergence. En fait, le principe de l'émergence est l'existence d'un comportement global nouveau par rapport aux composants du système, et l'auto-organisation est un comportement adaptatif qui procure et maintien d'une manière autonome un ordre croissant.



**Figure 2.4.** Auto-organisation versus Emergence  
 (a) auto-organisation sans émergence; (b) émergence sans auto-organisation;  
 (c) émergence et auto-organisation combinées

Dans ce qui suit, les similarités et les différences entre les deux concepts sont présentées.

### 2.5.1 Les similarités

La principale similarité entre l'émergence et l'auto-organisation est l'existence d'une dynamique des processus évoluant dans le temps. L'autre caractéristique que partagent l'émergence et l'auto-organisation est la robustesse. Cependant l'émergence est robuste dans la flexibilité de ces éléments, causant cette émergence (l'échec d'un élément dans le système n'induit pas une défaillance complète du système).

La robustesse d'un système auto-organisé représente son adaptation aux changements survenus et la capacité de maintenir de l'ordre. Aussi, l'émergence et l'auto-organisation s'intéresse chacun au comportement du système.

## 2.5.2 Les différences

La différence se distingue dès lors qu'un des deux phénomènes se produit sans l'autre. Dans ce qui suit, chaque concept est présenté sans la présence de l'autre.

### 2.5.2.1 Auto-organisation sans présence de l'émergence

La figure 2.3, montre une idée sur la différence entre l'émergence et l'auto-organisation. Dans 2.3 (a), le système génère sans contrôle central une forme d'organisation à partir de la dynamique des éléments.

Les propriétés spécifiques d'émergence et qui ne sont pas nécessaire pour un processus auto-organisé sont : la nouveauté, les effets micro macro, la flexibilité des entités, et le contrôle décentralisé. Enfin, un système auto-organisé, est un système autonome qui maintient un ordre croissant à travers les interactions.

### 2.5.2.2 Emergence sans présence de l'auto-organisation

L'émergence sans auto-organisation est également possible. La figure 2.3 (b) illustre une telle situation. Le système n'a pas de contrôle externe. Le système peut exhiber des comportements chaotiques, qui émergent à travers les interactions de ses éléments.

## 2.6 Approche Prometteuse : Combinaison entre approches

La combinaison de l'émergence et de l'auto-organisation constitue une approche prometteuse et riche pour la production des comportements complexes désirés. Dans des systèmes distribués, ouverts et fortement dynamiques, il est nécessaire de garder les entités relativement simples. Mais, un individu simple ne peut pas diriger le système seul, et le comportement global cohérent doit émerger des interactions entre les individus.

Un système peut solliciter la production de comportement émergent car il est impossible d'imposer une structure initiale au système avec une complexité croissante. De ce fait, la combinaison des deux phénomènes s'explique par la nécessité du système d'acquérir des structures auto-organisées afin que le phénomène émergent puisse survenir.

### 2.6.1 Lien Auto-organisation- émergence

Combiné l'auto-organisation et l'émergence dans un seul système impose le fait de savoir comment les deux phénomènes sont liés l'un à l'autre. Pour cela différentes pistes ont étaient considérées.

- La première piste considère l'auto-organisation comme étant une cause. L'émergence dans les systèmes complexes est le résultat d'un processus auto-organisé combinés d'une manière préprogrammé ou sélective vers des comportements émergents [Camazine et al. 2001].
- La deuxième piste considère l'auto-organisation comme étant un effet. L'émergence d'un comportement se produit dans le processus d'auto-organisation. Ainsi l'auto-organisation est une propriété émergente et dans ce cas elle se produit au niveau macro, comme le montre la figure 2.3 (c).

Ce dernier point de vue est montré comme étant un résultat caractérisant l'auto-organisation, c'est-à-dire un besoin dans la croissance dans l'ordre.

### 2.6.2 Etude des phénomènes auto-organisateurs et émergents

Les notions d'émergence et d'auto-organisation sont étroitement liées. Pourtant on trouve abondamment dans la littérature des travaux se focalisant exclusivement sur cette seconde notion et évoquant peu la première. Une hypothèse pouvant expliquer cela serait peut-être le caractère abstrait de la notion d'émergence la rendant difficile à appréhender et de ce fait délicate à manipuler, en opposition à celle d'auto-organisation, plus concrète de par sa description en terme de mécanismes et donc plus aisément exploitable

Aussi, les propriétés émergentes sont liées à la complexité du système obtenue grâce aux interactions. Ces propriétés ne peuvent être étudiées que par une étude globale du système matérialisée par des niveaux hiérarchiques d'organisation. Dans le niveau micro, la dynamique est toujours très compliquée et désordonnée. L'augmentation dans l'ordre peut se produire à l'intérieur de niveau macro. Dans ce cas, l'émergence devient de plus en plus organisée. Ce point de vu peut être exprimé par «l'auto-organisation implique une croissance dans la complexité, tandis que l'émergence la réduit » [Shalizi, 2001].

### 2.6.3 Évaluation des propriétés émergentes d'un système

L'évaluation des propriétés émergentes d'un SMA est une tâche délicate mais nécessaire dès lors que l'on désire analyser la solution fournie par le système dans le cadre de la résolution de problèmes. Ainsi, dans certains cas complexes, il est nécessaire d'avoir

recours à des indicateurs permettant une meilleure analyse du système qu'une simple observation.

Dans la plus part des cas, cette évaluation des propriétés émergentes revient à mesurer une distance par rapport à un objectif à atteindre. Néanmoins, il est nécessaire de construire des indicateurs permettant de mesurer les propriétés du système. Les mesures doivent prendre en compte les propriétés locales et globales du système. Ainsi, les indicateurs utilisés peuvent être classés selon trois catégories [Gechter, 2013]:

- **Indicateurs basés sur le cadre applicatif** : Ce sont les indicateurs les plus simples à construire puisqu'ils s'appuient sur une évaluation pragmatique du phénomène émergent relativement à l'application visée.
- **Indicateurs basés sur une évaluation globale de l'organisation émergente** : Ce type d'indicateurs est également très répandu dans la littérature. Il s'agit alors de trouver une caractérisation globale de la configuration spatiale ou temporelle de la population d'agents.
- **Indicateurs globaux basés sur les mécanismes locaux d'interaction** : Cette catégorie est la plus délicate à mettre en œuvre car il faut à la fois prendre en compte les mécanismes locaux d'interaction et leurs résultats au niveau global. Parmi les indicateurs usuels entrant dans cette catégorie, nous pouvons noter l'entropie et les indicateurs s'appuyant sur la notion d'énergie.

#### 2.6.4 Contraintes et difficultés de contrôle des SMA

Actuellement, il est de plus en plus difficile de contrôler correctement l'activité des SMA situées dans des environnements de plus en plus dynamiques. A cause de l'évolution technologique, telle l'utilisation de processeurs et de moyens de traitement de l'information dans la majorité des équipements qui nous entourent, l'ouverture et la distribution s'imposent comme caractéristiques essentielles dans les systèmes informatiques de nos jours.

Ces systèmes nécessitent la réalisation d'applications caractérisées par une phase de spécification incomplète pour les raisons suivantes [Gleizes et al., 2001] :

- Le système doit évoluer dans un environnement dynamique et il est impossible de spécifier totalement toutes les situations auxquelles il aura à faire face ;
- Le système est ouvert et complexe et l'organisation interne du système n'est pas connue, a priori ;

- Il n'existe pas de solution algorithmique connue pour résoudre le problème.

### 2.6.5 Problèmes liés au développement des systèmes ouverts et dynamiques

Un système ouvert repose sur la notion de l'environnement. Il peut être défini comme un système situé qui interagit, communique et fait des échanges avec d'autres systèmes de son environnement.

Aussi, l'ajout et la suppression d'éléments sont parmi les propriétés prépondérantes de ces systèmes. Cependant, l'arrivée d'une nouvelle entité nécessite une coordination avec celles déjà existantes de façon à ce que le système conserve son bon fonctionnement.

De même, la suppression d'une entité dans un système implique que celui-ci dispose de mécanismes de gestion de départ ou de sortie, ainsi que d'adaptation de façon à maintenir une certaine stabilité. Un système ouvert est donc un système auquel on peut ajouter ou enlever de nouvelles fonctionnalités en cours d'exécution.

Le contrôle du système ouvert est étendu sur son fonctionnement pour s'assurer que tous les éléments constituant ce dernier accomplissent correctement leurs rôles. Il est préférable d'avoir une gestion décentralisée du contrôle, c'est-à-dire, plusieurs entités de contrôle interviennent dans cette gestion. En effet, une gestion de contrôle centralisée dans une seule entité met en cause la robustesse et la fiabilité du système particulièrement dans les cas de défaillance de l'entité de contrôle ou dans les cas de surcharge de cette dernière.

L'imprévu étant inhérent à la vie de ces systèmes, et pur faire face à ces difficultés, les agents doivent en effet être capables de réagir aux changements et perturbations de leur environnement complexe et dynamique. L'unique solution consiste à laisser plus d'autonomie à ces systèmes afin qu'ils évoluent au mieux aux imprévus. L'adaptation devient ainsi un moyen pour parvenir à surmonter les perturbations éventuelles de l'environnement et réaliser des systèmes robustes et intelligents.

## 2.7 Les systèmes adaptatifs

Pour être plus conforme avec les environnements d'exécution qui ne cessent de changer constamment, le développement de système exhibant les propriétés d'adaptation devient alors une exigence.

L'adaptation repose essentiellement sur le fait qu'il soit constitué d'entités séparables, c'est-à-dire qu'il doit être décomposable. Elle repose aussi sur une autre caractéristique très importante, il s'agit de la capacité du système à s'observer et donc à répondre à des changements de son état.

### 2.7.1 Dimensions l'adaptation

L'adaptation des systèmes englobe différentes dimensions. Des études [Grondin, 2008 ; Senart, 2003] ont permis d'identifier les axes autour de lesquels tourne cette notion d'adaptation.

Chaque axe représente une réponse pour l'une des questions suivantes :

- Dans quels buts adapte-t-on un système?
- Sur quels éléments du système peut porter une adaptation ?
- À quels moments peut-on adapter le système ?
- Qui décide d'adapter le système ?
- Comment réaliser cette adaptation?

#### a. Première dimension : Dans quels buts adapte-t-on un système?

Nous pouvons définir et extraire les raisons pour lesquelles une application est amenée à subir des adaptations. Généralement la communauté des chercheurs dans ce domaine semble d'accord sur la définition de quatre types d'adaptations qui se distinguent par des motivations différentes tout en notant que ces différents types ne sont que des points de vue et qu'ils ne sont pas exclusifs : l'adaptation corrective, l'adaptation adaptative, l'adaptation évolutive et l'adaptation perfective.

- *Adaptation corrective* : le but de l'adaptation est de corriger les erreurs de fonctionnement d'une entité au sein du système.
- *Adaptation adaptative* : le but de l'adaptation est de faire évoluer le système en fonction d'un éventuel changement (nouveau contexte par exemple).
- *Adaptation évolutive* : ajouter des fonctionnalités à cause de l'évolution des besoins du client.
- *Adaptation perfective* : L'objectif est d'optimiser les performances du système. Pour ce faire, on peut améliorer le comportement de certaines parties du système pour résoudre une tâche plus efficacement.

### **b. Deuxième dimension : Sur quoi porte l'adaptation ?**

On peut apercevoir un système comme une architecture composée d'un ensemble d'éléments reliés entre eux. L'adaptation d'un élément peut aller d'un simple paramétrage à une transformation complète. Par un élément on désigne une partie d'un système qui peut s'agir d'une fonction, d'une classe, d'un composant, ou toutes autres entités logicielles. Pour une liaison, l'adaptation consiste à modifier la nature de la liaison ou de l'appliquer sur d'autres couples d'éléments. Une adaptation peut donc porter sur un élément, sur une liaison entre deux éléments ou bien sur l'architecture entière.

Au niveau de l'architecture, l'adaptation consiste à ajouter, supprimer ou remplacer un élément ou une liaison ; comme il peut s'agir d'un redéploiement qui consiste à modifier la répartition d'un système sur un environnement d'exécution composé d'un ou de plusieurs sites de déploiement.

### **c. Troisième dimension : À quels moments peut-on adapter le système ?**

L'adaptation d'une application peut être effectuée à des moments différents et qui recouvrent toutes les étapes de son cycle de vie. L'adaptation peut être une *adaptation statique* et qui concerne dans ce cas que les étapes d'analyse, de conception et d'implémentation. Par exemple, les modifications des spécifications d'un élément qui est en cours de développement engendrent une adaptation statique de son implantation.

Nous parlons d'une *Adaptation semi-dynamique* pour désigner l'adaptation qui peut apparaître à l'étape de déploiement de l'application. Dans ce cas, on s'intéresse au processus d'installation du système. Des opérations d'adaptation peuvent avoir lieu pendant que le processus d'installation de l'application est en cours. En effet, lorsque le contexte de déploiement d'une application n'est pas totalement connu à l'avance, il doit être découvert dynamiquement au moment du déploiement.

En dernier, l'adaptation est dite *Adaptation dynamique* : puisque elle intervient lors de la phase d'exécution. Il est important de noter qu'il existe des systèmes critiques qui doivent s'exécuter de manière continue car ils doivent être disponibles à tout moment. Dans ces cas, il faut envisager d'adapter dynamiquement ces derniers.

Il existe encore d'autres cas où les adaptations doivent être faites sans arrêter l'application concernée par les adaptations. Par exemple, dans les cas des systèmes dont les environnements d'exécution changent constamment, il est inadapté d'arrêter les applications à chaque modification ; de même pour les applications dont l'arrêt est coûteux pour les entreprises.

#### **d. Quatrième dimension : Qui décide de l'adaptation du système ?**

L'adaptation est un processus qui se déroule en plusieurs étapes. Chacune de ses étapes fait intervenir des acteurs particuliers. Plusieurs acteurs humains ou logiciels peuvent intervenir pour déclencher le processus de l'adaptation.

Si le concepteur a anticipé l'adaptation, le système peut initier sa propre adaptation automatiquement sans intervention humaine. Dans ce cas des entités logicielles par exemple des composants ou des capteurs physiques peuvent jouer ce rôle d'initiateur d'adaptation.

Ceci devient possible si les critères de déclenchement de l'adaptation sont clairement identifiés et donc facilement automatisables. D'autre part, le déclenchement d'une adaptation peut aussi être initié par un acteur humain c'est-à-dire le concepteur ou l'administrateur.

#### **e. Cinquième dimension : comment mettre en œuvre cette adaptation ?**

Il existe plusieurs approches pour réaliser l'adaptation. Les techniques d'adaptation peuvent être basiques comme un simple *re-paramétrage* ou une reconfiguration à condition que l'adaptation soit anticipée. Dans ce cas, le système à adapter est construit selon un ensemble de paramètres prédéfinis dont les valeurs peuvent être spécifiées après la construction du système. Toujours dans le cas d'adaptation anticipée, celle-ci peut être réalisée par la *transformation de code*.

Les techniques utilisées pour mettre en œuvre l'adaptation aux niveaux d'applications informatiques peuvent aussi être catégorisées par d'autres critères. Une classification peut se faire selon l'élément sur lequel agit le processus d'adaptation. Selon cette vision, on a des techniques qui agissent sur les liaisons reliant les différents éléments d'une application, et d'autres techniques qui se contentent d'agir sur les éléments ainsi que d'autres techniques qui agissent sur l'architecture globale du système.

Pour les techniques d'adaptation qui agissent sur les éléments ou les entités du système, l'adaptation consiste à modifier l'état d'une entité ou son code. Les travaux qui s'intéressent à l'adaptation de l'architecture complète, sont basés sur des procédures de configuration et de reconfiguration.

Cette activité consiste en une recomposition d'éléments à l'intérieur de l'architecture. Cela peut donc être réalisé soit par retrait, remplacement, paramétrage de certaines entités et/ou liaisons, soit par ajout de nouvelles entités.

### 2.7.2 Conception de système adaptatif

La problématique de conception des systèmes adaptatifs n'est pas nouvelle. Depuis très longtemps, elle est abordée avec de nombreux travaux de recherches et la propriété recherchée d'adaptation a été traitée avec une grande diversité de méthodes. Cette adaptation est vue comme l'aptitude de cette dernière de se conformer à des conditions nouvelles ou différentes.

Elle correspond au processus de modification du système, nécessaire pour permettre un fonctionnement adéquat de ce dernier dans un contexte donné. Le terme adéquat signifie que le système correspond parfaitement à ce que l'on attend dans un contexte précis. Ainsi, on note l'existence d'approches qui se base sur la capacité d'un système à s'observer et donc à répondre à des questions sur son état. Cette propriété est considérée comme essentielle pour l'adaptation de systèmes complexes.

Devant le changement de conditions, on distingue deux situations différentes, la première situation présente le cas où les changements sont déjà prévus alors une étape de reconfiguration du système peut répondre aux besoins d'adaptation. Le deuxième cas où le changement des conditions n'a pas été prévu, alors il faut adapter le système en modifiant une partie ou même sa totalité. Dans notre cas, cette modification est déployée autour des métaphores organisationnelles collectives plus précisément sur le principe d'auto-organisation.

## 2.8 Adaptation dans les SMA

La notion d'adaptation caractérise un système par un comportement adéquat face aux sollicitations de son environnement dans le but de servir ses objectifs. Le système peut alors modifier son comportement face aux évolutions de son environnement. En effet, un agent s'adapte au sein d'un SMA, si face à une situation imprévue par le concepteur, l'agent ne se bloque pas mais réagit.

Deux niveaux d'adaptation peuvent être définis :

- Premièrement, l'adaptation au niveau global du système. Tout changement de l'environnement peut modifier l'organisation. Si cette réorganisation est un moyen d'adaptation général pour tout système artificiel, elle doit être conduite en autonome par le système et non par l'extérieur. Ainsi, les parties deviennent actives sur leur propre processus de changement. Ainsi, ces systèmes ne sont pas stables et cherchent un état d'équilibre. C'est le phénomène d'auto-organisation.

- Deuxièmement, l'adaptation au niveau de l'agent. Cette caractéristique peut être associée à sa capacité d'ajuster et réagir face aux variations des contraintes de l'environnement et des besoins des utilisateurs ; afin qu'il soit capable d'atteindre ses objectifs malgré des changements de l'environnement. On parle alors d'agent adaptatif.

L'adaptation pour un agent peut être structurelle. De ce fait, il s'agit d'adapter sa structure à l'évolution de son environnement. Elle peut être aussi comportementale et le processus de décision de l'agent doit s'adapter à l'évolution de son environnement [Guessoum, 2003].

L'adaptation comportementale peut être statique ou dynamique. Contrairement à la forme statique, l'adaptation dynamique est considérée comme étant la capacité de l'agent à s'adapter aux changements imprévus de son environnement par son évolution continue sans l'interruption de son exécution. Cette dernière forme est la plus sollicitée.

## 2.9 Potentiel de l'auto-organisation dans les SMA adaptatifs

La fonction globale réalisée par SMA découle de l'organisation entre ses agents. Changer cette organisation par des mécanismes d'auto-organisation revient à changer cette fonction. Cependant, la question qui se pose est « Pourrait-on établir une organisation idéale pour un problème? ».

Un moyen d'apprendre pour un système qui est plongé dans un environnement changeant est de s'y adapter. Apprendre pour le système consiste donc à transformer sa fonction actuelle de manière autonome afin de s'adapter à l'environnement c'est-à-dire à changer l'organisation entre ses agents. On parle alors d'auto-organisation. Cet apprentissage doit lui permettre d'avoir une activité "correcte" dans l'environnement dans lequel il est immergé [Gleizes et al., 2001].

Sachant que les SMA sont composés d'entités autonomes interagissant en suivant des règles locales, le but est de produire des systèmes adaptatifs capables de réaliser des tâches données. De tels systèmes doivent exhiber le comportement attendu par l'utilisateur en fonction de l'interaction du système avec son environnement et faire ainsi émerger la fonction attendue.

L'auto-organisation, qui correspond à un changement décidé de manière autonome est alors un moyen essentiel pour que les SMA puissent évoluer et s'adapter dans un environnement dynamique.

L'adaptation basée sur l'auto-organisation implique que le système et son environnement tentent de s'ajuster mutuellement pour être en interaction coopérative et implique à un autre niveau que tous les agents du système tendent à être aussi en interaction coopérative.

La fonction réalisée au niveau du collectif est émergente car les agents du système n'ont pas la connaissance de la finalité globale. La condition supplémentaire pour parler d'émergence est que le moyen pour obtenir la fonction globale du système ne soit pas codé dans la connaissance des agents [Gleizes et al., 2001].,

Ainsi, l'adaptation automatique consiste en une capacité du système à identifier les circonstances nécessitant l'adaptation, et en fonction de celles-ci, il entreprend les actions appropriées. Ceci implique que le système a la capacité de contrôler ses interactions et enfin s'auto-organiser.

## Conclusion

Beaucoup de chercheurs se sont orientés vers la problématique des systèmes auto-organisés pour lesquels la finalité n'est pas à priori connue. Dans ces approches, les agents ne se réorganisent pas pour garantir l'obtention d'un résultat particulier mais pour s'adapter uniquement aux perturbations extérieures. Les éléments constituant le système poursuivent ainsi un objectif individuel et la collectivité fournit un résultat global jugé par un observateur extérieur.

L'auto-organisation et l'émergence sont deux concepts fondamentaux des systèmes complexes. Chacun s'intéresse d'une manière différente au comportement global. Dans la plupart des systèmes ces deux concepts co-existent. Pour les SMA, le fait de combiner les deux phénomènes est une approche prometteuse pour avoir des comportements adaptatifs cohérents. Cette combinaison impose le fait de savoir comment les deux phénomènes sont liés l'un à l'autre.

La théorie des systèmes complexes ouvre une nouvelle ère dans l'étude des organisations au niveau des systèmes. Néanmoins, il ne faut pas négliger que le fait de fixer des règles entre agents pose un véritable problème de spécification des modèles que l'on souhaite analyser.

Les recherches menées se fondent sur l'idée d'interaction entre entités modifiant leur comportement selon des règles simples. Le but étant de comprendre comment de simples règles auxquelles des agents en interaction sont soumis, peuvent mener à des ordres inattendus. Néanmoins, et pour faire face aux perturbations imprévues, qui ne peuvent être évitées ou masquées à la conception, des stratégies adaptatives consistent à les considérer comme faisant partie du système. Ce qui conduit aussi à une réorganisation fréquente de ce dernier.

Enfin, la complexité du monde, et en particulier des systèmes informatiques, est maintenant telle que seules les perspectives qui mettent en avant des conceptions locales et qui recommandent avant tout la capacité d'émergence et d'auto-organisation de ces systèmes seront capables d'apporter des réponses adaptables, durables et efficaces.

## **CHAPITRE 3**

# **Paradigmes de développement des systèmes multi-agents**

## CHAPITRE III

# Paradigmes de développement des systèmes multi-agents

### Introduction

Les systèmes multi-agents (SMA) constituent aujourd’hui une nouvelle technologie pour la conception et le contrôle des systèmes complexes. Les solutions proposées par ces derniers sont prometteuses et permettent d’obtenir des systèmes flexibles et évolutifs.

Pour aborder la résolution des problèmes complexes, un effort particulier a été porté ces dernières années sur les SMA qui se basent sur les métaphores de l’organisation collective. En effet, la complexité des systèmes s’est trouvée considérablement accrue du fait de leur distribution, la grande quantité d’informations qu’ils manipulent, leur aspect coopératif, leur ouverture ainsi que leur incertitude. Il devient alors possible de concevoir un système adaptatif en ne se basant que sur les interactions entre les parties du système et son environnement. Cependant, leur mise en œuvre reste difficile. Ceci est dû au manque de standardisation des techniques d’ingénierie adaptées à ce genre de système et qui permettent un développement fiable et cohérent.

D’autre part, les méthodes orientées agents qui fournissent un cadre conceptuel visant à guider les développeurs des SMA pour l’étude et la résolution des systèmes complexes sont nombreuses. Elles ont pour objectif la conception des systèmes où les compétences sont attribuées à des entités logicielles autonomes, qui interagissent dans un environnement commun pour accomplir des tâches relevant de leurs compétences.

La classification et la comparaison de ces méthodes orientées agents ont motivé la rédaction de nombreux articles tels [Iglesias *et al.*, 1999 ; Tveit, 2000 ; Wooldridge *et al.*, 2000], ou encore celui de [Shehory *et al.*, 2001].

Dans ce chapitre, nous présentons les principes et concepts utilisés pour la modélisation des SMA ainsi que des méthodes et des plates-formes de développement nécessaire pour leur mise en œuvre.

### 3.1 Modélisation et développement des SMA

Nous commençons par présentons le formalisme de développement de SMA conçu par Ferber [Ferber, 1997 ; Ferber et al., 2003 ; Ferber et al., 2005] et qui constitue une bonne base à notre travail de recherche.

#### 3.1.1 Formalisme de développement de SMA

Ferber considère un agent en interaction avec le monde comme un système composé de deux systèmes dynamiques couplés. Le couplage s'effectue au travers des perceptions que l'agent a du monde et des actions qui modifient ce monde.

Il représente un système mono-agent par le couple  $(a, \omega)$  où  $a$  est un agent et  $\omega$  un monde décrits comme suit:

$$a = (P_a, Percept_a, F_a, Infl_a, S_a)$$

$$\omega = (E, \Gamma, \Sigma, R)$$

-  $P_a$  représente la fonction de perception de l'agent. C'est la fonction qui permet à un agent, étant dans un état, de discerner l'ensemble de données qui peuvent influencer son comportement.

- $Percept_a$  : l'ensemble des stimuli et sensations qu'un agent peut recevoir.
- $F_a$  la fonction de comportement de l'agent qui détermine l'état de l'agent à partir de ses perceptions et de son état précédent.

- $Infl_a$  : la fonction d'action de l'agent. C'est-à-dire la fonction qui tend à modifier l'évolution du monde en produisant des influences [Ferber, 96] et ce à partir d'un état interne à l'agent.

- $S_a$  : l'ensemble des états interne de l'agent
- $E$  : l'espace dans lequel l'agent évolue
- $\Gamma$  : l'espace des influences produites par l'agent et ayant comme conséquence de modifier l'évolution du monde.
- $\Sigma$  : l'ensemble des états de l'agent.
- $R$  : la loi d'évolution du monde.

