



Contribution à une modélisation globale de la collaboration dans les systèmes multi-agents : Application aux entités mobiles intelligentes

Laurent Lucien

► To cite this version:

Laurent Lucien. Contribution à une modélisation globale de la collaboration dans les systèmes multi-agents : Application aux entités mobiles intelligentes. Intelligence artificielle [cs.AI]. Université Bourgogne Franche-Comté, 2018. Français. NNT : 2018UBFCD039 . tel-01988405

HAL Id: tel-01988405

<https://theses.hal.science/tel-01988405>

Submitted on 21 Jan 2019

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

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

THÈSE DE DOCTORAT DE L'ÉTABLISSEMENT UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ
PRÉPARÉE À L'UNIVERSITÉ DE FRANCHE-COMTÉ

École doctorale n°37
Sciences Pour l'Ingénieur et Microtechniques

Doctorat d'Informatique

par

LAURENT LUCIEN

**Contribution à une modélisation globale de la collaboration dans les
systèmes multi-agents. Application aux entités mobiles intelligentes**

Thèse présentée et soutenue à Besançon, le 9 novembre 2018

Composition du Jury :

BOUQUET FABRICE	Professeur, Université de Franche-Comté	Président du jury
AMBLARD FRÉDÉRIC	Professeur, Université de Toulouse 1 Capitole	Rapporteur
MANDIAU RENÉ	Professeur, Université Polytechnique Hauts de France	Rapporteur
EL FAOUZI NOUR-EDDIN	Directeur de Recherche, IFSTTAR	Examineur
SOCOLIUC MIHAI	Responsable des Communautés Scientifiques, Groupe PSA	Examineur
PHILIPPE LAURENT	Professeur, Université de Franche-Comté	Directeur de thèse
LANG CHRISTOPHE	Maître de Conférences, Université de Franche-Comté	Encadrant de thèse
MARILLEAU NICOLAS	Ingénieur de Recherche HDR, Institut de Recherche pour le Développement	Encadrant de thèse

Titre : Contribution à une modélisation globale de la collaboration dans les systèmes multi-agents. Application aux entités mobiles intelligentes

Mots-clés : Systèmes multi-agents, Collaboration, Architecture hybride, GAMA

Résumé :

Nous vivons aujourd'hui dans un monde de plus en plus complexe et interconnecté où de nombreuses entités, de plus en plus intelligentes, génèrent une multitude d'interactions qui peuvent contribuer à enrichir leurs capacités.

Nous nous intéressons à la collaboration qui favorisera la réalisation de tâches complexes par ces machines d'aujourd'hui et de demain en stimulant ces interactions structurées et en intégrant des processus de décisions intelligents. Par ce biais, cela contribuera à améliorer leur fonctionnement et pourra participer à leur perfectionnement (meilleures connaissances de leur environnement, rapidité d'action et de décision, apport de nouvelles compétences).

L'objectif principal de la thèse est donc de contribuer à la compréhension de ce qu'est la collaboration, de sa définition à sa mise en œuvre, en

mettant en lumière ses concepts sous-jacents. Nous proposons une méthode d'analyse du besoin et des contraintes ainsi qu'un modèle d'architecture d'agent collaboratif (HACCA) permettant d'intégrer toutes les caractéristiques des processus collaboratifs que nous présentons. Nous présentons également une première implémentation dans la plateforme multi-agents GAMA.

Dans le cadre de cette étude, nous nous intéressons à deux cas d'application d'entités mobiles : les drones et les véhicules connectés.

Ainsi nous contribuons également plus largement aux besoins d'autonomie et de décision des drones, des véhicules connectés et autonomes du futur, dans un contexte temporel contraint où la qualité des interactions est primordiale pour optimiser le processus de réalisation des objectifs.

Title: Contribution à une modélisation globale de la collaboration dans les systèmes multi-agents. Application aux entités mobiles intelligentes

Keywords: Multiagent systems, Collaboration, Hybrid architecture, GAMA

Abstract:

We live today in an increasingly complex and interconnected world where many entities, increasingly intelligent, generate a multitude of interactions that can contribute to enrich their capabilities.

We are interested in collaboration that will enable complex tasks to be performed by these machines of today and tomorrow by stimulating these structured interactions and integrating intelligent decision-making processes. In this way, it will contribute to improve their functioning and will be able to participate in their improvement (better knowledge of their environment, speed of action and decision-making, provision of new skills).

The main objective of the thesis is therefore to contribute to the understanding of what collaboration is, from its definition to its implementation, by

highlighting its underlying concepts. We propose a method of analysis (needs and constraints) and a collaborative agent architecture model (HACCA) to integrate all the characteristics of the collaborative processes that we present. We are also showing a first implementation in the GAMA multi-agent platform.

As part of this study, we are interested in two cases of application of mobile entities: drones and connected vehicles.

Thus we also contribute more to the autonomy needs and decision-making process of drones, connected and autonomous vehicles of the future, in a constrained temporal context where the quality of interactions is essential to optimize the process of achieving objectives.

REMERCIEMENTS

C E fut évidemment un cheminement long et tortueux, six années avec ses hauts et ses bas, ses doutes et ses hésitations, ses frustrations et ses joies immenses mais ce fut également une expérience tellement enrichissante, tant au niveau personnel que professionnel, avec cette volonté de se dépasser pour se montrer à la hauteur de ce challenge. Néanmoins il est clair que la difficulté de concilier le travail au quotidien et les activités de recherche en parallèle a souvent mis à mal ma motivation et ma production scientifique.

Evoluant dans le monde industriel, cette expérience m'a aidé à mieux appréhender le monde universitaire. J'ai désormais une meilleure vision des passerelles potentielles entre ces univers mais j'y vois aussi toutes les difficultés de mise en œuvre car il y a plusieurs niveaux de compréhension, des temporalités différentes (temps court pour l'industrie, temps long pour la recherche universitaire) et deux finalités différentes (trouver une rentabilité immédiate vs. faire progresser la science).

Je tiens donc à remercier très chaleureusement toutes les personnes que j'ai pu rencontrer tout au long de ces années et qui m'ont aidé à passer ce moment de mon existence :

- Merci à mes encadrants, Laurent Philippe, Christophe Lang et Nicolas Marilleau pour leur accompagnement et leurs remarques toujours constructives ;
- Merci à tous mes collègues du laboratoire (notamment Alban, Louis-Claude et Julien) pour les discussions parfois enflammées et leurs conseils.

Je remercie tout particulièrement mon entreprise, le groupe PSA, pour l'opportunité qu'elle m'a offerte et le soutien financier :

- Merci à Mihai Socoliuc pour son aide précieuse et sa confiance sans faille ;
- Merci à Jean-Pierre Dumoulin, Bruno Muller, Gilles Feuntun, Bertrand Leinen et Christophe Prigent de m'avoir laissé le temps nécessaire à la concrétisation de ce projet.

Bien évidemment, j'exprime aussi ma très grande gratitude à l'égard de ceux qui m'ont fait l'honneur de lire mon travail et de participer au jury :

- Merci au président du jury, Fabrice Bouquet ;
- Merci à mes rapporteurs, Frédéric Amblard et René Mandiau ;
- Merci à mes examinateurs, Nour-Eddin El Faouzi et Mihai Socoliuc.

Merci également à tous mes amis et à ma famille qui m'ont soutenu et rassuré pendant mes phases de doute ainsi qu'à toutes celles et ceux, enfin, que j'aurais malheureusement oubliés ici tant il est difficile de citer tout le monde.

A ma mère qui doit être fier de son rejeton où qu'elle soit,

A mon fils Killian, pour lui prouver qu'il n'y a pas d'âge pour faire des études,

A ma femme Rut que j'aime...

SOMMAIRE

0.1	Introduction générale	1
0.2	Contexte	2
0.3	Objectifs de la thèse	2
0.4	Questions de recherche	3
0.5	Plan de la thèse	4
0.5.1	Partie 1 : Systèmes multi-agents et Collaboration	4
0.5.2	Partie 2 : HACCA (Hybrid Architecture for Collaborative Commu- nicating Agent), Analyse et Conception	4
0.5.3	Partie 3 : Instanciation, Implémentation et Expérimentations	5
I	Systèmes multi-agents et Collaboration	7
1	Systèmes multi-agents	11
1.1	Concepts et définitions	11
1.1.1	Définitions d'un système multi-agents	11
1.1.2	Définitions d'un agent	14
1.1.3	Principales propriétés	15
1.1.4	Organisation	16
1.2	Taxonomie	17
1.2.1	Types d'agent	18
1.2.1.1	Agent cognitif	18
1.2.1.2	Agent réactif	19
1.2.1.3	Agent hybride	20
1.2.2	Classification	21
1.3	Principales plateformes	22
1.4	Ce qu'il faut retenir	23
2	Collaboration générale	27
2.1	Avant propos	27
2.2	Collaborer ou coopérer ?	28

2.3	Autres approches	29
2.3.1	Dans le monde du travail	29
2.3.2	Dans le monde de l'éducation et de l'apprentissage	31
2.4	Cycle de la collaboration	33
2.5	Ce qu'il faut retenir	34
3	Collaboration dans les SMA	37
3.1	Définitions	38
3.2	Cycle de la collaboration	39
3.2.1	Intentions	39
3.2.2	Interactions	40
3.2.3	Communication	42
3.2.4	Coordination	45
3.2.5	Négociations	46
3.2.6	Organisations	49
3.3	Architectures	50
3.4	Ce qu'il faut retenir	51
4	Drones et Véhicules connectés	53
4.1	Drones	53
4.1.1	Qu'est-ce qu'un drone ?	53
4.1.2	Différents types de drones	54
4.1.3	Domaines d'utilisation et enjeux	55
4.1.4	Drones et collaboration	56
4.1.5	Limites actuelles	57
4.2	Véhicules connectés	57
4.2.1	Quelques définitions	57
4.2.2	Niveaux d'autonomie	58
4.2.3	Propositions des constructeurs	60
4.2.4	Enjeux	61
4.2.5	Quelques verrous	62
4.2.6	Limites actuelles	63
4.3	Ce qu'il faut retenir	64

II HACCA (Hybrid Architecture for Collaborative Communicating

Agent) : Analyse et Conception 67

5 Analyse et verrous 69

5.1	Analyse	69
5.1.1	Cadre d'application	69
5.1.2	Problématiques	70
5.1.3	Pistes de réflexion	72
5.2	Verrous	72
5.2.1	Méthode d'analyse	73
5.2.2	Architecture	73
5.2.2.1	Représentation du monde	73
5.2.2.2	Organisation des données	74
5.2.2.3	Gestion des comportements	75
5.2.2.4	Gestion des communications et des protocoles	76
5.3	Exemple Fil rouge	76
5.3.1	Présentation	76
5.3.2	Implémentation NetLogo	77
5.4	Ce qu'il faut retenir	80

6 Collaboration : définition, qualification 83

6.1	Caractérisation générale	84
6.1.1	Définition	84
6.1.2	Questionnement	84
6.1.3	Formalisation	85
6.2	Gradient de collaboration : Caractérisation fonctionnelle	87
6.2.1	Complète	87
6.2.2	Partielle	88
6.2.3	Altruiste	89
6.2.4	Forcée	90
6.2.5	Induite	90
6.2.6	Synthèse	91
6.3	Types de Collaboration : Caractérisation technique	91
6.3.1	Locale ou distante	92
6.3.2	Synchrone ou asynchrone	93
6.3.3	Formelle ou informelle	94
6.3.4	Synthèse	94

6.4	Description du processus	94
6.4.1	Déclencheur	97
6.4.2	Dialogue	97
6.4.3	Mise en œuvre	98
6.4.4	Capitalisation	99
6.5	Influence du gradient sur les algorithmes	100
6.6	Ce qu'il faut retenir	101
7	Méthode d'analyse	105
7.1	Démarche	106
7.2	Présentation du cas d'application	107
7.3	Première itération : Analyse du problème et caractérisation	107
7.3.1	Données manipulées	107
7.3.2	Comportements	108
7.3.3	Interactions	108
7.3.4	Collaborations	109
7.3.5	Constat	109
7.4	Deuxième itération : Amélioration des comportements	109
7.4.1	Données manipulées	110
7.4.2	Comportements	110
7.4.3	Interactions	111
7.4.4	Collaborations	111
7.4.5	Constat	112
7.5	Troisième itération : Optimisation	114
7.5.1	Données manipulées	114
7.5.2	Comportements	114
7.5.3	Interactions	114
7.5.4	Collaborations	115
7.5.5	Constat	115
7.6	Formalisation	115
7.7	Ce qu'il faut retenir	117
8	HACCA	119
8.1	Architecture générale	119
8.1.1	Constat et justification	120
8.1.2	Modèle conceptuel	120

8.1.3	Principes de fonctionnement	121
8.2	Description détaillée	122
8.2.1	Modules de perception et d'action	123
8.2.2	Module d'enregistrement des données	123
8.2.3	Trois couches de traitement	123
8.2.4	Règles de comportement	124
8.2.5	Connaissances de l'agent	126
8.2.6	Communications	128
8.3	Cinématique et exemple d'application	129
8.4	Ce qu'il faut retenir	132
III	Instanciation, Implémentation et Expérimentations	135
9	Instanciation du modèle urbain	137
9.1	Scenario	137
9.1.1	Discussion préalable	137
9.1.2	Description	138
9.2	Méthode d'analyse	139
9.2.1	Analyse du problème et caractérisation	139
9.2.2	Amélioration des comportements	140
9.2.3	Optimisation	143
9.3	Synthèse	144
9.3.1	Comportements	144
9.3.2	Connaissances	145
9.4	Ce qu'il faut retenir	146
10	Implémentation et expérimentations du modèle urbain	147
10.1	Modeleur GAMA	147
10.1.1	Justification du choix	148
10.1.2	Présentation générale	148
10.1.3	Concepts retenus	149
10.2	Implémentation	149
10.2.1	But de la modélisation	150
10.2.2	Entités, Variables d'état, Echelles	150
10.2.3	Revue des processus	150

10.2.4 Concepts d'implémentation	151
10.3 Expérimentations	153
10.4 Difficultés	153
10.5 Ce qu'il faut retenir	154
11 Conclusion et Perspectives	157
11.1 Conclusion générale	157
11.2 Perspectives	158
11.2.1 Dans le domaine des systèmes multi-agents	158
11.2.2 Dans le monde réel	159

INTRODUCTION

0.1/ INTRODUCTION GÉNÉRALE

Nous vivons aujourd'hui dans un monde de plus en plus complexe et interconnecté, en présence de nombreux objets ou entités électroniques, de plus en plus intelligentes. C'est que nous appelons souvent les objets connectés du quotidien (montre, réfrigérateur, etc). Que ce soit en domotique ou en robotique, nous sommes confrontés à cette avalanche d'objets qui font partie d'un tout que nous nommons maintenant l'Internet des Objets (Internet of Things ou IoT).

Ce nouveau monde des objets intelligents commence à générer une multitude d'interactions. En effet, tous ces objets évoluent dans un même environnement (ou espace physique) et adressant des fonctionnalités variées, ils peuvent être amenés à se rencontrer sur des domaines fonctionnels connexes avec la nécessité de partager des informations. Ces interactions pourront contribuer à enrichir les possibilités de ces objets soit en leur apportant des informations complémentaires pour mieux adapter leur comportement, soit en créant via un réseau d'interactions, un fonctionnement plus intégré de différents appareils pour former un environnement technologique cohérent pouvant proposer des offres de services adaptés. Par exemple, un réfrigérateur connecté détectant le manque prochain d'un aliment cherchera à interagir avec un site de supermarché en ligne. Deuxième exemple, dans la maison, un détecteur de présence (pouvant être associé à une caméra) pourra piloter l'ouverture ou la fermeture des volets roulants ainsi que le système de chauffage ou de climatisation en fonction de l'heure de la journée, de l'occupant et de ses habitudes pour adapter l'environnement de vie.

Il y a donc une nécessité impérieuse de communiquer. Pour que les informations soient échangées, il faut le faire avec des dispositifs adaptés incluant des moyens électroniques, informatiques et des protocoles pour se comprendre. Des stratégies sont à mettre en œuvre pour pouvoir organiser ces échanges et pour favoriser une communication efficace entre entités. Cela passe par l'organisation des données, la définition des protocoles de création et de suivi des échanges.

Afin de pouvoir gérer efficacement ces échanges structurés dans cette multitude d'interactions, il faut optimiser les performances des entités individuellement et collectivement. En effet, l'objectif est d'améliorer l'efficacité propre de l'entité, sa capacité à effectuer son travail. Se faisant, des interactions mieux organisées constituent l'une des clés d'un système plus stable, plus efficace.

Toutes ces interactions pourront, le cas échéant et en fonction du type d'objet, permettre également l'extension des capacités et des possibilités via l'échange d'informations contextualisées. En effet, au fur et à mesure des communications, il sera possible d'améliorer par apprentissage le système de connaissances propres de l'objet pour qu'il puisse proposer un service mieux ciblé et plus efficace, prenant en compte l'environnement complet.

La collaboration ou plus prosaïquement les processus collaboratifs pourront, en structurant les échanges autour d'un objectif commun accepté et partagé, être utilisés pour favoriser cette amélioration générale de l'efficacité d'exécution et de traitement des entités évoluant dans ce système complexe que constitue le monde connecté qui nous entoure.

0.2/ CONTEXTE

Comme évoqué dans l'introduction générale, les objets connectés deviennent de plus en plus intelligents et présents dans notre environnement quotidien. Ce sont des entités (mobiles ou non) qui tendent à devenir totalement autonomes et devant faire face à des situations complexes d'une grande diversité pour évoluer correctement et accomplir leur mission. Il est donc intéressant d'étudier pratiquement comment des processus collaboratifs pourront contribuer à améliorer leur fonctionnement.

Dans le cadre de cette étude, nous allons nous intéresser à deux cas d'application d'entités mobiles : les drones et les véhicules connectés.

Drones

De jour en jour, nous pouvons observer l'avènement de ces appareils volants, roulants et flottants, pouvant remplir des missions diverses d'observation ou d'intervention sans mettre en péril l'être humain ou tout simplement, pour l'aider à prendre des décisions. D'abord pilotés, ces drones sont de plus en plus automatisés et doivent donc, notamment dans le cadre de missions éloignées de l'opérateur humain, prendre des décisions de plus en plus complexes. Nous sommes à la croisée de plusieurs domaines scientifiques : sciences sociales, sciences cognitives, sciences de l'information, robotique, etc.

Véhicules connectés

Autre cas d'application, les véhicules connectés (communicants et à terme totalement autonomes), véritable enjeu économique et technique du moment, représentent, à l'instar des drones, un défi technologique et informatique majeur. La course actuelle à l'automatisation des déplacements sans intervention humaine repose la question de la prise de décisions en milieu ouvert où les interactions avec l'environnement rendent parfois difficile la caractérisation de celui-ci et donc le jugement sur les actions nécessaires à accomplir.

0.3/ OBJECTIFS DE LA THÈSE

Nous pensons que les processus collaboratifs aideront à favoriser la réalisation de tâches complexes par ces machines d'aujourd'hui et de demain en stimulant des interactions structurées et en intégrant des processus de décisions intelligents. Par ce biais, cela contribuera à améliorer leur fonctionnement et pourra participer à leur perfectionnement (meilleures connaissances de leur environnement, rapidité d'action et de décision, apport de nouvelles compétences). Nous utiliserons les systèmes multi-agents car cela nous semblent un outil idéal pour modéliser ce type d'entités au sein d'un environnement. De plus, nous pourrons faire le parallèle entre des systèmes multi-agents simulés et réels.

Cependant cela passe tout d'abord par un **état des lieux contextualisé** de l'existant, tant au niveau agent qu'au niveau de la collaboration dans sa définition générale. Cette ana-

lyse contribue à justifier notre point de vue sur la collaboration et de poser notre **définition**. Il en découle une liste de **verrous scientifiques** dans le cadre d'application que nous avons choisi. Puis cela nous amène à proposer notre **modèle d'architecture d'agent** intégrant les contraintes d'exécution que nous expérimentons afin d'en tester la validité.

L'objectif principal de la thèse est donc de contribuer à la compréhension de ce qu'est la collaboration dans le contexte des entités mobiles intelligentes et autonomes, de sa définition à sa mise en œuvre, en mettant en lumière ses concepts sous-jacents. Ainsi nous contribuerons plus largement aux besoins d'autonomie et de décision des drones, des véhicules connectés et autonomes du futur, dans un contexte temporel contraint où la qualité des interactions est primordiale pour optimiser le processus de réalisation des objectifs. Nous proposons une approche méthodologique d'analyse du besoin et des contraintes et un **modèle d'architecture d'agent collaboratif** permettant d'intégrer toutes les caractéristiques des processus collaboratifs que nous présentons de manière structurée en nous appuyant sur nos propres définitions. Nous utiliserons cette démarche et ce modèle pour instancier des agents mobiles intelligents (drones et véhicules connectés).

0.4/ QUESTIONS DE RECHERCHE

Question 1

Qu'est-ce que la collaboration ? Qu'est-ce qui la caractérise ?

Il est nécessaire de se poser la question de la définition de la collaboration, de ses propriétés, ses caractéristiques et ses enjeux pour voir son rôle dans l'amélioration des processus de décision et d'exécution des missions des objets connectés et communicants décrits plus hauts, c'est-à-dire des entités mobiles intelligentes et autonomes.

Question 2

Comment mettre en œuvre la collaboration ?

Les processus de collaboration sont a priori complexes. Il faut en étudier les mécanismes afin de comprendre comment les mettre en œuvre concrètement. Cela passe aussi par l'analyse du problème à modéliser ou du phénomène à mettre en lumière.

Question 3

Quelle architecture pour porter la collaboration ?

Une fois que les processus de collaboration sont bien définis et compris, il faut pouvoir les mettre en application au sein d'une architecture qui permettra de les manipuler de manière explicite. Cette architecture sera abordée sur un plan conceptuel.

Question 4

Comment implémenter un agent collaboratif sur une plateforme de modélisation de systèmes multi-agents ?

Une fois la collaboration comprise sur un plan conceptuel et fonctionnel, comment implémenter techniquement ces concepts dans une architecture d'agent adapté au sein d'un modeleur en sachant que nous pouvons avoir à faire à des agents également réels non-simulés comme dans un environnement multi-robots de type drones.

0.5/ PLAN DE LA THÈSE

0.5.1/ PARTIE 1 : SYSTÈMES MULTI-AGENTS ET COLLABORATION

Le **chapitre 1** propose un aperçu de ce qu'est un système multi-agents en général et un agent en particulier. Nous présentons les différents types d'agent ainsi qu'une classification afin de mettre en avant le meilleur type d'agent susceptible d'être utilisé pour servir de base à la modélisation des échanges collaboratifs dans le contexte de l'étude (drones et véhicules connectés).

Dans le **chapitre 2**, nous passons en revue les différentes définitions des mots collaboration et coopération pour tenter d'en montrer les différentes significations et les différences. Puis, en prenant des exemples dans les domaines de l'éducation et de l'apprentissage mais aussi du travail, nous listons les caractéristiques essentielles de la collaboration.

Le **chapitre 3** aborde ce qu'est la collaboration dans le domaine des systèmes multi-agents avec ces principales caractéristiques (avec les notions d'interaction et d'intention), les architectures connues et les problèmes que cela pose.

Dans le **chapitre 4**, nous faisons un focus rapide sur les drones et les véhicules connectés avec quelques définitions, les contextes d'utilisation et les perspectives pour ces deux types de machine.

0.5.2/ PARTIE 2 : HACCA (HYBRID ARCHITECTURE FOR COLLABORATIVE COMMUNICATING AGENT), ANALYSE ET CONCEPTION

Le **chapitre 5** présente les différents verrous observés dans le contexte de la discussion qui sera ouverte sur la possibilité d'implémentation de processus collaboratifs.

Contribution 1

Freins et enjeux de la mise en application du concept de collaboration dans les systèmes multi-agents

Fort des chapitres précédents et en s'appuyant sur la discussion proposée au chapitre précédent, nous posons dans le **chapitre 6** notre propre définition de la collaboration et nous proposons une qualification des différents types de processus autour d'algorithmes les explicitant en utilisant un exemple fil rouge issu du monde des drones autonomes et connectés.

Contribution 2

Définition d'une collaboration, catégorisation et description des différents types de collaboration

Le **chapitre 7** propose un cheminement intellectuel pour arriver à décrire et mettre en place des processus collaboratifs. Nous utilisons notre exemple fil rouge d'une patrouille de drones de détection d'incendie.

Contribution 3

Méthode d'analyse de problème pour mettre en évidence et décrire des comportements collaboratifs pour des entités mobiles intelligentes

Le **chapitre 8** présente HACCA, notre proposition d'architecture d'agent hybride collaboratif. Les différents modules et fonctionnalités de cette architecture sont décrits ainsi que la méthode d'organisation des données et les processus de communication.

Contribution 4

Présentation détaillée de l'architecture répondant aux verrous exposés dans le chapitre 5, dans un contexte d'agent autonome

0.5.3/ PARTIE 3 : INSTANCIATION, IMPLÉMENTATION ET EXPÉRIMENTATIONS

Dans le **chapitre 9**, nous proposons une instanciation du modèle HACCA dans un environnement urbain constitué d'entités mobiles autonomes (véhicules connectés) en réutilisant également la méthode d'analyse décrite précédemment.

Dans le **chapitre 10**, nous proposons une première implémentation de l'architecture HACCA en utilisant le modeleur GAMA.

Contribution 5

Proposition d'une implémentation du modèle d'architecture d'agent hybride collaboratif en langage GAMA

Puis nous présentons les expérimentations réalisées pour vérifier le concept de collaboration et sa pertinence dans le domaine des véhicules connectés évoluant dans un milieu urbain dense. Nous parlons des résultats obtenus en présentant les graphes et courbes ainsi que l'analyse de ceux-ci.

La figure 1 résume l'organisation des différents chapitres de ce rapport.

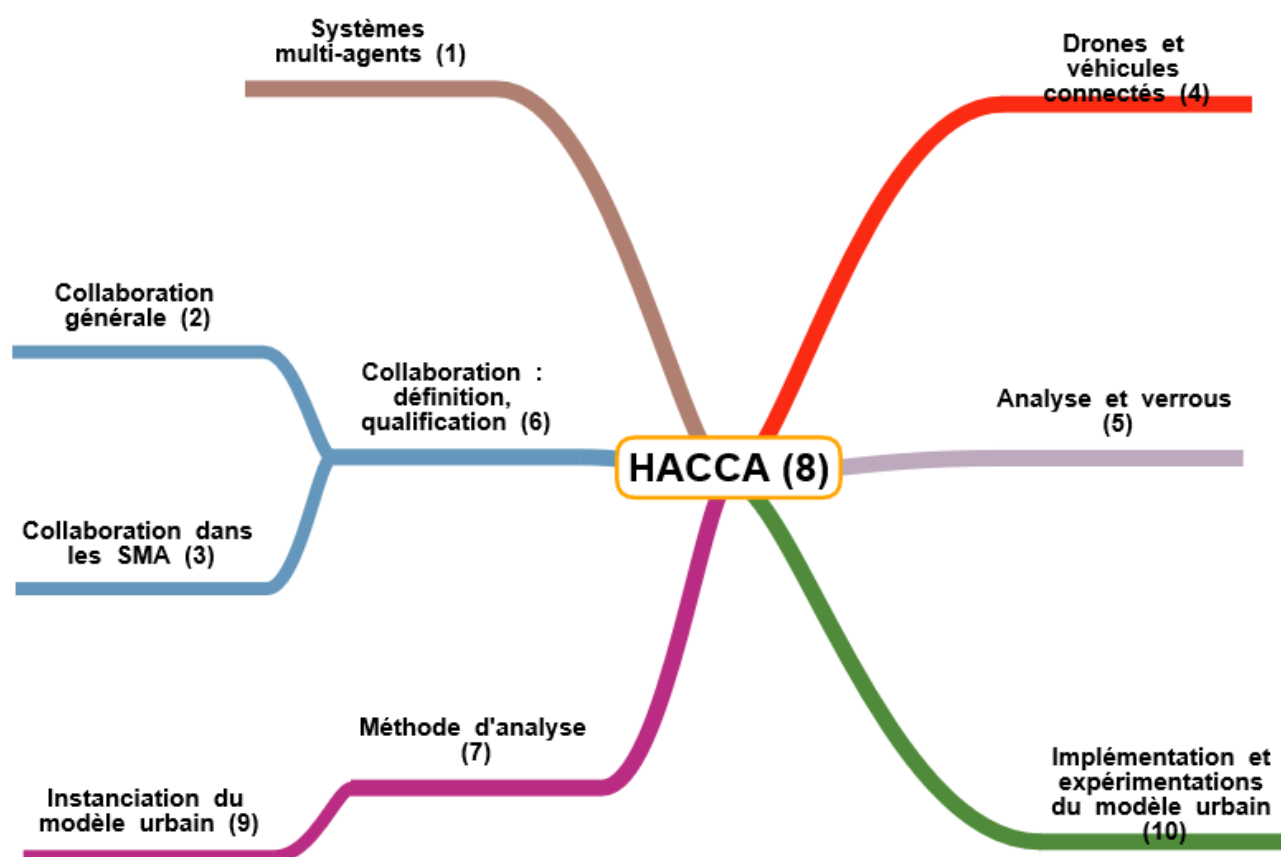


FIGURE 1 – Organisation des chapitres (entre parenthèses, le numéro du chapitre)



SYSTÈMES MULTI-AGENTS ET COLLABORATION

LE monde dans lequel nous vivons s'avère, de jour en jour et au fur et à mesure de nos découvertes, toujours plus complexe. Dans de nombreux domaines (géographie, biologie, sociologie, économie, etc), nous avons besoin de modéliser et de simuler des situations issues de notre environnement direct (par exemple, terrestre) ou indirect (cosmos), mettant en interaction des entités vivantes (par exemple, une colonie de fourmis, des rats des champs, les habitants d'une ville, etc) ou non (des flocons de neige, des gouttelettes d'eau ou des cours de bourse).