Tel que :

$$P_a : \Sigma \rightarrow Percept_a$$

$$Infl_a : S_a \rightarrow \Gamma$$

$$F_a : S_a \times Percept_a \rightarrow S_a$$

$$R : \Sigma \times \Gamma \rightarrow \Sigma$$

Ces fonctions satisfont les équations suivantes qui décrivent la dynamique de l'agent en interaction avec son environnement. L'état interne d'un agent évolue au cours du temps en conséquence des perceptions qu'il reçoit à l'instant  $t$ . Ainsi que son état global qui évolue au cours du temps en conséquence des influences qu'ils produits au même instant  $t$  :

$$S_a(t+1) = F_a(s_a(t), P_a(\sigma(t))) \quad (\text{Eq. 1})$$

$$\sigma(t+1) = R(\sigma(t), \text{Infl}_a(S_a(t))) \quad (\text{Eq. 2})$$

où  $s_a$  est un élément  $S_a$  et  $\sigma$  un élément de  $\Sigma$

Dans un SMA, les différentes actions des agents sont combinées par l'intermédiaire d'un opérateur de combinaison d'influence  $\Pi$  qui prend les résultats des actions des agents et les combinent de manière simple (union des influences, sommation vectorielle, etc). On peut définir alors un SMA par un triplet  $(A, \omega, \Pi)$  :

- $A$  est un ensemble d'agents décrit comme précédemment
- $\omega$  un monde
- $\Pi$  un opérateur de combinaison d'influences

La dynamique du système est alors donnée par le système des  $n+1$  équations suivantes :

$$s_1(t+1) = F_1(s_1(t).R(\sigma(t))) \quad (\text{Eq. 3})$$

...

$$s_n(t+1) = F_n(s_n(t).R(\sigma(t))) \quad (\text{Eq. 4})$$

$$\sigma(t+1) = R(\sigma(t), \Pi_{i=1..n} \text{Infl}_i(s_i(t))) \quad (\text{Eq. 5})$$

Ainsi, pour un SMA composé de  $n$  agents, l'état interne de chaque agent, à un instant  $t+1$ , dépend de son état interne à l'instant  $t$  et de son évolution dans son environnement en suivant la même loi que suivent les autres agents du système. L'état global du système à l'instant  $t+1$  évolue en fonction de l'état précédent à l'instant  $t$  et de la combinaison des différentes influences produite par tous les agents du système. Il est primordial dans un SMA que tous les agents suivent la même loi d'évolution c.à.d. qu'ils les mêmes règles au sein du groupe. Dans ce cadre, il est possible de modéliser un problème multi-agents.

En fait, il existe une fonction de satisfaction globale (ou coût)  $h(\sigma)$  définie sur l'ensemble des états du monde (les éléments de  $\Sigma$ ) que l'on cherche à maximiser (ou à minimiser). On définit pour chaque agent une fonction de satisfaction qui associe une valeur d'un espace de satisfaction à un état interne de l'agent et que l'on note  $\mu_a(s)$ .

Donc, résoudre un problème distribué, connaissant la dynamique de l'environnement  $w$ , l'opérateur de combinaison d'influences  $\Pi$ , le nombre  $n$  d'agents, leur fonction de perception  $P_i$  et de production d'influences  $Infl_i$ , ainsi que la fonction de satisfaction globale  $h$ , revient à déterminer l'ensemble des comportements des agents, c'est-à-dire les fonctions  $F_i$ , de manière à maximiser (ou à minimiser)  $h(\sigma(t))$ .

Généralement, il est impossible non seulement de résoudre directement les équations précédentes, mais aussi de parcourir l'ensemble des états de manière à approximer ces fonctions. On fait alors l'hypothèse que l'on cherche à maximiser non pas une fonction globale sur l'état du monde, mais des fonctions de satisfactions locales propres aux agents.

Ainsi, pour résoudre ce problème, connaissant la dynamique de l'environnement  $w$ , l'opérateur de combinaison d'influences  $\Pi$ , le nombre  $n$  d'agents, leur fonction de perception  $P_i$  et de production d'influences  $Infl_i$ , ainsi que leur satisfaction locale  $\mu_i$ , et avec  $A^0$  l'ensemble des agents auxquels on s'intéresse ( $A^0 \subseteq A$ ), c'est trouver l'ensemble des  $F_i$ , de manière à maximiser  $M(A^0)$ .

$$M(A^0) = \sum_{i \in A^0} \mu_i(s_i(t)) \quad (\text{Eq. 6})$$

Etant immergé dans un SMA, un agent doit agir sur son environnement qu'il partage avec d'autres agents, interagir avec ceux-ci et obéir aux règles d'une organisation. La problématique centrale est de coordonner les agents entre eux afin d'établir un comportement global cohérent du système tout en permettant aux agents de contrôler localement leur comportement (autonomie). Quand il entre dans son organisation, un agent devient le sujet de plusieurs contraintes imposées par les entités externes (d'autres agents, l'organisation même). Ces contraintes limitent sa prise de décision et son comportement.

L'hypothèse de l'autonomie de l'agent revient à faire en sorte que le comportement d'un agent, ou la fonction  $F_i$ , dépende non seulement de la perception et de l'état interne, mais aussi de  $\mu_i(s)$ . On peut dire alors que les agents sont dirigés par leur satisfaction ou encore par leurs buts.

En d'autre terme, développer un SMA pour la résolution de problème, consiste donc à répondre un certain nombre de contraintes que l'on peut décrire ainsi:

1. Quelle est la forme de la fonction  $F_i$ , et peut-on définir une théorie générale permettant de relier le comportement à ces satisfactions?
2. Quelles sont les formes des interactions permettant à plusieurs agents de maximiser leur satisfaction ?

3. Comment faire évoluer le comportement des agents et quelles en sont les conséquences sur le comportement collectif qui en découle?
4. Enfin, comment réaliser de tels systèmes ?

Enfin, il existe plusieurs outils qui permettent le développement des SMA et qui sont décrites ci-dessous.

### **3.1.2 Aspect méthodologique**

Ces dernières années, le besoin de fournir des méthodologies, des modèles et de plates-formes se fait sentir afin de faciliter la prise en compte de la complexité des systèmes à concevoir. Les deux grandes familles de travaux concernant les méthodes orientés agents étendent les concepts soit vers des méthodes orienté objets, soit vers des méthodes issues de l'ingénierie des connaissances.

Souvent, il est d'usage de confondre les termes méthodologie et méthode et de les utiliser comme signifiant la même chose. Cependant, ces termes sont différents. Une méthode correspond en réalité à une description étape par étape de ce qui doit être fait pour accomplir une tâche ; alors qu'une méthodologie désigne un ensemble de principes généraux qui guident le choix d'une méthode adaptée aux besoins d'un projet ou d'une tâche.

Nous retiendrons la définition proposée par [Booch, 1992], qui est plus explicite et qui stipule qu'une méthodologie est : « un ensemble de méthodes appliquées tout au long du cycle de développement d'un logiciel, ces méthodes étant unifiées par une certaine approche philosophique générale ».

Elle est aussi définie comme « un processus rigoureux permettant de générer un ensemble de modèles qui décrivent divers aspects d'un logiciel en cours de développement en utilisant une certaine notation bien définie », [Arlabosse, et al., 2004].

Une méthode de développement de système est constituée donc d'un processus, d'une notation et d'outils pour servir de support à ce processus et à ces notations et/ou pour aider le développeur. Par contre, une méthodologie est un ensemble de guide qui couvre tout le cycle de vie du développement d'un logiciel : ces guides sont à la fois techniques et gèrent le projet, [Arlabosse, et al., 2004]. La méthodologie doit donc amener le concepteur à réfléchir sur la décomposition de son système et l'aider à décider si un composant sera un agent ou un simple objet.

### **3.1.3 Phases de conception**

Actuellement, il est largement admis que les phases principales d'une méthodologie sont l'analyse des besoins, la conception, le développement (appelé aussi implémentation) et la maintenance.

Ces phases constituent les différents aspects du cycle de vie d'un logiciel.

– *L'analyse des besoins*, appelée aussi spécification dans certaines méthodes, est l'étape au cours de laquelle les fonctionnalités du système sont exprimées du point de vue des utilisateurs. Cette étape peut à son tour être divisée en deux sous-étapes: les besoins préliminaires et les besoins finaux.

Les besoins préliminaires représentent un travail d'intercompréhension et/ou une description consensuelle du problème du cahier des charges entre utilisateurs et concepteurs sur ce que doit faire le système, ses limites et ses contraintes.

Les besoins finaux fournissent un modèle de l'environnement. Les objectifs sont définis et leurs priorités aussi.

– *La conception* a pour objectif de définir les entités et les mécanismes qui donnent le comportement défini par la phase d'analyse et l'architecture détaillée du système. Elle doit permettre de fournir la description des besoins de l'utilisateur relativement au domaine et à la fonction du système. Elle permet de régler les problèmes liés à l'implémentation.

– *Le développement* correspond aux phases classiques de codage, de test et d'intégration d'un logiciel. L'objectif de cette phase est de produire un code compilable ou interprétable.

– *La phase de maintenance* consiste soit à corriger d'éventuelles erreurs, soit à ajouter des fonctionnalités demandées par l'utilisateur.

## **3.2 Principales méthodes de conception des SMA**

Plusieurs méthodes de conception de SMA existent. Elles peuvent être distinguées en des méthodes de haut niveau et des méthodes de conception. Dans cette section, les principales méthodes orientées agents sont exposées. Pour la plupart, elles sont disponibles en ligne.

Les principales caractéristiques telles que l'autonomie, la réactivité, l'interaction, l'adaptation et l'organisation sont soulignées.

#### *a. GAIA*

Cette méthode est une extension des approches d'ingénierie logicielle classique [Wooldridge et al., 2000 ; Zambonelli et al., 2003]. Elle bénéficie d'une large reconnaissance dans le domaine multi-agents. Elle utilise une approche centrée sur l'organisation pour analyser et concevoir un SMA. Elle est considérée comme une étant générique et permet le développement de tous types de systèmes.

Gaia prend en considération deux niveaux : un macro-niveau (qui modélise une société d'agents) et un micro-niveau (qui se focalise sur l'agent). Grâce à la richesse de ses modèles, Gaia rend facile la manipulation d'un grand nombre de concepts multi-agents. Les agents modélisés, pouvant être hétérogènes, sont des systèmes computationnels qui vont essayer de maximiser une mesure globale.

L'organisation est statique dans le temps. La propriété d'autonomie des agents est exprimée par le fait qu'un rôle encapsule sa fonctionnalité, qui est interne et non affectée par l'environnement. La réactivité et la proactivité sont peu exprimées. le processus de développement de Gaia ne couvre que quelques phases de développement : la collecte des besoins, l'analyse et la conception.

La méthodologie Gaia oriente le développement vers la spécification des organisations, ce qui est considéré comme une manière plus "naturelle" d'appréhender le développement des SMA [Ferber et al., 2003]. Les modèles proposés par Gaia rendent la manipulation des concepts de base tels que l'agent, l'environnement, l'organisation et l'interaction facile ce qui a certainement fait sa popularité.

Par contre, plusieurs inconvénients peuvent être relevés [Azaiez, 2007]. Tout d'abord la notation adoptée dans Gaia est limitée. Elle peut être adaptée pour l'analyse des systèmes mais ne peut couvrir une conception détaillée. Nous pouvons aussi noter que la gestion de la complexité organisationnelle ou fonctionnelle n'est pas prise en compte. Gaia est limitée aux applications à agents à forte granularité, peu nombreux, et avec une organisation statique, ce qui rend le passage à l'échelle difficile. De plus, elle ne peut permettre l'application des théories d'émergence, d'auto-adaptation et d'intelligence or ces théories constituent une forte valeur ajoutée des systèmes multi-agents.

Le processus de développement couvert par Gaia est très limité et ne dépasse pas la phase de conception. Il n'y a donc pas de phase de vérification ou de validation.

### *b. MaSE*

MaSE (*MultiAgent Software Engineering*) est un exemple d'approche complète de développement de SMA, de l'analyse au déploiement, avec de nombreux modèles graphiques et une approche logique [Deloach et al., 2001]. MaSE manipule des modèles très proches de ceux utilisés en conception orientée objet (la hiérarchie, les diagrammes de séquences, les classes d'agents, les diagrammes de déploiement).

L'autonomie des agents est, comme dans Gaia, assurée par l'encapsulation des fonctionnalités dans les rôles. De même, la proactivité est spécifiée par des automates. Par contre, la réactivité et les propriétés sociales (groupe, organisation) ne sont pas clairement définies.

MaSE présente l'avantage de couvrir la totalité du cycle de vie du système, malgré un manque de gestion de qualité. De plus, malgré la large couverture du processus, MaSE n'impose l'utilisation d'aucun langage de programmation particulier et s'applique à n'importe quel domaine.

### *c. INGENIAS*

Dans INGENIAS [Gomez et al., 2002], l'approche générale pour spécifier un SMA consiste à diviser le problème en plusieurs aspects plus concrets qui forment les différentes vues du système. C'est un projet qui étend la notation UML.

INGENIAS est fondée sur la définition de méta-modèles décrivant les différents aspects d'un système multi-agents ainsi que leurs relations. Cette approche considère cinq perspectives (ou vues) dans un SMA : l'agent, l'organisation, les interactions, les tâches et les buts et l'environnement. Aussi, elle couvre les phases d'analyse, de conception, de codage et d'implémentation. Comme dans Gaia, l'autonomie des agents est assurée par l'encapsulation des buts. La réactivité des agents est obtenue grâce à la spécification d'événements et d'actions dans les modèles d'agent/rôle et d'interaction. De même, la proactivité est reliée aux buts. Les concepts sociaux propres aux SMA sont exprimés, telles les notions d'organisation ou de collaboration.

D'autre part, INGENIAS offre une manière très flexible pour le développement des SMA. Elle applique une technique d'ingénierie basée sur l'approche IDM (Ingénierie Dirigée par les Modèles) qui permet le passage des modèles vers le code. Un des grands avantages est d'avoir un environnement de développement offrant les outils nécessaires pour son application.

Cependant, on lui reprocher le manque de formalisme permettant de spécifier et de vérifier les propriétés relatives aux SMA telles que les propriétés de vivacité et sûreté. Ces propriétés, bien qu'elles soient considérées comme des propriétés comportementales devraient être prise en compte très tôt au niveau du cycle de vie.

#### *d. PASSI*

PASSI (Process for Agent Societies Specification and Implementation) est une méthode pas à pas intégrant des modèles de conception et concepts provenant de l'ingénierie orientée objet et de l'intelligence artificielle en utilisant UML [Cossentino, 2001].

L'autonomie des agents est liée aux différents rôles qui leur sont attribués. La réactivité des agents est définie dans un ensemble de modèles comportementaux, ainsi que la proactivité. Les aspects sociaux sont intégrés dans le modèle de société d'agents. PASSI est générique, pouvant s'appliquer à n'importe quel domaine et a la particularité de prendre en compte la modélisation des agents mobiles. Elle couvre pratiquement la totalité des phases du cycle de développement. Son utilisation est facile puisqu'elle est basée sur une extension du langage UML. Son application ne nécessite donc pas une expertise particulière

On peut reprocher à PASSI, le manque de formalisme qui rend difficile l'expression de certaines propriétés telle la sûreté. Aussi, le fait d'être basée sur un langage semi-formel rend les phases de tests et de vérification peu efficaces.

#### *e. PROMETHEUS*

[Padgham et al., 2003] ont développé PROMETHEUS comme une méthode complète (processus, notations et outils) pour des agents cognitifs (BDI), bien qu'aujourd'hui elle ait évolué et se dise indépendante de toute architecture. On retrouve dans PROMETHEUS les mêmes concepts que dans Gaia ou MaSE. Les agents sont à gros grains et fortement proactifs. L'autonomie des agents est assurée par l'encapsulation des buts et des plans. Bien que la notion de groupe soit avancée, les notions de rôles sont inexistantes et leur gestion non abordée. Prometheus définit des notations dédiées et réutilise UML.

Prometheus couvre une grande partie du cycle de développement, même si les besoins ne sont que peu abordés. Les phases de test, validation, déploiement et maintenance sont totalement absentes, ainsi que la gestion de qualité et de conduite de projet. Cette approche convient à des applications orientées utilisateurs, et non à la résolution de problèmes ou à la simulation.

*f. **ADELFE***

La méthode ADELFE (Atelier de DÉveloppement de Logiciels à Fonctionnalité Emergente) pour but d'aider tout développeur à concevoir des logiciels à fonctionnalités émergentes. Leur conception impose une méthode rigoureuse qui se distingue de l'approche globale descendante. Elle utilise les concepts issus de la théorie des AMAS (Adaptive Multi-Agent Systems) [Picard et al., 2004]. L'adaptation dans ces systèmes est obtenue par des interactions coopératives entre les agents autonomes. Par contre, l'autonomie d'un agent est assurée par l'encapsulation de ses compétences et de ses représentations. La réactivité est obtenue par l'utilisation de machines à états finis pour transcrire les comportements dynamiques et sociaux. Aussi, les interactions sont prises en compte dans le module « interaction » via les actions et les perceptions.

De manière générale, un agent, dans la théorie des AMAS, est doté de compétences, de possibilités de communication ou d'interaction avec les autres agents et/ou avec son environnement, de croyances sur certains autres agents du système, d'aptitudes qui lui permettent de raisonner et d'une attitude sociale basée sur la coopération. Le modèle adopté par ADELFE met en évidence ce type d'agents qui ont une représentation du monde. Les règles de coopération sont reliées à des situations non coopératives, c'est-à-dire des situations qui ne sont pas prises en charge. Celles-ci peuvent être de plusieurs types : incompréhension, ambiguïté, incompétence (la requête ne peut être satisfaite), improductivité (ne sait pas traiter les informations perçues), concurrence et conflit. Quand l'agent est face à ces situations, il devrait être capable de les résoudre, en cas d'incompréhension par exemple, il pourrait envoyer la requête vers un autre agent qu'il croit capable de la résoudre.

ADELFE est une approche intéressante puisqu'elle décrit d'une manière assez complète les caractéristiques des agents utilisés. Cependant, elle reste très spécifique aux systèmes basés sur la coopération des agents. Son modèle n'est donc pas générique. De plus, ADELFE ne peut pas être considéré comme méthode exhaustive puisqu'il est principalement centré sur la coopération entre les agents. Il ne décrit pas les autres concepts qu'on peut trouver dans la littérature tels que l'organisation, les plans, les tâches, les actions.

*g. **Approche VOYELLES***

Voyelles est une approche de haut niveau (peu de directives techniques sont données), difficile à classer en tant que méthode, mais très souvent citée car reposant sur des principes purement multi-agents [Demazeau, 2001].

Elle repose sur la décomposition de la vue d'un système suivant quatre dimensions (ou voyelles) : Agent, Environnement, Interaction, Organisation. Cette approche permet un jeu d'écriture par associativité, afin d'exprimer le type d'approche choisie pour le développement d'un système.

Les concepts comme l'autonomie, la réactivité, la proactivité ou les aspects sociaux ne sont pas clairement exprimés, mais présupposés. L'approche Voyelles laisse les concepteurs libres d'utiliser les formalismes, les notations ou langages de leur choix pour spécifier chaque lettre du système. Cependant quelques exemples permettent de les guider. Ce processus est complet, mais comme son niveau d'abstraction est élevé, aucune des phases n'est réellement détaillée.

### **3.3 Comparaison des méthodes présentées**

Globalement, aucune méthode n'est « parfaite » pour la conception des SMA. Chaque méthode possède ses spécificités en termes d'architecture, de formalisme ou de modèles. La méthode Voyelles est certainement l'approche méthodologique qui se distingue le plus des autres et qui est difficilement comparable. Aussi, on peut constater que les interactions ou les notions de rôles sont communément abordées. Par contre, les notions comme l'adaptation, l'ouverture ou l'environnement sont très souvent absentes.

Principalement, les méthodes citées se concentrent sur les phases d'analyse et de conception. Pour la plupart, leur processus s'inspire de processus orientés objet. Cette limitation peut s'expliquer par le fait que la technologie agent n'en est qu'à ses débuts et certains concepts, comme l'autonomie ou la socialité, restent difficiles à implanter. Même le concept d'agent est sujet à polémique. En effet, ceci implique le manque de formalisme et donc de validation.

De plus, le domaine de la programmation orientée agent, qui propose de nouveaux paradigmes de programmation, non forcément inspirés de l'objet, souligne la nécessité de se munir de nouveaux langages de programmation afin de pallier les difficultés des concepts agents et multi-agents.

Le tableau ci-dessus (tableau 3.1) synthétise et compare les différentes méthodes présentées.

	METHODES	GAMA	MASE	INGENIAS	PASSI	PROMETHEU	S	ADELFE	VOYELLE
PROPRIETES	Autonomie	+	+	+	+	+	+	+	±
But	±	++	++	±	++	++	+	+	±
Organisation	++	±	++	+	+	+	+	+	++
Ouverture	--	±	±	±	±	-	++	+	+
Proactivité	+	+	+	+	+	+	+	+	±
Réactivité	+	-	+	+	+	±	++	±	±
Rôle	++	++	++	++	++	-	+	+	+
Groupe	±	-	±	+	+	+	±	±	+
Environnement	±	-	-	±	±	+	+	+	+

**Notation :** (++) pour les propriétés pleinement et explicitement prises en charge ; (+) pour les propriétés prises en charge de manière indirecte ; (±) pour des propriétés potentiellement prises en charge ; (-) pour des propriétés non prises en charge ; (--) pour des propriétés explicitement non prises en charge.

**Tableau 3.1.** Synthèse de la comparaison des méthodes de conception [Beron et al., 2009]

Pour faciliter l'implémentation des SMA modélisés. Il est nécessaire d'utiliser des plateformes de développement adaptées. Dans ce qui suit nous présentons quelques-unes.

### 3.4 Plates-formes orientées Agent

La notion de plateforme est liée à l'implémentation des SMA. Elle constitue un réceptacle au sein duquel les agents peuvent évoluer. Les plateformes peuvent être classées en trois groupes :

- *Les plateformes de simulation* : elles servent à reproduire l'environnement et le comportement d'un système afin d'en étudier la dynamique (Swarm).
- *Les plateformes d'exécution* : elles fournissent des outils d'implémentations basé sur des modèles particuliers, mais elles ne sont pas associées à des méthodes (Jade et Voyager).
- *Les plateformes de développement* : elles servent de support à une méthode en fournissant des outils pour assister une démarche de conception par exemple : Agent Builder, Madkit et Zeus.

Dans ce qui suit, nous présentons ces plateformes. Cette liste n'est pas exhaustive. Elle représente cependant les plateformes les plus utilisées et les plus citées dans la littérature.

#### **a. La plate-forme Swarm**

Initialement développé au Sante Fe Institut, Swarm est un environnement de simulation de société d'agents à grande échelle et permettant de faire des "expérimentations informatiques" décentralisées à évènement discrets de systèmes complexes. C'est une plateforme générique constituée d'une bibliothèque logicielle (en Objective C, avec une interface Java) qui permet de développer des simulations à base d'agents, reconnue dans le domaine de la vie artificielle. Son utilisation passe par les étapes de création du monde virtuel, c.-à-d. environnement artificiel spatial et temporel où évoluent des entités, puis la création d'agents d'observation qui sondent le monde virtuel précédent. Enfin, le fonctionnement de l'ensemble (monde virtuel + observateurs) se fait dans une simulation discrétisée en temps avec des horloges synchronisées.

#### **b. La plate-forme Jade**

Jade [Bellifemmine, 1999], pour Agent DEvelopment framework, est un middleware écrit en langage Java et se conformant aux spécifications de la Fipa (Foundation for Intelligent Physical Agents).

Cet environnement simplifie le développement d'agent en fournissant les services de base et un ensemble d'outils pour le déverminage et le déploiement. JADE, la plus utilisée par la communauté des SMA, permet de développer et d'exécuter des applications distribuées basées sur le concept d'agents et d'agents mobiles. Les agents dans JADE sont implémentés selon six propriétés :

- Autonomie : les agents ont leurs propres threads de contrôle qui leur permet de contrôler leurs actions, de prendre leurs propres décisions afin de réaliser leurs buts mais aussi de contrôler leur cycle de vie.
- Réactivité : les agents peuvent percevoir les évènements de leur environnement et réagissent en fonction de ces évènements
- Aspects sociaux : les agents exhibent des aspects sociaux qui leur permettent de communiquer et d'interagir entre eux. La communication se fait à travers le passage de messages asynchrones. La communication est considérée comme un type d'actions et peuvent de ce fait intégrer un plan d'actions. Les messages ont une sémantique et une structure définis par le standard FIPA.

- Dynamité : les agents ont la possibilité de découvrir dynamiquement d'autres agents et de communiquer avec eux.

- Offre de service : chaque agent offre un ensemble de services, il peut enregistrer ses services et les modifier. Il a aussi la possibilité de chercher des agents qui offrent les services dont il a besoin.

- Mobilité : les agents dans Jade ont la possibilité de se déplacer. Les agents sont implémentés dans des conteneurs et ils peuvent se déplacer. La plate-forme peut être donc répartie sur un ensemble de machines et configurée à distance. Et la configuration du système peut évoluer dynamiquement.

Jade assure la sécurité en offrant aux applications des systèmes d'authentification qui vérifient les droits d'accès des agents. Plusieurs méthodes la considèrent comme plateforme cible lors de la génération de code tel que Gaia et PASSI.

#### **c. La plate-forme Agent Builder (Reticular systems Inc.)**

Agent Builder est une suite intégrée d'outils permettant de construire des agents intelligents. Cette plate-forme est adaptée pour tous types de systèmes. Cette suite logicielle commerciale en Java est composée de deux modules : Toolkit pour définir les agents et les communications et le Run-Time system qui est un interpréteur du système.

Les agents sont décrits avec le langage Radl (Reticular Agent Definition Language), proche du langage Agent-0 proposé par [Shoham, 1993], qui permet de définir les règles du comportement. Ces règles se déclenchent en fonction de certaines conditions et sont associé à des actions.

Cette plateforme est basée sur la notion d'agent, de groupe et de rôle. Elle permet la construction d'agent en spécifiant une classe d'agent abstraite. Chaque agent peut tenir différent rôles au sein de différents groupes. Cette plateforme est intéressante pour l'approche organisationnelle qu'elle met en avant lors de l'analyse et de la conception d'un SMA

#### **d. La plate-forme MadKit**

MadKit (Multi-Agents Developement kit) [Ferber, 2000] a été développée à l'Université de Montpellier II. Bien qu'elle puisse supporter le développement de divers systèmes, elle semble bien adaptée pour les applications de simulation. MadKiT est basée sur les concepts agent, groupe et rôle. Quasiment aucune contrainte n'est posée sur l'architecture interne ou le modèle de comportement de l'agent. Ce dernier est simplement décrit comme une entité autonome communicante qui joue des rôles au sein de différents groupes.

Cet outil, écrit en JAVA permet l'utilisation de différents langages de communication (KQML, XML, ...). Il est construit suivant trois principes :

- Un micro-noyau qui est très léger (gestion des groupes, rôles, cycle de vie des agents et messages)
- Une agentification des services qui utilise la notion d'agent de la plateforme elle-même pour effectuer un certain nombre de services du système. Ces agents ont des rôles dans des groupes spécifiques.
- Un modèle graphique componentiel, basé sur des JavaBeans.

#### **e. La plate-forme Zeus**

Zeus [Collis et al., 1998] est une plate-forme dédiée pour la construction rapide d'applications à base d'agents collaboratifs. Elle se prête bien aux systèmes économiques qui utilisent des applications de planification ou d'ordonnancement. Ses concepts s'appuient sur les notions d'agents ; de buts ; de tâches (que doit réaliser l'agent pour atteindre son but) ; et des faits (ce que l'agent considère comme vrai). Un agent possède une définition; Il appartient à une organisation ; Il perçoit et modifie son environnement. Il possède un cycle de vie et s'appuie sur un protocole d'interaction.

Un agent dans Zeus est constitué en trois couches : la couche de définition, qui contient les capacités de raisonnement et des algorithmes d'apprentissage, la couche organisationnelle, qui contient la base de connaissances et des accointances de l'agent, et la couche de coordination, qui définit les interactions avec les autres agents.

Zeus propose aussi un ensemble d'agents utilitaires (serveur de nommage et facilitateur) pour faciliter la recherche d'agents. Cette plateforme spécifie quatre phases pour la construction d'agents :

- l'analyse du domaine : consiste à modéliser des rôles. A ce stade aucun outil logiciel n'est fourni et le langage UML est utilisé.
- la conception : consiste à spécifier des buts et des tâches qui permettent de les réaliser. Cette spécification se fait par le langage naturel et n'est supportée par aucun outil.
- le développement : Cette phase est réalisée en quatre étapes : création de l'ontologie, création des agents, configuration de l'agent utilitaire, configuration de l'agent tâche et implémentation des agents. Des outils supportant des notations graphiques sont fournies afin d'aider à l'élaboration de ces étapes.

- le déploiement : dans cette phase, le système multi-agents est lancé. Un outil de visualisation permet le suivi de l'exécution. Cet outil permet de visualiser l'organisation, les interactions ayant lieu dans la société d'agents, la décomposition des tâches et l'état interne des agents.

En plus, ZEUS fournit un environnement de développement d'agent grâce à un ensemble de bibliothèques Java que les développeurs peuvent réutiliser pour créer leurs agents.

La majorité des plateformes ont été développées pour exploiter ou démontrer un concept ou une idée en particulier. Cependant, le développement de ces outils négligent, volontairement ou non, le développement de plusieurs dimensions essentielles à l'implémentation d'un SMA. Peu de travaux ont jusqu'ici été publiés sur l'évaluation des différents outils et environnements de développement de SMA. Toutefois, un constat est évident : les outils existants ne permettent pas encore le développement complet, et de façon relativement simple, de SMA appliqués à des systèmes réels d'envergure complexes.

Nous présentons ci-dessous quelques travaux qui ont utilisé l'approche adaptative dans la conception des SMA.

### **3.5 Travaux connexes**

Dans l'approche des systèmes multi-agents adaptatifs décrite dans [Cardon et al. 2000], les auteurs proposent un modèle de systèmes artificiels adaptatifs se basant sur l'auto-organisation, dont la principale caractéristique est la satisfaction de besoins généraux qui orientent leur comportement de manière décisive. Ces besoins multiples et contradictoires sont appelées tendances fondamentales. Le modèle proposé est une organisation multi-agents composé de deux parties une statique et l'autre adaptative. Cette dernière est doté d'un composant connexionniste (basé sur les réseaux de neurones de type cartes de Kohonen) qui permet de représenter les tendances fondamentales sur la base des comportements des agents. Un agent adaptatif est par conséquent un agent capable de générer et d'interpréter implicitement des indications relatives aux tendances fondamentales du système.

[Landau et al. 2002] propose un framework SFERES d'évolution artificielle et de simulation multi-agents permettant de faciliter l'intégration de l'apprentissage par algorithme évolutionniste dans la conception de systèmes multi-agents adaptatifs. Le framework est destiné à des concepteurs de SMA, il offre un outil dédié à une forme particulière d'apprentissage, l'évolution artificielle. Les outils offerts permettent au concepteur d'évaluer

la pertinence de plusieurs techniques pour un même problème ou celle de plusieurs architectures d'agents pour une même technique.

La théorie des AMAS (Adaptive Multi-Agent Systems) [Picard et al. 2004] est une approche théorique et pratique pour concevoir des systèmes adaptatifs. Le but de cette théorie est de permettre à un système, utilisant des critères de réorganisation locale au niveau de ses entités, d'atteindre un comportement global du système qui doit être adapté à la tâche courante. Cette théorie s'intéresse en particulier aux relations entre les entités constituantes, elle se base sur la coopération entre elles. Les situations imprévues du point de vue local de l'agent sont prises en compte comme des situations de non coopérations au niveau de l'agent qui va chercher à modifier ses relations avec les autres. Ces décisions sur la coopération sont locales aux agents et ne doivent pas être dictées par la fonction globale.

Dans les travaux [Georgé, 2004], l'auteur propose la réalisation d'un environnement de programmation émergente, permettant de concevoir des systèmes complexes adaptatifs conformément à la théorie des AMAS. L'objectif général est alors de réaliser un langage complet de programmation dans lequel chaque instruction est un agent autonome cherchant à être coopératif avec les autres agents du système, ainsi qu'avec l'environnement du système. En dotant ces agents-instructions de mécanismes d'auto-organisation, le système obtenu est capable de s'adapter continuellement à la tâche requise par le programmeur.

Gaston [Gaston, 2005] propose de doter les agents de capacités d'adaptation qui agissent sur la structure d'un réseau social dans des sociétés d'agents, il démontre que ces derniers peuvent améliorer la performance de l'organisation par adaptation de la structure du réseau utilisant seulement l'information locale et des méthodes relativement simples basées sur le Q-learning. L'adaptation s'opère sur les connexions, ajouter ou supprimer une connexion, ou par mémorisation des agents avec qui il y a eu de bonnes interactions. Les majeures considérations dans la conception des stratégies des agents est de décider quand adapter la connectivité et décider quelles connexions changer. La clé du framework proposée est de doter les agents de la capacité de détecter quand d'autres agents sont en train de faire des décisions de connectivité qui sont bénéfiques ou non.

Dans [Charrier, 2009] l'auteur propose un système multi-agents auto-organisateur adaptatif basé sur les réseaux d'itérations couplées (RIC). Ces réseaux sont constitués de cellules comportant chacune une application non linéaire à temps discret que l'on itère en la couplant avec les cellules voisines. Ce modèle de calcul est intégré dans le paradigme agent pour donner naissance à SMAL (Système Multi-agents Logistique), dont le générateur de la

dynamique décisionnelle interne des agents est basé sur ce calcul. Le modèle SMAL a été adapté à l'intelligence en essaim, où l'auteur a mis en œuvre des méta-heuristiques (intelligence en essaim) pour aborder des problèmes d'optimisation à l'aide des phénomènes d'auto-organisation et d'adaptation intégrées dans le système.

L'objectif des travaux de [Simonin, 2010], est de modéliser les comportements d'auto-organisation pour mieux les comprendre et les utiliser dans la définition du processus de résolution de problème ou de contrôle de flottilles de robots. Dans ce cadre, les recherches sont menées selon deux approches d'interactions entre agents. Dans un premier temps, l'auteur considère les interactions directes entre agents, essentiellement par signaux/perceptions, avec coordination réactive pour la résolution de problèmes spatiaux. Dans un deuxième temps, les modèles basés sur la coopération indirecte sont examinés. Aussi, l'étude s'est focalisée sur l'implémentation réelle en robotique des processus d'auto-organisation engendrés par ce principe.

Dans le travail de [Perozo, 2011], une architecture multi-agents pour systèmes émergents et auto-organisés (MASOES) est définie. Cette architecture permet la possibilité de modéliser un système émergent et auto-organisé à travers une société d'agents (homogène ou hétérogène), qui travaillent de manière décentralisée, avec différents types de comportement: réactive, imitative et cognitive. En outre, ils sont capables de modifier dynamiquement leur comportement en fonction de leur état émotionnel, de sorte que les agents peuvent s'adapter dynamiquement à leur environnement, en favorisant l'émergence de structures. Pour cela, un modèle à deux dimensions affectives avec des émotions positives et négatives est proposé. L'importance de ce modèle affectif est son utilité pour étudier certains aspects de l'interaction sociale multi-agents (influence des émotions dans les comportements individuels et collectifs des agents).

Le système multi-agents adaptatif conçu et développé au cours des travaux de [Boes, 2015] répond au défi du contrôle des systèmes complexes. Dans ce système, les agents ignorent la tâche globale que le collectif doit effectuer. Chacun se concentre sur son propre but local, tout en gardant un comportement coopératif pour résoudre les problèmes (locaux) qui se présentent à lui. Ces résolutions locales provoquent des changements au niveau des agents (ajustements de paramètres, réorganisation des interactions, création ou suppression d'agents) qui se répercutent sur le comportement global du système.

Deux types d'agents en particulier sont au centre de l'activité du système : les Agents Contrôleurs et les Agents Contextes. Les premiers sont majoritairement responsables du contrôle proprement dit, tandis que les seconds apprennent les réactions de l'environnement. Les Agents Contrôleurs se basent sur les indications des Agents Contextes pour effectuer des actions, tandis que l'apprentissage des Agents Contextes dépend directement des choix que font les Agents Contrôleurs. Ainsi, apprentissage et contrôle sont couplés et se construisent mutuellement.

## Conclusion

L'autonomie absolue, dans le sens d'indépendance à tout critère, n'existe que pour une entité isolée. A partir du moment où il y a des échanges, que ce soit de ressources, ou d'informations, le comportement de cette entité dépend de ces échanges. Quand on parle d'autonomie de décision pour les agents, le cadre est alors un peu moins strict : cela signifie que la conduite de l'agent n'est pas dictée par une autre entité. L'autonomie est alors plus une frontière subtile entre deux extrêmes : l'isolation et le contrôle total extérieur.

De plus, l'autonomie de l'agent n'est pas en soi une garantie pour l'apparition d'auto-organisation. Si chaque agent, bien qu'il soit autonome et fonctionnant avec des règles locales, possède un plan préétabli de son rôle, de son comportement et des interactions à effectuer afin que le système atteigne un but précis, l'ensemble des agents pourra se réorganiser, mais ce ne sera pas là non plus de l'auto-organisation.

Pourtant, obtenir un système répondant aux changements dans l'environnement, et de la façon la plus adaptative possible, c'est vouloir utiliser le concept d'auto-organisation. Ceci amène alors à définir une influence subtile de l'environnement sur le système au travers des agents le composant et en lui fournissant des informations utiles pour la convergence du système vers un fonctionnement spécifique, sans que ces informations explicitent ce but ou contraignent inopportunément le comportement du système. Ceci est couplé à des règles de comportement locales pour chaque agent qui n'intègrent pas en leur sein un plan explicite vers le but du système.

Pour fournir au système une fonctionnalité particulière, on peut donc avoir recours à une autre méthode que celle utilisée traditionnellement et qui consiste à procéder à une décomposition fonctionnelle du problème en un ensemble de primitives qui seront implantées dans les agents. Les avantages sont un renforcement de la robustesse du système - il est auto-organisé et adaptatif - et une simplification de la conception - il n'est pas nécessaire que le concepteur envisage toutes les possibilités pour chaque situation.

## **Partie II**

## **Contributions**

## **CHAPITRE 4**

**Proposition d'un modèle de conception d'un  
SMA adaptatif pour la résolution de problème**

# CHAPITRE IV

## Proposition d'un modèle de conception d'un SMA adaptatif pour la résolution de problème

### Introduction

Malgré, l'évolution du paradigme multi-agents, peu de méthodes existent pour son application au cadre général de la résolution de problème. Dans ce chapitre, nous abordons cette problématique en se focalisant sur un modèle de conception de SMA auto-organisés. Ces derniers présentent les propriétés intéressantes recherchées telles la simplicité d'agents, l'émergence de solution, l'adaptation et aussi la robustesse.

L'approche adaptative est particulièrement adéquate lorsqu'il s'agit d'aborder des problèmes liés à l'étude de systèmes complexes, à la conception de systèmes coopératifs ou à la résolution de problèmes distribués et ouverts. Les systèmes distribués sont caractérisés par leurs grandes complexités et par le caractère inhérent du non prédictibilité des services qu'ils fournissent. Cette complexité est due à l'impossibilité de la mise en œuvre de contrôle « *Top-Down* » du fait que les nœuds qui fournissent les services sont des systèmes indépendants qui n'obéissent pas à une seule autorité.

Le développement des applications distribuées et plus particulièrement leurs conceptions s'articulent sur cette propriété d'ouverture. De même, les méthodes de modélisation de ces systèmes à base d'agents réactifs relèvent des difficultés énormes et qui sont dues au nombre important d'agents en interactions.

L'imprévu étant inévitable à ces systèmes, l'adaptation, qui correspond à un changement décidé de manière autonome, devient un moyen adéquat pour parvenir à surmonter les perturbations éventuelles de l'environnement. La réorganisation peut être vue alors comme un processus dynamique à travers lequel ces systèmes peuvent rechercher, acquérir et maintenir une structure cohérente sans contrôle.

Dans ce qui suit, nous allons situer les SMA dans le processus de résolution de problème puis nous présenterons l'impact de l'auto-organisation dans ce processus et enfin nous exposerons le modèle générique proposé pour la conception de SMA adaptatif dans ce cadre.

#### **4.1 Résolution de problèmes et Système Multi-Agents**

Construire un SMA et l'utiliser pour la résolution d'un problème ou pour la construction d'un mécanisme de prise de décisions est naturel notamment lorsque le problème à résoudre est distribué d'une façon fonctionnelle ou topologique, ou lorsque certains couplages élémentaires entre les entités du problème sont déjà connus. De ce fait, les SMA, de par leur décentralisation et leur dimension sociale, sont à la fois :

a. *Paradigme de modélisation de problèmes exhibant de telles propriétés :*

En effet, résoudre un problème consiste à le modéliser dans un cadre formel adapté aux spécificités de ce problème. Puis à concevoir des techniques de résolution (algorithmes ou protocoles d'interaction) afin de trouver une solution à ce problème. Cependant, choisir un cadre de modélisation du problème n'est pas suffisant pour développer un SMA. Il est nécessaire d'identifier les agents du système, puis les techniques et les mécanismes d'interaction à mettre en place pour garantir le comportement souhaité du système.

b. *Champ d'application avec des problèmes spécifiques pour lesquels des solutions centralisées ne se montrent pas adaptées :*

De manière complémentaire, certaines techniques de résolution de problèmes ont été conçues à l'origine pour répondre à des problèmes spécifiques à la communauté multi-agents.

#### **4.2 L'auto-organisation dans le processus de résolution de problème par SMA**

Dans le cadre de la résolution de problèmes, le mécanisme auto-organisationnel est d'autant plus intéressant qu'il s'agit d'un processus continu, qui ne sollicite pas de reprendre la résolution d'un problème suite à une modification de ses spécifications.

En fait, cette auto-organisation des agents peut intervenir dans les deux cas suivants:

(i) *lorsque l'organisation est le résultat du processus de résolution.*

(ii) *lorsque l'organisation est un moyen de modéliser le problème qui change au cours de l'exécution.*

Dans le premier cas, l'organisation est modifiée ou adaptée par le processus de recherche d'une solution. Le résultat de l'activité du système est alors une organisation stable représentant une solution adéquate au problème posé.

Dans le second cas, le système doit être capable de fournir des solutions, alors que l'environnement et/ou l'organisation interne du système sont en train de changer. On parle alors d'adaptation à la dynamique.

### 4.3 Modèle proposé de résolution de problème par les SMA adaptatifs

Dans le cadre de cette recherche, la conception de SMA pour la prise de décision et la résolution de problèmes, les méthodes recherchées reposent sur les approches auto-organisées. En général, l'adaptation dans les SMA passe par une modification de l'organisation.

#### 4.3.1 Point de vue adopté

La conception de SMA adaptatifs se distingue de la conception des autres systèmes par au moins deux aspects :

1. **Le rôle de l'environnement**, à la fois, en tant que lieu d'inscription et en tant qu'ensemble de contraintes sur la dynamique des systèmes multi-agents.
2. **L'importance des interactions** entre agents qui surpassé l'importance de l'architecture des agents elle-même, si l'on veut que le tout soit plus que la somme de ses constituants;

Si l'on considère ces deux points, nous pouvons souligner l'existence de trois approches différentes de la conception de SMA:

- *L'approche orientée agents*, qui cible les agents individuels et propose différents formalismes de spécification de leurs comportements.
- *Les approches organisationnelles*, qui traitent de la spécification des interactions au travers de la notion de rôle, de relations entre rôles et groupes, soit pour spécifier les réseaux d'interaction statiquement ou de façon plus dynamique.
- *Les approches émergentistes*, qui distinguent un niveau micro d'agents en interaction, d'un niveau macro où un phénomène global est produit. Ce phénomène pouvant être une structure organisationnelle, la réalisation d'une tâche ou la construction d'une solution à un problème.

Ces approches sont toutes complémentaires et fondamentales lorsqu'on conçoit un SMA adaptatif. Néanmoins, par « adapter », on entend agir sur les agents ou la structure organisationnelle afin de réorganiser le système.

Ainsi, pour concevoir un SMA adaptatif et résoudre un problème de manière distribuée, il est nécessaire de considérer l'auto-organisation comme équivalente à une méthode de résolution par recherche locale décentralisée dans laquelle le système fournit toujours une solution même si celle-ci n'est pas optimale.

Le système va améliorer cette solution, par propagation locales, pour converger vers un optimum et les agents autonomes seront les moteurs de cette adaptation. Les décisions des agents s'appuient sur des informations à la fois locales et globales permettant de réduire les conflits et répondre efficacement à la dynamique de l'environnement. Dans le cas, par exemple, d'un SMA ouvert, il est nécessaire de lui donner les moyens de se réorganiser dynamiquement afin de satisfaire au mieux ses objectifs. Et de ce fait, la réorganisation est prise en charge par l'ensemble des membres de l'organisation.

### 4.3.2 Principe de conception

Au cœur de la conception d'un SMA adaptatif se trouvent les agents. Ces entités autonomes sont munies de capacités de décision et ne possède qu'une vision locale de leur environnement. Le comportement d'agent étant très simple, c'est dans et par l'environnement que le système calcule, communique et construit la solution du problème.

Par conséquent, cette solution ne peut être formulée qu'à travers le modèle de l'environnement et les interactions des agents. L'environnement joue un rôle fondamental en faisant le lien entre le problème à résoudre, ses caractéristiques statiques et dynamiques, et le processus de résolution.

De ce fait, le modèle présenté ici correspond à des stratégies de changement de l'environnement c.-à-d. changement émergent de l'organisation, dans laquelle les agents évoluent ensemble et n'auront qu'une vision limitée sur l'état de cette dernière. C'est sur cette notion d'organisation que va s'articuler le processus d'adaptation.

L'organisation est spécifiée de manière descendante avant l'implémentation des agents. Elle existe comme un modèle défini et formalisé par le concepteur du système et les agents ne connaissent rien de cette organisation. Ils s'y conforment simplement. Le schéma de coopération est préalablement fixé.

Changer d'organisation ou réorganiser le système peut se faire par des changements à différents niveaux: configuration spatiale, voisinages, rôles, différentiation/spécialisation des individus, etc. Par conséquent, l'adaptation s'effectue en changeant ces configurations de telle sorte que le système se comporte de manière plus adaptée à la pression de l'environnement perçue ou subie.

Toutefois, une organisation peut être modifiée à différents moments du cycle de vie du système. Le processus de réorganisation est une conséquence du fonctionnement du système. Cela signifie que si les agents ne vérifient pas certains critères (but, performance ou encore utilité) l'organisation est changée.

De manière générale, afin de modifier une organisation, nous pouvons identifier un processus générique qui sera implémenté en fonction de l'application ciblée. Ce processus, inscrit dans une boucle, est normalement composé de deux phases: le contrôle, et la réplique (figure 4.1).

1. *La phase de contrôle* est destinée à détecter les problèmes d'inadéquation entre le système et son organisation c.-à-d. les agents et l'environnement. Il convient donc de définir les situations de non adaptation.

2. Une fois qu'un besoin de modification est détecté lors la première phase de contrôle, *la phase de réplique* met en œuvre un processus de réorganisation pour que le système en cours d'exécutions retrouve un fonctionnement optimal.

Pour cela :

- Une première étape de choix d'alternative pour modifier l'organisation. Le principal problème est donc la définition de critères d'évaluation des solutions les plus prometteuses.
- La seconde étape correspond à la mise en œuvre de l'alternative choisie précédemment.

La dynamique du système représente l'organisation globale de ce dernier et la dynamique locale représente le processus réorganisationnel des agents. De ce fait, la conception des systèmes adaptatifs ne se focalise pas seulement sur les agents mais principalement sur les relations les liant et la façon dont ces liens sont réalisés.

Si le système ne sait pas quand l'organisation va changer, il connaît les conditions nécessaires de déclenchement d'un changement, qui sera effectué en suivant une procédure spécifiée par le concepteur.

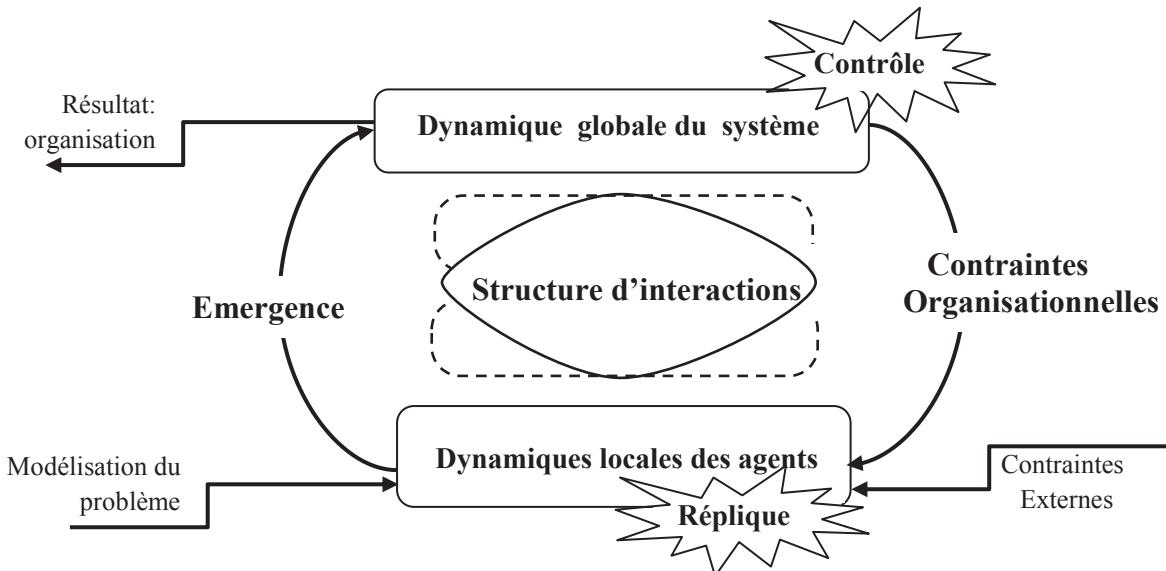


Figure 4. 1. Modèle générique proposé pour la conception d'un SMA adaptatif

Des spécifications organisationnelles peuvent être utilisées pour fixer des bornes comportementales au phénomène émergent résultant, en définissant, des contraintes que les agents ne pourront enfreindre ou des conditions que les agents utiliseront afin d'explorer de nouvelles configurations organisationnelles.

Cette démarche proposée de conception des SMA adaptatif établit le lien entre la représentation du problème, exprimé sous forme de contraintes environnementales, et les comportements des agents qui doivent être des éléments de contrôle pour la réorganisation du système. En outre, il s'agit d'étudier comment réaliser ce lien entre le système à modéliser ou le problème à résoudre et le SMA adaptatif à concevoir.

#### 4.4 Conception d'environnement

Nous proposons donc de classer les enjeux pour la conception d'un environnement en deux catégories :

1. Les aspects structurels (organisation).
2. Les aspects dynamiques (perception, action).

#### **4.4.1 Les aspects structurels**

Un des rôles fondamental de l'environnement est de structurer l'ensemble des entités d'un SMA (les objets et les agents). Pour cela il doit définir les règles qui permettent d'établir et de faire évoluer des relations entre les entités

Incontestablement, l'organisation de l'environnement a une grande influence sur la perception que les agents peuvent en avoir et sur les actions qu'ils peuvent effectuer. Ainsi, les agents ne peuvent que tenter de faire évoluer l'environnement d'une certaine manière puis ensuite constater comment celui-ci a évolué.

Formellement, cette évolution de l'environnement est bien conforme à certaines lois prédefinies. Ce dernier aspect est essentiel: il permet d'une part de délimiter l'autonomie des agents en établissant ce qu'ils peuvent faire, et d'autre part de favoriser la sécurité du système car aucun agent ne peut contourner les lois établies par l'organisation.