Dans son ouvrage d'introduction à la complexité, Edgar Morin ([Morin et al., 1990]) décrit celle-ci en ses termes :

Définition 1 : Qu'est-ce que la complexité ? par EDGAR MORIN

Au premier abord, la complexité est un tissu (complexus : ce qui est tissé ensemble) de constituants hétérogènes inséparablement associés : elle pose le paradoxe de l'un et du multiple. Au second abord, la complexité est effectivement le tissu d'évènements, actions, interactions, rétroactions, déterminations, aléas, qui constituent notre monde phénoménal. Mais alors la complexité se présente avec les traits inquiétants du fouillis, de l'inextricable, du désordre, de l'ambiguïté, de l'incertitude... D'où la nécessité, pour la connaissance, de mettre de l'ordre dans les phénomènes en refoulant le désordre, d'écarter l'incertain, c'est-à-dire de sélectionner les éléments d'ordre et de certitude, de désambiguïser, clarifier, distinguer, hiérarchiser. . .

Edgar Morin explique clairement que la complexité vient finalement du paradoxe entre la connaissance intime d'une entité identifiée et l'impossibilité de comprendre les phénomènes induits par un ensemble de ces mêmes entités évoluant dans un environnement commun. En effet, toutes les interactions (événements, actions, rétroactions, etc) entre ces entités provoquent une évolution non-prédictive du système. En d'autres termes : ce n'est pas parce qu'on connaît parfaitement une entité que l'on comprend le système composé de ces entités en interaction.

La complexité semble être un fouillis inextricable. Nous cherchons donc à le comprendre en modélisant et en simulant des environnements reprenant les caractéristiques de ces entités avec le but principal d'augmenter notre capacité de compréhension générale d'un phénomène mais également d'accroître notre capacité d'anticipation en mettant en lumière de possibles comportements émergents.

Une distinction claire doit être faite entre la modélisation qui formalise un système et la simulation qui permet de tester et d'animer « grandeur nature » la modélisation. Le modèle est une représentation de la réalité tandis qu'une simulation est une animation du modèle.

Dans le cadre des systèmes dits complexes, la modélisation de l'ensemble du système n'est pas aisée voire impossible du fait des interactions nombreuses entre entités et seule la simulation permet de visualiser le ou les phénomènes étudiés. Le paradigme agent peut aider à la construction d'une modélisation adaptée à ces contraintes.

Afin de comprendre cette complexité, son étendue et la dynamique générale du système, nous allons nous attarder tout d'abord sur le comportement d'une entité de base (appelée **Agent**), bien connue et bien modélisée. Pour étudier le système complet, il faut pouvoir considérer un ensemble d'entités et les faire interagir dans un même environnement, lui-même connu (un **Système Multi-Agents**). La simulation résultante permet d'appréhender

la compréhension d'un phénomène dans son ensemble et déterminer le cas échéant des pistes de réflexion en vue d'une modélisation plus efficace du système. Nous allons donc naturellement utiliser le paradigme agents pour cette étude.

Dans cette première partie, nous décrivons ce qu'est un système multi-agents et quels sont les différents types d'agent que nous pouvons rencontrer en insistant sur leurs caractéristiques principales ([chapitre 1](#)). Nous nous interrogeons ensuite sur le concept de collaboration en partant de sa définition ([chapitre 2](#)) puis dans le domaine du travail, de l'éducation et de l'apprentissage. Enfin nous étudions comment la collaboration est considérée dans le monde des systèmes multi-agents ([chapitre 3](#)). Enfin nous présentons le cadre d'application que nous avons choisi et abordé sommairement dans l'introduction : les drones et les véhicules connectés ([chapitre 4](#)). Ces deux exemples vont guider notre approche et nous permettre de mettre en évidence le chemin que nous allons emprunter.

SYSTÈMES MULTI-AGENTS

Sommaire

1.1 Concepts et définitions	11
1.1.1 Définitions d'un système multi-agents	11
1.1.2 Définitions d'un agent	14
1.1.3 Principales propriétés	15
1.1.4 Organisation	16
1.2 Taxonomie	17
1.2.1 Types d'agent	18
1.2.2 Classification	21
1.3 Principales plateformes	22
1.4 Ce qu'il faut retenir	23

L'objectif de ce chapitre est de présenter, dans le cadre de la modélisation et simulation de systèmes complexes, ce qu'est un système multi-agents en général et un agent en particulier.

Après quelques définitions, les différents types d'agent et leurs propriétés sont listés. Nous essayons alors de n'en retenir que l'essentiel pour éclaircir notre propos. Une classification est ensuite proposée afin de montrer le type d'agent pouvant servir à la mise en œuvre d'un processus collaboratif. Afin d'être complet, quelques structures d'organisation des systèmes d'agents sont mis en lumière. Enfin nous balayons les différents types de plateformes multi-agents existants aujourd'hui.

1.1/ CONCEPTS ET DÉFINITIONS

1.1.1/ DÉFINITIONS D'UN SYSTÈME MULTI-AGENTS

Dans son ouvrage de référence ([Ferber, 1995]), Jacques Ferber pose la définition suivante :

Définition 2 : Système multi-agents vu par JACQUES FERBER

On appelle système multi-agents (ou SMA), un système composé des éléments suivants :

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

Il s'agit ici d'une définition générale. En effet, selon les cas rencontrés, certains ensembles décrits ci-dessus sont nuls. Cette définition pourra donc être adaptée selon le problème et le type d'agent à modéliser, par exemple des agents réactifs ou cognitifs.

Faisons un focus sur la notion d'environnement car c'est un élément important du système multi-agents. Il est généralement défini comme étant tout ce qui entoure les agents ([Weyns et al., 2004]) mais il peut également avoir sa propre dynamique. Dans tous les cas de figure, cet environnement dans lequel les agents évoluent va les influencer plus ou moins directement.

Il est possible de caractériser les différents types d'environnement ([Russell et al., 1995]) :

- accessible (en intégralité par les agents) ou inaccessible
- déterministe (tout changement est déterminé par l'état de l'environnement et les actions des agents) ou indéterministe (deux résultats différents pour une même action)
- statique (pas d'évolution, même sous l'action des agents) ou dynamique (changements d'états possibles, règles d'évolution, avec ou sans action des agents)
- discret ou continu en fonction du processus de perception des agents

Dans sa définition, Jacques Ferber indique que l'environnement possède généralement une métrique. Il s'agit de la représentation d'un modèle physique. Il faut donc inclure toutes les règles et propriétés permettant d'appliquer des principes physiques ([Gechter et al., 2006]). Dans le cadre des simulations d'insectes par exemple ([Müller, 2003], [Parunak, 1997] ou [Drogoul, 1993]), l'environnement contient des zones spécifiques comme les nids, les zones de nourriture, etc en plus des données physiques du terrain modélisé. Il faut alors distinguer deux types d'aspects pour les environnements ([Weyns et al., 2004]) : les aspects structurels (métrique, propriétés fondamentales, etc) et les aspects dynamiques (perception, action ou dynamique intrinsèque).

Toutefois, on peut également faire la distinction entre l'environnement physique et l'environnement de communication ([Odell et al., 2002] ou [Weyns, 2007]). L'environnement physique est un ensemble de lois, de règles et de contraintes qui permet l'existence physique des objets et des agents. L'environnement de communication supporte les principes et processus d'échange d'information (idées, connaissances, etc) et permet la création de modèle de communication autour de la notion de groupes et de rôles ainsi que les différents protocoles d'interaction associés à cette organisation. On peut ainsi combiner les deux types d'environnement ([Galland et al., 2014] ou [Galland et al., 2016]) afin d'obtenir une vision à plusieurs dimensions des agents et de leurs interactions (existence physique dans l'environnement et couche sociale).

D'autres définitions d'un système multi-agents ont été également proposées par Edmund Durfee ([Durfee, 2012]) ou Nicholas Jennings ([Jennings et al., 1998b]).

Définition 3 : Système multi-agents vu par EDMUND H. DURFEE

Un système multi-agents est un réseau fortement couplé d'entités qui travaillent ensemble pour résoudre des problèmes qui dépassent les capacités individuelles ou les connaissances de chaque entité.

Ici Edmund Durfee met l'accent sur la résolution collective de problèmes dans un environnement où les agents en interaction disposent de capacités limitées.

Définition 4 : Système multi-agents vu par NICHOLAS R. JENNINGS

Un système multi-agents est composé de plusieurs composantes autonomes :

- Chaque agent possède des capacités incomplètes pour résoudre le problème.
- Il n'y a pas de système de contrôle global.
- Les données sont décentralisées.
- Le calcul est asynchrone.

Nicholas Jennings reprend la notion de résolution de problèmes mais insiste sur la notion d'autonomie des agents.

Il convient donc de mettre l'accent sur l'autonomie de l'agent avec ses propriétés et connaissances intrinsèques mais aussi sur le processus d'interaction dans l'environnement qui vient jouer un rôle fondamental dans la dynamique de l'agent. Cette mise en contact avec d'autres agents ou stimuli crée ainsi une dynamique de réaction et d'évolution du système dans son ensemble et permet des résolutions de problème au niveau individuel mais aussi collectif. On parle alors d'émergence ([Drogoul, 1993], [Ferber, 1994]). Un des exemples les plus courant est la modélisation d'une colonie de fourmis. Toutes les fourmis en interaction vont progressivement et collectivement trouver le chemin le plus court vers les sources de nourriture (grâce à la dépose des phéromones dans l'environnement). Toutes ces interactions et mises en relation créent une sorte de réseau d'acointances où les réactions et les décisions s'entremêlent au sein d'un système global.

Les domaines d'application sont très nombreux, en voici quelques exemples : en géographie (pour l'occupation des terres agricoles, le développement urbain, etc), en biologie (compréhension des évolutions de colonies d'insectes ou d'animaux), en logistique (gestion des flux dans une usine), etc. Dans ce rapport, nous nous intéressons à des agents en interaction intelligente où leur mise en œuvre permet d'améliorer le processus

de décision et donc leur autonomie.

1.1.2/ DÉFINITIONS D'UN AGENT

Il existe de nombreuses définitions d'un agent. Nous allons en citer plusieurs afin de lister les propriétés qui nous semblent essentielles.

Tout d'abord, Jacques Ferber ([Ferber, 1995]) propose une définition détaillée qui accentue les propriétés de l'agent liées à l'action, la perception, la communication, la vision locale, les tendances et objectifs, les ressources et les compétences.

Définition 5 : Agent vu par JACQUES FERBER

On appelle agent une entité physique ou virtuelle :

1. qui est capable d'agir dans un environnement,
2. qui peut communiquer directement avec d'autres agents,
3. qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire 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,
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.

Jennings et Wooldridge ([Wooldridge et al., 1995], [Jennings et al., 1998b], [Jennings, 2000], [Wooldridge, 2009]) proposent une définition commune dans laquelle nous retrouvons les notions de situation, autonomie et flexibilité.

Définition 6 : Agent vu par MICHAEL WOOLDRIDGE et par NICOLAS JENNINGS

Un agent est un système informatique encapsulé qui est situé dans un environnement et qui est capable d'effectuer des actions autonomes et flexibles afin d'atteindre les objectifs pour lesquels il a été conçu.

Roberto Flores-Mendez ([Flores-Mendez, 1999]) met l'accent sur le fait qu'un agent est une entité conceptuelle qui prend des initiatives. Elle dispose donc d'un système de gestion des connaissances et de représentation de son environnement.

Définition 7 : Agent vu par ROBERTO FLORES-MENDEZ

Un agent est une entité interactive qui existe en tant que partie d'un environnement partagé par d'autres agents. C'est une entité conceptuelle qui perçoit et agit avec initiative ou en réaction dans un environnement où d'autres agents existent et interagissent les uns avec les autres sur la base de connaissances partagées de communication et de représentation.

Il convient toutefois de noter que toutes ses propriétés ne se retrouveront pas forcément ensemble dans tous les types d'agent (cf. section 1.2.1 sur les types d'agent) mais cette définition permet de donner le cadre général de l'agent.

1.1.3/ PRINCIPALES PROPRIÉTÉS

En accord avec les définitions pré-citées, les principales propriétés d'un agent sont sa capacité de perception de son environnement, sa capacité d'action sur tout ou partie de cet environnement (agents ou objets) et sa capacité de communication (directe ou indirecte selon le type d'agent).

Dans le contexte d'application, nous nous intéressons aux entités mobiles ayant besoin de processus de décision intelligents en vue d'augmenter leur autonomie. Par exemple, les drones et les véhicules connectés doivent réagir à des changements de leur environnement mais aussi faire preuve d'adaptation de leur comportement afin d'être flexibles et pro-actifs pour anticiper certaines actions. Afin d'optimiser leurs actions, ils doivent également adopter des comportements sociaux pour communiquer leurs objectifs, atteindre des buts et exécuter des tâches complexes.

Voici donc les propriétés que nous retenons en particulier ([Jennings et al., 1998a], [Flores-Mendez, 1999]) :

- **Autonomie** : Le comportement de l'agent est influencé par ses seules perceptions qui agissent en conséquence sur son état (environnement, objets de l'environnement, autres agents). Pas de processus de supervision (c'est le propre du système multi-agents).
- **Réactivité** : Au-delà de son fonctionnement interne, l'agent est en contact permanent avec son environnement et réagit donc aux changements qu'il perçoit dans celui-ci.
- **Proactivité** : Selon ses capacités et ses connaissances, l'agent peut prendre des initiatives afin de satisfaire ses buts. Il agit en fonction de ses motivations et de ses intentions. Il peut disposer aussi de capacité inférentielle (agir avec des spécifications abstraites de tâches).
- **Flexibilité/Adaptabilité** : L'agent adapte son comportement à sa perception de son environnement et peut participer à des organisations afin de mieux satisfaire son but. Il peut avoir la capacité d'apprendre et de s'améliorer avec l'expérience mais aussi d'avoir une certaine continuité temporelle (persistance d'une identité et d'un état sur une longue période).
- **Sociabilité** : En complément de sa capacité à s'adapter et à prendre des initiatives, l'agent peut développer des interactions complexes (coopération, collaboration, coordination) pour atteindre ses buts ou pour aider d'autres agents dans leurs activités. Il peut avoir une certaine capacité de communication pour échanger des informations. Ainsi il peut avoir et/ou développer une certaine personnalité.

1.1.4/ ORGANISATION

Un système multi-agents peut être considéré comme une organisation si les agents qui le composent possèdent une connaissance des autres agents du système et suivent les règles fixées par le système global tout en gardant leurs capacités d'autonomie ([Mandiau et al., 1999]). Ce système peut être analysé essentiellement de deux manières :

- **Analyse fonctionnelle** afin de déterminer les principales activités et rôles des agents selon plusieurs dimensions (physiques, sociales, relationnelles, environnementales, etc). Par exemple, un agent peut avoir le rôle de médiateur, planificateur, ordonnanceur, décideur, fournisseur, etc.
- **Analyse structurelle** afin de déterminer les éventuelles interactions entre les agents ainsi que leur ordre et leur évolution dans le temps. Par exemple : relation d'accointance (connaissances d'un agent sur les autres), de communication, de subordination (dépendance), d'information, de conflit, de compétition, etc.

Un modèle d'organisation définit ainsi les relations entre les différentes entités qui composent le système mais aussi leurs rôles, leurs règles d'interaction, leurs activités ainsi que les buts et les critères d'évaluation de ces activités.

Quelques exemples d'organisation :

- **AGR (Agent Groupe Rôle)** ([Ferber et al., 1998], [Ferber et al., 2003])

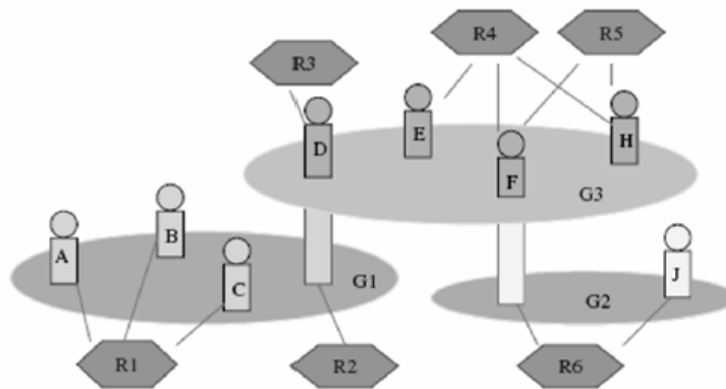


FIGURE 1.1 – Représentation AGR Agent Groupe Rôle ([Ferber et al., 1998])

L'agent est une entité active et communicante qui joue un ou plusieurs rôles dans un ou plusieurs groupes (cf. figure 1.1). Il n'y a pas de contrainte sur l'architecture interne de l'agent (réactif, cognitif ou hybride) ou sur ses facultés mentales. Le groupe représente un ensemble d'agents partageant un ensemble de caractéristiques et est utilisé pour créer des sous-organisations. Pour communiquer, deux agents doivent appartenir à un même groupe. Le rôle est la représentation abstraite de la fonction d'un agent dans un groupe. L'agent peut avoir un ou plusieurs rôles selon la nature du groupe. Les rôles sont locaux aux groupes et sont attribués à un agent sur sa demande. Un même rôle peut être joué par plusieurs agents.

- **AGRE (Agent Groupe Rôle Environnement)** ([Ferber et al., 2004]) Par extension,

AGRE permet de décrire des environnements sociaux (comme AGR) mais aussi des environnements physiques. Ces deux types d'environnements forment des espaces où les agents sont situés et où ils peuvent percevoir et agir au travers des modes (manifestation d'un agent dans un espace). Un mode dans un espace physique est un corps alors qu'un mode dans un espace social est un rôle. Cette organisation permet donc de combiner interaction physique et interaction sociale sans toutefois les relier.

- **AGREEN** ([Báez-Barranco et al., 2006]) Le modèle AGREEN étend encore le modèle AGRE pour relier les espaces sociaux et les espaces physiques en s'appuyant sur la notion d'institution.
- **AGRS (Agent Groupe Rôle Service)** ([Mansour et al., 2007]) Nouvelle extension du modèle AGR avec cette fois-ci la notion de service qui permet aux agents d'exprimer leurs besoins et compétences au travers de descriptions de rôles faciles à publier et rechercher. Dans ce modèle, le rôle décrit les services qu'il fournit. Ces services sont ensuite exécutés par les agents "joueurs" et utilisés par des agents "utilisateurs". Les services fournis par un rôle sont publiés dans le cadre de la description de son groupe.
- **MASQ (Multi Agent System based on Quadrants)** ([Tranier, 2007], [Ferber et al., 2009]) Cette organisation repose sur quatre concepts primitifs que sont l'esprit, l'objet, l'espace brut et la culture. Ces concepts sont ensuite attribués à chacun des quadrants selon des dimensions internes ou externes, individuelles ou collectives (cf. figure 1.2). Cette formalisation a été introduite afin de généraliser, intégrer et mettre en relation des modèles spécifiques d'agent, d'environnement, d'organisation, de norme ou encore d'institution. MASQ permet également de spécifier séparément les agents, l'environnement et les aspects culturels, ce qui le rend intéressant pour modéliser des systèmes ouverts.
- **Approche holonique** (étude phénomènes naturels avec [Koestler, 1968], Gerber pour les approches multi-agents ([Gerber et al., 1999]), ou des travaux plus récents ([Adam et al., 2002], [Gaud, 2007])) Cette approche permet la modélisation plus aisée de populations d'individus (cf. figure 1.3). Par exemple, on peut considérer des entités atomiques de type piéton puis un holon sur une zone géographique donnée formant un échantillon de personnes et enfin un super holon fusionnant les différents sous-holons pour former une population complète.

Certains aspects et caractéristiques sont détaillés plus précisément dans le [chapitre 3](#) avec la présentation d'une classification des organisations.

1.2/ TAXONOMIE

Dans cette section, nous allons introduire les différents types d'agent communément cités dans la littérature et présenter deux classifications (une version "classique" puis une autre selon le niveau de cognition et de sociabilité).

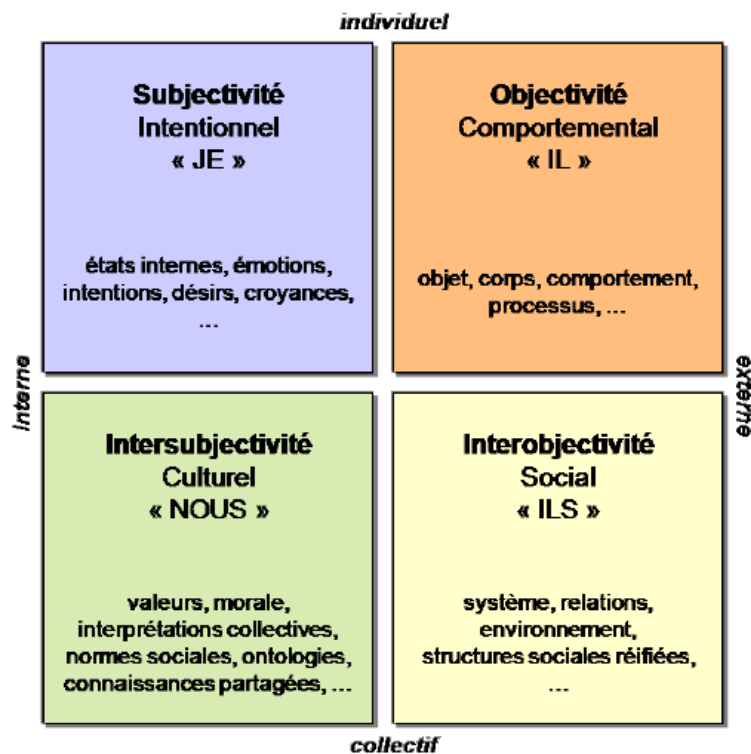


FIGURE 1.2 – Approche MASQ par quadrants ([Tranier, 2007])

1.2.1/ TYPES D'AGENT

En informatique, un agent est un processus cyclique composé de différentes phases (comportements) définies en fonction de la nature même de l'agent. Cependant, quelque soit cet agent, on trouve en premier lieu une phase de perception qui peut être la résultante d'une simple observation ou d'une interaction (simple ou structurée) avec un objet de l'environnement ou un autre agent.

On classe ([Ferber, 1995], [Wooldridge et al., 1995], [Briot et al., 2001]) souvent les types d'agents en fonction de leur comportement individuel et deux grandes catégories d'agents sont alors généralement distinguées : les agents réactifs et les agents cognitifs. Dans les faits, on parle de plus en plus aujourd'hui d'agents hybrides avec des degrés de cognition variable, qui sont à la frontière entre les deux grandes catégories.

1.2.1.1/ AGENT COGNITIF

Ce type d'agent s'inspire des êtres humains dont il cherche à reproduire le mode de pensée dans la continuité des premiers travaux sur l'intelligence artificielle distribuée. Nous pouvons citer quelques travaux fondateurs : Lenat ([Lenat, 1975]), Minsky ([Minsky, 1988]) ou Hayes-Roth ([Hayes-Roth et al., 1979]). Comme il est censé représenter un être pensant, il possède des propriétés avancées de représentation symbolique de son environnement mais aussi des capacités d'abstraction assez fortes. Également, cet agent peut mémoriser ses expériences et en tirer partie pour établir de nouveaux objectifs pour plus de flexibilité et d'adaptation à son environnement.

Ces agents sont construits sur un cycle de type : perception / délibération / action.

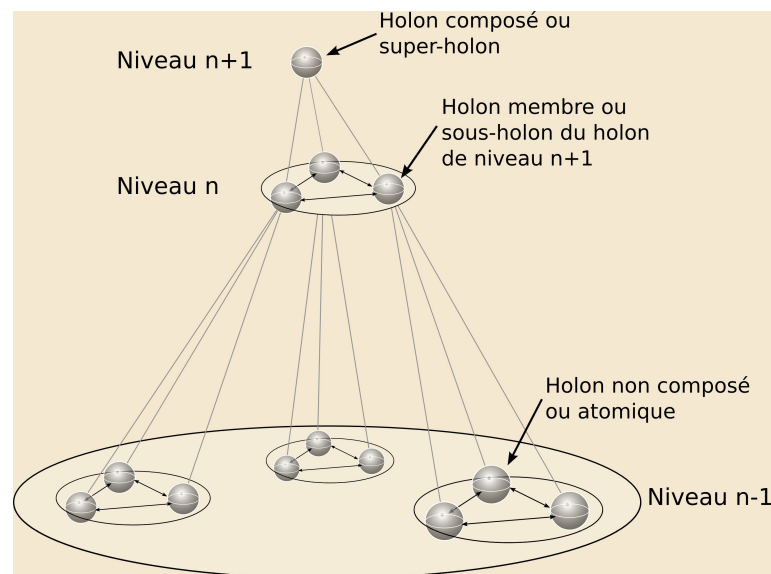


FIGURE 1.3 – Système holonique ([Gaud, 2007])

La phase de délibération détermine le choix de l'action à exécuter. C'est la phase la plus importante dans la mesure où elle est définie par les caractéristiques comportementales de l'agent. L'approche anthropomorphique conduit généralement à concevoir les agents cognitifs comme des entités possédant des croyances, des intentions, des désirs ([Rao et al., 1995]) et même dans certains cas des principes éthiques [Shoham, 1993] et [Wooldridge et al., 1995]. Comme ils disposent de capacité de représentation et de raisonnement, les agents cognitifs peuvent travailler indépendamment mais aussi, dans certains cas, se comporter comme des organisations d'individus devant se coordonner pour résoudre leurs conflits [Jennings et al., 1998b].

Il en résulte de grandes difficultés de mise en œuvre car cet agent doit pouvoir inférer avec une intelligence artificielle plus développée. Son implémentation n'est pas aisée en raison des connaissances à enregistrer, des interactions complexes et des mécanismes de coopération à mettre en œuvre pour satisfaire les objectifs.

On trouve également d'autres appellations pour les agents cognitifs, par exemple agents rationnels ou délibératifs. Ces derniers ont été introduits par Wooldridge ([Wooldridge et al., 1995]). L'agent délibératif ne va pas aussi loin qu'un agent cognitif "pur" inférant sur ces connaissances personnelles. Les mécanismes de réflexion sont un peu plus simple.

1.2.1.2/ AGENT RÉACTIF

Les agents réactifs sont construits sur un cycle simple de type : perception / réaction. L'exécution de l'agent est directement liée à ses perceptions limitées de l'environnement par une fonction réflexe selon le principe d'action/réaction ([Ferber, 1995]), ou stimulus/réponse ([Bonabeau et al., 1999]).

Les principales propriétés de ce type d'agent sont des comportements élémentaires avec une absence de but explicite, des interactions élémentaires (avec l'environnement ou les autres agents), des situations élémentaires avec une absence de représentation de l'en-

vironnement, une absence de représentation des autres agents. Toutes les actions sont exécutées dans le présent, il n'y a pas de planification, pas d'historisation, pas d'enregistrement des expériences. Nous pouvons citer par exemple les travaux de Ferber ([Ferber, 1995]), d'Agre et Chapman ([Agre et al., 1987]). En outre, contrairement aux agents cognitifs, il n'y a pas de représentation de but global du système ni de planification permettant de prévoir des propriétés d'émergence [Drogoul, 1993] ou [Parunak, 1997].