#### **4.4.2 Les aspects dynamiques**

##### **A- Perception**

La perception est définie comme la capacité d'un agent à observer son voisinage, ce qui se traduit par la construction de percepts de l'environnement qui sont transmis aux agents. Un percept décrit l'environnement sous une forme compréhensible pour les agents.

Les agents utilisent ces percepts directement pour prendre leurs décisions. C'est le rôle de l'environnement de déterminer la perception que peut avoir un agent suivant ses propriétés.

##### **B- Action**

Le traitement de l'action est aussi une responsabilité de l'environnement. C'est l'environnement qui doit établir pour les agents quelles actions ils peuvent effectuer, et quel est l'effet de ces actions. Cet aspect est parfois traité de manière implicite en encodant les actions dans la structure interne des agents. Cette solution pose un problème de conception de manière générale car il n'est pas évident de garantir que les actions définies au niveau local de l'agent respecteront bien les contraintes que l'on souhaite avoir sur l'environnement.

Ainsi, l'évolution du système est obtenue par l'intermédiaire de nombreuses interactions agent-agent et agent-environnement.

### C- Processus Contrôle/réplique

Le traitement de l'action d'un agent est un aspect fondamental incombe à l'environnement. Dans le cadre d'un SMA, elle ne peut pas être considérée comme une transformation d'état de l'environnement, et cela pour deux raisons :

- a. *Un agent n'est pas seul à agir.* Les activités des autres agents peuvent interférer avec la sienne ; le résultat d'une action ne peut donc pas être déterminé sans considérer l'ensemble des actions qui ont été réalisées au même instant. C'est le problème de la simultanéité de l'action.
- b. *Un environnement est régi par des lois et des règles qui déterminent son évolution.* Un agent n'a pas nécessairement connaissance de toutes ces lois, il n'est donc pas en mesure de prévoir toutes les conséquences de ses actes. L'environnement doit donc assurer que l'évolution de son état est conforme aux lois qui ont été préétablies à sa conception. D'autre part, le dynamisme de l'environnement provient des agents qui agissent de manière autonome ainsi que des objets qui ont leur propre évolution. Il transforme alors son état en prenant en compte les actions simultanées produites par les agents tout en préservant son intégrité. Ce dernier aspect est essentiel : il permet d'une part de délimiter l'autonomie des agents en établissant ce qu'ils peuvent faire, et d'autre part de favoriser la sécurité du système car aucun agent ne peut contourner les lois établies par l'environnement.

Cette phase de réaction, durant laquelle l'environnement met à jour son état, engendre aussi l'émission de percepts à destination des agents.

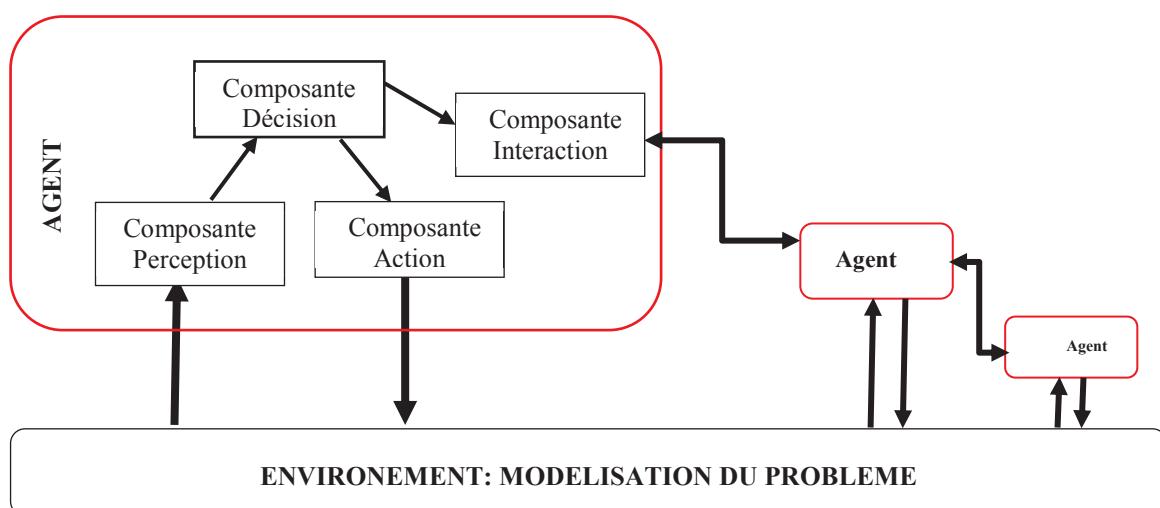


Figure 4.2. Cadre général d'interactions dans un SMA adaptatif

Une fois que l'environnement a été défini, il est nécessaire de déterminer les comportements des agents, et notamment concernant la modification de l'environnement.

#### **4.5 Conception du comportement auto-organisé des agents**

Les agents ont bien sûr leurs objectifs individuels, et ne vont évoluer dans l'environnement que dans leur domaine de valeurs admissibles. Une fois l'organisation définie, il faut déterminer quels sont les informations émises et comment les agents réagissent aux changements éventuels.

Ces agents élémentaires sont tenus d'éviter les situations de conflit avec des règles génériques de réorganisation afin d'assurer au collectif une fonction globale adéquate et adaptée.

##### **4.5.1 Conception d'interaction**

Dans le cadre des agents réactifs, il est habituel de considérer que l'intelligence n'est pas contenue dans les agents mais dans l'environnement et les interactions. Il est alors naturel, après s'être intéressé à l'environnement, de se concentrer sur la notion d'interaction.

Une interaction peut être définie comme une mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques. Elles peuvent être considérées comme des actions ayant une influence sur la perception des autres agents induisant ainsi une modification de leurs actions.

La définition et la mise au point de mécanismes d'interaction permettant d'obtenir des états fonctionnels adéquats.

##### **4.5.2 Réorganisation dirigée par une utilité : la fonction du système**

Pour obtenir le comportement collectif auto-organisé souhaité, les parties qui composent le système ont besoin d'un critère local de réorganisation afin de fournir, au final, une fonction plus adaptée aux demandes de l'environnement.

De ce fait, l'adaptation se fait par modification au niveau des spécifications organisationnelles, à l'initiative des agents après détection de dysfonctionnements ou de perturbations.

Sachant que pour une organisation interne du système correspond une fonction produite par le système, les concepteurs doivent spécifier ce critère et donner aux agents des capacités à le manipuler.

Pour ce faire, l'environnement doit planter cette perception des agents sur l'organisation. Ces derniers raisonnent sur la spécification de l'organisation. Ils construisent ainsi les étapes à suivre de la réorganisation.

#### **4.6 Conditions sollicitées dans le processus de réorganisation**

L'approche par réorganisation permet l'adaptation aux changements de l'environnement et aux dysfonctionnements de l'organisation qui est représentée explicitement via l'environnement. Les agents doivent être dotés de comportements pour contrôler le système dans lequel ils évoluent, soit au niveau de leur environnement, soit au niveau de l'organisation.

Toutefois, une réorganisation appropriée doit prendre en compte l'aspect structurel et fonctionnel. Aussi, elle doit être distribuée sur l'ensemble d'agents et convergente pour mettre en évidence la structure émergente. L'évaluation de l'organisation obtenue est à prendre en considération pour la réorganisation.

#### **4.7 Dualité de l'autonomie et du contrôle du système**

Dans les SMA adaptatifs, il faut souligner l'existence d'une dualité entre le besoin d'autonomie au niveau des agents et le besoin de contrôle au niveau du système. L'autonomie est l'essence des agents, c'est parce que les agents agissent de manière autonome que le système résultant est capable de s'adapter aux situations nouvelles et d'évoluer. C'est aussi l'autonomie qui rend possible l'émergence nécessaire à l'auto-organisation d'un système. Mais plus l'autonomie des agents est importante et plus il est difficile de prévoir le résultat de leurs interactions.

Un SMA adaptatif pour la résolution de problème, comme tout système informatique, est conçu pour répondre à certains objectifs et il est donc nécessaire d'avoir un certain niveau de contrôle pour garantir certaines propriétés au niveau global.

Il s'agit de trouver le bon compromis entre le niveau d'autonomie et le niveau de contrôle. Si le niveau de contrôle est absolu alors le système devient totalement prévisible. Or l'intérêt du paradigme multi-agents est justement d'apporter des réponses pour les problèmes dont on n'est pas en mesure de prévoir à l'avance une bonne solution. Mais à l'inverse, si l'autonomie des agents est trop importante le système risque alors d'être chaotique et non performant.

De manière générale, plus l'autonomie des agents est importante, et plus la complexité de la coordination de leur comportement est importante, ce qui inévitablement réduit la performance globale du système. L'intérêt se limite donc à empêcher les comportements dont on a la certitude au moment de la conception du système qu'ils seront néfastes en toutes circonstances.

#### **4.8 Lien entre les niveaux macro et micro**

Il est particulièrement difficile de prévoir au niveau macro le résultat d'un ensemble d'activités autonomes qui se produisent au niveau micro. De manière duale, il est très complexe de définir un système répondant à certaines exigences uniquement à partir des spécifications des entités du niveau micro.

Il est donc nécessaire de se donner des moyens permettant de relier l'activité locale et autonome d'un agent aux propriétés attendues au niveau global du système. De plus les organisations ont des buts qui leurs sont propres.

#### **4.9 Etapes de conception de SMA auto-organisé pour la résolution des problèmes**

De l'étude précédente, nous dégageons les étapes de conception de SMA pour la résolution de problème, qui se décline en 4 repères :

- 1) **Formulation du problème à travers l'environnement:** Définir un modèle de l'environnement qui représente le problème à résoudre sous formes de contraintes, de structure, de dynamisme et d'organisation).
- 2) **Détermination des perceptions des agents :** Les agents doivent être capables de percevoir les différents états de l'environnement et être influencer par les perturbations qui résultent de leurs actions. Les conditions de déclenchement de réorganisation sont explicitées, et seront effectuées en suivant une procédure spécifiée par le concepteur.
- 3) **Modélisation des mécanismes d'interaction :**
  - a. Déterminer les réactions locales selon la perception (agent-agent ou agent-environnement). Le code des agents est généré à partir de contraintes organisationnelles.

**b.** Prévoir les actions destinées à réduire les conflits qui peuvent générer une mauvaise performance du système. Si le système ne sait pas quand l'organisation va changer, il doit connaître les conditions nécessaires à la réorganisation.

**c.** Il faut établir un lien explicite entre le but du système et l'activité des agents qui intègre l'organisation de ce dernier.

4) **Evaluation du résultat:** La structure organisationnelle obtenue est la conséquence des interactions. L'évaluation et la mesure des propriétés émergentes du SMA sont des tâches délicates mais nécessaires dès lors que l'on désire analyser la solution fournie par le système dans le cadre de la résolution de problèmes ou de la prise de décision. Ainsi, dans certains cas complexes, il est nécessaire d'avoir recours à des indicateurs ou des mesures permettant une meilleure analyse du système qu'une simple observation.

L'évaluation des propriétés émergentes revient souvent à mesurer une distance par rapport à un objectif à atteindre. Pour cela, il est nécessaire de construire des indicateurs permettant de mesurer les propriétés du système. Ces mesures peuvent prendre en compte les propriétés locales et globales du système et peuvent être selon le problème une valeur de fitness ou une fonction d'utilité. Elles peuvent aussi considérer l'évaluation globale de l'organisation émergente.

Néanmoins, il est plus judicieux de construire un indicateur basé sur le cadre applicatif. Il est simple à construire puisqu'il s'appuie sur une évaluation pragmatique du phénomène émergent relativement à l'application visée et ceci par rapport à des systèmes de contrôle existants ou à des algorithmes.

#### **4.10 Résolution de problème dirigée par une auto-organisation**

Le modèle présenté de construction de SMA auto-organisé pour la résolution de problème s'appuie sur la notion d'environnement et d'interaction. Ceci incite donc à établir le lien entre la représentation du problème, exprimé sous forme de contraintes organisationnelles environnementales, et les comportements des agents qui doivent être des éléments de contrôle et de réplique aux changements.

L'environnement est défini en entrée. Il formalise la topologie et les variations du problème devant être perçues par les agents qu'il contient. L'organisation émergente est la sortie du système. Le mécanisme de réplique est défini par l'intermédiaire des agents et de leurs interactions.

Ces interactions peuvent être séparées en deux catégories distinctes :

- Les interactions agent-agent, d'une part, composent le feedback positif.
- D'autre part, les interactions agent-environnement caractérisent la boucle de retour c.-à-d. le feedback négatif.

L'environnement est transformé à la fois par la dynamique du problème à résoudre et par la réorganisation dynamique liée aux interactions des agents.

## Conclusion

L'analyse des processus collectifs organisés, nous a permis de dégager un modèle de conception de SMA adaptatif pour la résolution de problème. La méthode proposée spécifie que l'environnement doit être modélisé avant les agents. En effet, celui-ci permet de représenter le problème, alors que les agents définissent le processus de résolution des contraintes du problème. Ce processus est construit aux niveaux individuel et collectif, en définissant tout d'abord les perceptions des agents, puis les interactions qui en découlent.

Toutefois, la définition des mécanismes de réorganisation et d'adaptation qui dépendent du problème est aussi laissée au concepteur.

## **CHAPITRE 5**

### **ETUDE DE CAS**

**Conception d'un SMA adaptatif pour le  
contrôle intelligent de la signalisation dans un  
réseau routier**

## **CHAPITRE V**

### **ETUDE DE CAS**

#### **Conception d'un SMA adaptatif pour le contrôle intelligent de la signalisation dans un réseau routier**

##### **Introduction**

L'approche multi-agents a montré son intérêt pour les problèmes distribués. Le découpage d'une application en entités décentralisées et coopérantes déplace les problèmes d'analyse d'un niveau global à un niveau local et réduit ainsi la complexité de sa conception.

Cependant, les systèmes fortement distribués, pour lesquels il est difficile de prévoir toutes les situations rencontrées, mettent en jeu de nombreuses entités, souvent autonomes d'un point de vue énergétique. Ces entités peuvent être à la fois matérielles et logicielles et chargées de tâches complexes et de natures très diverses : tâches d'acquisition d'information, de calcul, d'action et de communication.

Dans ce chapitre, nous montrons que l'approche proposée permet de gérer efficacement ce type de système complexe en l'adaptant au contrôle intelligent de la signalisation de trafic routier avec un réseau sans fil de capteurs. Les spécificités de ces systèmes réels immergés dans des environnements ouverts et dynamiques rendent les techniques multi-agents encore plus attractives. L'utilisation de l'auto-organisation permet d'assurer la gestion du contrôle avec un caractère hautement adaptatif.

Nous précisons tout d'abord la problématique du trafic routier en soulignant l'application des systèmes intelligents de contrôle de trafic. Nous exposons ensuite quelques travaux connexes utilisant les SMA puis nous formulons le modèle adaptatif de gestion de signalisation du trafic utilisant un SMA auto-organisé. Avant de conclure, nous présentons une analyse comparative des résultats quantitatifs obtenus sur une plateforme de simulation.

## 5.1 Problématique du contrôle du trafic routier

Le trafic routier implique un nombre variable d'usagers différents et un réseau de capacité limitée. La tâche de conduite est généralement non coopérative et les conducteurs ont des objectifs changeants. Tout ceci implique que le "système de trafic routier" est hétérogène, ouvert, compétitif et dynamique. De plus, l'information perçue par les conducteurs est géographiquement limitée et incomplète. La différence majeure entre un réseau autoroutier et un réseau urbain est la présence dans ce dernier d'intersections présentant des croisements de flots. L'étude générale d'un trafic urbain nécessite de se focaliser sur une problématique plus restreinte qui est la coordination du flux aux intersections pour éviter les situations d'interblocage. Aussi, le trafic routier urbain s'est accru d'une façon exponentielle, augmentant ainsi les problèmes engendrés qui sont nombreux et qui coûtent quotidiennement temps, argent, santé et qualité environnementale: embouteillages, accidents, pollution ou encore infractions.

Dans le contexte abordé dans ce travail, les mécanismes de coordination se situent principalement dans l'utilisation d'une approche auto-organisée à base d'agent et notamment dans la conception d'une stratégie adaptative de contrôle des feux de circulation qui sont les éléments clés du contrôle du trafic.

En effet, la combinaison de plusieurs contraintes ne permet pas d'atteindre un ordonnancement optimal en temps réel. Ceci est particulièrement le cas de plusieurs intersections où les conditions de circulation des intersections voisines ont un impact dynamique sur la stratégie locale. Incontestablement, ces flux voisins devraient faire partie de l'écoulement de la circulation locale après une certaine période et par conséquent, la coordination entre les intersections adjacentes doit également être prise en compte.

## 5.2 Systèmes de transport intelligents (STI)

Aujourd'hui, la gestion du trafic routier s'inscrit dans le domaine des systèmes de transport intelligents (STI), qui visent à proposer des outils et modèles afin de gérer les aléas de ce dernier, ceci par le biais d'équipements dynamiques. Les STI apparaissent comme étant l'application des technologies de l'information et de la communication au domaine des transports.

Le terme système est vague et se décline en un ensemble de moyens mis en place pour gérer au mieux les contraintes liées au trafic routier. La plupart de ces moyens permettent de recueillir des données propres à la circulation routière afin de tenter de la contrôler.

Parmi ces moyens, les réseaux de capteurs sans fil, appartenant à la famille des réseaux mobiles ADHOC et se composent d'un large ensemble de capteurs à capacité et énergie généralement limitées.

Dans de nombreux cas, les capteurs sont constitués des unités suivantes [Knaian , 2000] :

- Une unité d'acquisition, permettant le recueil de données environnementales et la conversion analogique vers numérique.
- Une unité de calcul, permettant le lancement de procédures, protocoles et autres.
- Une unité de communication, permettant la connexion au réseau (émission et réception). Ceci, en sans-fil et souvent en multi-sauts pour s'affranchir des inconvénients filaires (temps d'installation et facilité d'accès).
- Une unité d'énergie, qui permet la répartition de l'alimentation entre les différents constituants.

### 5.2.1 Unité d'acquisition

Pour mesurer les différentes grandeurs physiques liées à la situation du trafic, une large gamme de capteurs peut être utilisée comme détecteurs de données dans le réseau de circulation telles les boucles inductives, les caméras, magnétomètres, les capteurs sans fil, et les radars.

Cependant, l'utilisation de capteurs sans fil, dans la plupart des œuvres, est motivée par diverses caractéristiques telles: faible coût de fabrication, petite taille, facilité et rapidité d'installation, réactivité, communication et mémorisation. Les capteurs sans fil peuvent être nombreux et situés sur diverses parties de la voie mais la distance entre deux capteurs doit être suffisante pour avoir un échantillonnage correct.

Le rôle de chaque capteur est de récolter un ensemble de données dans son environnement, et le transmettre de proche en proche jusqu'à atteindre généralement une station de base, qui peut jouer le rôle de coordinatrice du réseau et communiquer vers l'extérieur les données importantes recueillies.

L'application de tels systèmes va avoir de multiples objectifs, parmi lesquels la fluidification du trafic, la détection d'incidents, la surveillance temps-réel du trafic, la diffusion d'informations ou de consignes variables aux automobilistes ou encore la réduction en conséquence de la pollution et des bruits. A une échelle plus locale, ces technologies sont utilisées à des endroits stratégiques du réseau, comme les intersections qui constituent les éléments de base dans la gestion de feux de circulation.

### 5.2.2 Modélisations microscopique et macroscopique

Il existe dans la littérature, deux grands types de modélisation du trafic routier qui sont la modélisation microscopique et la modélisation macroscopique. Tandis que la modélisation microscopique s'attache à décrire le trafic en modélisant chaque véhicule individuellement, la modélisation macroscopique consiste à regrouper les véhicules de mêmes caractéristiques en une seule entité, qui, sous certaines conditions, est assimilable à un fluide, dont on modélise les propriétés dans l'espace et dans le temps.

Dans le cas d'une modélisation microscopique, on attribue à chaque véhicule un ensemble de variables telles que la position et la vitesse, qui vont évoluer en fonction du trafic et de certains paramètres du modèle, tel que le comportement du conducteur.

### 5.2.3 Grandeur physiques associés

Les grandeurs classiques liées à cette entité sont le débit, la concentration et la vitesse de l'écoulement, qui sont des fonctions du temps et de l'espace. On appelle débit, en un point, le nombre de véhicules qui passent en ce point par unité de temps. Deux types de débits sont utilisés : le débit en entrée de tronçon, que l'on appellera débit des arrivées et le débit en sortie de tronçon, que l'on appellera débit de sortie.

Une simplification commune à beaucoup de systèmes de régulation est de considérer que le débit de sortie est égal au débit de saturation c.à.d. au débit maximal possible en sortie de tronçon en cas de présence d'une file d'attente infinie et feu au vert. En effet, le débit de sortie peut s'avérer complexe à calculer, car il dépend, en particulier, de gènes sur le carrefour, difficilement prévisibles. Les débits des arrivées pour un carrefour isolé, en revanche, peuvent être estimés à partir de prévisions statistiques plus fiables, par exemple en fonction de l'heure des arrivées.

La capacité d'une voie (entrante) est le débit maximal que celle-ci est capable de laisser passer. En général, cette grandeur est utilisée lorsque la régulation se fait via un plan de feux. La capacité d'un carrefour, critère usuel pour la conception de carrefour, est la somme des capacités de ses voies.

### 5.3 Système de gestion des feux de circulation

La gestion des feux de circulation est un problème qui a commencé à être étudié au début des années 1970. Au fil des années, ces systèmes se sont diversifiés, et ont connu trois générations de contrôle. Aujourd'hui, ces générations peuvent être utilisées chacune en fonction des moyens mis en place sur l'infrastructure, et de la connaissance de cette dernière :

#### a. Contrôle à temps fixe

Le contrôle de temps fixe est la stratégie de contrôle la plus couramment mis en œuvre dans le monde. Le signal de synchronisation est préalablement calculé sur la base des conditions de circulation qui est prévu (par jour et heure). Une grande quantité de données historiques de trafic doit être collectée, enregistrées et ensuite utilisé pour tirer quelques modèles de trafic. Ces modèles se basent sur l'emplacement géographique des intersections et les périodes de temps (par exemple, les heures de pointe).

#### b. Contrôle à temps dynamique

Dans ce système, des détecteurs sont utilisés afin de recueillir les données du trafic pendant des périodes définies. Ces données peuvent être utilisées afin d'optimiser ou mettre en place un plan de feu. Ce système fonctionne en adaptant le volume de véhicules sur la route et définir en conséquence la fréquence et l'intervalle de feux de circulation.

#### c. Contrôle adaptatif

Ce type de contrôle sert à programmer dynamiquement les plans de feux en se servant des paramètres recueillis sur le terrain, et en calculant les valeurs telles que le temps de cycle, des phases ou encore leur ordonnancement. Les modèles adaptatifs efficaces permettent essentiellement à coordonner les signaux d'intersections contrôlées de manière à améliorer les performances et réduire la congestion dans les réseaux urbains. L'opération est effectuée de manière adaptative, c'est à dire en quasi-temps-réel.

L'avantage de ces méthodes est qu'elles peuvent s'adapter à de multiples situations, mais elles restent la plus compliquée à mettre en place. En effet, elle nécessite la recherche des informations utiles pour les utiliser à programmer dynamiquement les plans de feux de circulation en se basant sur des théories parfois lourdes. Néanmoins, les avancées en outils ont été permises grâce à l'introduction de plusieurs solutions novatrices en termes de gestion intelligente du trafic routier.

Une première caractéristique importante d'un système adaptatif de contrôle est sa capacité à prendre en compte de l'information sur le trafic de manière régulière et à pouvoir utiliser cette information sur ses prises de décision. En d'autres termes, il s'agit de sa capacité à s'adapter à la situation de trafic courante.

Ainsi, l'adaptation se compose des deux éléments suivants :

- l'acquisition des informations considérées par le système et la fréquence de révision de ces informations.

- la flexibilité du système, qui sera sa capacité à exploiter l'ensemble des solutions admissibles, en termes de commutations des feux, par rapport aux conditions du trafic aux carrefours à feux.

#### 5.4 Principaux systèmes adaptatifs existants

Trois issues représentent les principaux systèmes adaptatifs de gestion du trafic routier utilisés dans le monde :

1- ***Split Cycle Offset Optimization Technique (SCOOT)*** [Robertson et al., 1991] est l'un des principaux système de contrôle adaptatif développé par le Trafic Research Laboratory (TRL). Dans ce système, l'ensemble des informations recueillies sur le terrain vont à un centre de gestion, qui s'occupe de traiter les informations et renvoyer des indications directement aux intersections.

Les véhicules sont détectés par des dispositifs pouvant être placés à divers endroits sur les voies : au niveau des feux ou à une certaine distance afin de pouvoir mesurer le débit du trafic. SCOOT mesure en permanence le volume de véhicules de chaque côté de l'intersection et change la durée des phases en fonction d'un indice de performance, calculé par rapport au délai d'attente moyen, de la longueur des queues. Ainsi, ce modèle génère des plans de feux en fonction de la demande des utilisateurs. En addition, SCOOT utilise des informations en ligne issues de bases de données (historique ou autres).

2- **Sydney Coordinated Adaptive Traffic System (SCATS)** [Sims et al., 1980] a été à l'origine développé pour Sydney et d'autres villes australiennes. Ce système est pour sa part adaptatif et de manière similaire à SCOOT, il ajuste le temps des cycles et autres paramètres en fonction des données recueillies afin de diminuer le délai d'attente et les arrêts. Cependant, il n'utilise pas la même stratégie : les valeurs recueillies permettent la sélection de plans de feux parmi une large librairie, sur lesquels le système va se baser pour proposer des plans adaptés. De plus, contrairement à SCOOT, les détecteurs sont uniquement placés au niveau des feux de circulation.

3- Un autre mécanisme adaptatif classique connu sous le nom de la « *vague verte* » est largement utilisé dans de nombreuses villes. C'est aussi une stratégie de contrôle adaptative qui affecte un décalage fixe à une séquence de feux de signalisation. Ceci permet aux véhicules regroupés en pelotons de tailles variées, déterminé par la synchronisation de signal, de progresser à travers les feux verts (vague verte) à une vitesse uniforme sans s'arrêter. Le décalage calculé définit le temps nécessaire pour les véhicules de se déplacer d'une intersection à l'autre à une vitesse constante attendue.