Brooks a introduit le concept de subsomption ([Brooks, 1991a], [Brooks, 1991b]) où le processus de décision d'un agent correspond à la réalisation d'un ensemble de tâches pour réaliser des comportements. Un comportement reçoit en continu des percepts et y associe une action à réaliser. Plusieurs comportements peuvent être exécutés en même temps. Afin de réaliser un choix, les comportements sont hiérarchisés par ordre de priorité. Les comportements sont indépendants et peuvent être ajoutés ou retirés en fonction de la structure de l'agent et de ses missions.

Les agents réactifs, de part leur nature et leurs propriétés limitées, connaissent certaines limitations. L'agent n'ayant pas de représentation structurée de son environnement local doit pouvoir accéder à suffisamment d'information pour réagir convenablement. Il est également difficile de développer des agents avec de multiples comportements, la dynamique des interactions devenant alors trop complexe. En résumé, qui dit "Agents bêtes" dit "Concepteur intelligent".

1.2.1.3/ AGENT HYBRIDE

En fait, pour de nombreux problèmes, la frontière n'est pas aussi nette entre la notion d'agent réactif et d'agent cognitif. Par extension, une architecture purement réactive ou purement cognitive est rare. Ainsi une architecture hybride peut combiner les avantages des deux. Un agent hybride possède donc des composants réactifs et aussi des composants cognitifs pour garantir un raisonnement de qualité [Aknine et al., 1998], [Chaib-draa, 1996], [Sycara, 1998], [Wooldridge, 2009].

On peut distinguer plusieurs types d'architecture :

- **Architecture horizontale** : on trouve un module de perception en entrée, plusieurs couches de traitement (en fonction d'un arbitrage à la sortie du module de perception) et un module d'action en sortie. Cette architecture a une conception assez simple : autant de couche que de comportements attendus en fonction des stimuli de l'environnement. Inconvénient notoire : il n'y a pas de communication entre les couches. On peut citer par exemple l'architecture TouringMachines ([Ferguson, 1992]).
- **Architecture verticale** : elle peut exister en plusieurs versions : en une passe (du module de perception, en passant par les différentes couches de traitement, vers le module d'action) ou en deux passes (perception, aller et retour dans les différentes couches en interaction avant d'agir). Cette architecture est plus difficile à mettre en œuvre mais il y a un certain raffinement des raisonnements : passage d'une couche à l'autre pour traiter un problème plus complexe avec une communication entre les couches. Exemple d'architecture verticale en deux passes : InteRRap ([Müller et al., 1993]).
- **Architecture modulaire** : chaque module va englober des fonctions de percep-

tion/délibération/action contextualisées englobant des possibilités de communication inter-modules. Exemple : DIMA ([Guessoum, 1996]).

1.2.2/ CLASSIFICATION

Depuis plusieurs décennies, la communauté scientifique a l'habitude de distinguer l'agent réactif de l'agent cognitif et d'en proposer une classification "classique" (voir Table 1.1) où les principales caractéristiques sont présentées. L'agent réactif est un "agent qui ne dispose que d'un protocole simple et d'une capacité de communication réduite afin de répondre seulement à un stimulus par une action." [Labidi et al., 1993]. L'agent cognitif est "un agent possédant une base de connaissances avec toutes les informations et l'expertise nécessaires pour mener à bien sa tâche et sachant gérer les interactions avec d'autres agents et son environnement"[Ferber, 1995].

Agents réactifs	Agents cognitifs
Pas de représentation explicite de l'environnement	Représentation explicite
Pas de mémoire locale	Peut enregistrer des évènements et les exploiter
Comportement stimulus/action	Comportement complexe

TABLE 1.1 – Agents réactifs contre agents cognitifs [Ferber, 1995]

L'expérience a montré la nécessité de développer des agents intermédiaires qui ne sont ni purement réactifs ni purement cognitifs. Dans son ouvrage d'introduction au monde des agents, Wooldridge [Wooldridge, 2009] présente ce type d'agent en décrivant quelques architectures hybrides et un nouveau mode d'organisation de raisonnement (avec un agent doté d'un raisonnement déductif et un agent pourvu d'un raisonnement pratique). Aujourd'hui, il existe de nombreux exemples d'architecture réactive, cognitive et hybride [Ferber, 1995, Wooldridge, 2009, Briot et al., 2001]. Chacun d'eux donne un point de vue sur la réaction et la cognition ainsi que les moyens de les concevoir.

Boissier [Briot et al., 2001] propose un autre point de vue en comparant les différents types d'agents avec une vision raisonnement (agent réactif, hybride et délibératif) et coordination (agent autonome, interagissant ou social) dans la Table 1.2. Il énumère les différents modèles et plateformes connues au moment de la rédaction de l'article. Cependant il reste prudent sur la notion d'agent cognitif car seuls les agents délibératifs sont introduits dans ses travaux. Il les considère comme des agents cognitifs n'exécutant que des plans d'actions prédéterminés. Nous pouvons également remarquer que, pour lui, il n'existe pas encore d'agents hybrides avec un comportement social.

A travers le cas d'étude (drones et véhicules connectés) que nous avons présenté en introduction, nous pouvons en déduire que nous serons plus sur des agents hybrides car il faut pouvoir réagir promptement à des stimuli de l'environnement (du fait de leur nature mobile, parfois à haute vitesse) mais aussi pouvoir anticiper certaines actions dans le but de résoudre des tâches complexes ou des objectifs plus ambitieux. De plus, nous avons insisté sur les aspects de communication en vue de réaliser des échanges élaborés, surtout dans le cadre de la mise en place de processus collaboratifs.

Raisonnement Coordination	Agents réactifs	Agents hybrides	Agents délibératifs
Agent autonome	Subsumption, PENGI, MANTA	Touring Machines	<i>IRMA,</i> <i>PRS/dMARS</i>
Agent interagissant	PACO, SMARRPS	<i>InteRRap</i>	IMAGINE, AR- CHON, <i>AOP, COSY,</i> <i>GRATE</i>
Agent social	ECO, PA- CORG		<i>ADEPT, ASIC,</i> STARS, SAM, DIMA, CIA

TABLE 1.2 – Classification de Boissier des modèles et plateformes agents selon leur niveau de cognition et de sociabilité (les architectures de type BDI sont en italique) [Briot et al., 2001]

1.3/ PRINCIPALES PLATEFORMES

Progressivement, au fur et à mesure du déploiement des technologies agents et leur acceptation par la communauté, de nouvelles plateformes ont vu le jour afin de répondre aux attentes des chercheurs. Tout d'abord destinées à des non-informaticiens, ces plateformes sont de plus en plus accessibles à des non-spécialistes.

Une plateforme n'est pas seulement un objet technique pour exécuter un modèle. C'est un environnement intégré disposant d'outils et de capacités de développement d'agents et d'environnement de simulation, avec plus ou moins de possibilités d'abstraction. Chaque domaine peut avoir sa plateforme de prédilection en fonction des attendus et du niveau des modélisateurs de systèmes (informaticiens ou non-informaticiens, spécialistes ou non-spécialistes).

Dans [Banos et al., 2015], une classification des plateformes en 3 catégories est proposée :

- Plateformes utilisant un langage conventionnel de programmation issue de l'Informatique (comme Java, C++ ou Python), demandant de bonnes compétences de programmation et plutôt réservées à des informaticiens. On peut citer Madkit ([Gutknecht et al., 2000]), Mason ([Luke et al., 2004]), Cormas ([Bousquet et al., 1998]), Repast ([Collier, 2003]) ou encore Swarm ([Bonabeau et al., 1999]).
- Plateformes utilisant un langage agent de programmation dédié et simplifié, plus faciles d'accès pour des non-informaticiens. Quelques exemples : NetLogo ([Wilensky, 1999],[Tisue et al., 2004]), Gama ([Amouroux et al., 2007],[Taillandier et al., 2012]), Janus ([Galland et al., 2010]) ou encore SARL ([Rodriguez et al., 2014]). A noter que l'objectif de SARL à terme est d'offrir un environnement graphique pour permettre à tous de modéliser facilement.

- Plateformes proposant un langage graphique de programmation comme Star-Logo ([McCaffrey, 2006]), Repast Symphony ([North et al., 2007]) ou Magéo ([Langlois et al., 2013]). Toutefois, il n'est pas toujours possible de développer des modèles complexes.

Nous pouvons aussi signaler l'existence de la plateforme Fractal PMAS ([Rousset et al., 2016]) utilisant les "nested graphs" et destinée à paralléliser massivement les simulations sur des grilles de calcul.

On peut trouver également quelques articles de synthèse listant et comparant les différentes plateformes comme [Bordini et al., 2006] ou [Berryman, 2008].

1.4/ CE QU'IL FAUT RETENIR

La figure 1.4 donne une vue synthétique du concept de systèmes multi-agents. Ils sont utilisés pour modéliser et simuler les systèmes complexes mettant en œuvre un nombre important d'entités interagissant au sein d'un environnement, objets connectés en général et, dans le cadre de cette thèse, des entités comme les drones ou les véhicules connectés et communicants. Ces derniers exemples permettent justement de préciser qu'il existe donc des systèmes multi-agents simulés (qui peuvent être transposés dans le monde réel) mais aussi réels (qui peuvent être également simulés). Nous retiendrons les propriétés suivantes :

- réseau fortement couplé ;
- Multitude d'interactions ;
- Pas de contrôle global ;
- Données décentralisées.

La figure 1.5 présente un résumé graphique sous forme de carte mentale de l'état de l'art sur les différents types d'agents. Il convient de noter l'importance croissante des agents hybrides avec leurs architectures permettant de combiner les caractéristiques de deux types d'agents principaux dans le but de réaliser des objectifs (individuels ou collectifs) plus complexes nécessitant des interactions structurées comme des processus coopératifs ou collaboratifs.

Maintenant que le concept d'agent est précisé, nous pouvons aborder ce qu'est la collaboration au sens général dans le chapitre suivant.



FIGURE 1.4 – Ce qu'il faut retenir : Systèmes multi-agents



FIGURE 1.5 – Ce qu'il faut retenir : Agents

COLLABORATION GÉNÉRALE

Sommaire

2.1 Avant propos	27
2.2 Collaborer ou coopérer ?	28
2.3 Autres approches	29
2.3.1 Dans le monde du travail	29
2.3.2 Dans le monde de l'éducation et de l'apprentissage	31
2.4 Cycle de la collaboration	33
2.5 Ce qu'il faut retenir	34

L'objectif de ce chapitre est de montrer ce que nous comprenons quand nous parlons de collaboration et de coopération, sur la base de définitions classiques ainsi qu'une approche sémantique et comparative. En effet, ces deux termes (collaboration et coopération) sont très proches et nous les utilisons très souvent dans le langage courant en confondant leurs sens exacts et ce qu'ils sous-entendent. Quelques exemples sont ensuite présentés (dans le monde du travail, de l'éducation, de l'apprentissage) afin de mettre en lumière les principales caractéristiques de la collaboration et de montrer finalement les différences qu'il peut y avoir avec la coopération. Nous souhaitons effectivement différencier les deux mots afin de clarifier notre propos dans les chapitres suivants.

2.1/ AVANT PROPOS

Selon Maslow ([Maslow, 1943]), les besoins sont hiérarchisés (cf. figure 2.1). Cela va des besoins vitaux (physiologique, de protection et de sécurité) à des besoins plus évolués et complexes comme la reconnaissance des autres et l'accomplissement personnel en passant par les besoins sociaux (d'appartenance notamment). On notera l'importance des interactions sociales car elles interviennent dans 3 des 5 niveaux de la pyramide.

Cela souligne donc bien la prépondérance des interactions en vue de l'accomplissement personnel (développement de ses capacités intrinsèques ou résolution de problèmes complexes) qui représente le besoin le plus élevé selon Maslow. Toutefois, ces interactions ne sont pas forcément de simples échanges. En effet, plus on monte dans la pyramide des besoins et plus les échanges doivent être codifiés, encadrés et doivent prendre en considération les attentes et besoins des autres interlocuteurs ou acteurs.

Typiquement, afin de résoudre des problèmes complexes, comme il est évoqué au niveau 5 de la pyramide des besoins, il faut mettre en œuvre des moyens d'échanger avec

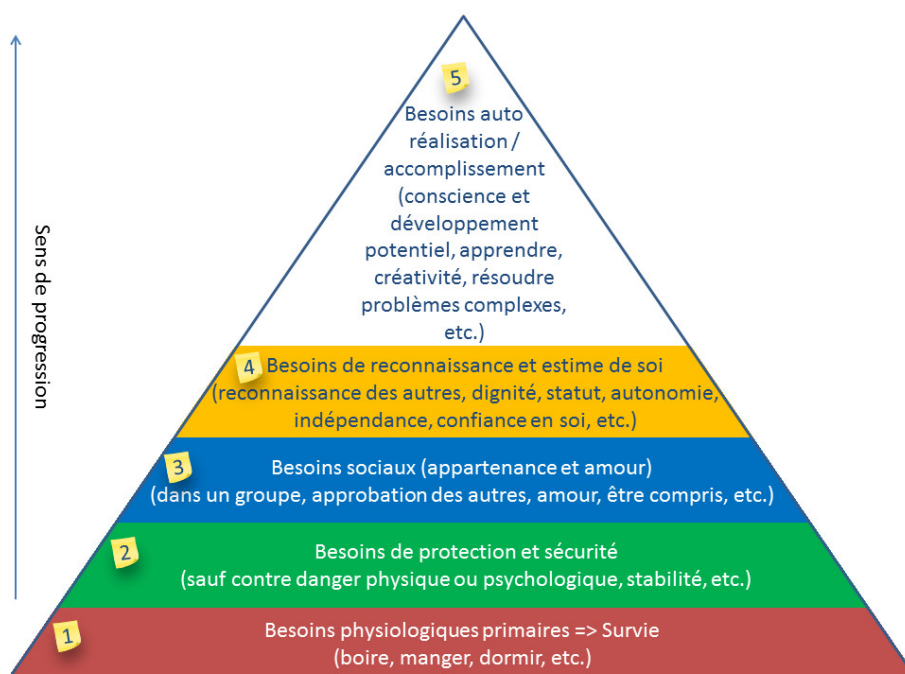


FIGURE 2.1 – Pyramide des besoins de Maslow (illustration communément admise par la communauté)

les différents acteurs qui permettent cette résolution. La collaboration et la coopération sont de bons candidats pour satisfaire ce besoin grâce à la formalisation des échanges sociaux entre congénères et des tâches à réaliser pour y arriver. Cependant s'il existe deux mots pour désigner ce qu'il nous semble au quotidien comme une évidence, c'est qu'il y a quelques subtilités. Nous allons maintenant nous employer à distinguer les points communs et les différences pour mettre en lumière ce qu'implique réellement le concept de collaboration.

2.2/ COLLABORER OU COOPÉRER ?

Nous commençons simplement par l'évocation des définitions classiques que nous pouvons trouver dans les dictionnaires. Ainsi selon le Larousse ¹ :

Définition 8 : Collaboration et Coopération, en français

- Collaboration : Action de collaborer, de participer à une œuvre avec d'autres.
- Collaborer : Travailler de concert avec quelqu'un d'autre, l'aider dans ses fonctions ; participer avec un ou plusieurs autres à une œuvre commune.
- Coopération : Action de coopérer, de participer à une œuvre commune ; collaboration, concours.
- Coopérer : Prendre part, concourir à une œuvre commune ; contribuer, participer.

1. <http://www.larousse.fr/dictionnaires/francais>

On peut remarquer l'ambiguïté des définitions françaises qui ne font que peu de distinction entre la collaboration et la coopération. Il est toujours fait référence à une œuvre commune mais la subtilité réside dans le verbe décrivant l'action : participer pour collaborer et concourir pour coopérer.

Si l'on reprend les origines latines des verbes, la différence est un peu plus explicite :

Définition 9 : Collaboration et Coopération, en latin

- Collaborer (*co laborare*) : travailler ensemble
- Coopérer (*co operare*) : opérer ensemble

Il s'agit donc de distinguer la qualité du travail en commun. Soit il s'agit d'un véritable travail en commun (collaboration) soit il s'agit de mener des opérations communes pour élaborer un tout plus grand (coopération).

Selon Oxford dictionaries² :

Définition 10 : Collaboration et Coopération, en anglais

- Collaboration : The action of working with someone to produce or create something.
- Cooperation : The process of working together to the same end.

Les définitions basiques anglo-saxonnes ne donnent pas forcément plus de clarté. Cependant elles introduisent une nuance : "produce or create something", produire ou créer quelque chose. Là où les définitions françaises parlent globalement de "prendre part à une œuvre commune", les définitions anglaises sont plus concrètes et pragmatiques.

Finalement ces quelques considérations sémantiques n'apportant pas de distinction claire entre les deux termes, il nous a paru important de définir explicitement, en nous appuyant sur d'autres approches, ce que nous entendons exactement par collaboration (toujours en comparaison avec la coopération) pour clarifier le périmètre et les objectifs de notre travail.

2.3/ AUTRES APPROCHES

Avant de définir ce qu'est la collaboration dans le contexte de notre étude, nous donnons dans un premier temps un point de vue général en nous appuyant sur des définitions du domaine et sur différentes approches.

2.3.1/ DANS LE MONDE DU TRAVAIL

Dans [Blanquet, 2009], Marie-France Blanquet décrit les différences entre un travail coopératif et un travail collaboratif. Elle propose également une définition :

2. <http://oxforddictionaries.com>

Définition 11 : Collaboration et Coopération vu par MARIE-FRANCE BLANQUET

Travail coopératif ou à chacun sa partie du tout. Travail collaboratif ou le tout pour tous.

Le travail coopératif est accompli par une division du travail dans laquelle chaque personne est responsable d'une partie de résolution d'un problème.

La collaboration implique un engagement mutuel des participants dans un effort coordonné pour résoudre ensemble le problème.

Selon l'auteur, la distinction se fait clairement sur la nature des relations qu'entretient chaque individu avec les autres membres du groupe :

Pour la collaboration :

- Liberté d'engagement ;
- Responsabilité constamment partagée par rapport aux actions ;
- Capacité à influencer sur la définition et l'enchaînement des actions permettant d'atteindre l'objectif assigné au groupe.

Pour la coopération :

- Notion d'obligation ;
- Responsabilité déléguée par rapport aux actions ;
- Peu ou pas de capacité à influencer sur la définition et l'enchaînement des actions permettant d'atteindre l'objectif assigné au groupe.

Toujours selon l'auteur, on peut donc parler de travail collaboratif quand au moins deux personnes travaillant en mode synchrone (ou asynchrone) dans le même milieu (ou des lieux différents), échangent des informations, organisent leur travail collectivement et définissent des objectifs communs afin de construire ensemble. La collaboration est un processus de co-construction et de maintien d'une conception partagée d'un problème. L'activité est synchronisée et coordonnée.

A la différence, on parle ainsi de travail coopératif quand deux ou plusieurs personnes travaillent conjointement pour un même objectif mais chacun ayant à sa charge une part bien définie du travail à réaliser sans connaissance nécessairement préalable du contexte général ou des intérêts de chacun. C'est une responsabilité déléguée généralement par l'une ou l'autre des parties.

Il convient toutefois de moduler le propos. Tout n'est pas que collaboration ou que coopération. En effet, cette distinction entre coopération et collaboration n'est pas définitive : une certaine division du travail existe toujours dans toute collaboration mais celle-ci est spontanée, basée sur un engagement, un désir de participation libre, alors que dans la coopération, cette division du travail est demandée et raisonnée. Ce qui sépare définitivement la collaboration et la coopération relève plus de la mentalité de ceux qui sont appelés à travailler ensemble. Il y a donc un aspect intentionnel à ne pas négliger.

Marie-France Blanquet explique donc les différences entre les deux types de travail en insistant sur ce qui sépare les deux notions. Là où le travail coopératif apparaît presque mécanique ou automatique avec une répartition dirigée des tâches à accomplir, le travail collaboratif implique une analyse fine et commune du problème à résoudre avec une volonté d'échanger de l'information à tout moment dans le cadre d'un engagement mutuel, d'un contrat moral entre les collaborateurs.

Par exemple, dans le cadre d'un projet informatique, le chef de projet peut répartir les

tâches de programmation auprès des développeurs. Chaque développeur a une tâche spécifique à réaliser (une ou plusieurs fonctions) dans un cadre contraint par le chef de projet. Le développeur n'a pas à se préoccuper des tâches des autres développeurs. La cohérence est assurée par le chef de projet. Il s'agit ici d'une coopération.

Autre exemple, dans le cadre de projets multi-domaines (en auto gestion, sans responsable bien défini), il faut que les chefs de projet partagent leurs préoccupations et leurs contraintes pour réaliser le projet global. Il faut coordonner les actions et les décisions pour que tout se mette en place le mieux possible au niveau le plus élevé : plannings, ressources, contraintes, gestion des risques. Il s'agit dans ce cas d'une collaboration.

2.3.2/ DANS LE MONDE DE L'ÉDUCATION ET DE L'APPRENTISSAGE

Sur le site Edutech³, on retrouve une définition / comparaison de la collaboration et de la coopération.

Définition 12 : Collaboration et Coopération vu par EDUTECH

La coopération correspond à une répartition des tâches entre les apprenants, où chacun peut accomplir de manière autonome et responsable sa part de travail. Dans cette structure d'interdépendance des apprenants, la tâche est donc divisée de manière hiérarchique en sous-tâches totalement indépendantes les unes des autres.

La collaboration demande une implication mutuelle des apprenants dans un effort de coordination afin de réaliser la tâche. Ce type de structure d'interdépendance des apprenants ne veut pas dire qu'il n'y a pas de division du travail, mais que cette division ne se fait pas de la même manière. Les sous-tâches restent très entrelacées les unes aux autres, de telle sorte qu'un effort de coordination demeure tout au long de l'activité, ce qui implique une conception partagée constante du problème.

Avec cette approche, on retrouve généralement ce qui a été décrit dans le paragraphe précédent. D'après l'auteur de cet article, la collaboration est donc un processus qui va plus loin qu'une "simple" coopération. On retrouve le partage des tâches, une certaine coordination des efforts et donc une synchronisation des résultats mais surtout une "conception partagée". Cela sous-entend que ce processus demande un travail de découpage du problème et surtout une compréhension réciproque des tenants et aboutissants de la part des intervenants.

Effectivement, dans le contexte éducatif, entre la coopération et la collaboration, la différence est nuancée. F. Henri et K. Lundgren-Cayrol ([Henri et al., 2001]) analysent la différence entre ces deux notions.

Selon ces auteurs, les concepts d'apprentissage coopératif et d'apprentissage collaboratif sont souvent confondus tant dans le monde de l'apprentissage que du travail. Là où Marie-France Blanquet indiquait qu'il pouvait y avoir des processus coopératifs dans un processus collaboratif plus global, Henri et Lundgren-Cayrol vont plus loin en ne distinguant plus catégoriquement l'un ou l'autre des processus. Ils les considèrent plutôt semblables à des pôles qu'à des catégories exclusives. Pour les différencier, il est important de bien saisir les nuances relatives à l'autonomie et au contrôle (lors de la phase

3. <http://edutechwiki.unige.ch>

d'engagement et de suivi des actions), aux moyens utilisés pour atteindre le but et effectuer la tâche, et de bien différencier le niveau d'interdépendance existant entre les participants en fonction du type de relation (par exemple hiérarchique).

En règle générale, les groupes coopératifs et collaboratifs travaillent bien tous les deux en vue d'atteindre un but commun ou partagé. Mais c'est dans la manière de partager le travail que la différence est la plus visible. La façon d'atteindre le but par la coopération repose sur la distribution des tâches et des responsabilités au sein du groupe tandis que la collaboration exige une responsabilité individuelle pour atteindre le but.

Le tableau ci-après compare les 2 approches.

Apprentissage	Coopératif	Collaboratif
Buts d'apprentissage	But commun imposé, atteint collectivement par le groupe	But commun partagé, atteint individuellement
Contenu	Contenu structuré, généralement présenté par le formateur	Structure à découvrir, à explorer et à élaborer par l'apprenant sur une base individuelle ou en groupe
Situation d'apprentissage	Apprentissage en équipe; on utilise la réalisation d'un travail collectif pour apprendre	Apprentissage individuel, mais partage des ressources de l'environnement. On utilise la dynamique et le travail de groupe pour apprendre
Tâches	Distribuées entre les différents apprenants	Réalisées individuellement de façon personnelle, sans exclure la possibilité de réaliser une tâche coopérative
Participation	Obligatoire	Volontaire et spontanée
Responsabilité de l'apprenant	Il est responsable de la tâche confiée, dont l'exécution est coordonnée à celle des autres	Il est responsable de son apprentissage, mais il est également engagé envers le groupe
Évaluation	Habituellement sommative, portant sur la production finale de l'équipe. Notation de groupe. Toutefois, il peut y avoir une évaluation formative par un retour sur la démarche du groupe	L'évaluation formative est plus appropriée et porte surtout sur les connaissances acquises (diagnostic individuel des stratégies et du processus d'apprentissage). Les apprenants contribuent à l'évaluation en donnant leur appréciation sur le fonctionnement du groupe (cohésion et productivité)

TABLE 2.1 – Synthèse et comparaison sélective des apprentissages coopératifs et collaboratifs ([Henri et al., 2001])

De nombreux critères sont étudiés par les deux auteurs mais nous allons insister seulement sur ceux qui nous paraissent importants. La table 2.1 présente les éléments constitutifs des apprentissages collaboratifs et coopératifs. Cette synthèse donne une première approche de notre point de vue sur la collaboration, par opposition à la coopération : buts d'apprentissage, contenu, situation d'apprentissage, tâches, participation, responsabilité de l'apprenant et évaluation du résultat. Tous ces éléments vont permettre l'élaboration de la définition générale du [chapitre 6](#).

Pour résumer, l'apprentissage coopératif est mis en œuvre pour atteindre un but commun imposé et atteint collectivement avec un contenu structuré. Les tâches sont distribuées et la participation est obligatoire. Chaque participant est responsable de sa tâche. L'apprentissage collaboratif est mis en œuvre pour atteindre lui aussi un but commun mais qui est partagé par tous les participants et qui est atteint individuellement. Le contenu dépendra beaucoup des échanges et des besoins entre les participants. Les tâches sont réalisées individuellement de manière volontaire et spontanée. Chaque participant, en plus d'être responsable de sa tâche, a un engagement envers le groupe.

Dans le cadre de notre étude sur les entités mobiles autonomes, nous allons désormais nous intéresser exclusivement à la collaboration qui permet la réalisation d'objectifs personnels dans une démarche collective.

2.4/ CYCLE DE LA COLLABORATION

Elaboré par "The Association for Image and Information Management : The Global Community of Information Professionals" et commenté par Frank Potter⁴, voici une proposition du cycle de la collaboration (cf. 2.2) pour illustrer les caractéristiques évoquées plus haut.

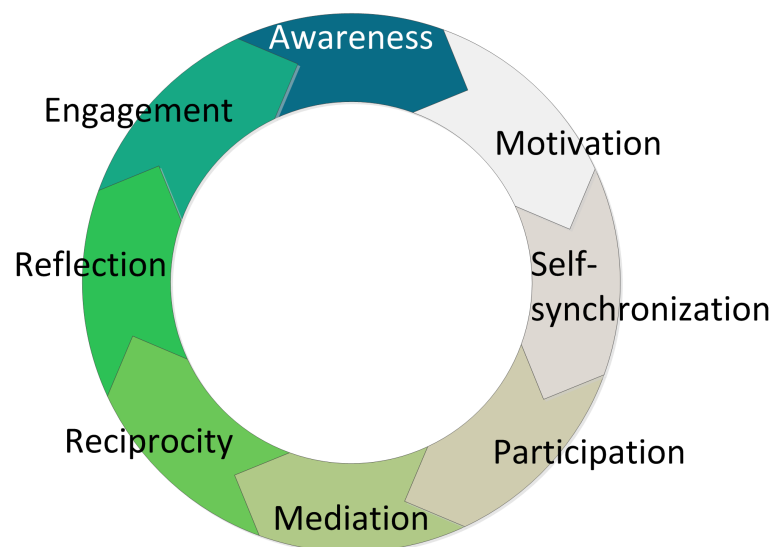


FIGURE 2.2 – Cycle de la collaboration proposé par l'AIIM (<http://www.aiim.org/What-is-Collaboration>)

A un niveau conceptuel, une collaboration implique :

- **Awareness / Conscience** : avoir conscience de faire partie d'un ensemble travaillant autour d'un objectif partagé ;
- **Motivation** : essayer d'obtenir un consensus dans la résolution ou le développement de problèmes ;
- **Self-synchronization / Auto-synchronisation** : décider en tant qu'individu lorsqu'un événement survient ;

4. <http://myorgbio.org/aiims-life-cycle-collaboration-model/>

- **Participation** : participer à une collaboration en espérant que d'autres participent également ;
- **Mediation / Médiation** : négocier et collaborer ensemble pour trouver un compromis ;
- **Reciprocity / Réciprocité** : partager et espérer un retour ;
- **Reflection / Réflexion** : penser et considérer les alternatives ;
- **Engagement** : s'engager de manière proactive plutôt que d'attendre et de voir ce qu'il va passer. Etre un acteur de la collaboration.

Dans ce cycle, on retrouve effectivement la plupart des propriétés qui ont été évoquées dans les paragraphes précédents. Cela tend à confirmer que toutes les définitions et les approches évoquées précédemment convergent bien.

2.5/ CE QU'IL FAUT RETENIR

Les sections précédentes nous ont permis de mettre en évidence les différences fondamentales entre une coopération et une collaboration en terme de partage d'informations, d'engagement, de responsabilité. On ne peut pas confondre les deux processus. Il n'y a pas d'ambiguïté. Nous allons maintenant mettre l'accent sur les seuls processus collaboratifs conformément à notre problématique d'étude.

La collaboration est un processus où tous les acteurs sont impliqués autour d'un but commun et d'une conception partagée et ce, quelque soit le domaine considéré (éducation, apprentissage plus largement ou le monde du travail). La figure 2.3 résume les éléments nous paraissant fondamentaux. Il s'agit d'un engagement mutuel où tous les participants ont une liberté de participation et d'action. Cependant ils ont tous une responsabilité dans la création et le maintien de l'organisation nécessaire à mettre en œuvre pour assurer le bon fonctionnement du processus collaboratif. A noter également que la collaboration, au travers des échanges, permet d'accroître de ses connaissances voire d'étendre ses compétences. On peut y voir une sorte d'apprentissage social.

Maintenant que la collaboration a été abordée généralement, le chapitre suivant va décrire comment est décrite la collaboration dans le domaine des systèmes multi-agents.

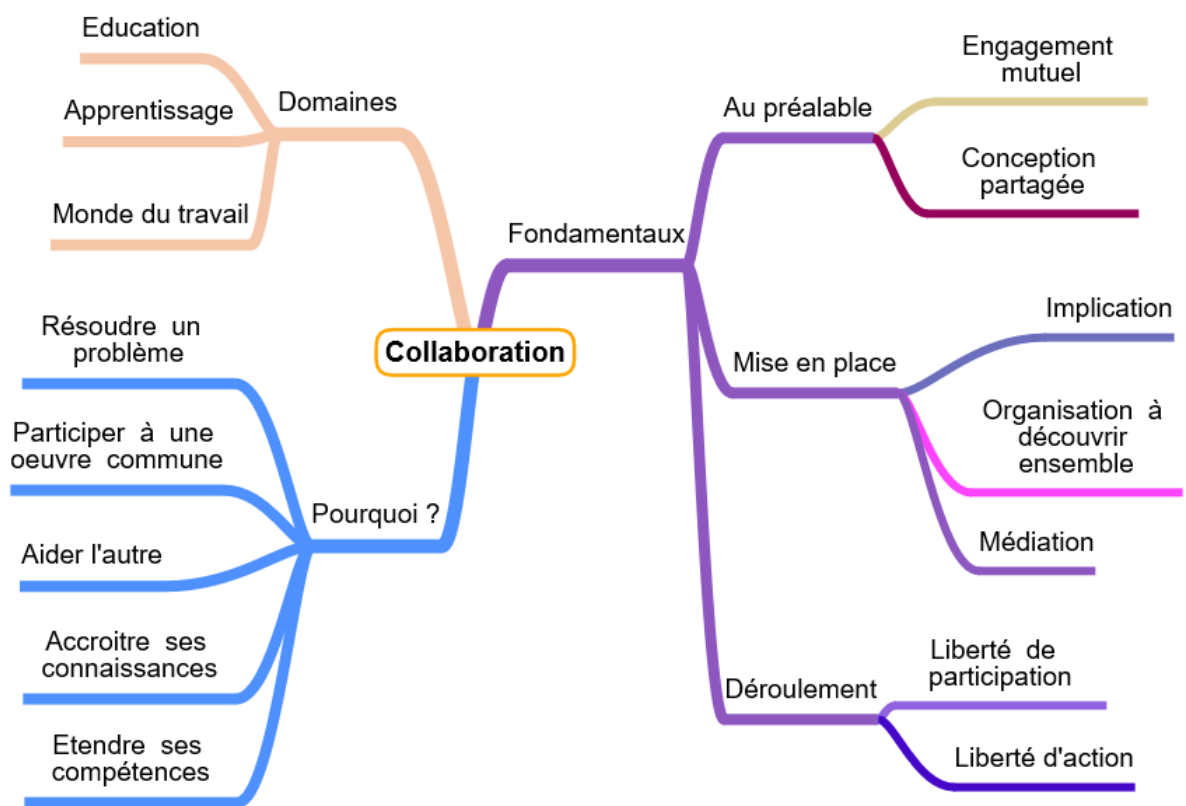


FIGURE 2.3 – Ce qu'il faut retenir : Collaboration

COLLABORATION DANS LES SMA

Sommaire

3.1 Définitions	38
3.2 Cycle de la collaboration	39
3.2.1 Intentions	39
3.2.2 Interactions	40
3.2.3 Communication	42
3.2.4 Coordination	45
3.2.5 Négociations	46
3.2.6 Organisations	49
3.3 Architectures	50
3.4 Ce qu'il faut retenir	51

L'objectif de ce chapitre est de présenter, dans le domaine des systèmes multi-agents, comment la collaboration est présentée conceptuellement et envisagée concrètement sous ses différents aspects.

Après une série de définitions, nous utilisons les grands principes présentés au [chapitre 2](#) pour évoquer la transposition de ces concepts dans le monde agents afin de structurer ce chapitre. Nous pouvons ainsi détailler le processus d'une collaboration : l'intention qui constitue la motivation, l'interaction permettant de mettre en œuvre les autres étapes du cycle, les processus de coordination avec l'utilisation de mécanismes de communication. Les protocoles d'échange et de négociation sont abordés. Puis les différents types d'organisation possibles seront listés. Enfin quelques architectures intégrant la possibilité d'implémenter des processus collaboratifs sont présentées pour permettre une première justification de la voie que nous souhaitons emprunter.

Dans le cadre d'application de notre étude (entités mobiles intelligentes comme les drones ou les véhicules connectés), nous nous intéressons seulement aux cas de collaborations intentionnelles, souhaitées et recherchées par l'agent. En effet, nous pourrions évidemment distinguer la collaboration au niveau de l'agent de la collaboration au niveau de l'observation globale du système (comme un processus émergent, [Drogoul, 1993]) mais nous nous concentrons sur les processus égocentrés.

3.1/ DÉFINITIONS

Dans le domaine des systèmes multi-agents, les définitions de la coopération et de la collaboration mettent en évidence l'interaction entre les agents et les principes de cognition. Quelques actions de coordination et des algorithmes de résolution des conflits pour réaliser les tâches sont également nécessaires.

Par contre, les points de vue divergent selon les auteurs. Parfois la collaboration est considérée comme une coopération de haut niveau (plus élevé que la coopération) et parfois la collaboration n'est qu'une composante de la coopération. En résumé, dans certains cas, on peut définir *Coopération* = *Collaboration* + *Coordination* et dans d'autres, *Collaboration* = *Coopération* + *Coordination*.

Par exemple, selon Jacques Ferber [Ferber, 1995], *"la collaboration est une forme d'interaction qui s'intéresse à la manière de répartir le travail entre plusieurs agents, qu'il s'agisse de techniques centralisées ou distribuées."* Cependant, il reste très ambigu sur la définition et la distinction entre coopération et collaboration car il dit également que *"la coopération demeure l'apanage des êtres capables d'avoir un projet explicite donc des agents cognitifs. Il est possible de parler d'une coopération de manière aussi bien réactive que cognitive si l'on envisage uniquement le résultat des actions et non les intentions des agents."* La notion d'intention semble donc attachée implicitement au concept de collaboration. Et pourtant, dans un autre article ([Ferber, 1994]), Ferber considère que c'est la coopération qui est vue comme une attitude intentionnelle, caractéristique d'une posture des agents. Les agents coopèrent s'ils s'engagent dans une action commune après avoir identifié et adopté un but commun. La constitution de l'association résulte de la prise de conscience de l'existence d'un but commun et de l'engagement de chacun de ses membres. Même s'il est troublant de constater que nous avons à faire à deux définitions presque opposées, il y a tout de même des éléments intéressants à prendre en compte : le caractère intentionnel, un but commun, des actions communes, une répartition des tâches, un engagement.

D'ailleurs le problème de l'engagement envers un but collectif est appelé une adoption de but ([Galliers, 1991], [Conte et al., 1991]), laquelle est considérée comme un élément essentiel de l'activité sociale. En particulier pour Julia Rose Galliers ([Galliers, 1991]), il y a coopération si les agents s'engagent dans une action et identifient un but commun.

Pour d'autres, plus simplement, la collaboration est considérée comme une coopération de haut niveau avec le développement d'une compréhension mutuelle, associée à un partage de point de vue sur les tâches à accomplir par plusieurs individus en interaction [Weiss, 1999].

A cause de cette ambiguïté entre les notions de coopération et de collaboration, nous allons désormais utiliser des articles traitant indifféremment des deux concepts pour asseoir notre raisonnement.

Nous poursuivons par les buts de coopération ou de collaboration (car cela dépend de la définition et varie en fonction des auteurs ainsi que du contexte) ([Durfée et al., 1989]) :

- Augmenter la vitesse de résolution des tâches par leur parallélisation,
- Augmenter l'ensemble ou la portée des tâches réalisables par le partage des ressources,
- Augmenter la probabilité d'achever des tâches (fiabilité) par la duplication des tâches et si possible par l'utilisation de différentes méthodes pour les réaliser,

- Diminuer l'interférence entre les tâches en évitant des interactions nuisibles.

La collaboration peut également être considéré comme un apprentissage social comme nous l'avions évoqué à la fin du [chapitre 2](#). Panait et Luke ([Panait et al., 2005]) ont établi un état de l'art sur cette capacité de la coopération/collaboration à servir de support à l'apprentissage. Cela va dans le sens des travaux de formalisation de F. Henri et K. Lundgren-Cayrol ([Henri et al., 2001]) évoqués également dans le chapitre précédent.

3.2/ CYCLE DE LA COLLABORATION

Conformément au cycle de la collaboration vu au [chapitre 2](#), nous pouvons juxtaposer quelques grandes étapes comme :

- Intention : prise de conscience d'un besoin ;
- Interaction : prise de contact avec un autre agent pouvant aider ;
- Communication : moyens pour échanger les informations ;
- Coordination : synchronisation des actions pour accomplir des tâches communes ;
- Négociation : médiation pour trouver un compromis ;
- Organisation : relation entre agents.

3.2.1/ INTENTIONS

Cette section fait écho à la prise de conscience et à la motivation du cycle de la collaboration évoqué précédemment. Cela peut se faire de plusieurs manières : soit une réflexion sur les objectifs, soit la confrontation à un événement venant influencer la prise de décision. Dans tous les cas, c'est une analyse des données, des croyances afin de déterminer une action.

Le modèle d'architecture BDI [Rao et al., 1995] par exemple met en œuvre les principaux aspects de la théorie de Michael Bratman sur le raisonnement pratique humain avec les notions de croyance, de désir et d'intention [Bratman, 1987].

D'autres évolutions de cette architecture basé sur ce modèle augmenté comme le modèle B-DOING ([Dignum et al., 2000]) peuvent prendre en compte des obligations et des normes. Les obligations sont les désirs particuliers des autres agents et les normes sont les désirs généraux de la société dans son ensemble.

Les auteurs considèrent deux types de normes : les normes qui interdisent la réalisation de certaines actions (interdiction de fumer dans un restaurant) et les normes imposant que certaines actions soient réalisées ou qu'un certain état soit atteint (obligation de payer avant de quitter le restaurant).

Voici une évolution de l'algorithme d'origine avec la gestion des engagements ([Cohen et al., 1990a]) :

Algorithme 1 : Interpréteur BDI avec la gestion des engagements

```

1 B ← B_init
2 I ← I_init
3 while true do
4   get_perception(perc)
5   B ← belief_revision(B,perc)
6   D ← options(B,I)
7   I ← filter(B,D,I)
8   plan = find_plan(B,I)
9   while not (empty(plan) or succeeded(I,B) or impossible(I,B)) do
10    action = next_action(plan)
11    execute(action)
12    get_perception(perc)
13    B ← belief_revision(B,perc)
14    if reconsider(I,B) then
15      D ← options(B,I)
16      I ← filter(B,D,I)
17    end
18    if not(sound(plan,I,B)) then
19      plan = find_plan(B,I)
20    end
21  end
22 end

```

Dans le contexte avec des entités mobiles intelligentes et autonomes, ce modèle peut être intéressant pour fixer un cadre d'exécution en utilisant une "culture" commune aux différents agents collaborants. Cela peut être utile notamment pour des agents hétérogènes car dans le cas d'agents homogènes, ils sont intrinsèquement identiques et donc partagent *a priori* les mêmes contraintes normatives (dépendantes de leur programmation).

3.2.2/ INTERACTIONS

Une des principales propriétés de l'agent dans un SMA est sa faculté d'interagir avec les autres agents. Ces interactions sont généralement définies comme toute forme d'action exécutée au sein du système d'agents et qui a pour effet de modifier le comportement d'un autre agent. Elles permettent aux agents de participer à la satisfaction d'un but global. Cette participation permet au système d'évoluer vers un de ses objectifs et d'avoir un comportement intelligent indépendamment du degré de complexité des agents qui le composent.

L'interaction est un processus dans lequel les agents s'engagent afin de s'assurer que, prises ensemble, leurs actions s'exécutent d'une manière cohérente. Ferber ([Ferber, 1995]) propose la définition suivante : *Une interaction est une mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques.*

L'interaction peut être décomposée en trois phases non nécessairement séquentielles ([Chaib-draa, 1996]) :

- Réception d'informations ou la perception d'un changement ;
- Raisonnement sur les autres agents à partir des informations acquises ;
- Emission de messages ou plusieurs actions (plan d'actions) modifiant l'environnement. Cette phase est le résultat d'un raisonnement de l'agent sur son propre savoir-faire et celui des autres agents.

Ce concept d'interaction est fondamental pour les SMA. En effet, sans interactions, aucune métaphore sociale n'est possible. Dans le cadre des SMA, une interaction peut être définie comme la mise en relation dynamique de deux ou plusieurs éléments du système par le biais d'un ensemble d'actions réciproques ([Gechter, 2003]). Ces éléments peuvent être des agents ou des éléments de l'environnement ou encore l'environnement lui-même. Les interactions peuvent être directes (Agent-Agent) ou indirectes (Agent-Environnement).

Par interaction directe, nous entendons une communication directe se faisant par envoi de message. Ce message est encodé par l'agent émetteur avant sa transmission à l'agent récepteur. Ce dernier décode le message reçu pour saisir son contenu. Il s'agit d'un acte délibéré de transfert d'informations vers un ou plusieurs autres agents.

L'interaction indirecte englobe l'utilisation de l'environnement ([Weyns, 2007]) et du *tableau noir* ([Hayes-Roth, 1985]). La *stigmergie* permet aux agents de laisser des traces dans l'environnement qui peuvent être également utilisées par d'autres afin de guider leurs actions.

A mi-chemin entre interaction directe et indirecte, nous pouvons citer [Decker et al., 1997] où un agent intermédiaire (une entité qui n'est ni demandeur, ni fournisseur d'un service) participe à l'interaction effective entre un demandeur et un fournisseur. Nous pouvons aussi parler de l'écoute flottante et de l'attention mutuelle (*mutual awareness*) ([Ricci et al., 2005], [Saunier et al., 2006]) où des agents qui ne sont pas protagonistes de la communication, peuvent participer en recevant certains messages qui ne leur sont pas destinés.

Le degré de complexité des connaissances nécessaires pour traiter les interactions dépend des capacités de raisonnement de l'agent et du fait que l'agent a connaissance ou non de l'objectif du système global. En effet, un agent qui poursuit un objectif individuel au sein du système, comme c'est le cas pour les agents réactifs, ne focalise pas son énergie pour interagir avec les autres même s'il y est amené. Par contre, un agent qui participe à la satisfaction du but global du système tout en poursuivant un objectif individuel, va passer une partie de son temps à coopérer ou à se coordonner avec les autres agents. Pour cela, il doit posséder des connaissances sociales qui modélisent ses croyances sur les autres agents.

La classification des situations d'interactions par Ferber ([Ferber, 1995]) (cf. Table 3.1) permet de poser une base de réflexion sur l'importance des processus collaboratifs.

Elle 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. Les tentatives de définition de la coopération/collaboration sont souvent subordonnées aux concepts de communication, de coordination, de négociation et d'organisation ([Bouron, 1992]). Nous les détaillons maintenant dans les sections suivantes.

Buts	Ressources	Compétences	Types de situation	Catégorie
Compatibles	Suffisantes	Suffisantes	Indépendance	Indifférence
Compatibles	Suffisantes	Insuffisantes	Collaboration simple	Coopération
Compatibles	Insuffisantes	Suffisantes	Encombrement	Coopération
Compatibles	Insuffisantes	Insuffisantes	Collaboration coordonnée	Coopération
Incompatibles	Suffisantes	Suffisantes	Compétition individuelle pure	Antagonisme
Incompatibles	Suffisantes	Insuffisantes	Compétition collective pure	Antagonisme
Incompatibles	Insuffisantes	Suffisantes	Conflits individuels pour des ressources	Antagonisme
Incompatibles	Insuffisantes	Insuffisantes	Conflits collectifs pour des ressources	Antagonisme

TABLE 3.1 – Classification des situations d'interactions ([Ferber, 1995])

3.2.3/ COMMUNICATION

Les communications, dans les SMA comme chez les humains, sont à la base des interactions et de l'organisation. Une communication peut être définie comme une forme d'action locale d'un agent vers d'autres agents. Les questions abordées par un modèle de communication peuvent être résumées par les interrogations suivantes : Pourquoi les agents communiquent-ils ? Quand ? Avec qui ? Comment ?

- **Pourquoi les agents communiquent-ils ?** La communication doit permettre la mise en œuvre de l'interaction et par conséquent toute forme de coopération / collaboration ainsi que la coordination éventuelle d'actions ;
- **Quand les agents communiquent-ils ?** En fonction des situations où le besoin d'interagir est recommandé pour l'atteinte de buts locaux ou globaux. La difficulté réside dans l'identification de ces situations ;
- **Avec qui les agents communiquent-ils ?** Les communications peuvent être sélectives sur un nombre restreint d'agents ou diffusées à l'ensemble des agents. Le choix de l'interlocuteur dépend essentiellement des accointances de l'agent (connaissances sur les autres agents) ;
- **Comment les agents communiquent-ils ?** La mise en œuvre de la communication nécessite un langage de communication compréhensible et commun à tous les agents. Il faut identifier les différents types de communication et définir les moyens permettant non seulement l'envoi et la réception de données mais aussi le transfert de connaissances avec une sémantique appropriée à chaque type de message.

La théorie des actes de langage est une théorie de la communication fondée dans les années 1960 par le philosophe J.L. Austin ([Austin, 1962]). Elle fut développée sous l'inspiration de L. Searle ([Searle, 1969]). Le postulat à la base de la théorie affirme que la production d'un énoncé s'assimile à l'accomplissement d'actions : si agir c'est transformer l'état des choses, parler c'est également transformer l'état mental des interlocuteurs. Une conséquence de ce postulat est qu'il doit être possible d'identifier les constituants élémentaires de la communication. Ceux-ci sont appelés actes de langage.

J.L. Austin s'est intéressé aux différents types d'actes de langage que constituent les actions intentionnelles effectuées au cours d'une communication. Il distingue trois types d'actes de langage : actes locutoires, actes illocutoires et actes perlocutoires.

- Les actes locutoires se rapportent à la formulation d'un énoncé. Ces actes sont satisfaits lorsque l'énoncé est correctement formulé ;
- Les actes illocutoires désignent l'action effectuée sur l'auditeur lors de la formulation de l'énoncé (par exemple : donner un ordre, poser une question). Ils sont accomplis avec succès lorsque l'effet attendu sur l'auditeur est obtenu ;
- Les actes perlocutoires rapportent aux conséquences indirectes visées par les actes locutoires et illocutoires. Ils ne sont pas codés dans l'énoncé et dépendent du contexte dans lequel s'effectue la communication. Ils sont satisfaits lorsqu'ils sont reconnus par l'auditeur (par exemple : convaincre, faire croire).

On trouve principalement deux standards : KQML ([Finin et al., 1994]) et FIPA/ACL ([Poslad et al., 2000]).

KQML (Knowledge Query and Manipulation Language) a été développé aux États-Unis ([Finin et al., 1994]). Il est fondé sur la théorie des actes de langage dans le but de permettre aux agents cognitifs de coopérer. Il est basé sur le fait de pouvoir coder explicitement dans les messages des actes illocutoires en terme de type de message (performatives) et repose sur les états mentaux des agents. Le contenu du message échangé est une expression spécifiée en KIF (Knowledge Interchange Format) utilisant le formalisme de la logique de premier ordre.

KQML est un langage de communication et un protocole de haut niveau pour l'échange de l'information, orienté messages et indépendant de la syntaxe du contenu et de l'ontologie applicable (vocabulaire des agents). Conceptuellement, nous pouvons identifier trois couches dans un message de KQML : contenu, communication et message.

- La couche *contenu* comporte la teneur réelle du message utilisant un langage de représentation propre au système ;
- La couche de *communication* code un ensemble de dispositifs au message qui décrivent les paramètres de communication les plus bas, tels que l'identité de l'expéditeur et du destinataire et un identifiant unique associé à la communication ;
- La couche *message* code un message qu'une application voudrait transmettre à une autre. Cette couche détermine les genres d'interactions au sein des agents dialoguant via KQML. La fonction de la couche message est d'identifier l'acte du langage (performative) que l'expéditeur attache au contenu. Cet acte de langage indique si le message est une affirmation, une question, une commande ou tout autre d'un ensemble de performatives connus (types de messages primitifs).

Les messages de KQML ne concernent pas seulement des phrases dans un langage quelconque mais sont enrichis d'une attitude sur le contenu (affirmation, désengagement, requête, question, etc). Ce langage propose une description d'un nombre important de performatives permettant les conversations entre agents mais manque de spécifications et de formalisation (cf. les critiques de Cohen et Levesque, [Cohen et al., 1995]).

Les performatives de KQML peuvent être classées en trois catégories :

- Discours (performatives utilisées pour l'échange d'informations et connaissances) : tel, untell, ask-if, ask-one, ask-all, stream-all, deny, insert, delete-one, delete-all, uninsert, undelete, etc ;

Élément	Signification
performative	type de l'acte communicatif
sender	l'émetteur du message
receiver	le destinataire du message
reply-to	participant à l'acte de communication
content	le contenu du message (l'information transportée par la performative)
language	le langage dans lequel le contenu est représenté
encoding	décrit le mode d'encodage du contenu du message
ontology	le nom de l'ontologie utilisé pour donner un sens aux termes utilisés dans le contenu
protocol	contrôle la conversation
conversation-id	identificateur de la conversation
reply-with	identificateur unique du message, en vue d'une référence ultérieure
in-reply-to	référence à un message auquel l'agent est entrain de répondre (précisé par l'attribut reply-with de l'émetteur)
reply-by	impose un délai pour la réponse

TABLE 3.2 – Structure d'un message FIPA/ACL

- Interconnexion entre les agents (pour faciliter l'obtention des informations et connaissances) : broker-one, recommend-one, recruit-one, broker-all, recommend-all, recruit-all, broadcast, register, unregister ;
- Exception (performatives modifiant le flux des informations et connaissances) : sorry, standby, error, ready, next, rest, discard, etc.

Autre standard plus récent, FIPA/ACL (Foundation for Intelligent Physical Agents / Agent Communication Language, [Poslad et al., 2000]) est également basé sur la théorie des langages : les messages sont des actions ou des actes communicatifs car ils sont prévus pour effectuer une certaine action en vertu de l'envoi. Les spécifications de FIPA-ACL se composent d'un ensemble de types de message (ayant des effets sur les attitudes mentales des agents communicants, émetteur et récepteur). Les spécifications décrivent chaque acte communicatif avec une forme narrative et une sémantique formelle basés sur la logique modale.

Les performatives de FIPA/ACL peuvent être classées en cinq catégories :

- Transmission d'informations : confirm, disconfirm, inform, inform-if, inform-ref ;
- Demande d'informations : query-if, query-ref, suscribe, cancel ;
- Négociation : propose, accept-proposal, reject-proposal, cfp ;
- Actions : agree, cancel, refuse, request, request-when, request-whenever ;
- Erreurs : failure, not-understood.

FIPA/ACL est assez semblable à KQML (cf. Table 3.2). Ainsi, il maintient l'approche de KQML de distinguer le langage externe du langage interne. Le langage externe définit la signification prévue du message. Le langage interne (contenu) porte les croyances, les désirs et les intentions des interlocuteurs.

Les deux langages diffèrent principalement dans les détails de leurs cadres sémantiques. En effet, il n'est pas possible de proposer une traduction systématique entre les perfor-

matives de KQML et celles complètement équivalentes de FIPA/ACL. Ils diffèrent aussi au niveau de ce qui constitue la description de la sémantique : pré-conditions, post-conditions, et conditions d'accomplissement pour KQML ; pré-conditions de faisabilité et effets rationnels pour FIPA-ACL.

Une autre différence entre les deux ACL est dans la gestion des agents (enregistrement dans le système, localisation, appartenance à un groupe, etc). Dans KQML, ces tâches sont associées aux performatives que le langage traite en tant que propositions de premier ordre. FIPA/ACL, prévu pour être un ACL pur, ne considère pas ces tâches individuellement comme des actes de communication. Au lieu de cela, il les traite en tant qu'actions rattachées à la primitive *request* et définit ainsi une gamme d'actions réservées qui couvrent les tâches de gestion dans le cycle de vie des agents telles que la tâche d'enregistrement.

3.2.4/ COORDINATION

Selon [Nwana, 1994] et [Jennings, 1996], il y a différentes raisons pour lesquelles des agents ont besoin de se coordonner :

- Empêcher un comportement chaotique du système. Par définition, aucun agent n'a de vue globale de tout le système. Dans cette situation, il est assez simple de tomber dans des configurations chaotiques, où le comportement du système en tant que tout n'est pas prévisible ;
- Satisfaire des contraintes globales. Ces contraintes existent très souvent et un groupe d'agents est considéré comme ayant réussi sa tâche s'il ne les viole pas ;
- Atteindre un objectif global que chaque agent isolé ne peut atteindre tout seul. Différents agents peuvent avoir des expertises et/ou sources d'information différentes, la mutualisation de ces expertises et de ces sources d'informations est nécessaire afin de parvenir à l'objectif global ;
- Respecter les dépendances entre les actions individuelles des agents ;
- Améliorer l'efficacité du système. Quand des agents découvrent des informations que les autres peuvent utiliser pour accomplir leurs tâches, les partager fait gagner du temps à tous, accélérant la résolution du problème traité.

Nous pouvons distinguer 4 types principaux de coordinations d'action entre agents :

- Coordination par synchronisation (par ajustements mutuels) ;
- Coordination par planification (de centralisé à distribuée) : l'action comme processus de coordination :
 - Planification centralisée pour agents multiples ;
 - Coordination centralisée pour plans partiels ;
 - Coordination distribuée pour plans partiels.
- Coordination par réglementation (standardisation) : lois ou conventions sociales (code de la route par exemple) ;
- Coordination réactive : de type stimulus-réponse (agents réactifs).