D'autre part, les techniques d'intelligence artificielle avec ses différents outils de modélisation ont été largement étudiées pour contribuer dans le contrôle adaptatif des feux de signalisation dans la circulation urbaine. Ainsi, les systèmes à logique floue, les réseaux de neurones artificiels, l'informatique évolutionnaire et l'intelligence en essaim ont constitué des outils de contrôle informatique efficaces dans le traitement de la complexité et de la dynamique du trafic [Pappis et al., 1977 ; Bingham, 2001 ; Sanchez et al., 2010 ; Aksaç et al., 2012].

Les SMA ont été aussi un axe privilégié de recherche et d'application vu qu'ils présentent les caractéristiques du mécanisme souhaité : ils sont distribués, prennent en compte les systèmes ouverts et permettent de gérer les conflits et la coordination. Dans ce qui suit, nous présentons quelques travaux connexes utilisant les SMA dans le contrôle du trafic.

## 5.5 Travaux connexes

Plusieurs études et recherches ont été menées dans le domaine de la modélisation du contrôle de trafic utilisant les SMA avec différentes échelles géographiques et à différents niveaux de complexité de comportements des agents [Ehlert et al., 2001 ; Chen et al., 2010; Benhamza, 2012; Bazzan et al., 2014].

Ainsi, [Roozemond, 1999] a présenté un modèle d'agents agissant de manière autonome tout en partageant les données pour faire les prévisions à court et long terme. Dresner et Stone [Dresner et al., 2008] ont abordé le problème en utilisant un système de réservation pour éviter les collisions à une intersection.

Cependant, dans ces travaux, il manque l'aspect de coordination entre les feux de circulation. [Bazzan, 2005] a utilisé aussi une approche décentralisée combinant les SMA et la théorie des jeux évolutionniste. Cette approche modélise chaque intersection comme un agent motivé individuellement qui doit se concentrer non seulement sur l'obtention de véhicules à travers l'intersection, mais aussi sur la réduction des temps de déplacement pour tous les véhicules.

Plusieurs applications se basant sur l'apprentissage par renforcement existent aussi comme dans les travaux [Wiering et al., 2003 ; Da Silva, 2006]. Ces modèles tentent de déduire des modèles de récompense pour les différentes paires d'état/action permettant l'interaction entre les agents de contrôle et l'environnement de la circulation. Dans un grand réseau, le problème du grand nombre de ces paires restreint l'utilisation de ces méthodes dans ce domaine (problème de convergence).

Gershenson [Gershenson, 2005] a présenté l'une des premières applications de systèmes auto-organisés au contrôle des feux de circulation. Par une modélisation simplifiée du réseau, des intersections et des véhicules, l'auteur a montré qu'une solution à base d'agents adaptatifs est capable de contrôler les feux de circulation de manière efficace.

Aussi, Lammer et Helbing [Lammer et al., 2008] ont présenté un autre système de contrôle auto-organisé qui a montré la manipulation efficace de plusieurs scénarios de trafic dans les réseaux de type grille hypothétique.

L'approche proposée par Xaio-Xie Feng [Xie et al., 2011], a présenté un modèle qui permet le calcul en temps réel des politiques de synchronisation du signal. Ces approches d'auto-ordonnancement donnent lieu à des systèmes efficaces et organisés.

## 5.6 Conception d'un système adaptatif de signalisation utilisant un SMA auto-organisé

Le modèle multi-agents peut aussi être utilisé pour faire face à des situations complexes réelles. Mais, il doit être plus adaptatif au changement environnemental. Ainsi, les systèmes auto-organisés sont les approches particulièrement appropriées à appliquer dans le contrôle du trafic.

L'ajustement de la configuration des feux de circulation est l'élément clé des modèles présentés dans les principaux travaux précédents. Cependant, ces paramètres de configuration sont fondés sur des données du trafic mis à jour fréquemment.

Les étapes de conception de SMA pour le contrôle et la gestion des feux de signalisation ont été dégagées et se déclinent en 4 repères :

### 5.6.1 Formulation du problème à travers l'environnement

Dans les réseaux urbains, une intersection est un espace partagé d'un réseau routier où les véhicules le traversent avec des directions opposées. Les véhicules accèdent une intersection via une voie d'entrée ou de sortie. Chaque voie possède un ensemble de mouvements tournants autorisés.

Chaque période de temps, durant laquelle une combinaison de feux verts pour un ensemble de mouvements autorisés pour chaque direction est activée, est appelée phase. Toutes les phases combinées ensemble constituent le cycle du signal.

Les feux de circulation fonctionnent en cycle à travers leurs phases de lumière : la lumière verte permettant aux véhicules de traverser l'intersection. La lumière jaune les obligeant à ralentir pour se préparer à s'arrêter. Enfin, la lumière rouge contraint les véhicules à stopper à l'entrée de l'intersection. Les feux de circulation sont gérés à l'intersection en alternant entre les différents flux de trafic de mouvements autorisés de véhicules.

L'intersection présentée dans la figure 5.1 est utilisée pour valider le modèle proposé. Elle est composée de quatre directions avec un nombre fixé de voies pour chacune. Les voies pour tourner à gauche sont séparées des voies allant tout droit, ces deux derniers mouvements étant confondus. Cette infrastructure sert de prototype de base à l'évaluation de la plupart des algorithmes et des principes de la littérature.

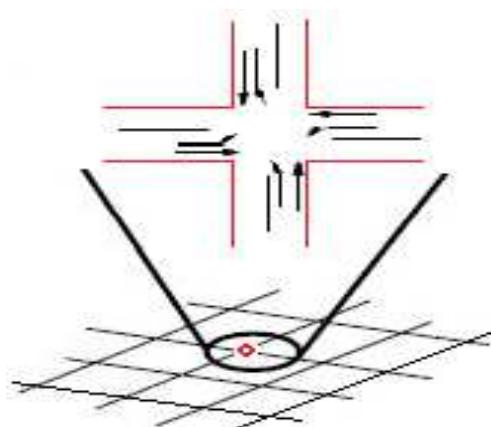


Figure 5. 1. Modèle d'Intersection

### 5.6.2 Modèle proposé

Le modèle proposé comprend trois placements différents de capteurs pour une intersection donnée (Figure 5.2). Ainsi, le trafic sur chaque voie entrante ou sortante est contrôlé par ces nœuds de capteurs [Benhamza et al., 2014a]:

- Un Capteur de Sortie (CS) situé près des feux de circulation comptant les véhicules sortants de l'intersection lorsque le feu est vert;
- Un Capteur Avancé (CA) placé à une distance appropriée avant les feux de circulation pour détecter les véhicules entrants.
- Un Capteur Mouvement du trafic (CM) situé à l'entrée des voies sortantes.

Un tel déploiement différencie mieux les véhicules sortant de l'intersection et permet d'informer les intersections voisines sur le flux de circulation à venir. Les capteurs mouvement (CM) permettent la collecte et l'agrégation du trafic de tous les flux des voies émettant leurs directions.

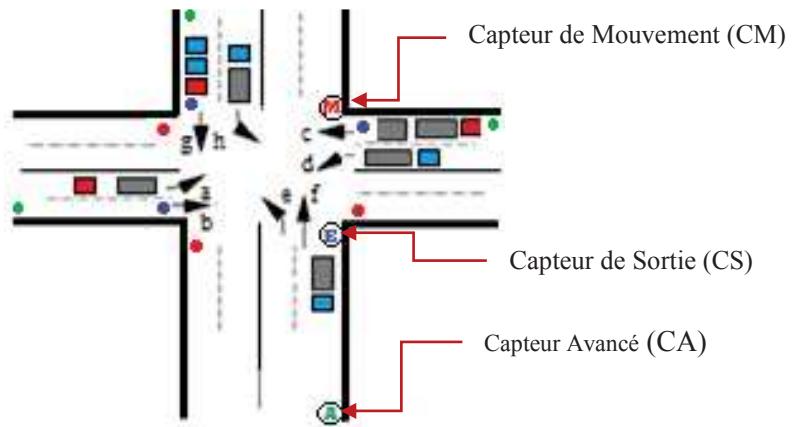


Figure 5. 2. Emplacement de détecteurs dans une intersection

Soumis aux règles de sécurité routière qui assurent à l'environnement une évolution conforme aux lois du trafic préétablies, il y a douze mouvements autorisés pour chaque intersection indiqués sur la figure 5.3. Il est à signaler qu'un mouvement est un déplacement de véhicules dont on distingue l'origine et la destination.

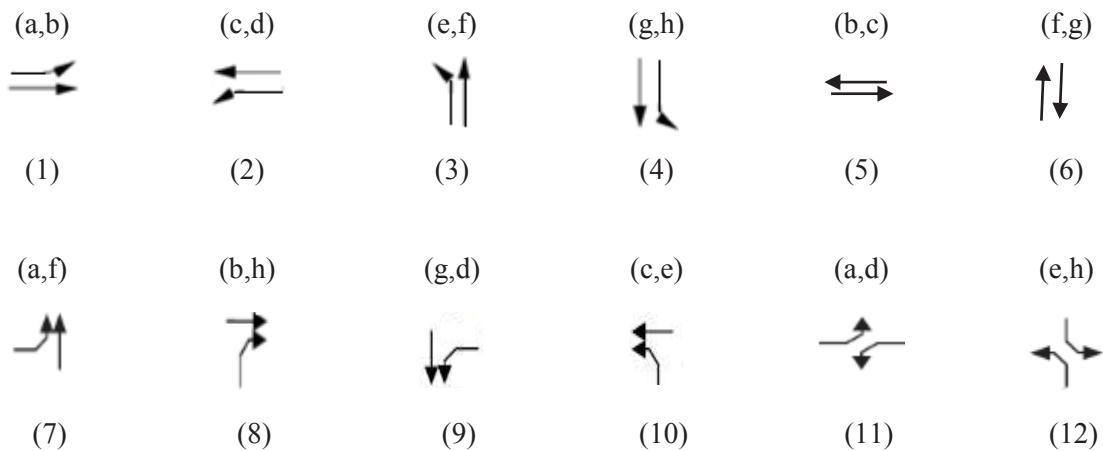


Figure 5.3. Mouvements autorisés

Ainsi, l'étape de décision est transformée en un choix de la séquence de feux adaptée et le réglage de la durée de lumière verte affectée à chaque séquence de mouvements [Benhamza et al., 2014b]. Les véhicules peuvent être modélisés en tant que SMA réactifs [Benhamza et al., 2010] ou en tant qu'objets dynamiques.

### 5.6.3 Objectif ciblé du modèle proposé

Il est à souligner que le système doit satisfaire un double objectif:

- L'objectif local qui consiste à améliorer le trafic dans chaque intersection.
- L'objectif global qui doit coordonner les signaux du trafic afin que les véhicules rencontrent successivement le maximum de feux verts et donc les pelotons générés de véhicules peuvent passer par une série continue de feux sans s'arrêter et ainsi améliorer le trafic dans le réseau.

### 5.6.4 Définition des agents

Selon le cadre proposé et décrit dans le chapitre précédent, trois éléments principaux (figure 5.4) sont définis dans le modèle proposé:

1. *Composante Capteurs*: Elle permet de collecter les informations de trafic en temps réel;
2. *Agent de contrôle de la circulation*: Il est chargé du contrôle autonome des feux de signalisation à l'intersection;
3. *Feu de signalisation* Dispositif qui définit le comportement alloué à l'intersection (contrôle de lumière et de sa durée).

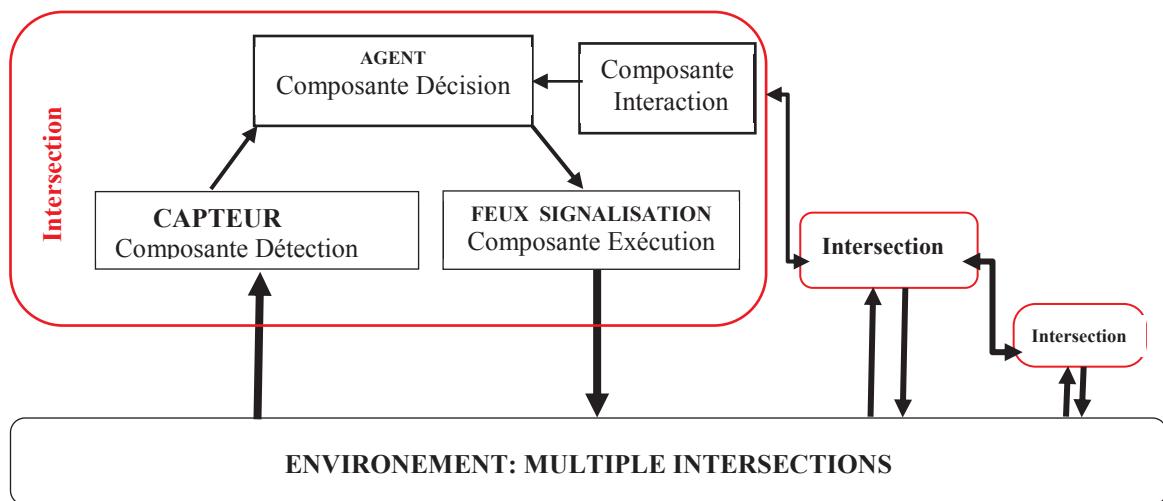


Figure 5. 4. Proposition du cadre multi-agents : composants d'Intersection

Etant donné que le trafic urbain dépend fortement de la réponse rapide dans la stratégie de contrôle du système de feux de circulation à chaque intersection, la stratégie de contrôle peut être décrite comme suit:

- Premièrement, la composante de détection (détecteur) est utilisé pour contrôler le trafic et transmettre les données vers la composante de décision (agent);
- Après avoir recueilli toutes les informations requises, la composante décision (agent) commence à générer la stratégie de contrôle du trafic de signal appropriée;
- Composante exécution de feux de signalisation est utilisée pour organiser la circulation.

Pour la composante de détection des données du trafic, les données du flux en temps réel sont collectées au moyen de capteurs (figure 5.2). Chaque capteur CA détecte les véhicules arrivés (leur nombre et leur type) et mesure l'espace entre les véhicules qui passent. En même temps, le capteur CE compte le nombre de véhicules sortants durant cette phase.

Lorsque la phase est terminée, le capteur CS transmet les données au capteur CM approprié qui gère la direction accessible par les véhicules des voies. CA et CS réinitialisent leurs compteurs à zéro.

Chaque capteur de mouvement CM calcule le nombre de véhicules correspondant au mouvement autorisé (figure 5.3) dans sa direction et la longueur totale des espaces existants dans le mouvement. Aussi, il devra garder une trace des taux de saturation de l'intersection voisine.

### 5.6.5 Définition des mécanismes d'interaction

En effet, pour augmenter le débit et réduire le temps d'attente moyen, la fréquence à laquelle un grand espace atteint l'intersection est minimisée. Une grande priorité est attribuée au mouvement avec les plus petits écarts vides et les plus éloignés aussi. De cette façon, l'utilisation du temps du feu vert serait optimisée par le passage de maximum de véhicules. Après avoir estimé les données d'intersection locales, il est nécessaire d'assurer la fluidité du trafic dans le réseau et la coordination entre les intersections voisines.

Deux étapes importantes sont conduites pour la sélection du prochain mouvement et le calcul de sa durée.

#### 1. Selection du mouvement:

L'agent feux de signalisation sélectionne les mouvements qui composent la phase suivante tandis que la phase courante est active en se basant sur les données formulées ci-dessus. En outre, un autre paramètre assez important et qui offre l'équité pour chaque mouvement doit être calculé et préservé: c'est le temps écoulé depuis la dernière sélection du mouvement qui doit être choisi dans un temps satisfaisant.

Par conséquent, la phase suivante est définie sur la base d'une priorité affectée au mouvement et calculée avec les paramètres résumés ci-dessous de chaque mouvement autorisé [Benhamza et al., 2015]:

- ✓  $TSel_{Mvt}$ : Le temps écoulé depuis la dernière sélection du mouvement
- ✓  $PrioVeh_{Mvt}$ : Présence de véhicules prioritaires
- ✓  $SNeigh_{Mvt}$ : taux de saturation de l'intersection voisine de la saturation
- ✓  $LGap_{Mvt}$ : Longueur totale de l'espace vide
- ✓  $N_{MVT}$ : Nombre total de véhicules

En fonction de l'importance du trafic et sa répartition, ces paramètres peuvent être ajustés et normalisés avec de poids ( $a_1, a_2, a_3, a_4$  et  $a_5$ ) qui peuvent être définis en fonction des informations de trafic.

$$R_{Mvt} = a_1 \times TSel_{Mvt} + a_2 \times PrioVeh_{Mvt} + a_3 \times SNeigh_{Mvt} + a_4 \times LGap_{Mvt} + a_5 \times N_{Mvt} \quad (\text{Eq. 7})$$

Enfin, le mouvement qui obtient le plus grand rang va se faire attribuer le prochain feu vert.

## 2. Définition la durée de mouvement:

Le temps vert  $t_{gl}$  destiné à vider la voie dans une approche dynamique est basé sur plusieurs aspects [Gordon et al., 2005]. Il est ajusté par la formule suivante:

$$t_{gl} = t_s + N_{max} \cdot t_H \quad (\text{Eq. 8})$$

où  $t_s$  est le temps que prend un véhicule pour commencer à bouger,  $N_{max}$  est le nombre de véhicules sur la voie la plus peuplée qui lui sera accordé le feu vert dans la phase et  $t_H$  est le temps moyen qui sépare deux véhicules.

Alors, le temps adaptatif du feu vert ( $T_{adpt\_gl}$ ) considérant les intersections voisines est

$$T_{adpt\_gl} = t_{gl} + t_{offset} \quad (\text{Eq. 9})$$

Le temps  $t_{offset}$  est défini comme la durée suffisante pour les véhicules autorisés à traverser l'intersection à partir de l'intersection voisine. Il est donné par l'équation ci-dessous:

$$t_{offset} = d / sp \quad (\text{Eq. 10})$$

où:  $d$  = distance entre intersections à feu.

$sp$  = vitesse moyenne du peloton.

À noter que, la période verte pour chaque phase a une durée variable limitée défini par des ingénieurs de la circulation pour éviter de capturer le feu vert.

Avec des feux coordonnés, l'interférence entre les feux de circulation serait réduite et la meilleure performance possible serait atteinte. Le système proposé sélectionne ainsi une direction de synchronisation basé sur la plus forte demande de mouvement sélectionné et attribut un temps de feu vert limite recommandé.

## 5.7 Evaluation du résultat de la structure auto-organisée obtenue

Les indicateurs utilisés sont des indicateurs basés sur un cadre applicatif de comparaison des résultats obtenus avec ceux d'autres modèles existants. Dans notre application, de nombreuses simulations ont été réalisées afin d'évaluer la performance du modèle proposé par rapport aux trois modèles cités précédemment: Contrôle à temps fixe, contrôle à temps dynamique et mécanisme de la vague verte. L'évaluation est effectuée en utilisant SUMO (Simulation de la mobilité urbaine) [Krajzewicz, 2010], qui est un simulateur de trafic routier open-source ( annexe 1).

Un réseau de trafic, illustré à la figure 5.5, est choisi comme réseau d'application afin d'évaluer l'approche proposée. Ce réseau d'essai est composé de cinq intersections interconnectées: un carrefour central avec quatre intersections voisines.

Toutes les expériences ont été réalisées avec un flux hétérogène de véhicules (simples, rapides, prioritaires et lents) et en utilisant des densités différentes (faible, moyen et élevé). Pour chaque densité de la circulation, une moyenne de dix simulations a été effectuée en utilisant les paramètres suivants de configuration. Ceci permet de couvrir les différents cas de figures :

- ✓ Temps de simulation (3600 secondes),
- ✓ Flux de trafic variable (taux d'arrivée = 0,6 véhicules/seconde)
- ✓ Trafic hétérogène (65% des voitures simples, 25% des voitures rapides, 4% de voitures prioritaires, 6% des voitures lentes).

Les indicateurs de performance comprennent le débit, le temps d'attente moyen et le nombre moyen d'arrêts. Les résultats de simulation obtenus ont été analysés puis représentés en annexe 2.

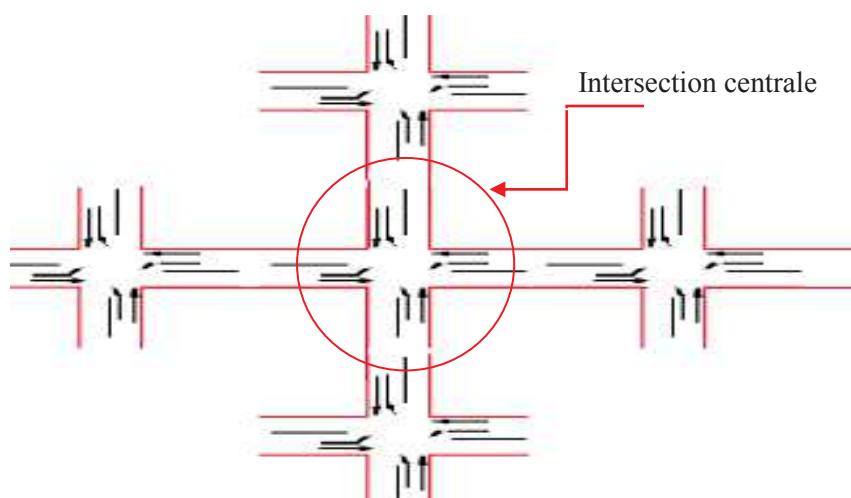


Figure 5. 5. Réseau de simulation

Les mesures de performance sont fonction du volume de trafic. La pression de volume est définie pour indiquer le degré occupé des voies. Pour évaluer le modèle proposé dans le réseau test, les trois niveaux de congestion sont définis comme suit:

- Faible pression : Pression de volume inférieure à 0,4.
- Pression modérée: Pression de volume : de 0,4 à 0,6.
- Pression élevée : Pression de volume supérieure à 0,6.

La moyenne calculée des résultats obtenus sont présentés sur les figures (figures 5.6, figure 5.7 et figure 5.8). Avec différentes intensités de la pression du volume, la discussion des résultats est regroupée selon les trois niveaux de congestion définis ci-dessus [Benhamza et al., 2015].

## 5.8 Discussion des résultats

### *a. Faible pression*

Avec une pression de faible volume, toutes les approches testées offrent de bons résultats. Le modèle proposé obtient les meilleures performances.

Comme le montre le graphique (figures 5.6, figure 5.7 et figure 5.8), le système proposé atteint, tout au long de la simulation, le plus haut débit, toujours avec le minimum de temps d'attente et le plus petit nombre d'arrêts.

### *b. Pression modérée*

En augmentant la pression du volume, le modèle proposé est toujours stable effectuant un débit proche de 100%. Le contrôle à temps fixe obtient les plus mauvais résultats, montrant des difficultés à traiter la pression du volume du trafic (le débit diminue rapidement). Aussi, les performances du contrôle à temps dynamique et le mécanisme de la vague verte se dégradent visiblement.

Aussi, le schéma proposé donne de bons résultats avec un temps d'attente moyen toujours inférieur à ceux du contrôle à temps dynamique et de la vague verte qui augmentent de façon exponentielle (Figures 5.6, 5.7 et 5.8).

De plus, le trafic se déplace facilement à travers le réseau, avec un maximum de 10% des voitures arrêtées. Le contrôle à temps dynamique offre un résultat entre 25% et 45% de voitures arrêtées; tandis que pour le contrôle de temps fixe, il y a seulement 40 à 60% des voitures en mouvement. Ainsi, il montre le caractère problématique du contrôleur à temps fixe.

## c. Pression élevée

Avec une pression de volume élevée, le modèle simple de contrôle à temps fixe bloque au début des simulations. Le débit obtenu par le schéma proposé commence à diminuer tandis que les autres approches obtiennent des débits très faibles (Figures 5.6, 5.7 et 5.8). Le temps d'attente moyen des méthodes comparées ainsi que le nombre d'arrêts ne cessent de croître et l'approche proposée offre toujours les meilleurs résultats.

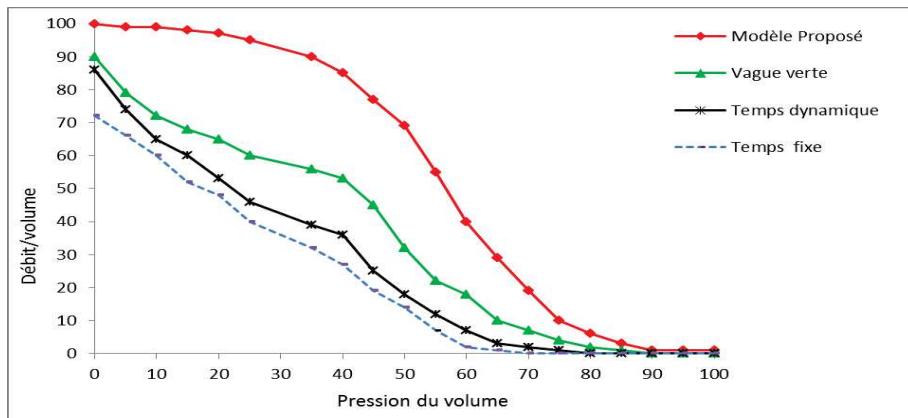


Figure 5.6. Courbe de comparaison du débit volumique

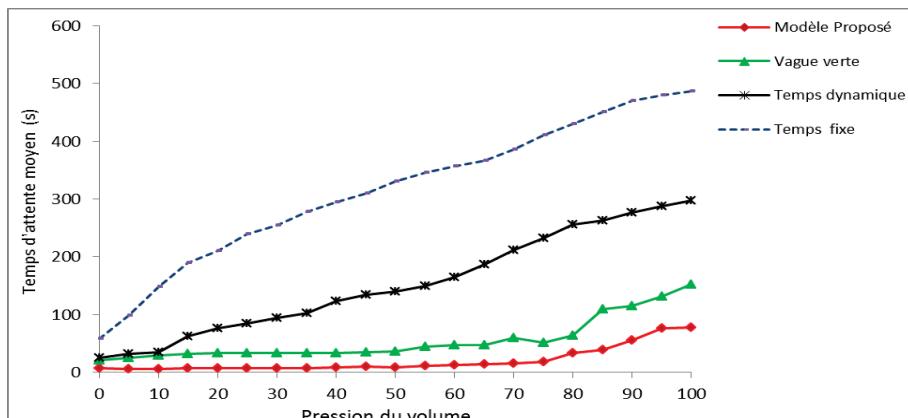


Figure 5.7. Courbe de Comparaison du temps moyen d'attente

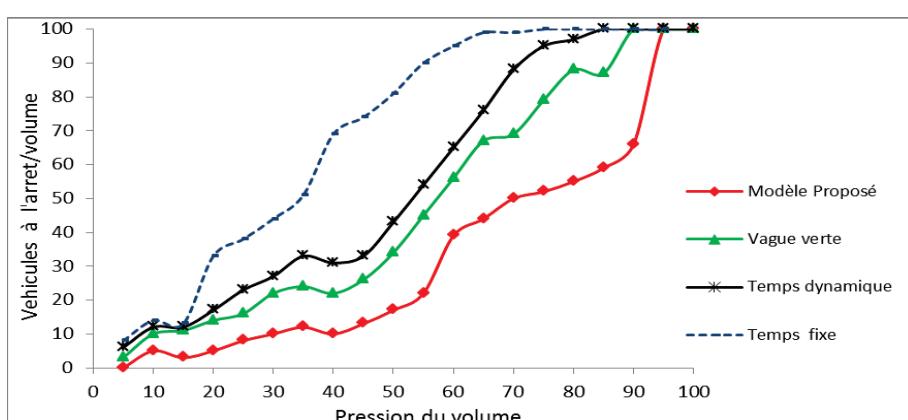


Figure 5.8. Courbe de Comparaison du nombre moyen d'arrêts

La méthode proposée de feux de circulation intelligents est mise en œuvre et comparée à trois méthodes de trafic existants. Après avoir testé plusieurs scénarios, il a été montré que cette méthode est en mesure d'améliorer la performance des feux de circulation en augmentant le flux de débit tout en réduisant les temps d'attente moyen et le pourcentage des voitures à l'arrêt. Ainsi, la performance du réseau est améliorée par le maintien d'une adaptation intelligente pour la synchronisation des feux de circulation.