Dans le contexte des entités mobiles intelligentes et autonomes, nous nous concentrons sur les agents cognitifs à tendance sociale. Deux approches sont possibles : définir des croyances mutuelles, des désirs et des intentions communes ou définir des normes et des conventions. Ainsi, pour la réalisation d'agents sociaux, Rao et Georgeff ([Rao et al., 1992]) définissent des agents qui ont les notions de croyances communes

et de buts et intentions communs. Ils utilisent des plans sociaux (au sens de planification) spécifiant comment ils doivent être menés et sous quelles conditions. Le processus de négociation ou de décision collective consiste alors à choisir un but commun puis sélectionner un plan social satisfaisant le but commun et finalement s'engager en commun à la réalisation de ce plan.

Woolridge et Jennings ([Wooldridge et al., 1994]), quant à eux, choisissent le double processus de l'engagement (promesses en vue de réaliser certaines actions) et des conventions (lois sociales partagées par tous les agents). Ainsi, pour eux, la réalisation d'un construit social chez les agents, se fait en plusieurs étapes. Dans un premier temps un agent réalise un travail de reconnaissance en identifiant une coopération potentielle. Puis l'agent va chercher à former un groupe ou une équipe en sollicitant de l'assistance. Il y a alors formation d'un plan social, l'équipe négocie alors pour un plan commun. Enfin suit l'exécution, phase au cours de laquelle les agents jouent les rôles définis par le plan social choisi.

Les conventions, quant à elles, décrivent les circonstances suivant lesquelles un agent doit reconsidérer son engagement, c'est à dire la manière dont un agent avertira les autres s'il se désengage ou si le statut de son engagement change. Ainsi, Woolridge et Jennings envisagent le processus de coordination comme étant la somme des engagements, des conventions sociales, et du raisonnement local. Enfin, Sichman ([Sichman, 1995]), et Conte et Castelfranchi ([Conte et al., 1992]) introduisent la notion de dépendance sociale entre agents. Nous considérons alors un contexte relationnel, les relations sociales étant préexistantes aux engagements. Il s'agit de simuler des agents socialement situés qui se trouvent dans un réseau de relations (réseau d'acointances). Il n'est alors plus nécessaire de raisonner sur un plan global pour réaliser que les agents coopèrent.

3.2.5/ NÉGOCIATIONS

Dans les systèmes multi-agents en général et dans le cas des agents intelligents en particulier, la négociation est une composante essentielle et contribue à leur autonomie ([Jennings et al., 2001]). Comme les objectifs des agents peuvent être différents, il n'y a pas de solution imposée à l'avance. Les agents doivent trouver des solutions conjointement et dynamiquement. C'est un processus déterminant pour tous les systèmes collaboratifs.

D'après [Rosenschein et al., 1994], la négociation s'appuie en général sur les quatre composants suivants :

- Un protocole ;
- Des stratégies ;
- Des information sur l'état des agents ;
- La recherche d'un équilibre.

Nous pouvons donner plusieurs définitions complémentaires de la négociation. Ainsi, Bussman et Miller [Bussmann et al., 1992] définissent la négociation comme *le processus de communication d'un groupe d'agents afin d'atteindre un accord mutuellement accepté sur un certain sujet*. David et Smith ([Davis et al., 1983]) disent que *par négociation, on entend une discussion dans laquelle des individus intéressés échangent des informations et arrivent à un accord en commun*. Sycara ([Sycara, 1989]) met l'accent sur le fait

que, afin de négocier effectivement, les agents doivent raisonner sur les croyances, désirs et intentions des autres agents. Cela se rapproche des concepts fondamentaux des architectures BDI. Quant à Pruitts ([Pruitt, 2013]), sa définition s'appuie sur des considérations psychologiques et pour laquelle le conflit est l'élément de base : *La négociation est le processus par lequel plusieurs individus prennent une décision commune. Les participants expriment d'abord des demandes contradictoires, puis ils essaient de trouver un accord par concession ou par la recherche de nouvelles alternatives.*

Si nous regardons de plus près les définitions, deux aspects de la négociation se dégagent : la communication et la prise de décisions, l'objectif étant de trouver un accord ou un compromis pour interagir et agir ensemble.

Nous pouvons distinguer plusieurs types de négociation :

- Aux enchères ;
- Par heuristique ;
- Par argumentation ;
- Par allocation de tâches.

La négociation aux enchères (*auctions*) repose sur des mécanismes d'interaction assez simples. Une enchère comprend un initiateur (*actioneer*) annonçant un objet à vendre et plusieurs participants (*bidders*) intéressés pour l'acheter. L'initiateur veut vendre l'objet au prix le plus élevé possible et les participants veulent l'acheter au prix le plus bas possible. Les participants font des offres (*bids*) à tour de rôle. Les offres des participants peuvent se faire une seule fois ou en plusieurs tours en fonction du protocole d'enchère. Parfois, l'initiateur impose une offre minimale acceptable (prix de réservation). A la fin, l'initiateur choisit le gagnant, les règles pour choisir le gagnant étant spécifiques au protocole. En fonction du protocole également, les enchères peuvent s'organiser autour d'offres publiques (*open cry*) ou d'offres cachées (*sealed bid*).

Dans les méthodes heuristiques de négociation, le problème central est de modéliser la décision de l'agent au cours de la négociation. Généralement, le protocole choisi ne prescrit pas une stratégie optimale pour l'agent. Dans ce cas, le succès relatif de deux agents est déterminé par l'efficacité de leur modèle de raisonnement. Les agents doivent découvrir d'une manière heuristique quelle est la meilleure proposition ou contre-proposition qu'ils doivent faire. Ils doivent découvrir cela en partant du modèle de décision qu'ils emploient. Meilleur sera le modèle et plus grande sera la récompense de l'agent. Le modèle de décision de l'agent est le dispositif que les participants emploient en conformité avec le protocole de négociation pour atteindre leurs objectifs. Le protocole contraint les types d'opérations qui peuvent être exécutées sur l'objet de la négociation et prescrit le comportement du modèle de décision de l'agent. Pour améliorer l'efficacité de la négociation, les agents doivent fournir des réactions plus utiles aux propositions qu'ils reçoivent. Ces réactions peuvent prendre la forme d'une critique ou d'une contre-proposition (proposition refusée ou modifiée). Une critique est un commentaire sur la partie de la proposition que l'agent accepte ou refuse. Une contre-proposition est une proposition alternative engendrée en réponse à une proposition. À partir de telles réactions, l'auteur doit être capable d'engendrer une proposition qui est probablement plus apte à mener à un accord.

Pendant la négociation heuristique, les agents peuvent essayer de changer le rejet ou la modification d'une proposition faite par un autre agent en utilisant des arguments. Ainsi, un agent peut essayer de persuader un autre agent de répondre favorablement à sa

proposition en cherchant des arguments qui identifient de nouvelles occasions, créent de nouvelles occasions ou modifient les critères d'évaluation. En plus d'engendrer propositions, contre-propositions et critiques, un agent cherche à faire la proposition plus attirante en fournissant une information supplémentaire sous forme d'arguments pour sa proposition. La nature et les types des arguments peuvent varier énormément. Gilbert ([Gilbert, 2011]) identifie quatre types différents d'arguments :

- Mode logique ressemblant à la preuve mathématique (nature déductive) ;
- Mode émotif utilisant les sentiments et les attitudes de l'agent ;
- Mode viscéral couvrant les aspects physiques et sociaux ;
- Mode kiscéral faisant appel à l'intuition notamment.

Nous pouvons citer deux exemples de systèmes d'argumentation : ANA (Agent de Négociation Automatisé) ([Kraus et al., 1998]) et PERSUADER ([Sycara, 1988]). Ce sont des agents egocentrés utilisant une méthode de négociation pour essayer de convaincre les autres d'accepter leurs propositions en cas de refus. Dans ce but, les agents doivent être capables de représenter leurs propres croyances, désirs et buts, de raisonner sur les croyances, désirs et buts des autres agents et d'essayer d'influencer les croyances et les intentions des autres agents du système.

Les différents protocoles de négociation évoqués plus hauts se destinent à des agents egocentrés ayant leurs propres buts. Mais il existe aussi des protocoles conçus principalement en vue de la coordination des agents coopératifs, ayant les mêmes buts, qui partagent et résolvent ensemble des problèmes comme le protocole réseau contractuel (*Contract Net Protocol*) qui a été une des premières approches utilisées dans les systèmes multi-agents pour résoudre le problème d'allocation des tâches ([Smith, 1980]). Les agents se coordonnent grâce à l'établissement de contrats afin d'atteindre des buts spécifiques.

Dans le protocole réseau contractuel, les agents peuvent prendre les rôles de gestionnaire ou de contractant. L'agent qui doit exécuter une tâche (le gestionnaire) commence par décomposer cette tâche en plusieurs sous-tâches. Le gestionnaire annonce chaque sous-tâche sur un réseau d'agents (les contractants). Les agents qui reçoivent une annonce de tâches à accomplir évaluent l'annonce. Les agents qui ont les ressources appropriées, l'expertise ou l'information requise pour accomplir la tâche, envoient au gestionnaire des soumissions qui indiquent leurs capacités à réaliser la tâche. Le gestionnaire rassemble toutes les propositions qu'il a reçues et alloue la tâche à l'agent qui a fait la meilleure proposition. Ensuite, le gestionnaire et les contractants échangent les informations nécessaires durant l'accomplissement des tâches. Depuis, ce protocole a été étendu au cas où les agents sont impliqués dans plusieurs négociations ([Aknine et al., 2004]).

Dans le contexte des entités mobiles intelligentes autonomes, tout laisse à penser que les protocoles de négociation heuristiques semblent les plus adaptés. En effet, les agents doivent prendre des décisions les concernant individuellement pour conduire leurs missions. Nous pouvons dire qu'ils sont egocentrés. Cependant, nous sommes également dans un contexte collaboratif et tout dépendra du type d'agents. Par exemple, si ce sont des agents homogènes, il est tout à fait envisageable d'avoir recours à des négociations de type Contract Net Protocol. L'objectif étant de trouver le meilleur compromis, il est acceptable de faire les concessions nécessaires pour éviter des échanges sans fin.

3.2.6/ ORGANISATIONS

Nous pouvons classer les différentes typologies des organisations dans les systèmes multi-agents en deux catégories : typologie relationnelle et structurelle ([Adam, 2000], [Mandiau et al., 2002]).

La typologie relationnelle dépendant de la typologie des interactions, cette classification ([Adam, 2000]) distingue les organisations multi-agents à structure hiérarchique de celles à structure hétérarchique (pas de supérieur hiérarchique, chaque entité prend ses propres décisions) :

- Structure hiérarchique : le contrôle est centralisé sur un agent de niveau supérieur (chef d'orchestre) qui est le seul à donner des ordres à des exécutants de plus bas niveau. Ce sont des structures rigides qui poursuivent un seul objectif opérationnel par niveau hiérarchique ;
- Structure hétérarchique : décomposition en trois catégories
 - Marché : structure faiblement décentralisée car le contrôle est partagé sur plusieurs agents coordinateurs hiérarchiques qui visent des objectifs opérationnels différents ;
 - Communauté : structure fortement décentralisée qui ne présente aucune forme de hiérarchie car tous les agents sont de même type, ont les mêmes capacités et partagent le même niveau de contrôle. Dans ce type d'organisation, il n'y a pas de règles d'interaction, il s'agit donc d'échanges de flux de données ;
 - Société : croisement entre les organisations hiérarchiques et les organisations de communauté (règles d'interaction à définir précisément). Structure décentralisée où le contrôle est partagé par plusieurs entités n'ayant pas forcément les mêmes capacités. Adam ([Adam, 2000]) parle d'ajustement mutuel entre les différentes unités par des principes de négociation.

Dans le contexte des entités mobiles intelligentes autonomes ayant des missions individuelles, nous pouvons distinguer deux cas de figure. Si ce sont des agents homogènes, la structure hétérarchique de type communauté est la plus adaptée. Il s'agit plus ici d'un objectif global que d'objectifs réellement individuels. Par contre, si nous sommes en présence d'agents hétérogènes, une organisation de type société semble clairement la plus adéquate. Tous les agents ont leurs propres buts, leurs propres capacités de réflexion et d'action. Les objectifs peuvent être antagonistes.

Les structures d'une organisation multi-agents peuvent être soit prédéfinies et connues à l'avance, soit découvertes *a posteriori* (structure émergente résultant des interactions entre agents). La classification suivante ([Mandiau et al., 2002]) distingue deux modèles d'organisation (selon leurs processus de formation) :

- Émergente : organisation à découvrir, c'est-à-dire qu'elle est définie progressivement avec l'évolution du système à travers les interactions inter-agents ;
- Support d'activités : organisation définie à l'avance (dès la conception du système). Elle impose le fonctionnement du contrôle global du système qui peut être soit centralisé (avec un agent coordinateur), soit décentralisé. Cette structure organisationnelle précise également pour chaque agent, sa fonction dans l'organisation, ses compétences ainsi que les possibilités d'interactions le concernant.

Dans le contexte de notre étude, il y a deux cas de figure. Si ce sont des agents homogènes, un modèle d'organisation par support d'activités peut être utilisé. Nous pouvons imaginer qu'à tour de rôle, en fonction des missions, les agents peuvent prendre

ou se donner le rôle d'agent coordinateur. Par contre, plus généralement, dans le cas d'agents hétérogènes, l'organisation retenue sera émergente. En effet, il est impossible de prédire quelle formation prendront les agents participants et interagissants dans le système.

3.3/ ARCHITECTURES

Il existe plusieurs architectures hybrides mettant en œuvre des possibilités de comportements délibératifs et collaboratifs comme les TouringMachines de Fergusson ([Fergusson, 1992]) et InteRRaP de Müller ([Müller et al., 1993]). D'autres modèles peuvent intégrer des possibilités de collaboration comme DIMA de Zahia Guessoum ([Guessoum, 1996]) mais il ne sera pas détaillé ici.

Le modèle de TouringMachines [Fergusson, 1992] (cf. la figure 3.1) est une architecture multi-couches avec une suite de couches dédiées et productrice de suggestions d'actions : une couche réactive, une couche de planification (pro-actif) ainsi qu'une de modélisation ou une couche coopérative (prise en compte des perceptions, mise à jour des croyances, prévisions de conflits avec les autres agents, changements des plans, buts).

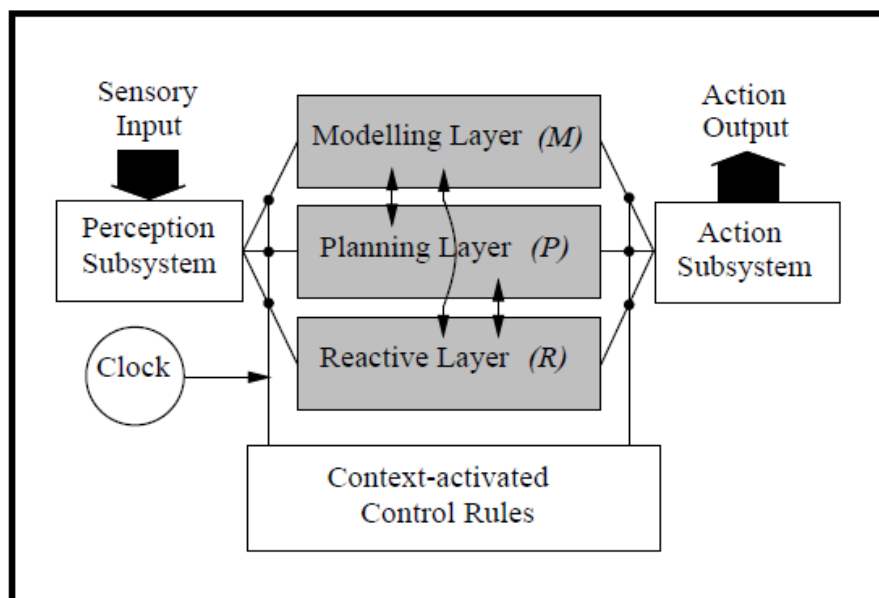


FIGURE 3.1 – TouringMachines (Fergusson)

Comme pour le modèle de TouringMachines, InteRRaP [Müller et al., 1993] (cf. la figure 3.2) est un modèle multi-couches avec une suite de couches dédiées : une couche réactive, une couche de planification ainsi qu'une de modélisation ou une couche coopérative. Chaque couche utilise une partie d'une base de données (de type hiérarchique) contenant toutes les informations requises pour la bonne exécution de celle-ci.

Plus les comportements requis pour répondre aux besoins de l'agent sont complexes, plus les informations sont reprises, raffinées et analysées par les couches supérieures.

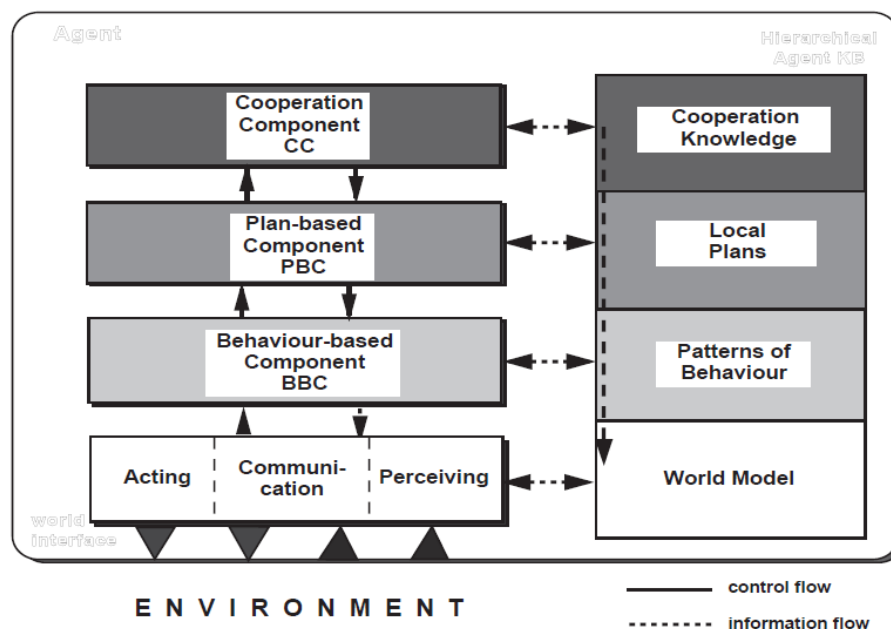


FIGURE 3.2 – InteRRaP (Müller)

En cas de blocage, la dernière couche dite de coopération déclenche un dialogue avec d'autres agents afin de constituer un plan commun autour d'un but défini. La communication est donc limitée à l'ultime couche de l'architecture. Le flux de contrôle redescend dans les couches basses du modèle pour exécuter les actions basiques. Cette architecture peut ne pas être adaptée à des agents devant réagir et communiquer rapidement des informations vers l'extérieur (environnement ou autres agents).

3.4/ CE QU'IL FAUT RETENIR

Ce chapitre a permis de faire un tour d'horizon de la notion de coopération / collaboration dans le domaine des systèmes multi-agents. Les définitions sont parfois très différentes mais les composantes principales sont toujours les mêmes. La figure 3.3 résume les éléments évoqués dans les sections précédentes.

L'intention guide le comportement d'un agent intelligent et autonome. Il évolue dans un monde où, par l'intermédiaire d'interactions avec d'autres agents ou d'éléments de l'environnement, il modifie son état mental et ses comportements pour satisfaire ses besoins et résoudre ses objectifs. Il cherche à coordonner ses actions avec les autres en négociant et en s'appuyant sur des outils de communication. Des organisations spécifiques peuvent se mettre en place au gré des actions ou plans à mettre en musique.

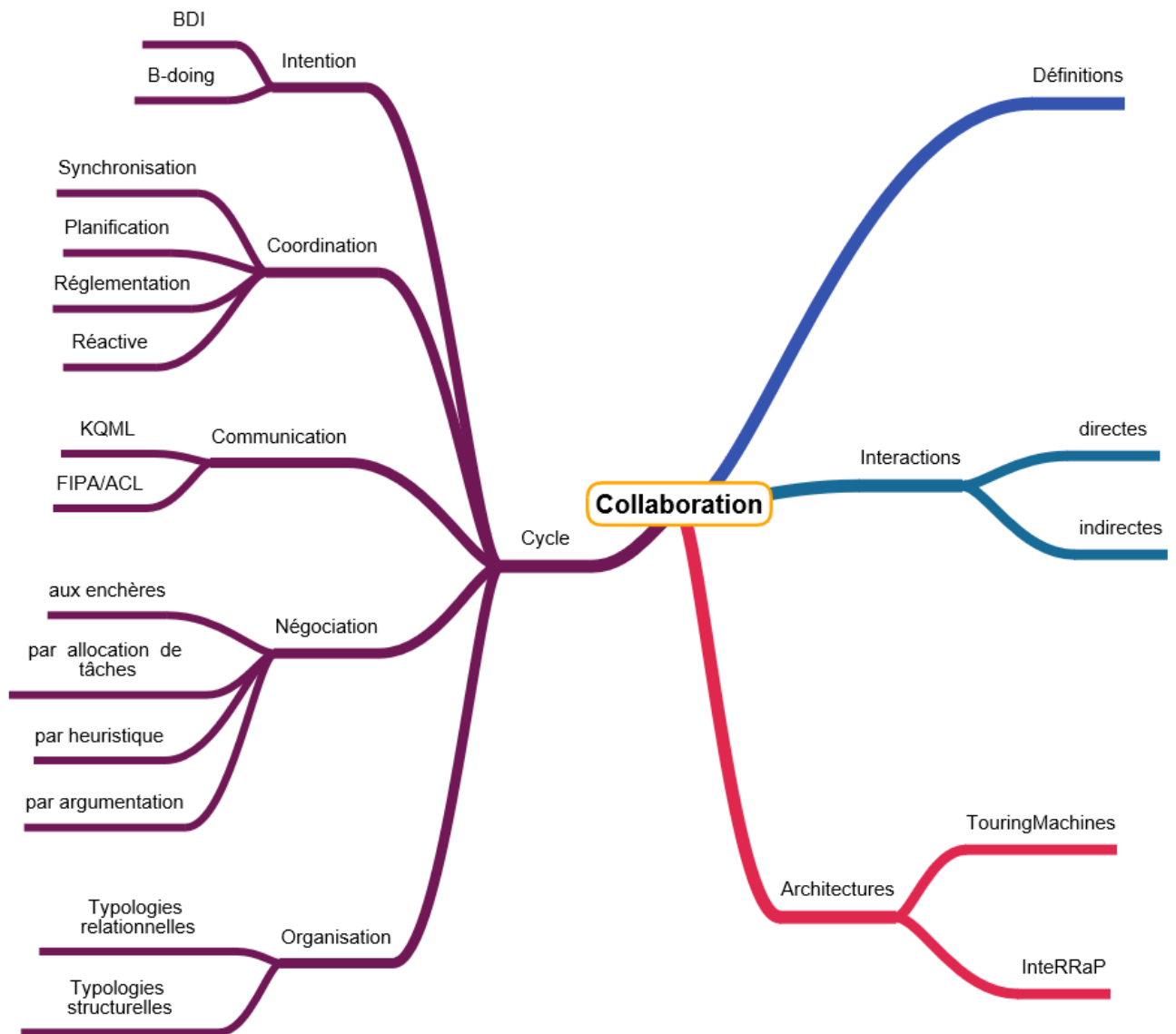


FIGURE 3.3 – Ce qu’il faut retenir : Collaboration dans les SMA

DRONES ET VÉHICULES CONNECTÉS

Sommaire

4.1 Drones	53
4.1.1 Qu'est-ce qu'un drone ?	53
4.1.2 Différents types de drones	54
4.1.3 Domaines d'utilisation et enjeux	55
4.1.4 Drones et collaboration	56
4.1.5 Limites actuelles	57
4.2 Véhicules connectés	57
4.2.1 Quelques définitions	57
4.2.2 Niveaux d'autonomie	58
4.2.3 Propositions des constructeurs	60
4.2.4 Enjeux	61
4.2.5 Quelques verrous	62
4.2.6 Limites actuelles	63
4.3 Ce qu'il faut retenir	64

L'objectif de ce chapitre est de présenter le contexte d'application de notre étude en évoquant les drones (qui vont constituer notre fil rouge technique) et les véhicules connectés (notre cas d'application).

Ce n'est pas une étude exhaustive car ce sont des domaines spécifiques, disposant d'une abondante production scientifique. Mais il nous semble néanmoins important de faire un focus rapide pour fixer les esprits.

4.1/ DRONES

4.1.1/ QU'EST-CE QU'UN DRONE ?

D'après le site Wikipedia¹ :

1. <https://fr.wikipedia.org/wiki/Drone>

Définition 13 : Drone

Un drone (de l'anglais « faux-bourdon »), également appelé UAV (pour Unmanned Aerial Vehicle), ou encore RPAS (Remotely Piloted Aircraft Systems), est un aéronef sans personne à bord, télécommandé ou autonome, qui peut éventuellement emporter une charge utile, destinée à des missions (ex. : de surveillance, de renseignement, d'exploration, de combat, de transport, etc).

Plus généralement, on utilise aujourd'hui le mot drone pour désigner tout dispositif téléopéré, semi-automatique ou automatique, qu'il soit terrestre, marin ou aérien.

On parle de plus en plus de système autonome ([Huang, 2007]) (Unmanned System, UMS) qui est une entité mobile possédant un degré variable d'autonomie dans ses décisions et ses actions possibles. On trouve ainsi depuis quelques années, un certain nombre d'applications impliquant notamment des robots terrestres (Unmanned Ground Vehicle, UGV), des drones volants (Unmanned Aircraft Vehicle, UAV) ou des drones marins (Automated Underwater Vehicle, AUV) développés aussi bien dans le domaine civil que militaire.

4.1.2/ DIFFÉRENTS TYPES DE DRONES

Comme nous l'avons abordé dans la section précédente, il y a toute une palette de drones et de dispositifs intelligents. Nous ne pouvons pas citer exhaustivement tous ces appareils mais nous pouvons en lister quelques-uns.

Dans le domaine des drones volants, il y a de nombreux exemples d'appareils militaires ou civils :

- Drones militaires (Figure 4.1) : engins volants généralement d'une taille importante de 2 à 8 mètres de longueur pour une envergure pouvant atteindre 15 m. D'une masse conséquente (plusieurs centaines de kilogrammes hors charge tactique), ils sont pilotés par un ou plusieurs pilotes au sol et appuyés par un ou plusieurs techniciens ;



FIGURE 4.1 – Drone militaire General Atomics MQ9 Reaper

- Drones civils (Figure 4.2) : engins beaucoup plus petits avec des missions plus pacifiques. Ces drones, l'instar des drones militaires, sont également souvent pilotés par une personne au sol même si un usage en autonomie partielle est possible (avec l'enregistrement préalable d'un parcours) ;



FIGURE 4.2 – Drone civil Delta H

- Mini ou micro-drones : pour une utilisation ludique ou à vocation scientifique dans le cadre de recherche.

Dans le domaine des drones marins, nous pouvons lister plusieurs catégories d'engin :

- ROV (Remotely Operated Vehicle) : télé-opéré, partiellement automatisé pour l'observation ou la manipulation ;
- Crawler : véhicule à chenille/roues et propulseurs, uniquement télé-opéré ;
- AUV (Automated Underwater Vehicle) : entièrement autonome ;
- Glider : planeur sous-marin à flottabilité variable ;
- Bio-inspirés : méduses, poissons, anguilles/serpents, etc ;
- ASV : Autonomous Surface Vehicle avec plusieurs modes de propulsion possible (moteur, vent, houle).

Dans le domaine des drones/robots terrestres, il s'agit principalement d'engins industriels disposant des roues ou de chenilles et généralement télé-opérés.

4.1.3/ DOMAINES D'UTILISATION ET ENJEUX

Les drones fournissent de nouvelles possibilités de changements économiques et sociaux dans de nombreux domaines comme l'exploration (cartographie et détection), la logistique, le monde agricole (culture ou élevage), etc.

Par exemple, dans le domaine des drones aériens, ce sont principalement des opérations de cartographie et de détection mais aussi des travaux d'inspection (comme la maintenance des pylônes électriques). Tout dépend du type de voilure : fixe ou tournante. Il y a un type de drones par domaine d'exploitation.

Plus spécifiquement, pour les drones marins, les domaines d'utilisation sont principa-

lement la recherche océanographique (observation, mesures), l'inspection de matériels, l'industrie offshore (notamment les activités pétrolières), l'enfouissement de câbles ou les opérations de déminage (dans le domaine militaire).

Pour les drones terrestres, il s'agit essentiellement d'opérations de détection, d'inspection ou de déminage.

Les applications sont donc nombreuses et de nouveaux besoins vont certainement encore émerger au fur et à mesure des évolutions technologiques.

L'utilisation des drones permet d'être plus efficace, plus rapide et moins coûteux qu'une intervention humaine. Ils sont d'ailleurs très utiles pour tous les travaux dangereux comme les opérations de déminage ou d'exploration en milieu hostile (comme la centrale nucléaire de Fukushima avec l'explosion de 2011).

4.1.4/ DRONES ET COLLABORATION

Avec la recherche sur les drones autonomes, apparaissent de plus en plus de projets de collaboration. Ces applications reposent sur l'utilisation de plusieurs engins, formant ainsi une flotte.

Nous pouvons ainsi citer le projet *Swarmies*² de la NASA qui utilise plusieurs engins terrestres afin d'optimiser l'exploration d'une zone inconnue. Le projet *Carus* ([Chaumette et al., 2011]) utilise quant à lui une flotte de plusieurs drones afin d'opérer la surveillance d'une zone.

Dans les exemples précédents, il s'agissait de drones homogènes de même type (terrestres ou aériens) mais il existe aussi d'autres projets cherchant à faire travailler conjointement des drones/robots de type aérien et terrestre. Avec ce type de coopération (Air-Ground Cooperation ou AGC), les véhicules aériens sans pilote (UAV) et les véhicules terrestres sans pilote (UGV) fonctionnent de manière coopérative pour effectuer une tâche donnée.

Par exemple, on considère un environnement avec un bâtiment endommagé par un tremblement de terre ([Michael et al., 2014]). On utilise et coordonne une équipe de drones volants (UAV) et terrestres (UGV) pour fournir une carte 3D des trois premiers étages. Le but principal de ce travail est de fournir un aperçu de degré de dommages à l'intérieur de ce bâtiment. L'équipe de robots (tous équipés de télémètre laser) a été capable de cartographier les trois étages en une après-midi.

Une autre application des AGC est la navigation coopérative où généralement le drone agit comme un capteur à distance ou une caméra volante ([Stentz et al., 2002]). Ces flottes de drones peuvent aussi surveiller une zone ([Grocholsky et al., 2006], [Hsieh et al., 2007]).

Nous pouvons encore citer, par exemple, la localisation collaborative ([Forster et al., 2013]) ou la saisie collaborative, ces projets utilisant également deux types de drones distincts (avec leurs propres caractéristiques physiques).

Il est impossible de citer tous les travaux actuels mais nous pouvons voir qu'il existe désormais de nombreuses recherches sur les moyens de faire collaborer des drones (de type homogène ou distinct) disposant d'une autonomie et d'une intelligence accrue.

2. <https://www.nasa.gov/content/meet-the-swarmies-robotics-answer-to-bugs>

4.1.5/ LIMITES ACTUELLES

Les limites et verrous actuels sont à la fois techniques et réglementaires.

Pour tout dispositif évoluant dans les airs, sous la mer ou sur terre, le problème principal est lié à l'exploitation des capteurs : collecte et fusion des données, interprétation. Par exemple, nous pouvons citer les capteurs acoustique (de type sonar) encore peu fiables et imprécis ou encore les capteurs liés à la vision qui sont sensibles à l'angle de vue ou au mode d'éclairage.

Comme nous l'avons vu dans la section précédente avec les recherches sur la collaboration, les difficultés sont liées à l'autonomie décisionnelle et à la sûreté de fonctionnement. Il faut une architecture logicielle distribuée permettant de faire coopérer une flottille hétérogène et tolérante aux fautes. Il y a des contraintes de supervision, de replanification éventuelle de missions. La communication entre les entités est essentielle mais difficile à mettre en oeuvre et à sécuriser. Il faut également gérer la perte éventuelle d'un membre et planifier au mieux les trajectoires.

Enfin il faut aussi à gérer efficacement la batterie.

Concernant les problèmes réglementaires, il y a la gestion des droits des personnes liée au risque d'utiliser des drones dans des espaces occupés par des êtres humains. Cela conduit à la gestion de la sécurité des déplacements de ces entités (surtout volantes) pour éviter les accidents. Enfin, il y a également la préservation des données privées. Le législateur propose déjà une régulation de l'utilisation des drones (notamment grand public). Nul doute que les lois vont encore évoluer au gré des évolutions technologiques.

Après ce rapide aperçu sur les drones, nous présentons maintenant quelques éléments sur les véhicules connectés.

4.2/ VÉHICULES CONNECTÉS

Dans cette section, nous proposons tout d'abord quelques définitions puis les différents niveaux d'autonomie sont présentés. Nous listons ensuite les propositions des constructeurs. Enfin nous évoquons les enjeux mais aussi les verrous techniques et les limites actuelles.

4.2.1/ QUELQUES DÉFINITIONS

Afin d'éclaircir le propos, nous allons expliquer ce que nous entendons par véhicule connecté. Pour être complet, il faut parler d'un véhicule intelligent, communicant et connecté. C'est une entité mobile disposant de moyens de communication qui permettent d'accéder à des services tiers pouvant être indépendants du constructeur et facilitant la vie du conducteur. La plupart du temps, ce type de véhicule embarque également une intelligence logicielle chargée d'orchestrer l'utilisation de ces services et de piloter l'interface utilisateur. Il y a plusieurs types de communications possibles. Par exemple : communication inter-véhicule (V2V) ou avec une infrastructure (V2I). Plus généralement, on parle de communication V2X ou C2X en anglais ([Eskandarian, 2012]).

Toutes les initiatives visant à connecter les véhicules entre eux et avec des infrastruc-

tures de services ou routières sont soutenues la plupart du temps par des subventions nationales ou européennes. Par exemple : SafeSpot, CVIS, Drive C2x, Sevecom, Scoref, SimTD, etc. Tous ces projets ont pour objectif à terme de créer, de tester et de standardiser de nouveaux protocoles, formats de messages, etc. Il existe aussi des regroupements européens pour la définition de ces standards comme le consortium Car2Car.

Toutes ces évolutions techniques ont pour but d'aboutir à un véhicule de plus en plus autonome et ce, pour plusieurs raisons. La voiture est moins synonyme de plaisir qu'hier. En effet, la jeune génération est de moins en moins attirée par la voiture telle que nous la connaissons depuis les années 50. Le modèle américain de la possession d'un objet participant au prolongement de son ego n'est plus en vogue. D'ailleurs les jeunes retardent voire ne passent plus l'examen du permis de conduire (surtout s'ils habitent en ville). De plus, la voiture est aujourd'hui perçue comme une contrainte (coût, temps, pollution, insécurité) et les habitudes changent progressivement (co-voiturage, auto-partage).

4.2.2/ NIVEAUX D'AUTONOMIE

Pour classer les véhicules selon le niveau d'autonomie géré par leur ordinateur de bord, il existe deux échelles : l'échelle américaine, qui s'étend de 0 à 4, et l'échelle européenne, plus précise puisqu'elle s'étend de 0 à 5.

La figure 4.3 présente l'actuelle norme SAE. Au-delà de la classification, c'est aussi une déclaration d'intention de la part des acteurs de la filière et une feuille de route pour les années à venir.

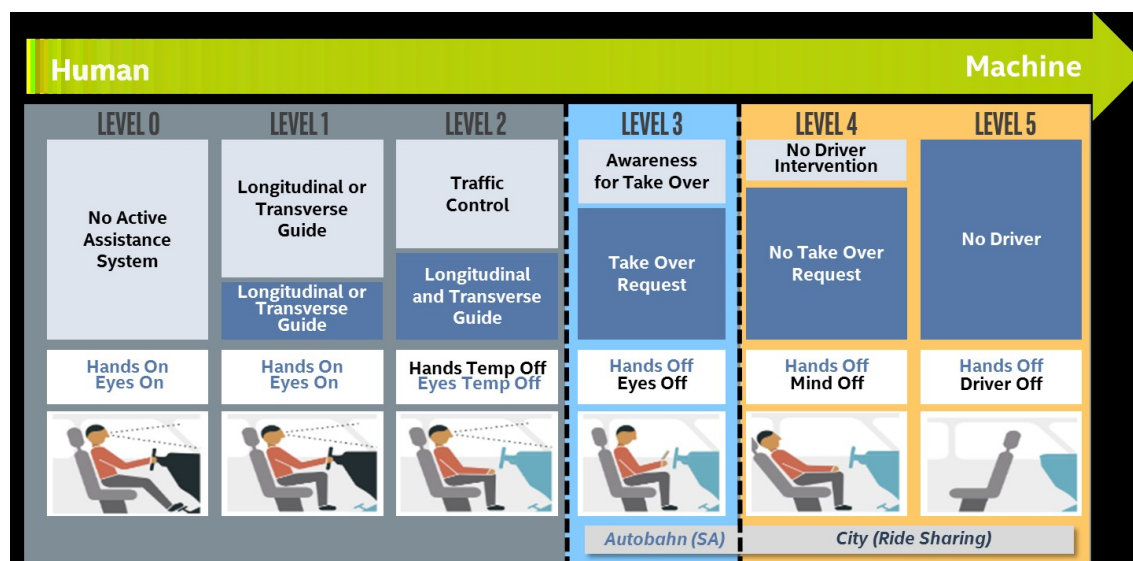


FIGURE 4.3 – Niveaux d'autonomie, norme SAE (<https://iq.intel.fr/voitures-autonomes-la-voie-de-lavenir/>)

- **Niveau 0** : le conducteur garde en toutes circonstances le contrôle des fonctions essentielles du véhicule telles que le freinage, la direction et l'accélération.
- **Niveau 1** : l'ordinateur de bord peut assister le conducteur en prenant le contrôle sur la direction ou la vitesse. Un exemple classique : le régulateur de vitesse.
- **Niveau 2** : l'ordinateur de bord est en mesure de gérer la direction et la vitesse en même temps. Les dispositifs d'assistance au stationnement sont à ce niveau. Le

conducteur reste toutefois responsable du bon déroulé des opérations.

- **Niveau 3** : l'ordinateur de bord prend le contrôle de toutes les fonctions du véhicules, mais uniquement dans des situations prédéfinies (par ex. embouteillages). Le conducteur doit pouvoir reprendre la main à tout moment.
- **Niveau 4** : le véhicule est capable de se déplacer seul sans son propriétaire, par exemple pour aller se garer tout seul sans personne à bord.
- **Niveau 5** : le véhicule peut s'auto-piloter en toutes circonstances et prendre seul des décisions critiques.

Comme le souligne [Silberg et al., 2012, Silberg et al., 2013, Hudda et al., 2013, Flury-Hérard et al., 2015], il n'y a pas de réelle autonomie si les véhicules ne peuvent pas coopérer ou collaborer avec d'autres entités. En effet, même si les véhicules sont amenés à utiliser de plus en plus de capteurs afin de modéliser leur environnement immédiat, ils ne pourront se mouvoir sans encombre qu'en échangeant de l'information notamment sur leur position, leur vitesse et leur direction. La Google Car est un véhicule "autonome" équipé de nombreux capteurs mais il ne communique pas avec son environnement. Nous pouvons considérer qu'il s'agit d'une exception et d'un dispositif de recherche expérimentale plus qu'un véritable futur véhicule de série.

La figure 4.4 propose différents cas d'application pour les véhicules intelligents en fonction du degré d'autonomie et de coopération. Tous ces systèmes électroniques d'assistance sont déjà en cours de développement chez la plupart des constructeurs ou font l'objet de recherches scientifiques.

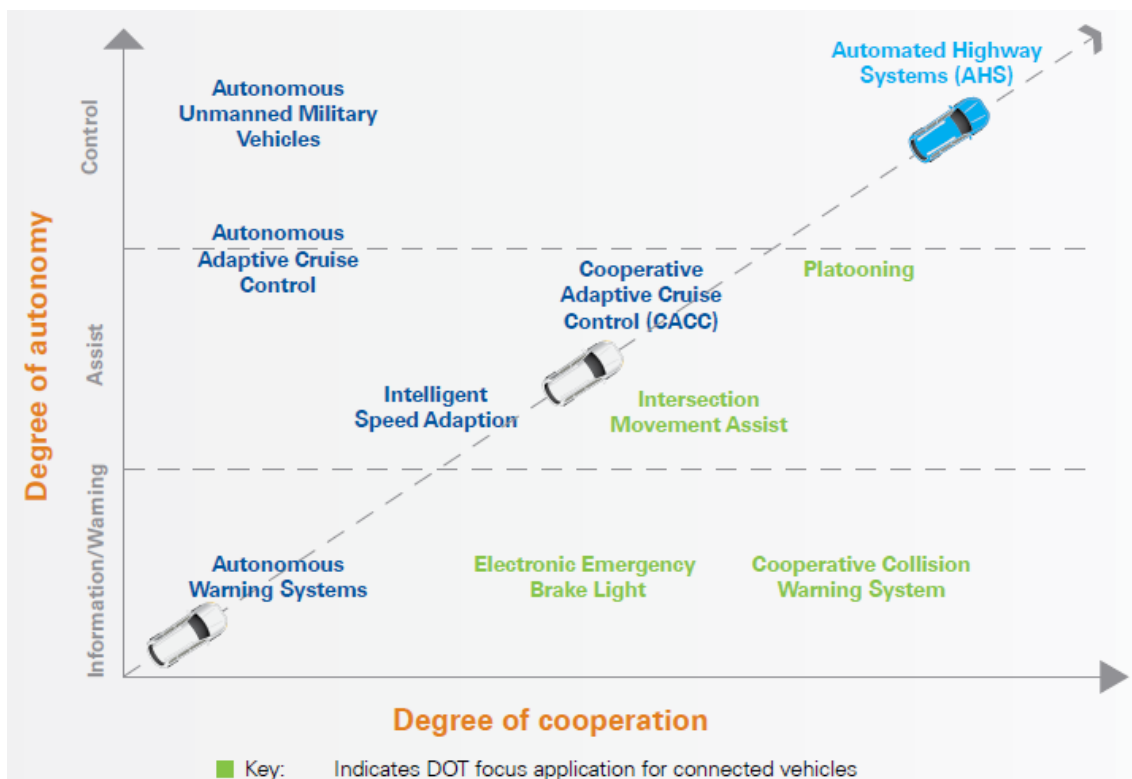


FIGURE 4.4 – Cas d'application pour les véhicules intelligents en fonction du degré d'autonomie et du degré de coopération ([Silberg et al., 2012])

Cela conduit évidemment à des évolutions importantes de l'architecture du véhicule tant sur le plan de l'électronique embarqué que des moyens débarqués et de communication

avec l'environnement. La figure 4.5 propose une synthèse des évolutions des architectures. Nous passons progressivement d'un modèle centré sur le conducteur à un modèle centré sur le véhicule lui-même. C'est en quelque sorte le pendant de la norme SAE évoquée plus haut. Nous pouvons constater l'importance croissante des algorithmes embarqués pour gérer des fonctions plus intelligentes prenant en considération de plus en plus de paramètres environnementaux et devant faire appel à des comportements plus adaptatifs. Le conducteur est progressivement relayé au rôle de passager.

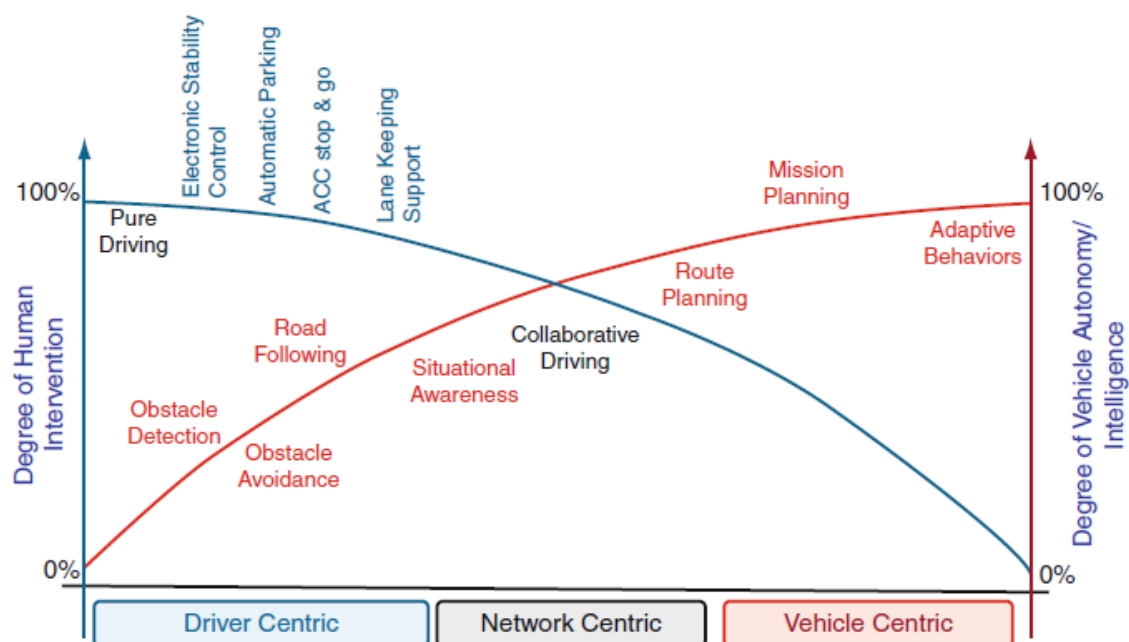


FIGURE 4.5 – Evolution des architectures automobiles ([Ibañez-Guzman et al., 2012])

Nous sommes désormais au milieu du gué. Les équipements embarqués sont de plus en plus intelligents. Les moyens de communication permettent maintenant des échanges structurés pour des fonctions de plus en plus complexes.

4.2.3/ PROPOSITIONS DES CONSTRUCTEURS

Tous les constructeurs proposent désormais des aides à la conduite (ADAS Advanced Driver-Assistance Systems) de plus en plus intelligents. Ce sont des systèmes de sécurité active d'information ou d'assistance du conducteur pour :

- Eviter l'apparition d'une situation dangereuse risquant d'aboutir à un accident ;
- Libérer le conducteur d'un certain nombre de tâches qui pourraient atténuer sa vigilance ;
- Assister le conducteur dans sa perception de l'environnement (détecteurs de dépassement, de piéton, etc) ;
- Permettre au véhicule de percevoir le risque et de réagir de manière anticipée par rapport aux réflexes du conducteur.

Le législateur modifie périodiquement la réglementation et oblige les constructeurs à intégrer certains systèmes de sécurité. Cela avait commencé dans les années 70 avec la ceinture de sécurité. Aujourd'hui ce sont les systèmes d'aide au freinage (ABS) ou de

maintien de la trajectoire (ESP) qui sont devenus obligatoires. D'années en années, au fil des évolutions technologiques, il est certain que la plupart des systèmes de sécurité en option seront montés de série dans les véhicules.

Sans faire un inventaire exhaustif, nous pouvons tout de même lister plusieurs catégories de systèmes d'assistance :

- Régulation et limitation de vitesse (passif, actif et intelligent) ;
- Aide au freinage (ABS) et au maintien de trajectoire (ESP) ;
- Alerte :
 - Détection de piétons, de fatigue ;
 - Franchissement involontaire de ligne (AFIL) ;
 - Détection de véhicules présents dans les angles morts ;
 - Alerte de Distance de Sécurité (ADS) ;
 - Fonction d'Indication de Limitation de Vitesse (ILV) à partir d'une lecture des panneaux de signalisation routière.
- Systèmes actifs : anti-collision, freinage automatique d'urgence ;
- Confort : aide à la navigation, avertisseur de zone dangereuse (travaux, accident, embouteillage), affichage tête haute des informations (vitesse, limitation de vitesse, etc) permettant de ne pas avoir besoin de détourner son regard de la route ;
- Aide au stationnement : radars et caméra de recul (détection de distance arrière et d'obstacles), mesure de place disponible et stationnement automatique ;

Les évolutions en cours portent essentiellement sur l'amélioration de ces systèmes existants. Cependant nous pouvons observer l'importance croissante des communications (entre véhicules et avec des infrastructures) pour augmenter l'efficacité des systèmes et permettre le développement de services connectés.

Des constructeurs comme Tesla ou Volvo ont proposé très tôt des systèmes de conduite semi-autonome. Dans un avenir proche, la plupart des acteurs de la filière automobile proposeront des aides à la conduite dans des conditions précises comme la conduite sur autoroute ou dans les embouteillages. Ces aides du niveau 3 de la classification SAE permettront de définir les bases techniques des futurs véhicules des niveaux supérieurs.

4.2.4/ ENJEUX

Différentes études [Silberg et al., 2012, Silberg et al., 2013, Hudda et al., 2013, Flury-Hérard et al., 2015] décrivent comment ces véhicules intelligents et autonomes changeront notre vie à travers trois objectifs principaux : améliorer la sécurité routière (perception augmentée, localisation et conduite coopérative), accroître la qualité de vie (en permettant de soulager le conducteur, services aux passagers) et permettre un accès à la mobilité à tout un chacun (dans le cadre des approches multimodales avec une coordination des moyens de transport).

Lors de la conférence Mobilis 2014 à Strasbourg, Franck Cazenave (Bosch) a évoqué ces objectifs en détaillant l'accès à la mobilité (cf. figure 4.6) notamment pour accroître l'autonomie des personnes âgées et des personnes à mobilité réduite.

Il est intéressant d'ajouter que le pilier "temps retrouvé" est particulièrement porteur. En effet, de nombreuses entreprises cherchent à valoriser cette libération en proposant des services commerciaux. C'est le cas des GAFA (Google Apple Facebook Amazon) qui cherchent à s'intégrer au cœur du véhicule pour mieux vendre leurs services, tout en



FIGURE 4.6 – Les 4 piliers de l'impact de la voiture autonome (Cazenave, Mobilis 2014)

recupérant des informations sur les habitudes du conducteur et des passagers.

4.2.5/ QUELQUES VERROUS

Pour rendre les véhicules intelligents, il faut les équiper de capacités accrues de perception, d'interprétation et de communication ainsi que d'une intelligence embarquée capable de prendre les bonnes décisions en fonction du contexte. Il existe de nombreux verrous dans ces différents domaines.

Pour les capteurs, il y a deux écoles : une approche traditionnelle où toute la chaîne est maîtrisée par un seul intervenant et une approche "boîte noire" qui est principalement adoptée par les constructeurs (les capteurs intelligents étant développés par les équipementiers). Ces capteurs servent à la détection et à la reconnaissance des obstacles (véhicules, piétons, etc). Il peut y avoir des approches télémétriques (lidar, radar) bien adaptées pour la détection et des approches visuelles. Toutes ces approches sont sensibles à la richesse de l'apprentissage.

En effet, dans le cadre du développement des véhicules autonomes, il faut intégrer des programmes intelligents qui vont venir interpréter les observations. Un des problèmes principaux est la variabilité des scènes. L'éclairage, l'orientation des objets, leurs déformations potentielles, les occultations, leurs mouvements peuvent empêcher une interprétation fiable où la sécurité et l'intégrité physique des passagers sont une obligation. L'interprétation des scènes passera certainement par une analyse de plus haut niveau faisant intervenir l'ensemble des acteurs de la scène (communication inter-véhicules).

Pour tenter de prédire les trajectoires et les comportements des autres véhicules, il est nécessaire d'avoir des informations détaillées sur la route. Cependant c'est une opération difficile si on ne connaît pas les intentions des conducteurs (ego-véhicule et obstacles). Dans le contexte des véhicules autonomes, le problème est simplifié si la trajectoire de l'ego-véhicule est planifiée, si les véhicules sont communicants et envoient leur trajectoire. Mais il reste toujours les obstacles non-coopératifs dont les trajectoires sont diffi-

ciles à prédire (piétons et cycles notamment).

Un autre point important est donc la localisation. C'est un élément clé de la conduite autonome qui permet de déplacer le problème de l'analyse de la scène. Là encore, les collaborations inter-véhicules permettront de rapidement créer et mettre à jour des cartes contextualisées quasi temps réel. Cependant il existe aussi quelques verrous comme :

- Disponibilité des cartes, formats compatibles avec l'embarqué et le temps-réel ;
- Mises à jour des cartes par plusieurs opérateurs (véhicules collecteurs, cartographes, cadastre, capteurs embarqués, etc) nécessitant une standardisation ;
- Evolution des bases de données spatio-temporelles.

Evidemment il est impossible de mettre en œuvre toutes ces évolutions si les communications sont difficiles ou inexistantes. La limitation de la bande passante peut compromettre l'envoi et la réception des messages nécessaires aux différents échanges. Il y a un risque de perte de messages ou de dissémination des données. De plus, il faudra arbitrer les communications V2V ou V2I (selon le contexte ou l'environnement physique). Enfin se pose plus largement la question de la sécurité des réseaux pour préserver la confidentialité et protéger la vie privée des usagers.

4.2.6/ LIMITES ACTUELLES

Outre les soucis sur la protection des données, la sécurisation des échanges (cybersécurité) et plus généralement les possibilités techniques, il y a deux limites au déploiement à ne pas négliger : l'acceptation du changement de paradigme par le grand public ou les problèmes de responsabilité en cas d'accident.

Dans une étude CSA de janvier 2014³, il apparaît que 71% français plébiscitent les véhicules connectés mais 78% pensent également que cela va nuire à la vie privée. De plus il n'est pas encore facile d'accepter de ne plus "piloter" son véhicule. D'ailleurs les accidents constatés dont certains mortels (avec des véhicules Tesla ou Uber plus récemment), même s'ils sont peu nombreux et restent exceptionnels, marquent les esprits et peuvent dissuader les gens de franchir le pas.

En même temps, il est nécessaire de faire évoluer la législation pour déterminer la chaîne de responsabilité et l'impact sur les systèmes d'assurances. Prenons la Convention de Vienne⁴ sur la circulation routière qui est un traité international conçu pour faciliter la circulation et pour améliorer la sécurité routière en harmonisant la réglementation routière entre les parties contractantes. Cette convention a été admise lors de la Conférence sur la circulation routière du Conseil Economique et Social des Nations Unies le 7 octobre 1968, signée à Vienne le 8 novembre 1968 et entrée en vigueur le 21 mai 1977. Elle fut complétée pour harmoniser les signaux et symboles routiers afin de faciliter la circulation routière internationale et accroître la sécurité sur la route.

Force est de constater que certaines de ces règles peuvent faire obstacle à la circulation de véhicules autonomes notamment celles qui imposent la présence d'un conducteur devant avoir le contrôle de son véhicule et d'en rester maître. Cela signifie que pour rester conforme à cette convention, dans le cas d'un véhicule autonome, le conducteur doit être en mesure de surveiller les fonctions du véhicule et de reprendre son contrôle

3. <https://www.csa.eu/media/1247/opi20131112-internet-des-objets-les-chiffres-cles.pdf>

4. <https://www.admin.ch/opc/fr/classified-compilation/19680244/>

à tout moment. Ces dispositions apparaissent comme une contrainte juridique dans le cadre des projets de développement des véhicules autonomes.

Pour autant, du point de vue de la sécurité du système de circulation, ces dispositions semblent pragmatiques et prudentes dans la mesure où il est difficile d'imaginer un système d'intelligence artificielle suffisamment fiable pour se substituer à un conducteur, du moins, pour le moment.

4.3/ CE QU'IL FAUT RETENIR

Dans ce chapitre, nous avons pu évoquer quelques généralités sur les drones et les véhicules connectés. Ces deux domaines d'applications sont très riches et dynamiques.

La figure 4.7 résume les éléments nous paraissant fondamentaux sur les drones. Ils sont de tout type (aériens, marins ou terrestres), télé-opérés ou autonomes. Leurs domaines d'utilisation sont nombreux comme l'exploration, l'inspection ou la cartographie. Ils sont plus efficaces, plus rapides et moins coûteux qu'une intervention humaine (sans risque d'accidents graves ou mortels). Des projets de collaboration se développent entre des drones autonomes de même type (homogènes) ou de type distinct (hétérogènes). Il reste néanmoins des limites et des verrous sur les capteurs, la gestion de la batterie et plus généralement sur l'architecture embarquée. Enfin il y a des contraintes réglementaires sur leur utilisation (surtout en milieux ouverts).