## **Conclusion**

Le problème de la congestion de la circulation constitue un sérieux problème dans la vie urbaine provoquant des troubles sociaux comme des retards, des pertes économiques et une pollution environnementale.

Dans ce travail, une approche adaptative pour le contrôle des feux de signalisation de multiples intersections est proposée. Un système multi-agents auto-organisé a été conçu pour gérer la coordination réactive dans le contrôle de la circulation. Le modèle d'auto-organisé contient deux étapes: détection de données réelles du trafic et décision d'attribution de la lumière verte.

Basé sur les données locales du trafic en temps réel et les conditions de circulation des intersections voisines, les séquences des feux verts sont auto-réglées. Aussi, l'efficacité du réseau routier ne se base pas uniquement sur l'efficience de chacun des feux de circulation, mais aussi sur leur collaboration et leur coordination qui a été prise en considération dans la conception du modèle.

Les étapes de conception d'un système multi-agents auto-organisé ont été suivies et appliquée dans la résolution du problème difficile de la gestion des feux de circulation. Un flux de trafic optimal peut être atteint par le réseau routier en adoptant un mécanisme d'auto-organisation des agents de signalisation proposé.

## **Conclusion générale et perspectives**

## Conclusion générale et perspectives

Ce travail se situe dans le cadre des systèmes multi-agents, et traite la conception des mécanismes d'auto-organisation pour la résolution de problème. L'objectif de cette recherche est de trouver alors comment faire – à la conception du système et lors de son fonctionnement – pour garantir l'obtention d'un comportement pour lequel il est construit.

L'approche multi-agents est un excellent paradigme de modélisation et d'implantation de systèmes possédant de telles propriétés. Notamment, les systèmes organisés qui proposent des mécanismes, riches et pertinents, pour l'adaptation et la résolution de conflits, et qui peuvent être vus comme autant de moyens efficaces à appliquer dans les systèmes complexes. De ce fait, ce travail de recherche vise à répondre à une problématique essentielle dans les systèmes ouverts et dynamiques : la mise en place de mécanismes auto-organisationnels permettant l'émergence de solutions cohérentes et adaptatives. Ceci permet de garantir des propriétés macroscopiques du système conçu en ne développant que des entités et leurs comportements à un niveau élémentaire microscopique.

La contribution dans ce travail consiste donc, à répondre à cette question, déclinée en deux parties :

- D'abord, la formulation d'un processus générique de conception de SMA auto-organisé. L'approche combine le facteur organisationnel au niveau global qui permet de déclencher la réorganisation au niveau local. Après identification de situations imprévues par le système, l'intérêt se focalise alors sur l'évaluation des interactions entre agents. A leur niveau local, ces derniers estiment leurs simples interactions et procèdent par conséquent à une adaptation de relations qui les relient. Un processus d'auto-organisation est alors déclenché.

Cette proposition devrait fournir des éléments de solutions pour répondre aux exigences des systèmes existants et aux caractères spécifiques imposés par les propriétés de distribution, d'ouverture et d'adaptabilité.

- Ensuite, une proposition de modélisation, à travers l'application de ce prisme multi-agents auto-organisé, pour la résolution du problème de contrôle de la signalisation dans un réseau routier.

Nous avons montré à travers une étude de cas que l'approche est efficace et qu'elle présente plusieurs avantages : Elle permet, sans intervention humaine, de fournir une politique qui soit optimale. Le caractère complexe, dynamique et ouvert du système a permis de tester la force de l'adaptation de l'approche en traitant les situations de perturbation et les comportements non souhaités.

Le principe proposé a été validé ainsi à travers le développement d'une technique de résolution dédiée au problème complexe et réel de contrôle du trafic. Il a été appliqué au développement d'un modèle de signalisation intelligente et adaptative dans un réseau à multiple intersections. Les expérimentations mises en œuvre sur une plateforme de simulation ont montré une réelle robustesse et une grande adaptabilité de la solution développée.

Le document est structuré en deux parties de cinq chapitres. La première partie consiste en un état de l'art sur les SMA. Elle met en évidence l'intérêt des SMA en ce qui concerne les problèmes complexes qui sont mis en exergue depuis plusieurs années.

Tout d'abord, le chapitre 1 présente une exploration du domaine des SMA de leurs définitions et caractéristiques aux concepts clés utilisés dans ce travail.

L'objectif du chapitre 2 est de fournir un cadre pour les travaux présentés dans la thèse. L'accent est mis sur les caractéristiques principales du système complexe, et plus particulièrement l'auto-organisation et l'émergence. Les deux concepts ont été longtemps considérés comme un seul, et parfois en confusion. De ce fait, la combinaison des deux phénomènes est une approche prometteuse pour la résolution et la manipulation des systèmes complexes.

Dans le chapitre 3, les méthodologies traitant de la conception des SMA pour l'étude et la résolution des systèmes complexes ont été exposées. La problématique de la construction des mécanismes d'adaptation et d'autonomie dans les systèmes ont été soulevées. Le potentiel de l'auto-organisation dans les SMA a été souligné et de ce fait, la proposition était de retrouver par les agents les indices pour reconnaître les situations de perturbations et de s'auto-organiser en conséquence.

Dans la deuxième partie, une approche générique de conception de SMA auto-organisé pour la résolution de problème a été dégagée à partir de l'analyse précédente. Cette méthode a été exposée dans le chapitre 4.

Elle spécifie que l'environnement représentant le problème doit être modélisé avant les agents. Avec leurs interactions, ces derniers définissent un processus de réorganisation pour le contrôle et l'adaptation. Ce processus est construit aux niveaux individuel et collectif, en définissant tout d'abord l'environnement, les perceptions des agents, puis les interactions qui en découlent.

Dans le chapitre 5, nous avons montré que l'approche proposée permet de gérer efficacement le système complexe de contrôle et de signalisation de trafic routier avec un réseau de capteurs sans fil. Les spécificités de ces systèmes complexes et réels immergés dans des environnements ouverts et dynamiques rendent les techniques classiques difficiles à appliquer. Les résultats obtenus et comparés appuient l'efficacité de l'approche. L'utilisation des SMA auto-organisé permet ainsi d'assurer la gestion et le contrôle avec un caractère hautement adaptatif.

Néanmoins, nous avons vu que le choix des comportements à modéliser a une influence pertinente sur les performances de la décision et la résolution de problème. Il faut donc concevoir un modèle qui, à partir d'une description locale du SMA, fait correspondre un comportement global recherché.

Ainsi, l'approche présente quelques difficultés qui feront l'objet de futurs travaux de recherches:

- la fonction du système peut évoluer en cours de fonctionnement par rapport aux besoins initialement exprimés.
- toutes les situations auxquelles le système peut faire face ne peuvent pas être spécifiées.
- il n'y a plus de contrôle local ou global sur le système en cours d'exécution.

En perspective, l'approche générique proposée doit contenir donc des éléments de contrôle tels les méthodes et les outils associés pour guider et converger le système rapidement vers l'état désiré et éviter les situations d'erreurs.

En effet, la fonction du système peut être codée directement dans les agents, par la définition du système à produire, mais reste à influencer le collectif d'agents pour atteindre la fonction adéquate. En effet, si le système est ouvert, il peut subir des perturbations qui l'amènent à présenter un comportement stable non souhaité. Le contrôle permet alors de sortir de cette situation.

L'idée est de profiter de l'ouverture du système pour ajouter des agents dédiés au contrôle, sans modifier les agents déjà présents. Ainsi, une extension de ce travail concerne la mise en œuvre et l'évaluation des performances qui s'appuie sur une estimation locale par rapport au comportement global.

## **REFERENCES BIBLIOGRAPHIQUES**

## Références bibliographiques

[Akça et al., 2012] A. Akça, E. Uzun and T. Özyer, A real time traffic simulator utilizing an adaptive fuzzy inference mechanism by tuning fuzzy parameters, *Applied Intelligence*, Volume 36, Issue 3, pp. 698-720, 2012.

[Aldewereld et al., 2001] H., Aldewereld, and D., Virginia, *OperettA: Organization-oriented development environment, Languages, Methodologies, and Development Tools for Multi-Agent Systems*. Springer Berlin Heidelberg, pp. 1-18, 2011.

[Ali et al., 1997] S.M. Ali, R.M. Zimmer et C.M. Elstob, The question concerning emergence: Implication for Artificiality. Dans *DUBOIS, D.M., éditeur : Computing Anticipatory Systems: CASYS'97 - First International Conference*, 1997.

[Arlabosse, et al., 2004] F. Arlabosse, M.P. Gleizes et M. Occello, *Méthodes de conception, Systèmes multi-agents*, vol 29, 137-171, 2004.

[Azaiez, 2007] S. Azaiez, *Approche Dirigée par les modèles pour le développement de systèmes multi-agents*, Thèse de l'Université de Savoie, Spécialité Informatique, 2007.

[Baez 2005] J.Baez, T.Stratulat et J. Ferber Un modèle institutionnel pour SMA organisationnel. *Proceedings of Journées Francophones sur les Systèmes Multi-Agents (JFSMA 05)*, Calais-France, 2005.

[Bazzan, 2005] A. L. Bazzan, A distributed approach for coordination of traffic signal agents, *Autonomous Agents and Multi-Agent Systems*, 10(2), pp.131-164, 2005

[Bazzan et al., 2014] A. L., Bazzan, and F. Klügl, A review on agent-based technology for traffic and transportation, *The Knowledge Engineering Review*, 29(03), pp.375-40, 2014.

[Boes, 2014] J. Boes, *Apprentissage du contrôle de systèmes complexes par l'auto-organisation coopérative d'un système multi-agent: application à la calibration de moteurs à combustion* Doctoral dissertation, Université de Toulouse, Université Toulouse III-Paul Sabatier, (2014).

[Bellifemmine, 1999] F. Bellifemmine, A. Poggi, and G. Rimassa. *JADE - A FIPA compliant agent framework*. In *4th International Conference and Exhibition on the Practical Application of Intelligent Agents and Multi-Agents*, 1999.

[Benhamza et al., 2010] K. Benhamza, S. Ellagoun, H. Seridi and H. Akdag, “Agent based Modelling for traffic Simulation”, proc., International Symposium on Modelling and Implementation of Complex Systems (MISC’2010), pp. 219-227, université Mentouri Constantine, Algérie, 2010.

[Benhamza et al., 2012] K. Benhamza, S. Ellagoune, H. Seridi, H. Akdag, Agent-Based Modeling for Traffic Simulation, revue « Courrier du Savoir » – N°14, pp.51-56, université de Biskra, Novembre 2012.

[Benhamza et al., 2014a] K. Benhamza, H. Seridi, C. Himeur, Self-organized Approach for Adaptive Traffic Signal Control, Colloque sur l'Optimisation et les Systèmes d'Information, COSI'2014, 8-10 Juin 2014, Béjaia, Algérie

[Benhamza et al., 2014b] K. Benhamza, H. Seridi, Intelligent traffic light control using an adaptive approach, In Proceedings of the International Conference on Information Technology for Organization Development, IT4OD 2014, pp. 246-250, Tebessa, Algérie.

[Benhamza et al., 2015] K. Benhamza, H. Seridi, Adaptive traffic signal control in multiple intersections network, Journal of Intelligent and Fuzzy Systems, vol. 28 no.6, pp 2557-2567, 2015

[Bertron et al., 2009] C. Bertron, M. P. Gleizes, & G. Picard, Méthodes orientées agent et multi-agent, Technologies des systèmes multi-agents et applications industrielles, chapter-2. 2009.

[Black 2007] G. Black et V. Vyatkin. On practical implementation of holonic control principles in baggage handling systems using IEC 61499. Holonic and Multi-Agent Systems for Manufacturing, pp. 314–325, 2007.

[Boissier 04] O., Boissier, S., Gitton, P. Glize, Caracteristiques des Systemes et des Applications. Systemes Multi-Agents, Observatoire Francais des Techniques Avancees, ARAGO 29, Diffusion Editions TEC & DOC, pp. 25-54, 2004.

[Bonabeau et al., 1997] E. Bonabeau, G. Theraulaz, JL Deneubourg, S. Aron, S. Camazine, Self-organization in social insects. Trends Ecology Evolution 12: 188–193, 1997.

[Bonabeau et al., 1999] E. Bonabeau, M. Dorigo, Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, 1999.

[Booch, 1992] G. Booch, Conception orientée objets et applications, Addison-Wesley, 1992

[Bourjot 1999] Bourjot, C., Chevrier, V., Bernard, A., Krafft, B., "Coordination par le biais de l'environnement : une approche biologique", Journées Francophones Intelligence Artificielle Distribuée et Systèmes Multi-Agents JFIADSMA '99, Editions Hermès 1999.

[Bingham, 2001] E. Bingham. Reinforcement learning in neurofuzzy traffic signal control. European Journal of Operational Research, 131(2) :232–241, 2001.

[Buffet et al., 2005] O. Buffet, A. Dutech, & F. Charpillet, Développement autonome des comportements de base d'un agent. Revue d'intelligence artificielle, 19(4-5), 603-632, 2005

[Calderoni et al., 1997] S. Calderoni, R. Courdier, S. Leman, P. Marcenac, « Construction expérimentale d'un modèle multi-agents », in J. Quinqueton, M.-C. Thomas, B. Trouse (eds), 5èmes Journées Francophones sur l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents, Hermès Science, Nice, p. 109-123, Avril, 1997.

[Camazine et al., 2001] S. Camazine, J.-L. Deneubourg, NR Franks, G. Theraulaz, E. Bonabeau, Self-organization in biological systems. Princeton University Press, Princeton 2001.

[Capera et al., 2003] D. Capera, J. P. Georgé, M.-P. Gleizes, and P. Glize. “The AMAS theory for complex problem solving based on self-organising cooperative agents”. International IEEE, International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, pp. 383-388, 2003.

[Castelfranchi, 1998] C. Castelfranchi. Modeling social action for AI agents. Artificial Intelligence, 103(1-2) :157–182, 1998.

[Collis et al., 1998] J. C. Collis, D. T. Ndumu, H. S. Nwana, and L.C. Lee. The ZEUS agent building toolkit. *BT Technology Journal*, 16(3), 1998.

[Cossentino, 2001] Cossentino M., Different Perspectives in Designing Multi-Agent System, Designing Multi-Agent System, AgeS'02 (Agent Technology and Software Engineering) Workshop at NodE'02, 2001.

[Chen et al., 2010] Chen, B., and Cheng, H. H., A review of the applications of agent technology in traffic and transportation systems. Intelligent Transportation Systems, IEEE Transactions on, 11(2), 485-497, 2010.

[Chevaleyre et al., 2005] Chevaleyre, Y., Dunne, P. E., Endriss, U., Lang, J., Maudet, N. et Rodríguez-aguilar, J. A., Multiagent Resource Allocation ». In : Knowledge Engineering Review 20.2, p. 143–149, 2005

[Da Silva, 2006] B.C. Da Silva, D. Oliveira, E.W. Basso and A.L.C. Bazzan, Adaptive Traffic Control with Reinforcement learning, Proceedings of the 4th Workshop on Agents in Traffic and Transportation (ATT 2006), Japan, 2006.

[De Weerdt et al., 2009] De Weerdt, M. et Clement, B., Introduction to Planning in Multiagent Systems, In Multiagent and Grid Systems – An International Journal (MAGS) 5.4, p. 345–355, 2009

[Deloach et al.,2001] Deloach S., Wood M., Sparkman C., Multiagent Systems Engineering, International Journal of Software Engineering and Knowledge Engineering, vol.11, n° 3, p. 231-258, World Scientific, 2001.

[Demazeau et al., 1990] Demazeau Y., Muller J.P., Decentralized AI Proc of the first European workshop on Modeling Autonomous Agent in Multi-agent Word, Elsevier Sc, Pub 1990.

[Demazeau, 1995] Demazeau, Y. : From interactions to collective behavior in agent-based systems. In proceedings of the First European Conference on Cognitive Science, pp. 117-132, 1995.

[Demazeau, 2001] Demazeau Y., VOYELLES, Mémoire d'habilitation à diriger des recherches, INP Grenoble, 2001.

[Dewolf et al., 2004] Dewolf T., Holvoet T. Emergence and self-organisation: a statement of similarities and differences. In Proceedings of the International Workshop on Engineering Self-Organising Applications, pp 96–110, 2004.

[Dewolf et al., 2005] Dewolf T., Holvoet T.. Emergence versus self-organisation: different concepts but promising when combined; Engineering Self organizing Systems pp. 77-91, 2005.

[Dondio et al. 2006] Dondio, P., Barrett, S., Weber S. et Seigneur J. M., Extracting trust from domain analysis : A case study on the wikipedia project, In : Autonomic and Trusted Computing. Springer, p. 362–373, 2006.

[Doran et al., 1997] Doran J. E., Franklin S., Jennings N. R., and Norman T. J., On cooperation in multi-agent systems. Knowledge Engineering Review, 12(3):309–314, 1997.

[Dresner et al., 2008] Dresner K. and Stone P., A multiagent approaches to autonomous intersection management, Journal of Artificial Intelligence Research, 31:591-656, 2008.

[Ehlert et al., 2001] Ehlert, P. A., and Rothkrantz, L. J., Microscopic traffic simulation with reactive driving agents. In Intelligent Transportation Systems, Proceedings, IEEE, pp. 860-865, 2001.

[Ferber, 1995] Ferber J., Les systèmes multi-agents, vers une intelligence collective. Edition Inter-editions, 1995.

[Ferber, 1997] J. Ferber, les systèmes multi-agents. Un aperçu général. Revue Technique et sciences informatiques, 16, 8, 1997.

[Ferber, 2000] J. Ferber and O. Gutknecht. Madkit: A generic multi-agent platform. In 4 th International Conference on Autonomous Agents., 2000.

[Ferber et al., 2003] Ferber J., Gutknecht O. et Michel F. From agents to organizations: an organizational view of multi-agent systems. Agent-Oriented Software Engineering IV, pages 443–459, 2003.

[Ferber et al., 2005] Ferber J., Michel F. et Baez J. AGRE : Integrating environments with organizations. Environments for Multi-agent Systems, pages 48–56, 2005.

[Gasser, 2001] Gasser, Les, Perspectives on organizations in multi-agent systems, Multi-Agent Systems and Applications, Springer Berlin Heidelberg, 1-16, 2001.

[Gechter, 2013] Gechter, Franck "Systèmes multi-agents réactifs d'inspiration physique pour la prise de décision et la résolution de problèmes." HDR thesis, Université de Franche Comté, 2013.

[Georgé, 2003] Georgé, J-P, L'émergence. Rapport interne IRIT/2003-12-R, Institut de Recherche en Informatique de Toulouse (IRIT), 2003.

[Georgé, 2004] Georgé, J.-P., Edmonds, B., Glize, P., Making Self-Organizing Adaptive Multi-Agent Systems Work - Towards the engineering of emergent multi-agent systems, Methodologies and Software Engineering for Agent Systems, Kluwer Publishing, 2004.

[Georgé et al., 2005] Georgé, J-P, and M-P Gleizes, Experiments in emergent programming using self-organizing multi-agent systems, Multi-Agent Systems and Applications IV. Springer Berlin Heidelberg, 450-459, 2005.

[Gershenson, 2005] C. Gershenson, Self-organizing traffic lights, Complex Systems 16(1): 29-53, 2005.

[Goldman et al., 2004] Goldman, C. et Zilberstein, S. (2004). Decentralized control of cooperative systems: Categorization and complexity analysis. Journal of Artificial Intelligence Research, 22:143-174.

[Goldstein, 1999] Goldstein, J. , Emergence as a construct: History and issues, Emergence volume 1, Issue 1, pages 49-71, 1999

[Gomez et al., 2002] Gomez S. J., Fuentes R., Agent Oriented System Engineering with INGENIAS, Fourth Ibero american Workshop on Multi-Agent Systems, Iberagents'02, 2002.

[Gleizes et al., 2001] Marie Pierre Gleizes, Carole Bernon, Valérie Camps, Pierre Glize « La conception de systèmes multi-agents adaptatifs : contraintes et spécificités » Atelier de

Méthodologie et Environnements pour les Systèmes Multi-Agents (SMA 2001), Plate-forme AFIA, Université Toulouse III – Paul Sabatier, 2001.

[Gleizes ,2004] Gleizes M.P. Vers la résolution de problèmes par émergence. Thèse d'habilitation de l'Université de Toulouse, 2004.

[Gordon et al., 2005] R. L. Gordon and W. Tighe, Traffic Control System Handbook. FHWA-HOP-06-006, U.S. DOT, FHWA, Washington, DC, 2005

[Grondin, 2008] Grondin G, «MaDcAr-Agent : un modèle d'agents auto-adaptables à base de composants », thèse, l'École Nationale Supérieure des Mines de Saint-Étienne, 2008.

[Guessoum, 2003] Z. Guessoum, « Modèles et architectures d'agents et de systèmes multi-agents adaptatifs », Université Paris 6, 2003.

[Gutknecht, 2001] O. Gutknecht. Proposition d'un modèle organisationnel générique de systèmes multi-agents. Thèse de doctorat, Université Montpellier II, Septembre 2001.

[Hassas 2003] S. Hassas. Systèmes complexes à base de multi-agents situés. University Claude Bernard Lyon, 2003.

[Heylighen, 2001] Heylighen F. The Science of Self-organization and Adaptivity, The Encyclopedia of Life Support Systems, (EOLSS Publishers Co. Ltd), 2001

[Iglesias et al., 1999] Iglesias, C., Garrijo, M. and Gonzale, J., A Survey of Agent-Oriented Methodologies, Proceedings of the 5th International Workshop on Intelligent Agents V (ATAL-98) ,Vol. 1555, Springer-Verlag p. 317-330, 1999.

[Ishida, 2004] Ishida, Y. Immunity-based systems: a design perspective. Springer, 2004.

[Jelasity et al. 2007] Jelasity, M., Voulgaris, S. Guerraoui, R., Kermarrec, A. et Van Steen, M., Gossip-based peer sampling, In : ACM Transactions on Computer Systems (TOCS) 25.3, 2007.

[Jennings et al., 1998] Jennings.N.R., Sycara,K., Wooldridge.M., A roadmap of agent research and development, International Journal of Autonomous Agents and Multi-Agent Systems (1), pp 7-38, 1998.

[Koestler, 1979] A. Koestler, The ghost in the machine. Hutchinson, London, the danube ed.2. Impr. édition, 1979.

[Krajzewicz, 2010] D. Krajzewicz, Traffic Simulation with SUMO – Simulation of Urban Mobility, In: Fundamentals of Traffic Simulation International Series in Operations Research and Management Science. Springer. Seiten 269-294. ISBN 978-1-4419-6141-9. ISSN 0884-8289, 2010.

[Knaian , 2000] A. Knaian. A wireless sensor network for smart roadbeds and intelligent transportation systems. PhD thesis, Citeseer, 2000.

[Lammer et al., 2008] S. Lammer and D. Helbing, Self-control of traffic lights and vehicle flows in urban road networks, Journal of Statistical Mechanics: Theory and Experiment, Vol. 2008, No.4, pp. 1-36, 2008.

[Langton, 1990] C. Langton, Computation at the Edge of Chaos – Phase Transition and Emergent Computation, Physica D, 4, 1990.

[Mostefaoui et al. 2003] Mostefaoui S., Rana O., Foukia N., Hassas S., Marzo-Serugend G. D., Van-Aart C., Karageorgos A. Self-organising applications: A survey. In International Workshop on Engineering Self-Organising Applications, 2003.

[Morin, 1977] Edgar Morin. La méthode 1 : La nature de la nature. Éditions du Seuil, 1977

[Malone, 1999] Thomas W. Malone. Tools for inventing organizations : Toward a handbook of organizational process. Management Science, 45(3) :425–443, March 1999.

[Moujahed et al., 2009] Moujahed S., Simonin O. et Koukam A., Location problems optimization by a self-organizing multiagent approach, Multiagent and Grid Systems 5.1, p. 59–74, 2009.

[Muller, 1998] Müller, J. P., Vers une méthodologie de conception de systèmes multi-agents de résolution de problème par émergence, In : JFIADSMA'98. Hermes, 1998.

[Muller, 2002] Müller J.P. Des systèmes autonomes aux systèmes multi-agents: interactions, émergence et systèmes complexes» Rapport d'HDR, Montpellier LIRMM, 8 novembre, 58p. (2002).