La figure 4.8 résume les éléments nous paraissant fondamentaux sur les véhicules connectés. Il s'agit d'un véhicule intelligent, communicant avec son environnement (inter-véhicules ou avec une infrastructure), porteur de nouveaux services connectés. Il y a 5 niveaux d'autonomie permettant aux constructeurs de se positionner les uns par rapport aux autres et de stimuler la recherche en définissant un objectif moyen/long terme. Les applications sont de plus en plus nombreuses et elles auront un impact sur la sécurité routière. Les véhicules connectés et autonomes permettront d'accroître la qualité de vie et donner un accès à la mobilité pour tous. Il y a aussi des verrous techniques (liés aux capteurs et aux architectures intelligentes embarquées ainsi que les problèmes liés à la gestion des communications). D'autres limites dans l'utilisation existent car il faut accepter de changer sa façon d'appréhender le déplacement et faire confiance à la machine. Des problèmes réglementaires au niveau national et européen sont encore à résoudre.

Pour ces deux exemples, il apparaît que la collaboration est un concept d'importance et peut être la clé du succès de certaines applications. Grâce aux échanges et à de nouveaux moyens d'extraire des données, un nouveau genre d'intelligence distribuée peut être étudiée et développée. La mise en œuvre de la collaboration est alors une des difficultés majeures de la conception de ces applications. En effet les réseaux d'objets intelligents sont des systèmes complexes dans la mesure où leurs dynamiques sont déterminées par un nombre considérable d'échanges.

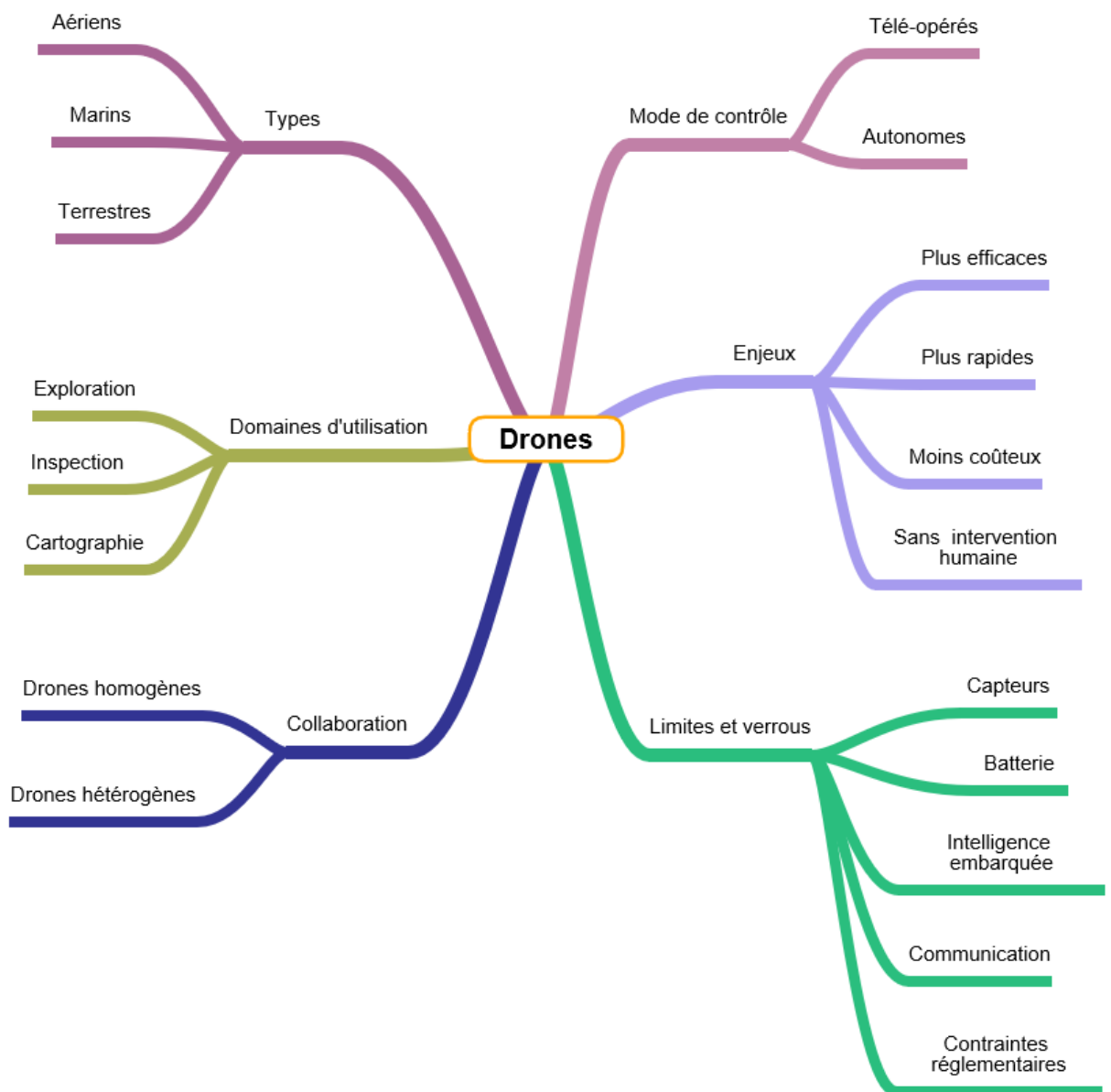


FIGURE 4.7 – Ce qu'il faut retenir : Drones

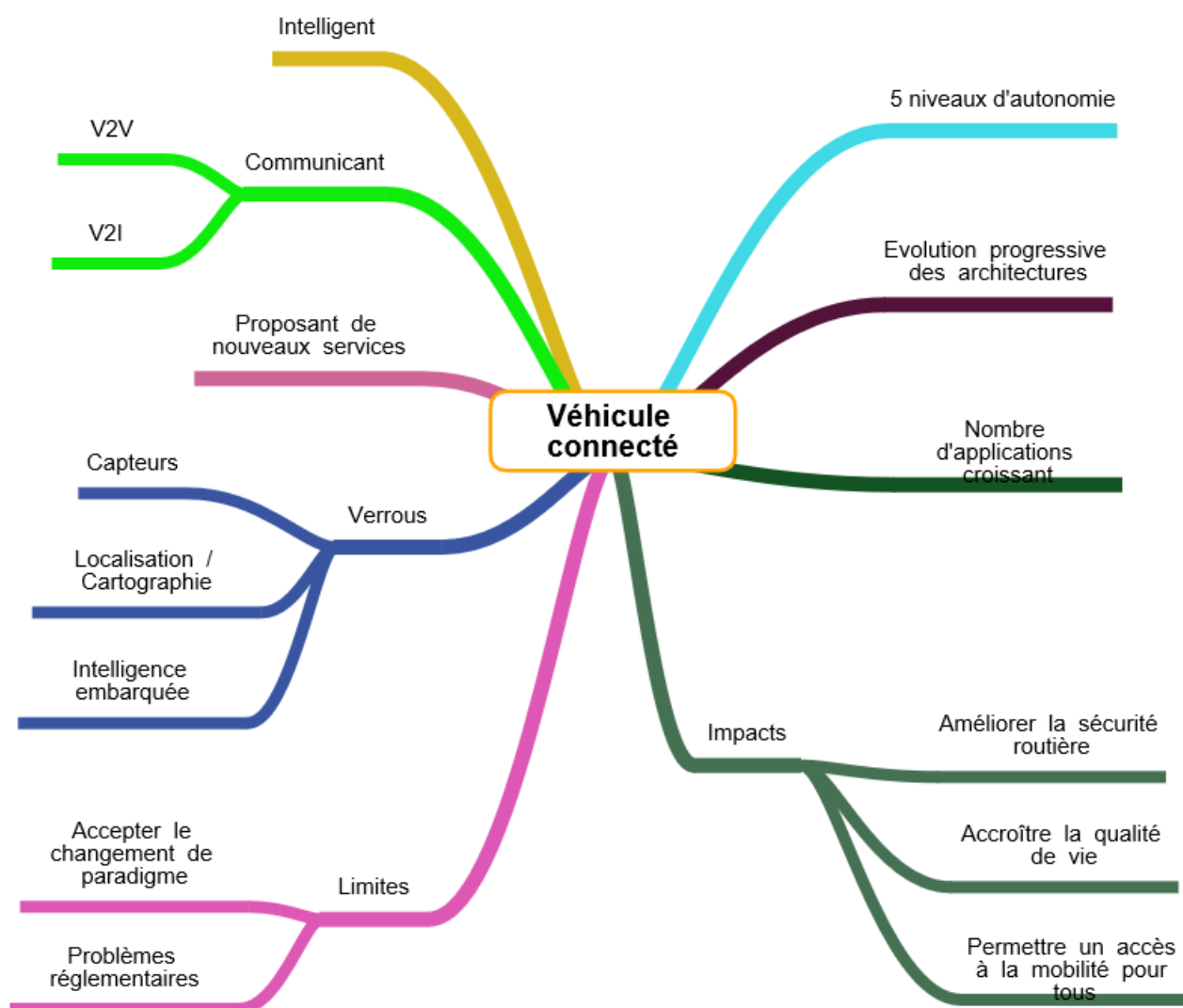


FIGURE 4.8 – Ce qu'il faut retenir : Véhicules connectés



HACCA (HYBRID ARCHITECTURE FOR COLLABORATIVE COMMUNICATING AGENT) : ANALYSE ET CONCEPTION

ANALYSE ET VERROUS

Sommaire

5.1 Analyse	69
5.1.1 Cadre d'application	69
5.1.2 Problématiques	70
5.1.3 Pistes de réflexion	72
5.2 Verrous	72
5.2.1 Méthode d'analyse	73
5.2.2 Architecture	73
5.3 Exemple Fil rouge	76
5.3.1 Présentation	76
5.3.2 Implémentation NetLogo	77
5.4 Ce qu'il faut retenir	80

Dans ce chapitre, nous discutons et analysons les enjeux de la collaboration et la manière de l'aborder dans le contexte des entités intelligentes dans les systèmes multi-agents physiques et logiciels. Cette approche méthodique conduit à des pistes de réflexion permettant d'identifier les verrous scientifiques et d'introduire les contributions via un exemple fil rouge.

5.1/ ANALYSE

Dans un premier temps, nous analysons le sujet de l'étude au travers des cas d'application Drones et Véhicules connectés, deux exemples d'entités mobiles intelligentes voire autonomes afin de mettre en évidence les problématiques principales. Puis nous les structurons en accord avec ce que nous avons déjà évoqué dans les chapitres précédents pour dégager des pistes de réflexion.

5.1.1/ CADRE D'APPLICATION

D'une manière générale, ces deux exemples d'entités mobiles, autonomes ou tendant à le devenir, vont devoir faire preuve d'intelligence et de réflexion pour répondre à leurs objectifs. Seulement, au regard de la complexité du monde qui nous entoure, il semble inconcevable que ces entités soient en mesure de satisfaire les besoins par elles-mêmes uniquement, sans connaissance ou découverte de leur environnement, sans prise de

conscience des autres entités mobiles ou immobiles autour d'elles, sans prise en compte des difficultés potentielles pouvant émerger de l'environnement direct ou indirect, sans capacité de prévision des aléas, sans anticipation.

Pour les drones, compte tenu de l'extension des périmètres d'exploitation, l'autonomie sera de rigueur. Il faudra donc développer, en plus des capacités physiques de gestion des déplacements automatiques, une intelligence artificielle qui devra prendre des décisions pour mener à bien les missions qui lui seront attribuées.

Pour les véhicules connectés, tout dépendra du niveau d'autonomie. Cela pourra aller d'un véhicule simplement connecté avec un conducteur au volant à un véhicule totalement autonome en passant par toutes les étapes intermédiaires comme décrit dans le chapitre précédent. Là aussi, on pourra parler de mission même si celles-ci seront le plus souvent dictées par l'être humain utilisant le moyen de transport.

5.1.2/ PROBLÉMATIQUES

Comme nous l'avons montré au [chapitre 2](#), la collaboration est un paradigme intéressant pour permettre la résolution de problèmes complexes pour les agents ainsi que pour augmenter leurs connaissances de leur environnement, voire permettre l'élaboration de plans d'actions inédits. En effet, les missions qu'ils doivent remplir seront peut-être impossibles à terminer sans une aide extérieure. Plus l'agent va être autonome dans ses actions et ses processus de décision et plus il va nécessiter des échanges avec les autres entités de son environnement pour piloter son action, adapter son comportement. Nous allons structurer cette réflexion en utilisant la notion de quadrants ([Tranier, 2007]) abordée dans le [chapitre 1](#). Cette représentation permet de positionner les différentes étapes au sein des 4 dimensions constitutives d'un agent. De plus, en combinant des aspects internes (esprit et corps) à des aspects externes et sociaux (espace brut et culture), cette méthode permet de représenter plus largement comment la collaboration s'inscrit dans un processus social global. La figure 5.1 illustre ce propos.

A l'origine du processus de collaboration d'un agent, il y a tout d'abord la conscience d'une limitation de ses capacités de décision et d'action, soit par manque d'information, soit par manque de temps. Cela implique un certain niveau de réflexion sur ses actes et ses objectifs. Il y a donc une dynamique interne de l'agent, une capacité d'introspection et d'interrogation sur son état, la possibilité de réaliser ses objectifs ainsi que leur niveau de réalisation. Il s'agit de l'**ESPRIT** de l'agent. Cette prise de conscience constitue la motivation principale afin de trouver une solution au problème rencontré. Selon le degré d'implication (si l'agent est à l'origine de la prise de conscience ou participant), cela permet à l'agent de résoudre un problème, de participer à une oeuvre commune (ou un problème d'un autre agent) ou tout simplement d'aider l'autre sans attente d'un retour potentiel comme nous l'avons vu dans le [chapitre 2](#).

Cette modification de l'état d'esprit se traduit par une modification du comportement de l'agent ainsi que d'une recherche d'interaction avec les autres agents ou entités de l'environnement. Les comportements fondamentaux de l'agent ne changent pas et répondent aux besoins de satisfaction des objectifs. Cependant des comportements plus spécifiques sont mis en œuvre pour aider à la réalisation des besoins initiaux. Ces comportements spécifiques peuvent être une recherche des autres agents dans l'environnement ainsi que l'exécution de processus de communication adaptés. Nous agissons ici sur le **CORPS** de l'agent.

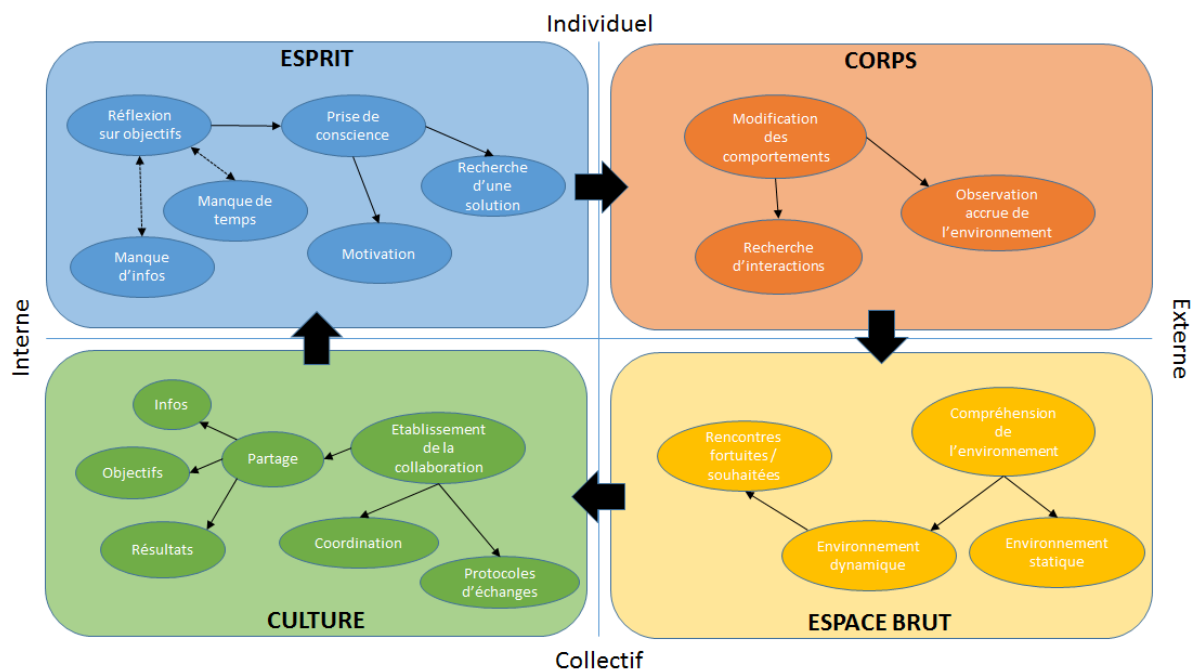


FIGURE 5.1 – Réflexion sur la collaboration à l'aide des quadrants

Au fur et à mesure de la découverte de l'environnement statique (objets inanimés ou éléments physiques de l'environnement) et dynamique (autres agents immobiles et mobiles), il faut organiser progressivement ces informations. Il s'agit de transcrire la perception de cet environnement sous une forme interprétable pour l'agent qui peut ainsi avoir une représentation symbolique de l'**ESPACE BRUT** dans lequel il évolue. Il est important de noter que cela ne constitue pas la réalité absolue de l'environnement, seulement un ensemble cohérent et compréhensible par l'agent, autrement dit, une vue subjective. Cela sert à alimenter les processus de réflexion et de décision au sein de l'**ESPRIT** de l'agent et qui peuvent influencer le **CORPS** de celui-ci.

Cela pourrait d'ailleurs s'arrêter à une simple reconnaissance de l'environnement, les autres entités (mobiles ou immobiles) étant finalement de simples constituants de celui-ci. Cependant ce sont des entités ayant des objectifs communs, propres ou antagonistes. Ce fait entraîne la nécessité d'agir et d'interagir. La gestion des processus d'interaction doit se faire avec l'objectif de réussir la collaboration pour atteindre ses objectifs propres. Il faut donc structurer les échanges pour favoriser leur réalisation. En cas de découverte d'autres agents et suite à l'établissement d'un contact, il faut expliquer ce que l'on attend de cette collaboration.

Nous entrons désormais dans le quadrant **CULTURE**. Dans la démarche d'établissement de la collaboration, après la prise de conscience, la modification des comportements pour chercher les interactions, la représentation mentale de l'environnement et les prémices d'échanges avec les autres agents, nous sommes maintenant dans le processus collaboratif concret avec le partage d'objectifs, d'informations de contextualisation et des résultats à atteindre. Cela représente le préalable de la collaboration : l'engagement mutuel et la conception partagée de la situation et de l'objectif. Maintenant la collaboration va s'appuyer sur un protocole de coordination et de communication afin de suivre la mise en oeuvre et le déroulement des actions appropriées nécessaires à la réalisation de l'objectif

convenu.

Toute nouvelle action ou nouvelle information influence l'**ESPRIT** de l'agent qui va, à chaque nouvelle mise à jour, remettre en question son raisonnement et ses comportements. Car il ne faut pas oublier qu'une collaboration est un processus choisi. Il y a une liberté de participation et d'action. L'agent doit s'interroger sur le bénéfice de son travail avec l'autre.

5.1.3/ PISTES DE RÉFLEXION

Les problématiques exposées dans la section précédente permettent de dégager quelques pistes de réflexion.

ESPRIT – Il faut bien organiser les informations au sein de l'agent afin de permettre une bonne interprétation du contexte interne et favoriser la prise de conscience de l'intérêt de rechercher une collaboration.

CORPS – Il paraît évident que pour accompagner la bonne gestion des processus collaboratifs, il est nécessaire de bien déterminer les comportements individuels à adopter en relation avec les informations enregistrées par l'agent.

ESPACE BRUT – Les perceptions et les interactions doivent pouvoir constituer la représentation du monde de l'agent. Celle-ci est cruciale pour analyser correctement une situation et permettre à l'agent de réagir de manière appropriée.

CULTURE – L'accent est mis sur la gestion des communications entre agents (coordination et protocoles) dans le but de mettre en œuvre la collaboration effective.

5.2/ VERROUS

La littérature montre une réflexion importante portant sur les différentes facettes de la collaboration dans les systèmes multi-agents. En effet, comme il est décrit dans le **chapitre 3**, de nombreux travaux ont déjà été menés pour caractériser les aspects importants d'un processus collaboratif comme la coordination d'actions ou la communication entre agents. Cependant, à notre connaissance, aucune d'entre elles ne donne une vision globalisante. Agentifier un processus collaboratif à partir d'un point de vue global constitue aujourd'hui un véritable verrou.

En résumé, quelque soit la nature intrinsèque de l'agent et du système, il s'agit de passer d'un problème à un modèle pouvant intégrer des comportements collaboratifs. Pour lever ce verrou, il est important d'adopter un point de vue méthodologique pour poser les principales caractéristiques de ces échanges. Cela passe donc par des outils de questionnement et d'analyse mais également par la définition d'une architecture portant l'ensemble des concepts et les intégrant.

Pour les aspects d'architecture et dans la continuité de la section précédente, voici donc les principaux axes à étudier : représentation du monde (vu par l'agent), organisation des données (au sein de l'agent), gestion des comportements (de l'agent), communication et protocoles (avec les autres agents).

5.2.1/ MÉTHODE D'ANALYSE

Considérer la collaboration dans son ensemble implique, lors de la modélisation d'un phénomène, de pouvoir définir les comportements de l'agent qui vont être susceptibles de nécessiter une aide extérieure voire un enrichissement du contexte d'exécution au contact des autres agents (apport de nouvelles connaissances, de nouvelles compétences ou de nouvelles missions). Cela passe par une analyse de la nature de l'agent, ses capacités d'interaction avec son environnement, ses possibilités de communication ainsi que ses comportements de base (définis par sa nature intrinsèque) et ses comportements liés à la réalisation de ses objectifs (si c'est un agent un peu plus complexe disposant d'une plus grande palette de comportements).

Il ne s'agit pas de prévoir toutes les interactions possibles (ceci est impossible car nous nous intéressons à un système complexe où l'ensemble des interactions rend toute prédiction impossible) mais de se focaliser sur les comportements qui vont pouvoir donner lieu à des collaborations. Il est donc nécessaire d'établir les liens entre les données et le monde de l'agent tel qu'il le perçoit et les mettre en rapport avec les informations et les caractéristiques de l'autre agent communicant.

En accord avec ce que nous avons montré au [chapitre 2](#), nous considérons un processus collaboratif comme étant un tout. Le problème doit donc être pris dans son ensemble : AGENTS, INFORMATIONS, COMPORTEMENTS, COMMUNICATIONS, OBJECTIFS, RESULTATS.

5.2.2/ ARCHITECTURE

Afin de modéliser des échanges collaboratifs entre entités mobiles, et plus particulièrement lorsqu'une caractéristique "temps réel" est requise (où l'on recherche une interactivité forte ou une certaine instantanéité), une architecture hybride des agents est essentielle : une partie réactive pour répondre aux stimuli de l'environnement dans un délai maîtrisé ; une partie délibérative pour la prise de décision, la planification des actions à réaliser et la collaboration. Cette architecture doit être jumelée à des moyens de stockage pour construire une base de connaissances. On doit également pouvoir enregistrer l'historique des interactions avec les autres agents ainsi que les caractéristiques de l'environnement : la façon d'enregistrer les informations (représentation du monde, objectifs à atteindre, interactions entre les agents) et le protocole de communication à utiliser dépendront de la plateforme de développement multi-agents.

Michon ([Michon, 1985]) propose une architecture pour les systèmes temps réels avec plusieurs couches en fonction de la criticité (cf. figure 5.2) : un niveau de contrôle pour les actions immédiates et automatiques (quelques millisecondes), un niveau de manœuvre pour les actions rapides et contrôlées (quelques secondes) et un niveau stratégique pour des plans plus généraux. Cette approche est intéressante pour servir de base à une architecture hybride dans un contexte d'agents mobiles.

5.2.2.1/ REPRÉSENTATION DU MONDE

Cette représentation est nécessaire pour aider l'agent à se positionner dans son environnement et le comprendre afin d'adapter son comportement et d'anticiper le cas échéant

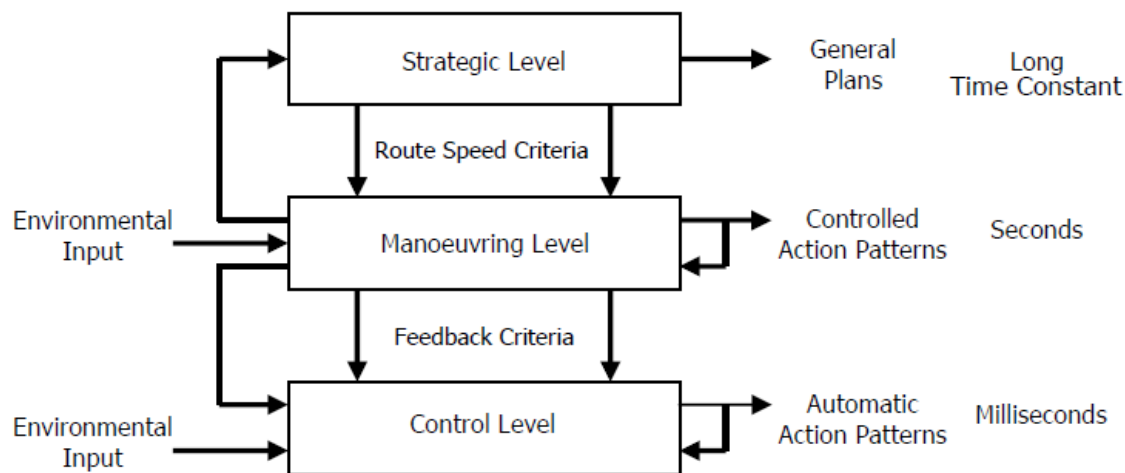


FIGURE 5.2 – Architecture proposée par Michon ([Michon, 1985])

des changements de contexte.

Nous pouvons distinguer différentes méthodes possibles : de simples variables pour les différents états de l'agent, un graphe (orienté ou non) pour la représentation mentale, une matrice pour la représentation de l'environnement ou une structure *ad hoc*. Ces structures de données doivent être choisies en fonction des environnements modélisés. Par exemple, l'emploi d'une matrice est judicieux pour le stockage d'environnement de forme rectangulaire mais l'utilisation d'un graphe permettra une meilleure matérialisation des liens entre les entités. Nous pouvons également créer une structure globale pour enregistrer toutes les informations liées : représentation du monde, entités rencontrées, objectifs à atteindre, etc. Une structure englobant toutes ces caractéristiques semble être intéressante pour regrouper toutes les informations dans un seul ensemble. De plus, les liens entre les actions, les connaissances, l'historique des communications, les agents rencontrés, etc peuvent être créés pour constituer de véritables grappes d'informations qui peuvent ensuite être échangées d'un agent à l'autre.

Le projet SafeSpot¹ (cf. figure 5.3) présente un bon exemple de carte dynamique pour structurer la connaissance du monde avec des éléments statiques comme le réseau routier, des éléments dynamiques avec des informations temporaires, etc.

5.2.2.2/ ORGANISATION DES DONNÉES

Selon le même principe de considération du processus collaboratif dans son ensemble, l'organisation des données doit aussi permettre de faciliter les échanges. Il est donc nécessaire de contextualiser chaque parcelle d'information. L'idéal est de mettre en relation les données internes de l'agent et sa propre représentation du monde. On peut ainsi supposer que la représentation du monde peut être incluse dans la somme d'informations que connaît l'agent.

La réponse dépend toutefois de la nature des agents utilisés et du cas d'application. Différents types de représentations (connaissances et faits) existent mais il ne sera peut-être pas pertinent de mélanger toutes les informations. Cependant il est évident qu'une

1. <http://www.safespot-eu.org/>

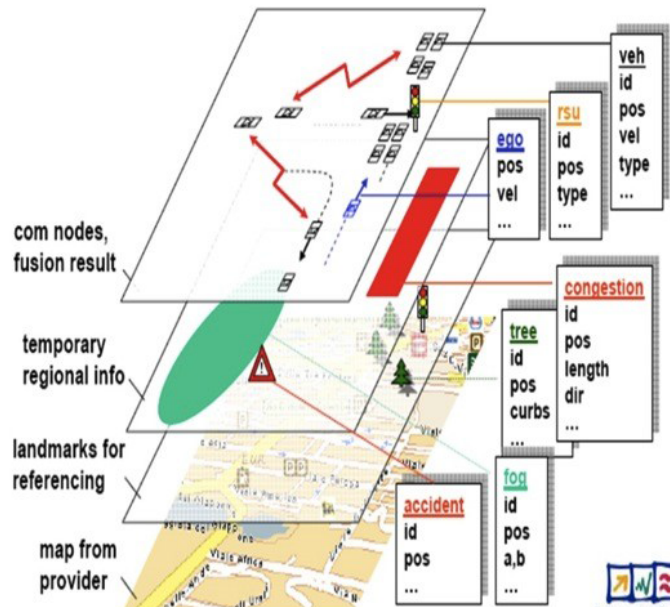


FIGURE 5.3 – Exemple de carte dynamique du projet SafeSpot (<http://www.safespot-eu.org>)

bonne structure de données permet une lecture facile des informations en vue d'améliorer la planification et le suivi des objectifs. Ce travail est d'autant plus nécessaire dans le contexte des échanges d'informations (avec des mises à jour réciproques des bases de faits et des connaissances des entités communicantes). La structure choisie pour les données devra intégrer toutes ces contraintes.