[Odell, 2002] Odell, J., Agents and complex systems, JOT, Vol. 1, pp. 35–45, 2002

[Padgham et al., 2003] Padgham, Lin, and Michael Winikoff, Prometheus: A methodology for developing intelligent agents, Agent-oriented software engineering III, Springer Berlin Heidelberg, 174-185, 2003.

[Pappis et al., 1977] C. P. Pappis and E. H. Mamdani, A fuzzy logic controller for a traffic junction, IEEE Trans. On Systems, Man and Cybernetics, Vol. SMC-7, No. 10, pp. 707-717, October 1977.

[Panait et al., 2005] Panait.L., Luke.S. (2005). Cooperative Multi-Agent Learning: The State of the Art. In Autonomous Agents and Multi-Agent Systems, 00, pp 1–48, Springer.

[Parunak, 2000] H.V.D. Parunak, Industrial and Practical Applications of DAI. Multiagent systems : a modern approach to distributed artificial intelligence, page 377, 2000.

[Parunak and al., 2004] H.V.D. Parunak and S. A. Brueckner, Engineering swarming systems. In Methodologies and Software Engineering for Agent Systems, pages 341–376. Kluwer, 2004.

[Perozo, 2011] N. Perozo, Modélisation multi-agents pour systèmes émergents et auto-organisés, doctoral dissertation, Université de Toulouse, Université Toulouse III-Paul Sabatier 2011.

[Picard et al., 2004] Picard, Gauthier, and Marie-Pierre Gleizes, The ADELFE methodology, Methodologies and Software Engineering for Agent Systems. Springer US, 157-175. 2004.

[Picard and al., 06] Picard, G., Glize, P. Model and Analysis of Local Decision Based on Cooperative Self- Organization for Problem Solving. In: Multiagent and Grid Systems, IOS Press, Vol. 2(3), 253-265, September 2006.

[Russell et al. 1995] Russell S., Norvig P. Artificial Intelligence: A Modern Approach, Prentice Hall, Englewood Cliffs, New Jersey, 1995.

[Russell et al. 2006] Russell S., Norvig P. Intelligence artificielle. Pearson Education 2e édition. 2006.

[Robertson et al., 1991] D. Robertson and R. Bretherton. Optimizing networks of traffic signals in real time-the scoot method. IEEE Transactions on Vehicular Technology, 40(1):11–15, Feb. 1991.

[Roozemond, 1999] D. A. Roozemond, Using intelligent agents for urban traffic control systems, In Proceedings of the International Conference on Artificial Intelligence in Transportation Systems and Science, pp. 69-79, 1999.

[Sanchez et al., 2010] J. Sanchez, M. Galan and E. Rubio, Traffic signal optimization in “la almozara” district in saragossa under congestion conditions using genetic algorithms, traffic microsimulation and cluster computing, Intelligent Transportation Systems, IEEE Transactions on, 11:132-141, March 2010.

[Scott, 1998] W. R. Scott. Organizations : rational, natural and open systems. Prentice Hall, 4edition, 1998.

[Senart, 2003] Senart A, «Canevas logiciel pour la construction d'infrastructures logicielles dynamiquement adaptables». Thèse de l'institut National Polytechnique de Grenoble, France, 2003.

[Serugendo and al., 2005] G.D.M. Serugendo, M.P. Gleizes et A. Karageorgos. Self-organisation and Emergence in Multi-Agent Systems. *The Knowledge Engineering Review*, vol. 20, no. 2, pages 165–189, 2005.

[Serugendo and al., 2005] G.D.M. Serugendo, M.P. Gleizes et A. Karageorgos. Self-organisation and Emergence in MAS: an overview, *Informatica* 30(1): 45-54 Slovene Society Informatika, 2006

[Shalizi, 2001] Shalizi, Cosma Rohilla. Causal architecture, complexity and self-organization in the time series and cellular automata. Diss. University of Wisconsin--Madison, 2001.

[Shehory et al., 2001] Shehory, O. and Sturm, A. Evaluation of modeling techniques for agent-based systems. In *Proceedings of the 5th International Conference on Autonomous Agents* (Montreal, Ont., Canada, June). ACM, New York, pp. 624–631. 2001.

[Schneider, 1994] Schneider, D. K.. Modélisation de la démarche du décideur politique dans la perspective de l'intelligence artificielle. Thèse de l'Unité Technologie de Formation et Apprentissage - Faculté de Psychologie et des Sciences de l'Education - Université de Genève, 1994.

[Shoham, 1993] Shoham Y., *Agent oriented Programming*, *artificial Intelligence*, 60, 1993

[Shoham, 2009] Shoham, Y. et Leyton-Brown, K. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2009.

[Simonin et al., 2006] Simonin, O., et Gechter, F, An environment-based methodology to design reactive multi-agent systems for problem solving, *Environments for Multi-Agent Systems II* (pp. 32-49), Springer Berlin Heidelberg, 2006.

[Simonin, 2010] Simonin, O. Contribution à la résolution collective de problème (Modèles d'auto-organisation par interactions directes et indirectes dans les SMA réactifs et robotiques), Doctoral dissertation, Université Henri Poincaré-Nancy I, 2010.

[Sims et al., 1980] A. Sims and K. Dobinson. The sydney coordinated adaptive traffic (scat) system philosophy and benefits. *IEEE Transactions on Vehicular Technology*, 29(2) :130 – 137, May 1980.

[Tveit, 2000 ] Tveit, A., *A survey of Agent-Oriented Software Engineering*. Report, Norwegian University of Science and Technology, 2000.

[Van de Vijver, 1997] G. Van de Vijver. Émergence et explication. *Intellectica*, 2(25) :7–23, 1997.

[Vercouter, 2004] Vercouter, L., MAST : Un modèle de composants pour la conception de SMA. 1ere Journée Multi-Agents et Composants (JMAC'04), Paris, France, 23 Novembre 2004.

[Versteegh 2010] F. Versteegh, M.A. Salido et A. Giret. A holonic architecture for the global road transportation system. *Journal of Intelligent Manufacturing*, vol. 21, no. 1, pages 133–144, 2010.

[Weyns et al., 2005] Weyns, D., Parunak, H. V. D., Michel, F., Holvoet, T., & Ferber, J. Environments for multiagent systems state-of-the-art and research challenges. In *Environments for multi-agent systems* (pp. 1-47). Springer Berlin Heidelberg, 2005.

[Weyns et al., 2007] Weyns, D., Omicini, A., & Odell, J. Environment as a first class abstraction in multiagent systems. *Autonomous agents and multi-agent systems*, 14(1), 5-30, 2007.

[Wiering et al., 2003] M.A. Wiering, J. Van Veenen, J. Vreeken and A. Koopman, Intelligent traffic light control, *ERCIM News*, European Research Consortium for Informatics and Mathematics, 53:40-41, 2003.

[Wooldridge et al., 1995] M. Wooldridge et N. R. Jennings: Intelligent agents Theory and practice. *The Knowledge Engeneering Review*, 10(2) : 115-152, 1995.

[Wooldridge et al., 2000] M. Wooldridge, N. R. Jennings., D.Kinny, The Gaia Methodology for Agent- Oriented Analysis and Design, *Journal of Autonomous Agents and Multi-Agent Systems*, vol. 3, n° 3, p. 285-312, Kluwer Academic Publishers, 2000.

[Xie et al., 2011] Xiao-Feng Xie, G.J. Barlow, S.F. Smith and Z.B. Rubinstein, Platoon-based self-scheduling for real-time traffic signal control, *Intelligent Transportation Systems (ITSC)*, 14th International IEEE Conference, pp. 879 – 884, 2011.

[Yokoo, 2001] M. Yokoo, *Distributed Constraint Satisfaction : Foundations of Cooperation in Multi-Agent Systems*. Springer (cité pp. 21–23, 47), 2001.

[Zambonelli et al., 2003] F. Zambonelli, N. Jennings, and M. Wooldridge. Developing multiagent systems: The Gaia methodology. *ACM Trans. Software Eng. Meth.* 12(3), pp. 417 – 470, 2003.

[Zambonelli et al., 2004] F. Zambonelli, M.-P. Gleizes, M. Mamei, and R. Tolksdorf. Spray computers: frontiers of self-organisation for pervasive computing. In 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'04), pp. 397-402, 2004.

[Zeghal et al., 1994] K. Zeghal, , J. Ferber A reactive approach for distributed air traffic control, proceedings of Avignon94, p. 381–390, 1994.

# ANNEXES

## ANNEXE 1

### Simulation of Urban MObility - SUMO

SUMO [Krajzewicz, 2010] est un simulateur open-source à temps discret, espace continu et microscopique entièrement réalisé en C++ permettant de modéliser l'écoulement du trafic routier, ceci en traitant une gamme de cas plus réalistes et plus larges que GLD (qui se limite principalement à l'optimisation du temps de parcours). Bien que plus complexe à mettre en place, SUMO possède également l'avantage d'être toujours maintenu par ses auteurs et d'être doté d'une documentation complète. La dernière version du logiciel date de novembre 2011 et est disponible sur la page officielle du projet (<http://sumo.sourceforge.net>). Globalement, l'utilisation de ce logiciel peut se décomposer en trois points détaillés ci-après.

1. **La construction d'un réseau :** La génération d'une carte (ou réseau) n'est pas nécessairement automatisée par une interface mais repose sur la construction d'un fichier XML de type `projet.net.xml`, qui peut être obtenu de multiples manières. SUMO peut mettre en place des structures plus complexes, incorporant le changement de voie, des boucles électromagnétiques, la création de types de véhicules, et bien d'autres.

Deux principaux outils existent pour générer de tels réseaux :

- NETGEN permet de générer en ligne de commande un réseau de plusieurs formes : en grille, en toile ou aléatoire.
- NETCONVERT possède la capacité de générer des cartographies adaptées pour SUMO à partir de différents formats.
- De base, il est possible de définir un ensemble de fichiers XML afin d'obtenir le réseau souhaité.

Cette démarche peut sembler lourde à faire manuellement, mais permet de mieux cerner le fonctionnement du logiciel, et reste suffisante pour de petits réseaux.

– Tout d'abord, il est nécessaire de décomposer la carte souhaitée en un ensemble de noeuds, qui peuvent être de plusieurs types : de bordures, fictifs pour la réduction ou l'augmentation de voies, ou encore d'intersections, avec ou sans feux (règle de la priorité à droite).

Ces nœuds et leurs coordonnées respectives prennent place dans un fichier de la forme `projet.nod.xml`

- Ensuite, un fichier projet.typ.xml permet de mettre en place les types de routes souhaitées, incluant pour chaque type des informations telles que le nombre de voies ou encore la vitesse limite. Ce fichier de types est optionnel, les paramètres pouvant directement être indiquées lors de la construction des routes.
- Puis, un fichier projet.edg.xml permet de relier les noeuds par des routes.
- Le fichier projet.con.xml permet pour sa part de mettre en place les directions possibles pour chaque voie.
- Le fichier projet.netc.cfg permet d'indiquer tous ces fichiers en vue de la création de la carte par le programme NETCONVERT, permettant d'obtenir le fichier projet.net.xml.
- Pour plus de confort, NETCONVERT possède l'avantage de prendre en charge de nombreux types de cartes existants, en provenance de certains simulateurs ou plateformes reconnues, telles que MatSim ou encore OpenDrive. Particulièrement, OpenStreetMap est également pris en charge : il s'agit d'une carte mondiale en partage et totalement libre, dont chacun peut contribuer à la construction ainsi qu'exporter des données, généralement au format .OSM.

Finalement, il est possible d'obtenir un fichier de type projet.net.xml à partir de formats .OSM via NETCONVERT. Plus loin encore, certaines structures décoratives telles que les rivières, les immeubles et autres éléments graphiques peuvent être importés à l'aide de POLYCONVERT, également inclus dans SUMO. Un exemple de création peut être illustré, qui vise à reproduire une ville, aussi bien au niveau de la cartographie que du trafic routier en lui-même (figure A.1).

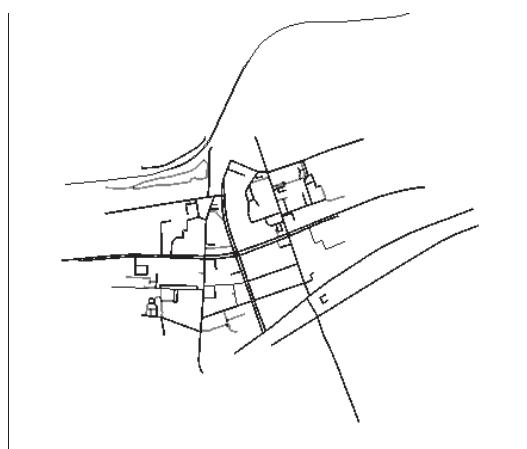


Figure A.1 : Illustration d'une carte

**2. Modélisation de la demande :** La génération d'un réseau routier pouvait se faire par plusieurs moyens : soit à la main directement, soit via les utilitaires fournis avec le

simulateur. Une fois ce réseau obtenu, l'étape suivante consiste à définir les informations à propos des véhicules, et notamment des routes qu'il est possible pour eux d'emprunter.

SUMO a besoin de connaître les routes possibles du réseau afin de générer les mouvements des différents véhicules. Le moyen le plus simple d'en obtenir est de les éditer manuellement, mais uniquement si leur nombre n'est pas trop élevé. La démarche manuelle va donc être de créer soi-même un fichier du type projet.rou.xml qui va se charger de définir les types de véhicules existant ainsi que les routes possibles.

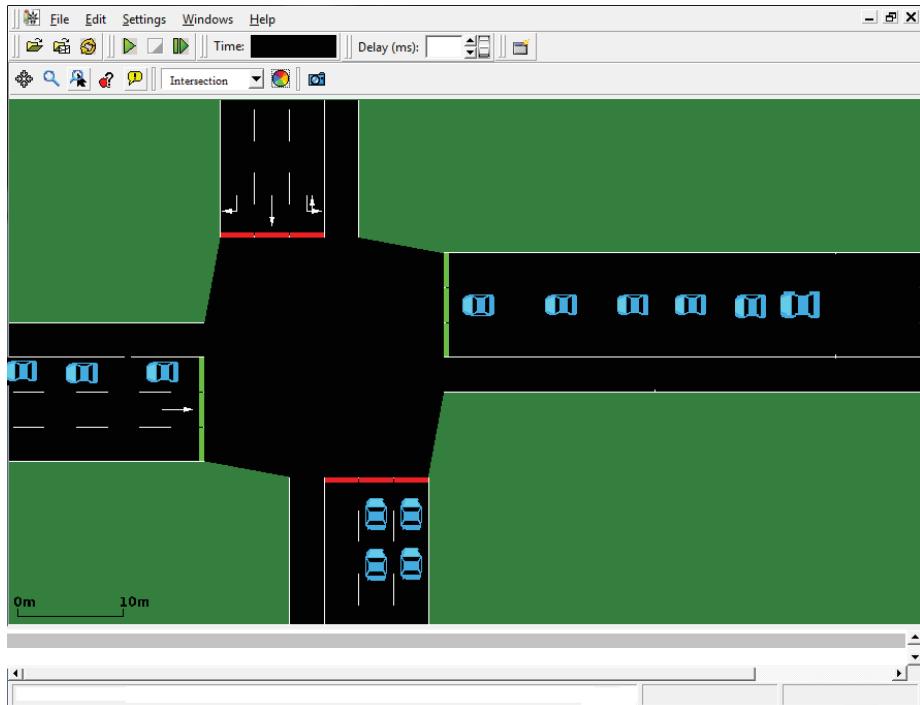


Figure A.2. Exemple d'intersection

Les véhicules en circulation sur la simulation vont être alors indiqués en utilisant ces deux paramètres, et de deux manières : soit sous forme de flots arrivant de manière continue pendant une certaine période, soit individuellement. Sur de larges réseaux, SUMO fournit plusieurs applications permettant de générer ce fichier en fonction de multiples critères :

- DUAROUTER : sur la base d'un ensemble de voyages, cette application obtient les routes correspondantes, en cherchant le plus court chemin entre le point de départ et d'arrivée suivant l'algorithme de Dijkstra.
- DJTRROUTER reprend des principes similaires, mais utilise des probabilités de tourner afin de générer les routes.
- DFROUTER s'appuie sur les mesures effectuées par les détecteurs présents.

– ACTIVITYGEN se repose sur une description de la population vivant sur le réseau : nombre d'habitants, de ménages, la probabilité pour un adulte d'avoir une voiture, d'utiliser les transports en commun, description des zones d'activité de la ville (ex : école), etc.

– Notons la possibilité de générer des routes aléatoirement en s'aidant d'un outil inclus dans SUMO et des deux premières applications.

3. **Simulations** : Une fois le réseau et les informations du trafic souhaité générées, il faut mettre en place le fichier de configuration correspondant à la simulation, de la forme projet.sumo.cfg.

Ce fichier va rassembler les deux fichiers projet.net.xml et projet.rou.xml, ainsi que d'autres éléments, tels que :

- Des éventuels fichiers supplémentaires (pour la définition de stops, de détecteurs,..).
- Le temps de simulation (début et fin, en pas de programme, ajustables en millisecondes).

La simulation ainsi définie est exécutable par l'intermédiaire de deux programmes : via une interface graphique (SUMO-GUI) mais également directement en ligne de commande (SUMO), où il est possible de passer certains arguments habituellement définis dans le fichier de configuration directement dans la commande. L'affichage de l'interface graphique apparaît propre, et de divers types : très basique, standard ou sous une vue plus réaliste. La possibilité de régler la vitesse d'exécution de la simulation existe toujours, ceci est ajustable en définissant le nombre de millisecondes auxquelles correspond un pas de programme.

Les paramètres d'affichage sont entièrement personnalisables : couleur des véhicules en fonction de leur vitesse, de leur provenance, etc.

Concernant le cas des feux de circulation, ces derniers possèdent ici les trois couleurs (rouge, orange, vert) et suivent un principe de cycle et de phase clairement définis et visibles. Par défaut, NETCONVERT et NETGEN s'occupent de générer les feux de circulation et leur fonctionnement lors de la mise en place du réseau.

Seulement, ces fonctionnements ne suivent pas nécessairement ce que nous pouvons retrouver dans la réalité. Ainsi, il est possible de charger des programmes supplémentaires afin de régir le fonctionnement des feux de circulation : l'utilisateur peut configurer à sa convenance le système de gestion des feux, l'opération se faisant par défaut manuellement avec un fichier de configuration fixe. Enfin, des détecteurs peuvent être mis en place dans des fichiers additionnels (manuellement, ou via des utilitaires intégrés).

Ces derniers peuvent servir au recueil de données et à l'évaluation des performances de feux de circulation. L'idée de base de cet outil est de donner l'accès à une simulation en cours, afin de récupérer des données en temps réel et éventuellement de modifier certains paramètres (Figure A.3).

En tout, trois classes de possibilités sont offertes :

- Action sur la simulation et l'interaction client/serveur : clore la connexion, effectuer un pas de programme, calculer certaines valeurs, etc.
- Obtention d'informations : en provenance des détecteurs, des feux de circulation, des voies, routes, véhicules, routes ou même sur la simulation en elle-même.
- Modification d'état : des feux de circulation, des voies ou encore des véhicules.

Aussi, ce simulateur fait preuve d'ouverture, en permettant par exemple à TraCI de s'interfacer avec des logiciels de tout type : solutions intégrées (TraCI en Python, TraCI4J en Java) ou simulateurs réseaux répandus (Middlewares Trans ou MOVE pour NS2, iTETRIS pour NS3, VEINS ou VSIMRTI pour OMNet++).

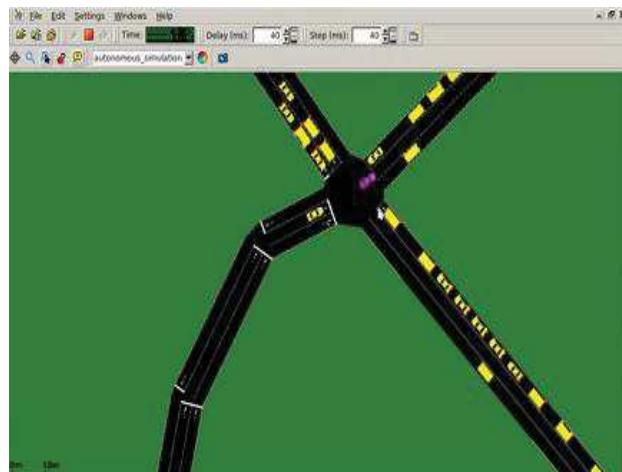


Figure A.3 Exemple de simulation

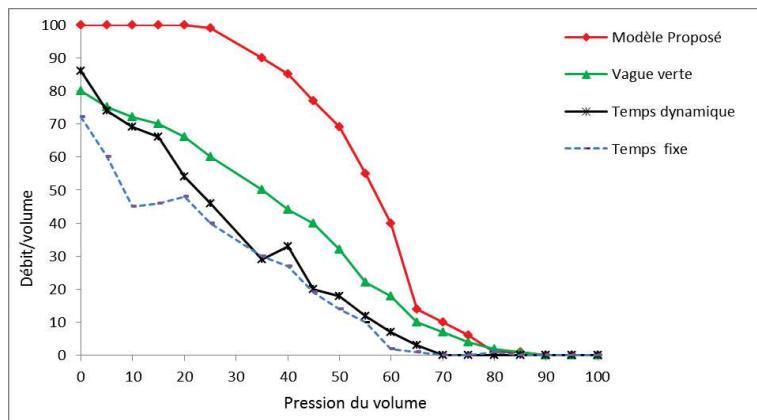
Enfin, les outils offerts sont nombreux, rendant l'utilisation du simulateur SIMO agréable, passée une phase d'apprentissage.

## ANNEXE 2

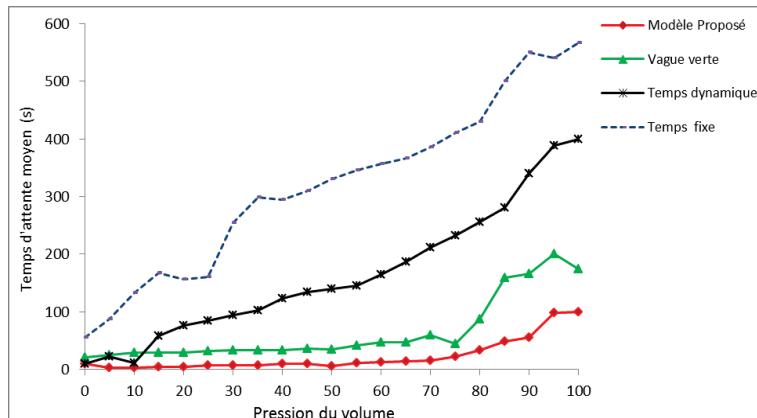
### Contrôleurs à Temps fixe, temps dynamique et vague verte Versus contrôleur adaptatif proposé

Cette annexe comprend les graphiques qui montrent la performance de signal adaptatif proposé sur les trois contrôleurs existants (Temps fixe, temps dynamique et vague verte).

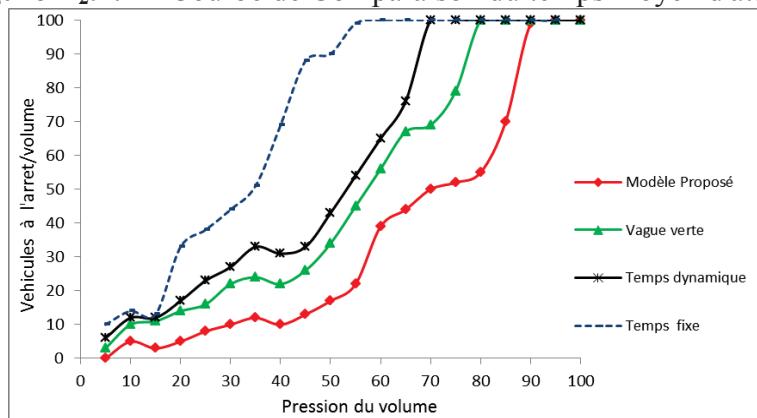
#### Simulation 1



**Figure A2.1.1** Courbe de comparaison du débit volumique

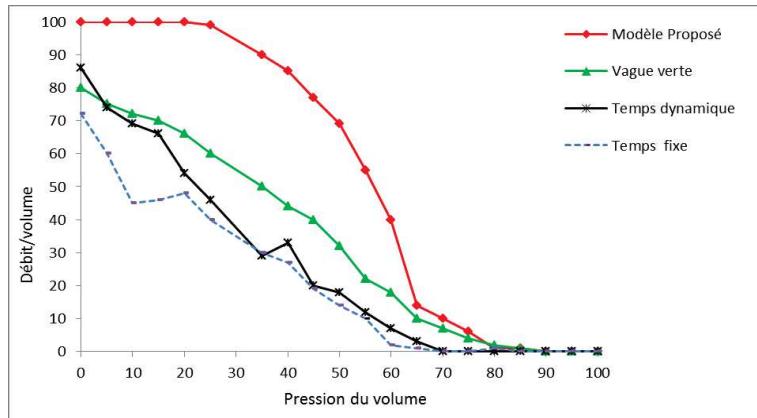


**Figure A2.1.2** Courbe de Comparaison du temps moyen d'attente

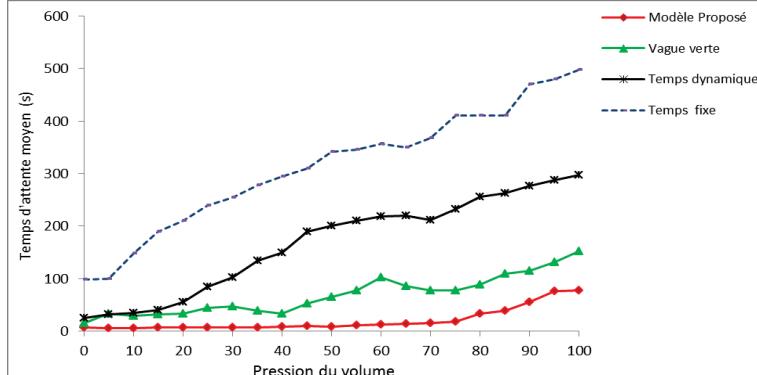


**Figure A2.1.3** Courbe de Comparaison du nombre moyen d'arrêts

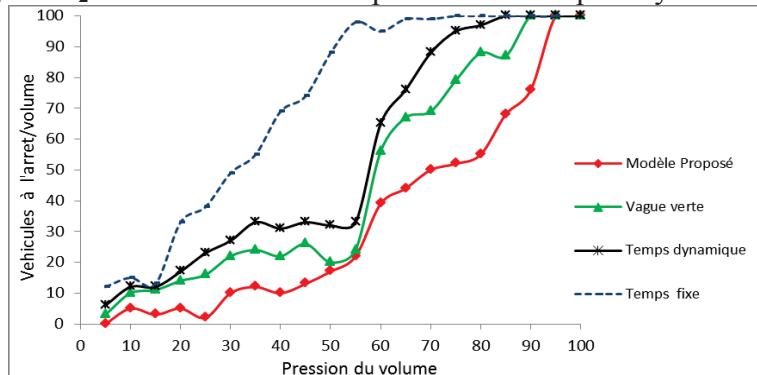
## Simulation 2



**Figure A2.2.1** Courbe de comparaison du débit volumique

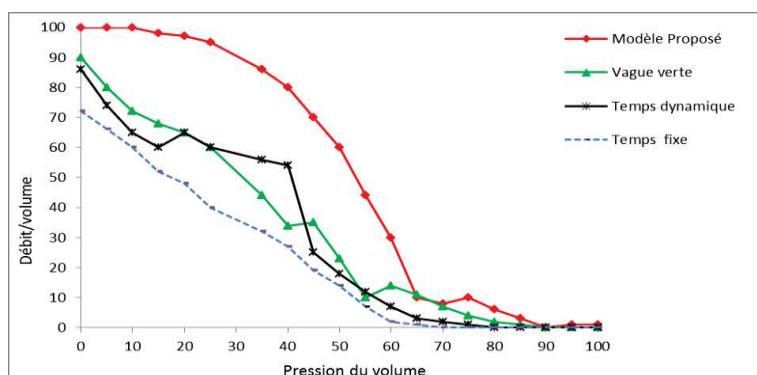


**Figure A2.2.2** Courbe de Comparaison du temps moyen d'attente

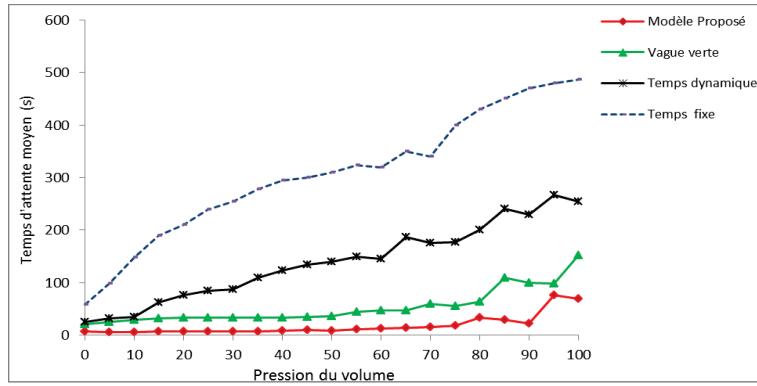


**Figure A2.2.3** Courbe de Comparaison du nombre moyen d'arrêts

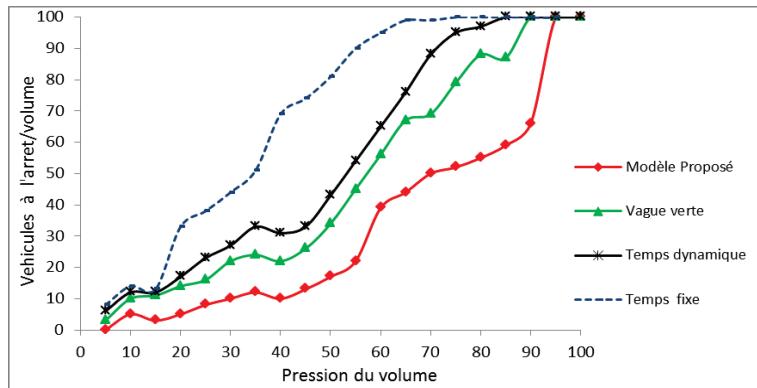
## Simulation 3



**Figure A2.3.1** Courbe de comparaison du débit volumique

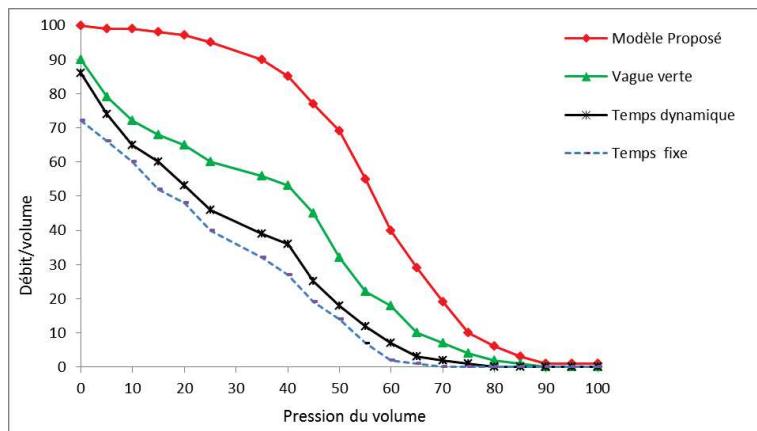


**Figure A2.3.2** Courbe de Comparaison du temps moyen d'attente

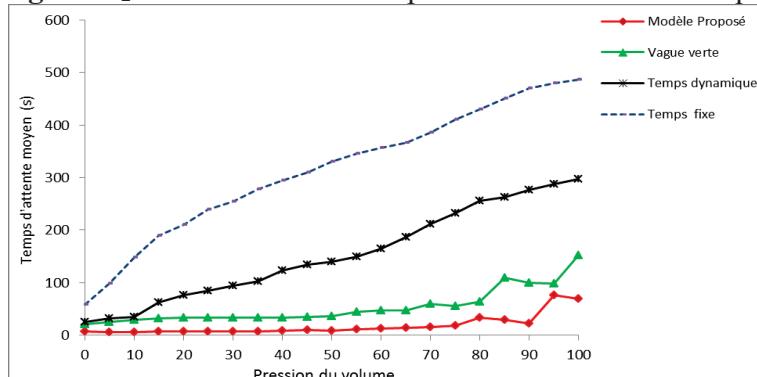


**Figure A2.3.3** Courbe de Comparaison du nombre moyen d'arrêts

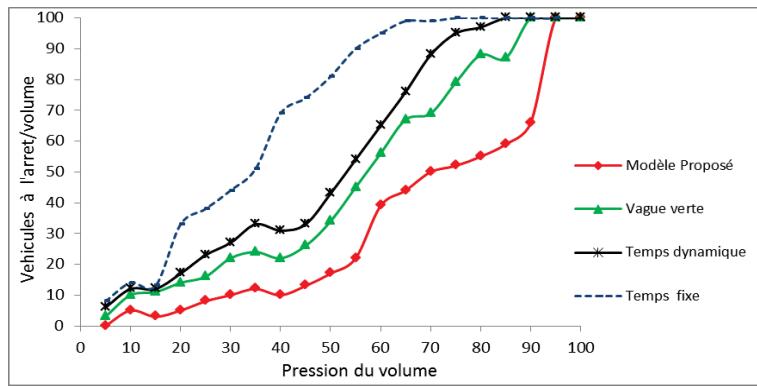
#### Simulation 4



**Figure A2.4.1** Courbe de comparaison du débit volumique

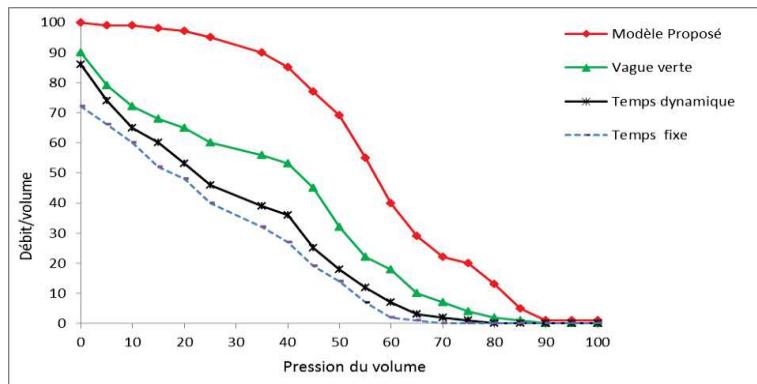


**Figure A2.4.2** Courbe de Comparaison du temps moyen d'attente

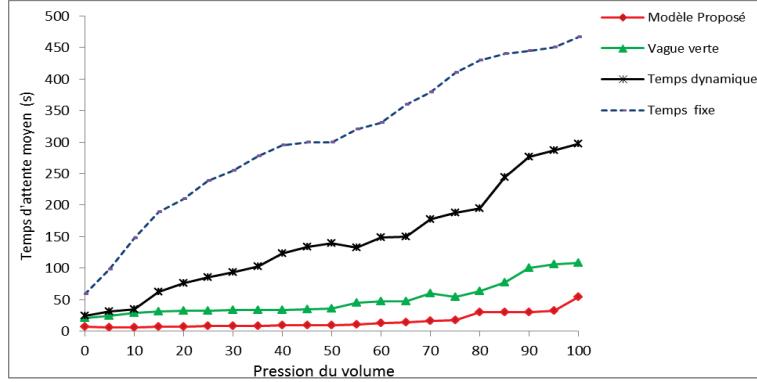


**Figure A2.4.3** Courbe de Comparaison du nombre moyen d'arrêts

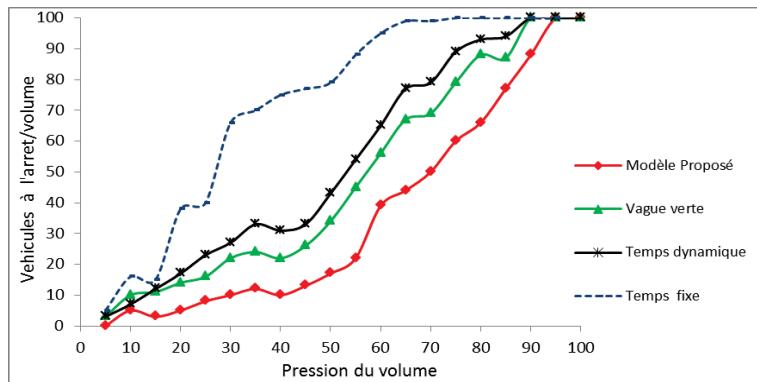
Simulation 5



**Figure A2.5.1** Courbe de comparaison du débit volumique

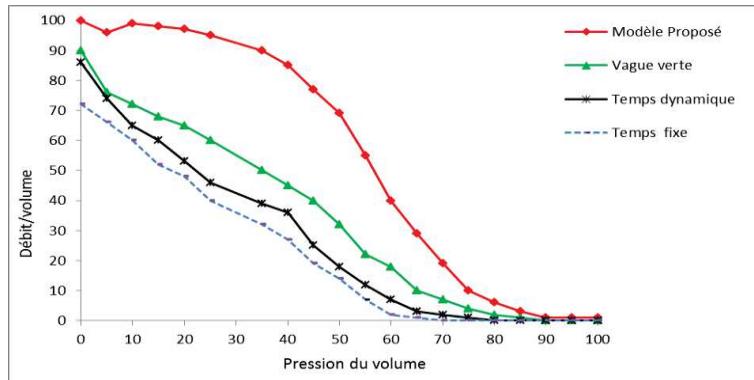


**Figure A2.5.2** Courbe de Comparaison du temps moyen d'attente

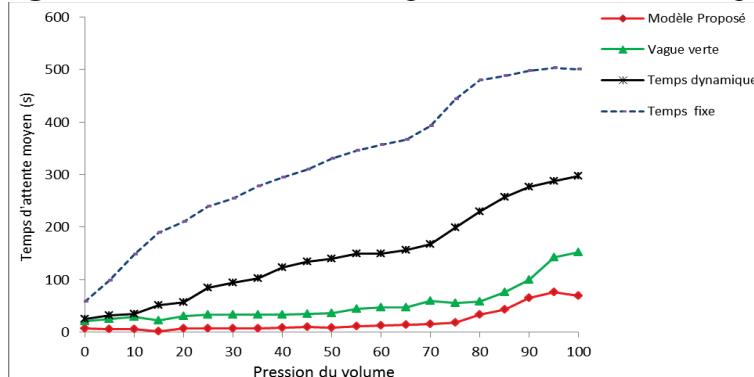


**Figure A2.5.3** Courbe de Comparaison du nombre moyen d'arrêts

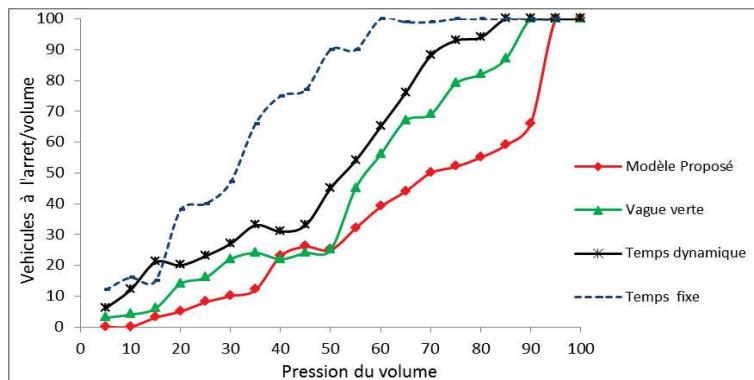
## Simulation 6



**Figure A2.6.1** Courbe de comparaison du débit volumique

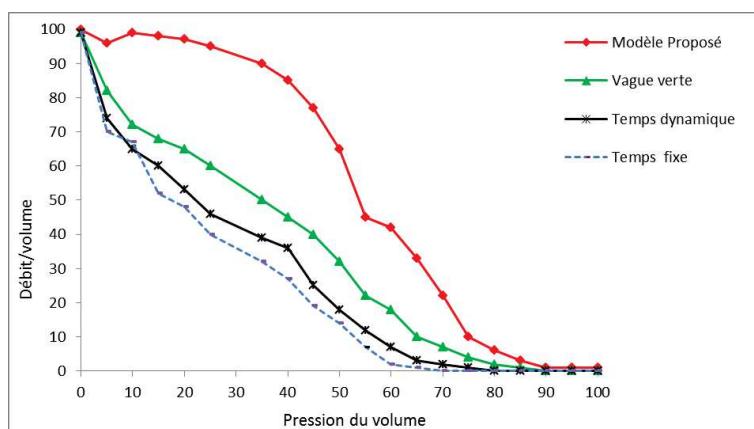


**Figure A2.6.2** Courbe de Comparaison du temps moyen d'attente

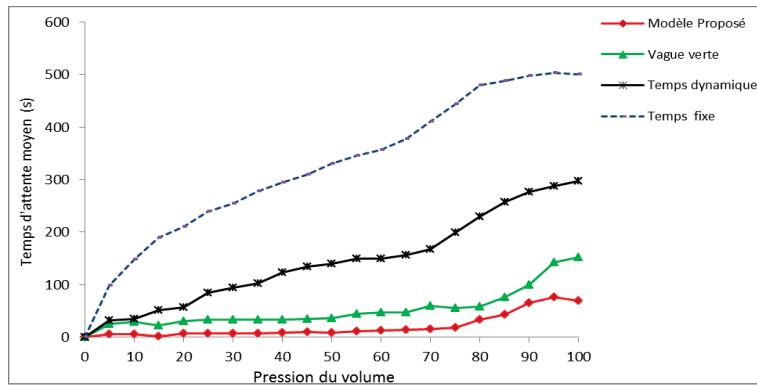


**Figure A2.6.3** Courbe de Comparaison du nombre moyen d'arrêts

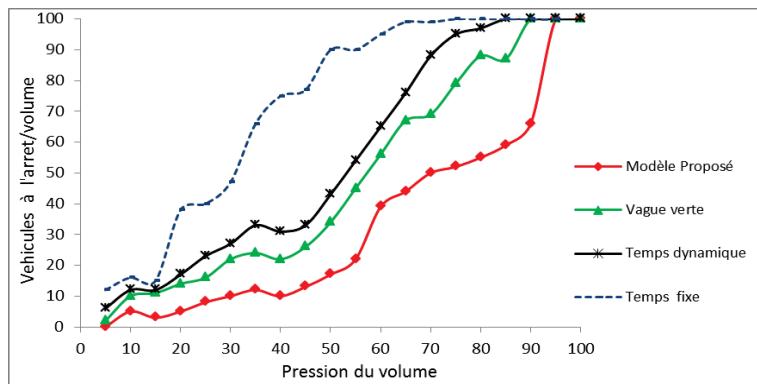
## Simulation 7



**Figure A2.7.1** Courbe de comparaison du débit volumique

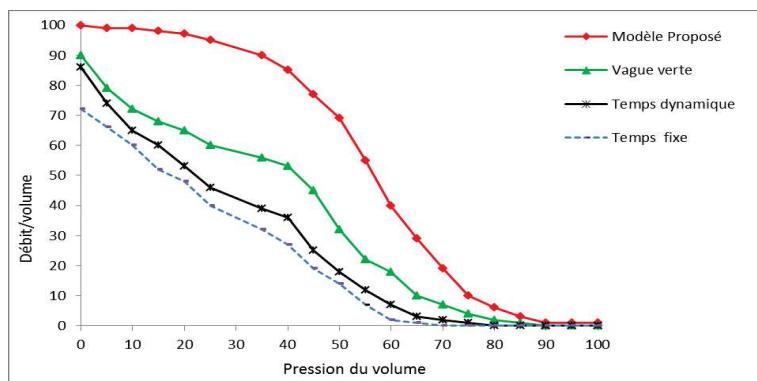


**Figure A2.7.2** Courbe de Comparaison du temps moyen d'attente

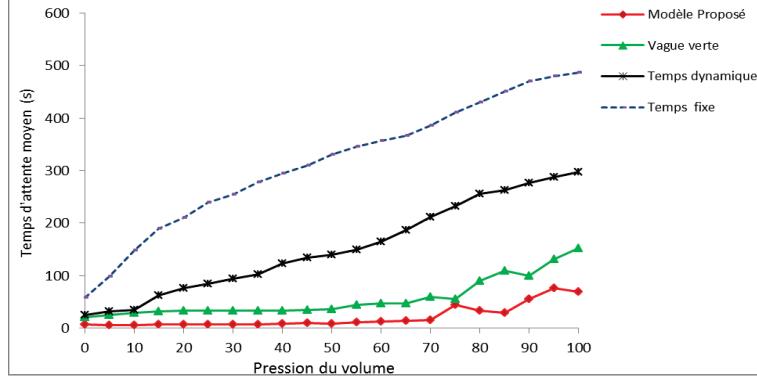


**Figure A2.7.3** Courbe de Comparaison du nombre moyen d'arrêts

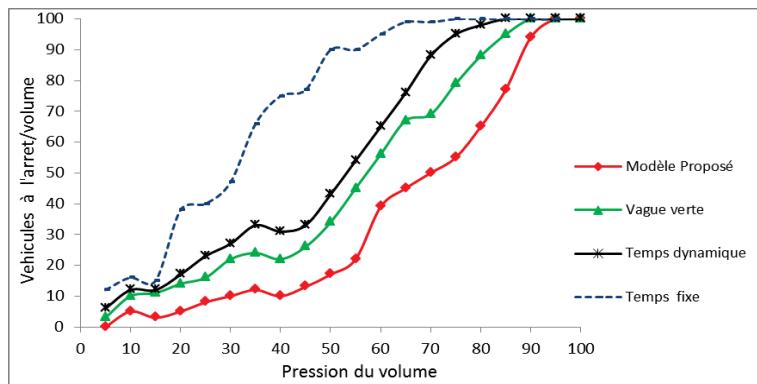
## Simulation 8



**Figure A2.8.1** Courbe de comparaison du débit volumique

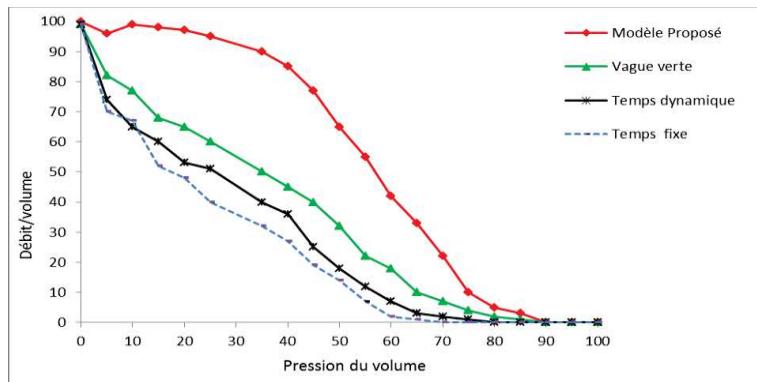


**Figure A2.8.2** Courbe de Comparaison du temps moyen d'attente

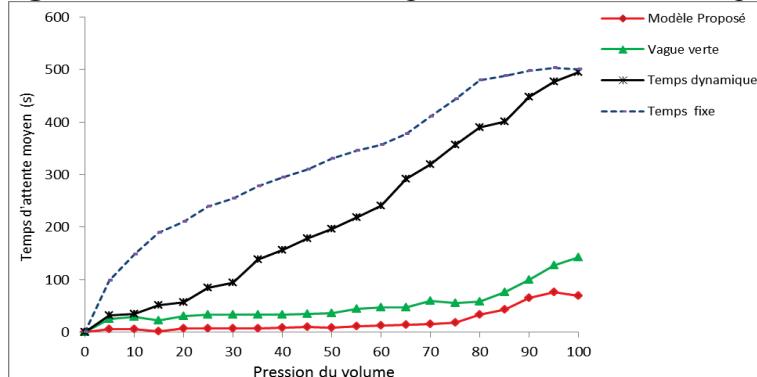


**Figure A2.8.3** Courbe de Comparaison du nombre moyen d'arrêts

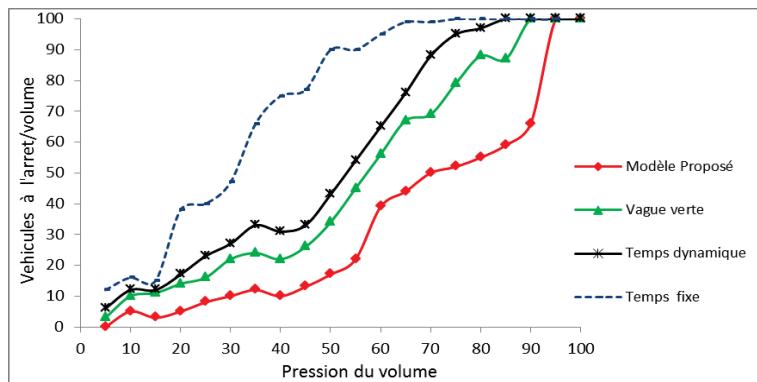
### **Simulation 9**



**Figure A2.9.1** Courbe de comparaison du débit volumique

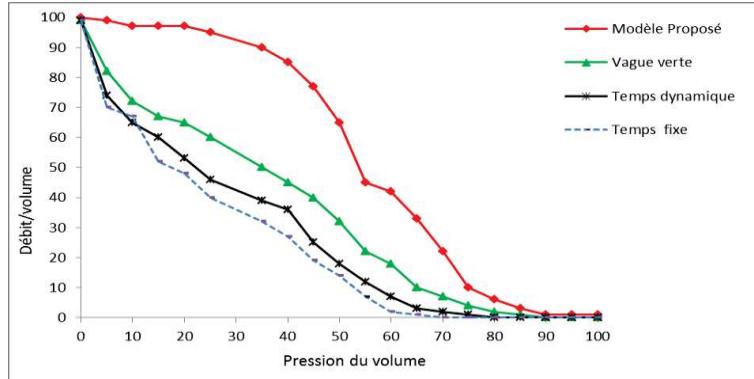


**Figure A2.9.2** Courbe de Comparaison du temps moyen d'attente

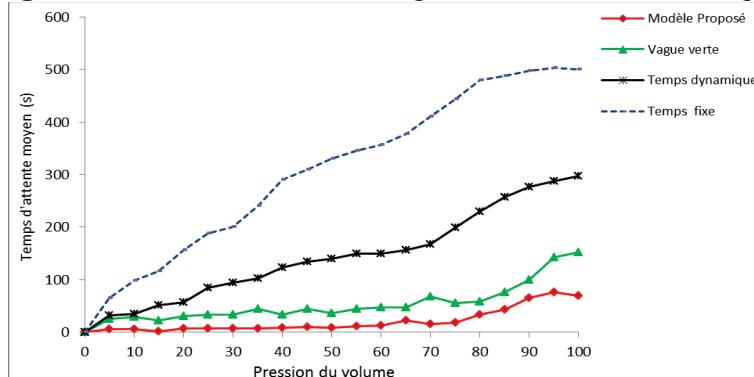


**Figure A2.9.3** Courbe de Comparaison du nombre moyen d'arrêts

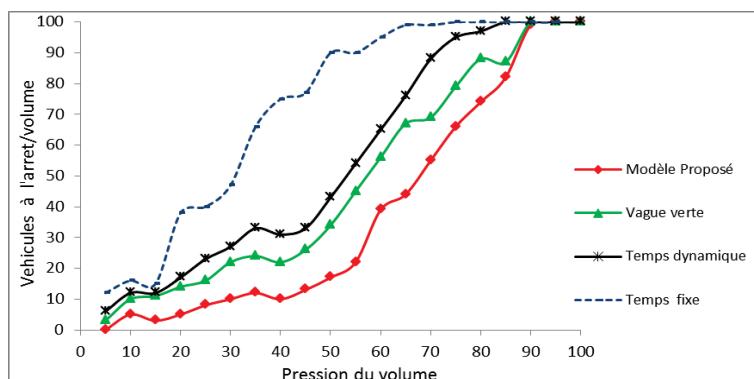
## Simulation 10



**Figure A2.10.1** Courbe de comparaison du débit volumique



**Figure A2.10.2** Courbe de Comparaison du temps moyen d'attente



**Figure A2.10.3** Courbe de Comparaison du nombre moyen d'arrêts