5.2.2.3/ GESTION DES COMPORTEMENTS

Comme nous l'avons vu dans les chapitres précédents, la collaboration est un processus social mettant en relation au moins deux agents. Après la prise de conscience d'une limitation, il y a une interaction directe ou indirecte et une intention de réaliser un travail en commun.

Il est donc nécessaire de faire le distinguo entre les comportements propres à l'agent (constitutifs et reflétant sa dynamique interne) et ceux qu'il va mettre en œuvre spécifiquement dans l'objectif de collaborer. La question à se poser, toujours en gardant en tête cette considération du processus collaboratif dans son ensemble, est comment peut-on formaliser et décrire ces comportements de manière globale ? Comment les mettre en relation avec les informations connues de l'agent et ses objectifs ?

Comme nous l'avons vu au chapitre 3, avec des entités mobiles autonomes, nous sommes plutôt dans une organisation de type société où la structure est émergente car évoluant en fonction du système et des interactions sociales entre les agents. Cela influence nécessairement la manière de décrire un comportement et cela veut dire également que les comportements collaboratifs ne doivent pas être figés mais s'adapter dans le temps en fonction des interactions et des échanges.

On peut donc établir une certaine hiérarchie de comportements en fonction de la finalité. Il y a des comportements basiques (ou briques atomiques de comportements) et des comportements plus élaborés basés sur ces éléments atomiques et mis en musique au gré des échanges.

5.2.2.4/ GESTION DES COMMUNICATIONS ET DES PROTOCOLES

Après la considération des informations de l'agent (données propres, représentation) et de ses comportements, nous finissons par la formalisation des communications et des échanges. En effet, comme évoqué dans la section précédente, le cœur de la collaboration (processus social) est la communication. Elle doit porter le contexte fonctionnel, l'intention de l'agent mais aussi le suivi des actions à mener en commun pour assurer la coordination et valider les résultats attendus pour répondre aux objectifs définis.

A l'heure actuelle, FIPA/ACL semble le meilleur moyen pour formaliser ces échanges entre agents car il intègre un caractère intentionnel. Dans un contexte décentralisé et une organisation de type société mais avec la volonté de collaborer, les processus de négociation se font essentiellement par allocation de tâches voire de manière heuristique selon le contexte d'exécution.

Dans certains cas de modélisation d'agents homogènes, il sera peut-être aussi possible de n'utiliser que des ontologies afin de ne communiquer que l'essentiel sans passer par un protocole particulier pouvant être relativement lourd à mettre en œuvre ou à traduire de manière concrète dans les modeleurs utilisés.

5.3/ EXEMPLE FIL ROUGE

Ici nous présentons rapidement l'exemple fil rouge à travers l'article de presse nous l'ayant inspiré. Puis, afin de commencer la réflexion sur les processus collaboratifs, nous montrons un premier exemple d'implémentation en NetLogo. Celui-ci nous permet de faire quelques constatations.

5.3.1/ PRÉSENTATION

Nous allons utiliser un exemple à base de drones afin d'explicitier les différents concepts théoriques. D'après l'article du journal Sud Ouest du 28/07/2012 (Le drone girondin veille sur la forêt des Landes) -², la sécurité civile utilise un drone d'observation afin de faciliter le travail des pompiers (cf. figure 5.4). La vitesse de l'appareil est d'environ 3 à 5 m/sec et il est léger (moins de 2 kilos). Il peut s'élever à 2000 mètres mais dès 150 mètres l'angle de la caméra embarquée couvre déjà 9 hectares. Il permet donc de faire des tours d'horizon à 360 degrés au-dessus des feux. Par contre, il a une autonomie très réduite de quinze minutes seulement.

En se basant sur l'exemple cité dans cet article de presse, nous considérons une flottille de drones pour explorer une zone géographique afin de la cartographier et de détecter des incendies de forêt. En cas d'observation de départ de feu, il y a une communication

2. <http://www.sudouest.fr/2012/07/28/le-drone-girondin-veille-sur-la-foret-des-landes-781215-4723.php>



FIGURE 5.4 – Drone SDIS pour la surveillance de la forêt des Landes

à la base pour alerter et envoyer les pompiers. Les contraintes sont liées au périmètre d'observation, à l'autonomie réduite des drones ainsi qu'aux possibilités réduites de communication.

5.3.2/ IMPLÉMENTATION NETLOGO

Pour nous permettre d'appréhender plus facilement l'exemple fil rouge et d'expliciter dans les prochains chapitres les concepts relatifs à la collaboration, un modèle jouet NetLogo a été implémenté. On y retrouve un environnement rectangulaire, une base située au milieu de la zone ainsi qu'une dizaine de drones évoluant sans formation particulière. Chaque drone dispose d'une capacité de déplacement dans l'environnement. Ces déplacements sont aléatoires et sont conditionnés au niveau de batterie constituant l'autonomie du drone. La mission est simple : déplacement aléatoire, observation et retour à la base si détection d'incendie ou besoin de recharger. Chaque drone dispose également d'un rayon maximum d'observation ainsi qu'un rayon maximum de communication.

La figure 5.5 présente une vue d'ensemble de l'interface principal de ce modèle jouet. Les cercles autour des drones représentent ici la surface maximum vue par chaque drone. La figure 5.6 fait état de la capacité de communication et les cercles autour des drones représentent ici la zone d'émission et réception des signaux. Les liens en rouge symbolisent les contacts entre drones.

La première implémentation met en évidence que le caractère aléatoire des déplacements ne permet pas de couvrir efficacement toute la zone autour de la base. Une deuxième implémentation, cherchant à améliorer les déplacements pour couvrir plus rapidement une surface plus importante, met en évidence la nécessité de mémoriser les zones découvertes pour ne pas revenir trop souvent aux mêmes endroits. La figure 5.7 montre pour chacun des dix drones, la surface découverte ainsi que les zones d'incendie. On peut noter que tous n'observent pas les mêmes choses. En effet, certains drones vont pouvoir détecter des zones d'incendie que d'autres n'auront pas nécessairement vu.

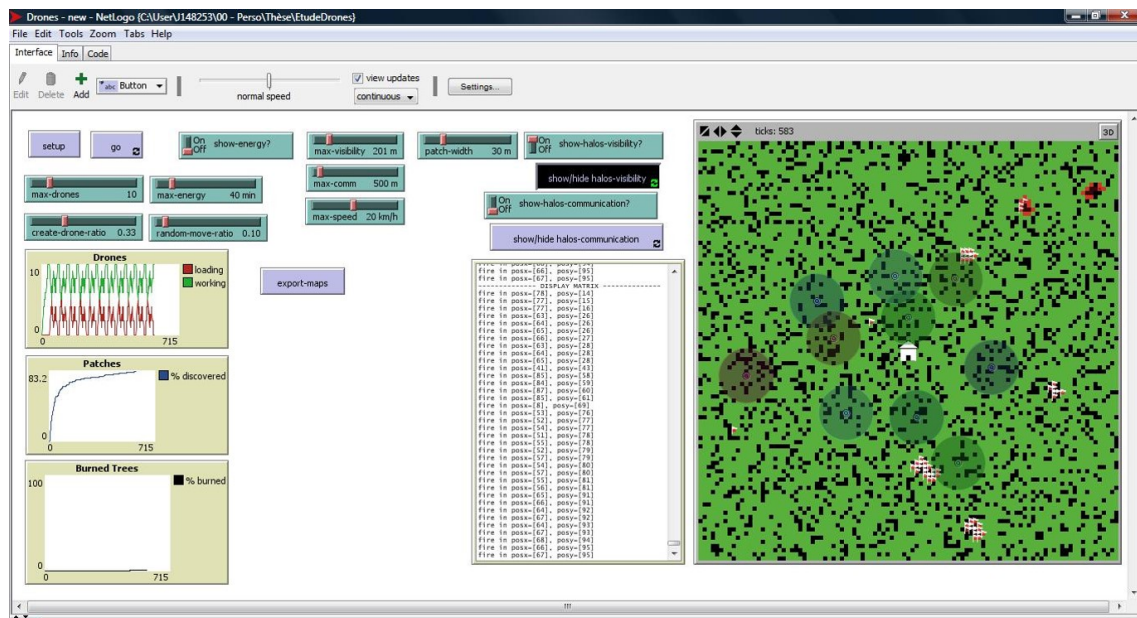


FIGURE 5.5 – Modèle jouet NetLogo : Exemple d'interface principale (zones d'observation)

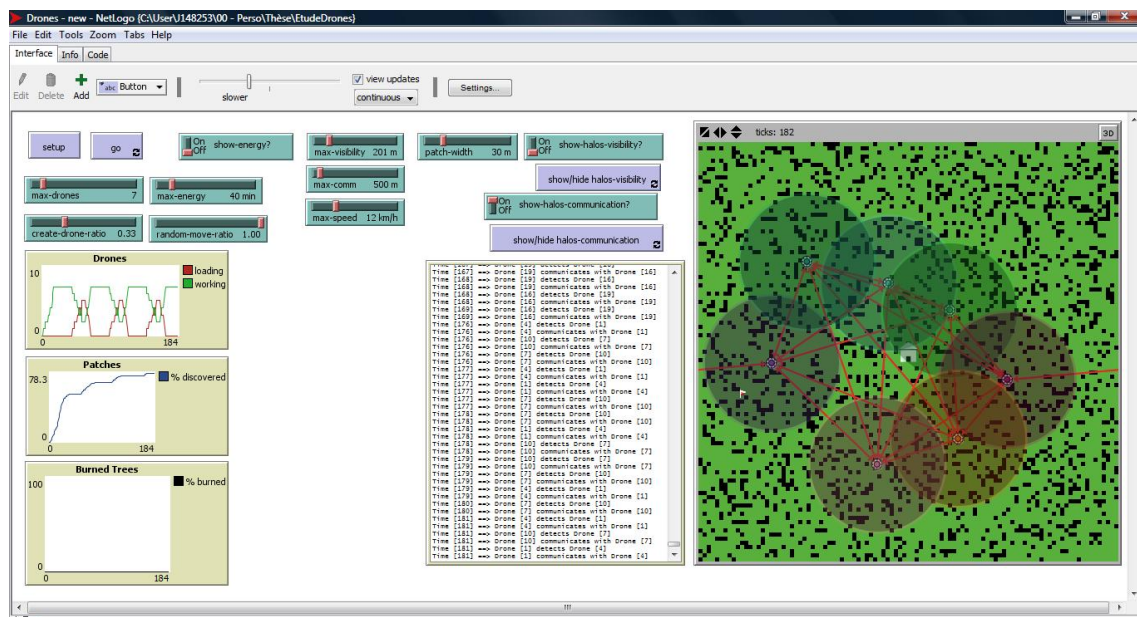


FIGURE 5.6 – Modèle jouet NetLogo : Exemple d'interface principale (zones de communication et lien entre drones)

Enfin dans une troisième implémentation, nous pouvons encore optimiser les déplacements et les communications vers la base en permettant des échanges entre drones pour mettre à jour les cartes internes. La figure 5.8 représente la fusion de toutes les cartes internes des drones. Il convient de faire remarquer que les communications sont assez coûteuses dans cet exemple. Nous avons pu observer un phénomène d'échanges quasi-permanent entre les drones si leurs zones de couverture se recoupent. C'est une sorte d'effet *salon de thé* où tous les convives s'expriment en même temps dans

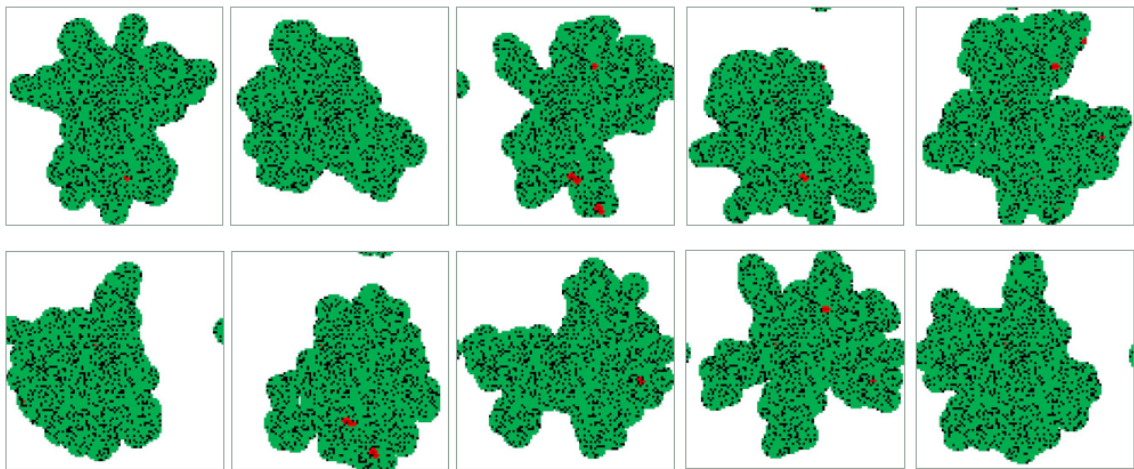


FIGURE 5.7 – Modèle jouet NetLogo : Cartes internes des dix drones

un espace restreint pour faire valoir leurs idées.

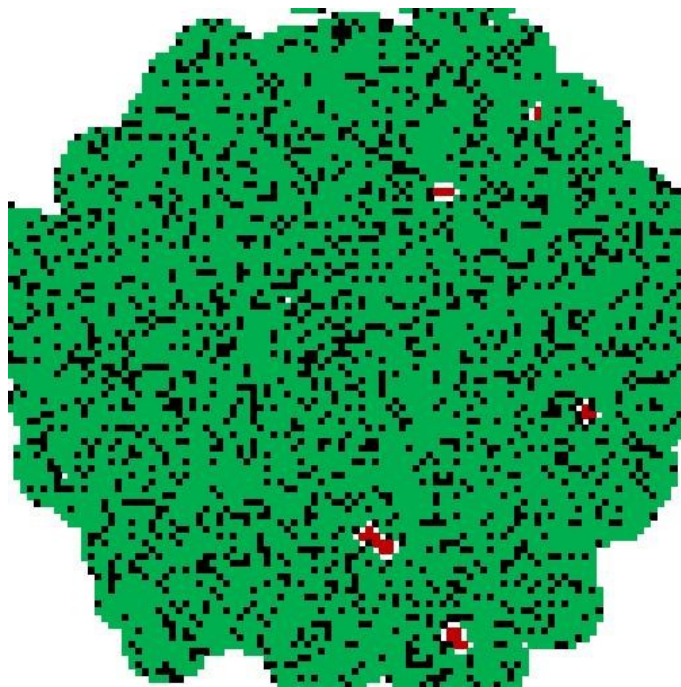


FIGURE 5.8 – Modèle jouet NetLogo : Fusion des cartes internes

Pour résumer, cet exemple jouet nous a permis une première vision et représentation de l'exemple fil rouge mais également nous a permis de commencer notre réflexion sur les différentes notions abordées dans ce chapitre d'analyse, notamment l'importance de la mémorisation des événements et des observations basés sur des processus de décision interne ainsi que de la communication entre agents avec une gestion des échanges et des informations pour en limiter les coûts.

5.4/ CE QU'IL FAUT RETENIR

La figure 5.11 résume notre analyse de la problématique au cœur de cette étude de la collaboration et son impact pour les systèmes multi-agents pour une considération globale du processus.

Les enjeux principaux sont d'améliorer notre compréhension de la collaboration car nous postulons qu'elle aidera les entités mobiles connectées à développer l'autonomie et leurs capacités de réflexion en favorisant les échanges et les contributions réciproques.

En développant la démarche évoquée dans le chapitre 2 et notamment la présentation du cycle de la collaboration, l'utilisation des quadrants permet de mieux organiser la présentation du processus dans le monde agents. Il s'agit d'une représentation fonctionnelle du processus dans son ensemble au sein de l'agent. Le quadrant ESPRIT regroupe les processus de réflexion sur soi et sur les objectifs ce qui constitue la **MOTIVATION**. Puis l'agent se donne les **MOYENS PHYSIQUES** et **SOCIAUX** de trouver une solution en modifiant ses comportements (CORPS) et sa gestion de l'environnement pour le comprendre et favoriser des interactions (ESPACE BRUT). Enfin la mise en œuvre effective au sein du quadrant CULTURE constitue le **FAIT** permettant de trouver la solution au problème soulevé initialement (cf. figure 5.9). La collaboration est un tout : un état mental qui conduit à des modifications comportementales pour permettre des échanges structurés afin de satisfaire des objectifs.

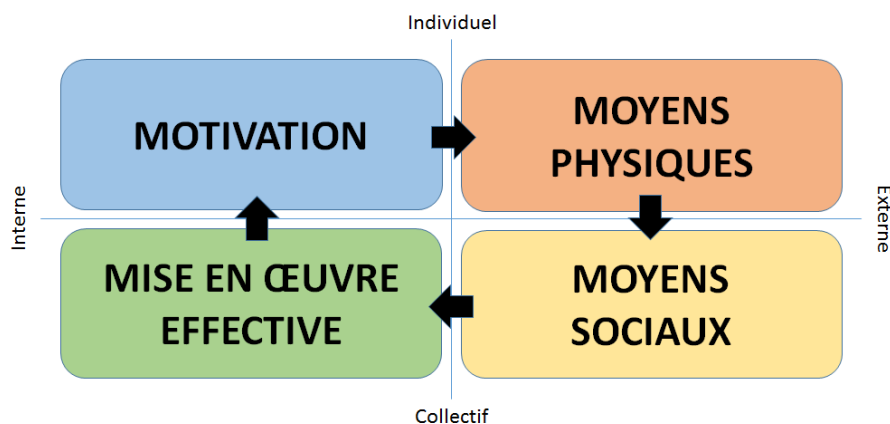


FIGURE 5.9 – Ce qu'il faut retenir : Synthèse du processus complet pour un agent

La figure 5.10 permet d'explicitier plus généralement les interactions entre 2 agents au minimum et les espaces de jointure. Les **RELATIONS** s'établissent dans les 2 quadrants ESPACE BRUT et la **COLLABORATION** se matérialise entre les 2 quadrants CULTURE.

Dans une approche globale et une volonté d'agentification de la collaboration, nous pouvons donc identifier les axes de réflexion suivants : une méthode d'analyse d'un système ou d'un agent afin de concevoir un mode de représentation du monde et plus largement d'organisation des données autour d'un certain nombre de comportements intrinsèques mais aussi de comportements spécifiques pouvant porter un processus collaboratif s'articulant autour de protocoles de communication, l'architecture de l'agent portant l'ensemble de ces concepts dans un tout cohérent. C'est la traduction technique de la vision fonctionnelle.

Nous considérons un processus collaboratif comme étant un tout, le problème doit donc

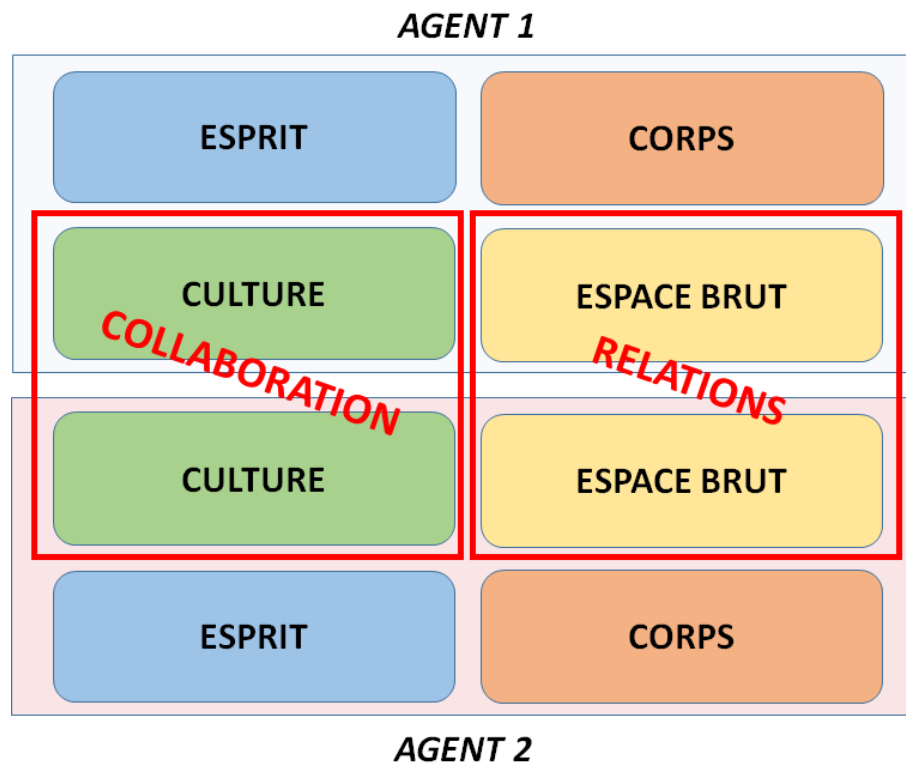


FIGURE 5.10 – Ce qu’il faut retenir : Interactions entre 2 agents (utilisation des quadrants)

être pris dans son ensemble : AGENTS, INFORMATIONS, COMPORTEMENTS, COMMUNICATIONS, OBJECTIFS, RESULTATS.

Dans les chapitres suivants, nous allons détailler notre proposition en l’illustrant avec notre exemple fil rouge de drones de surveillance d’incendie de forêt. Le [chapitre 6](#) présente notre définition de la collaboration et détaille ces caractéristiques. Le [chapitre 7](#) propose une méthode d’analyse d’un problème afin de déterminer les comportements pouvant faire l’objet d’une optimisation par processus collaboratif. Enfin le [chapitre 8](#) détaille notre contribution principale, une architecture intégrant le concept de collaboration entre entités mobiles autonomes et apportant une réponse aux verrous d’organisation des données et de représentation du monde mais également une proposition de gestion des comportements (notamment collaboratifs) et les moyens de communication à mettre en œuvre.



FIGURE 5.11 – Ce qu'il faut retenir : Vue synthétique de notre analyse et des verrous liés au concept de collaboration

COLLABORATION : DÉFINITION, QUALIFICATION

Sommaire

6.1 Caractérisation générale	84
6.1.1 Définition	84
6.1.2 Questionnement	84
6.1.3 Formalisation	85
6.2 Gradient de collaboration : Caractérisation fonctionnelle	87
6.2.1 Complète	87
6.2.2 Partielle	88
6.2.3 Altruiste	89
6.2.4 Forcée	90
6.2.5 Induite	90
6.2.6 Synthèse	91
6.3 Types de Collaboration : Caractérisation technique	91
6.3.1 Locale ou distante	92
6.3.2 Synchrones ou asynchrones	93
6.3.3 Formelle ou informelle	94
6.3.4 Synthèse	94
6.4 Description du processus	94
6.4.1 Déclencheur	97
6.4.2 Dialogue	97
6.4.3 Mise en œuvre	98
6.4.4 Capitalisation	99
6.5 Influence du gradient sur les algorithmes	100
6.6 Ce qu'il faut retenir	101

L'objectif de ce chapitre est de proposer notre définition de la collaboration et une formalisation de celle-ci. Puis nous étudions ce que cela implique conceptuellement. Nous présentons notamment cela selon une double caractérisation : fonctionnelle d'une part, avec un gradient de possibilité en fonction du contexte et technique d'autre part. Une description du processus général (étape par étape) d'une collaboration est également proposée puis mise en perspective en fonction du gradient.

6.1/ CARACTÉRISATION GÉNÉRALE

Dans cette section, nous proposons notre définition de la collaboration. Nous nous aidons ensuite de quelques questions générales pour nous permettre de guider notre réflexion afin de formaliser le concept.

6.1.1/ DÉFINITION

Dans le contexte présenté dans les chapitres précédents (la collaboration en général dans le [chapitre 2](#) et comment elle est aujourd'hui définie dans le domaine des SMA dans le [chapitre 3](#)) et au regard des constats évoqués dans le [chapitre 5](#), nous proposons la définition suivante :

Définition 14 : Collaboration

La collaboration est une interaction et un échange d'informations entre deux entités (ou plus) en vue d'atteindre un résultat partiel commun qui participe à la réalisation d'un objectif plus global. Il s'agit d'un processus intentionnel et cognitif, une volonté de la part des entités qui collaborent avec un effort de partage des informations nécessaires et suffisantes ainsi qu'une vision commune du résultat à atteindre.

Une interaction fait suite à une rencontre d'au moins deux entités dans un environnement commun. Cette interaction peut être fortuite ou provoquée. Nous détaillons les différentes interactions possibles dans la section 6.4.

L'échange d'informations est au cœur de la collaboration car il s'agit bien d'œuvrer pour atteindre un résultat partiel commun participant à la réalisation d'objectifs plus globaux. Pour se faire, il est primordial de communiquer aux autres entités les informations nécessaires et suffisantes pour aider à la prise de décision et comprendre les enjeux de la collaboration demandée.

Tout ce processus de collaboration fait partie d'un ensemble plus ou moins réfléchi. Si une entité ne parvient plus à répondre à ses objectifs, elle va chercher à trouver les informations, compétences ou ressources nécessaires pour les accomplir.

Pour reprendre ce qui a été décrit dans le [chapitre 2](#), la collaboration implique un engagement mutuel, une conception partagée et une organisation à mettre en œuvre mais aussi une liberté de participation et d'action. Ainsi toute entité impliquée dans une collaboration peut choisir de se retirer si les événements l'y contraignent et si elle choisit de le faire.

6.1.2/ QUESTIONNEMENT

Compte tenu de cette définition, répondre aux questions suivantes est une première façon d'illustrer et de qualifier la collaboration dans un système complexe composé d'objets intelligents :

- **Qui collabore ?** Au moins deux entités susceptibles de se comprendre pour échanger des informations et/ou mettre en commun des ressources. La notion de compréhension est essentielle pour engager le dialogue et structurer les communications. Il apparaît en effet incongru de vouloir faire collaborer deux

entités ayant des langages et protocoles intrinsèquement différents. Même si la collaboration repose sur la volonté réciproque de mettre en commun les savoirs et les compétences, il faut noter que ce processus peut être initié par une seule entité.

- **Pourquoi collaborer ?** Pour atteindre plus facilement et plus rapidement un objectif ou pour satisfaire un besoin ou un désir. En effet, si une entité ne parvient plus à répondre à ses impératifs, il y a un risque de blocage de son action. Elle va donc tout faire pour sortir de cette impasse en multipliant les interactions afin de trouver une autre entité capable de lui apporter l'aide nécessaire à la poursuite de la réalisation de ses objectifs. Il peut donc s'agir d'un besoin éventuel ou impératif mais également d'une volonté.
- **Quand collaborer ?** Quand survient simplement une occasion de collaborer notamment pour optimiser ou pour s'affranchir d'une situation de blocage. Il peut donc s'agir de profiter d'une situation, d'une interaction pour établir une communication et un échange d'informations qui va permettre d'augmenter ses connaissances et de réaliser plus rapidement ses objectifs.
- **Comment collaborer ?** Il faut nécessairement que les entités soient en mesure de communiquer efficacement entre elles. Cela veut dire, disposer d'un langage suffisamment évolué pour véhiculer les informations à échanger ainsi que les résultats à atteindre et formaliser les problèmes. Cela implique aussi que l'information soit suffisamment structurée et détaillée pour être transmise. Selon le type de communication (formelle ou informelle), le support de l'information (structure d'envoi) et son vecteur (mode d'envoi) peuvent être différemment envisagés.
- **Quelles informations doit-on transmettre ?** Toute information susceptible d'identifier un problème puis de participer à sa résolution. Il faut donc considérer chaque type d'information à échanger (degré de confidentialité selon les entités, caractéristiques des entités, etc).

6.1.3/ FORMALISATION

Avant d'aller plus loin, nous proposons de formaliser les agents de la manière suivante :

Proposition 1 : Formalisation d'un agent

Nous définissons un agent tel que : $Agent = (I, C, P, O)$

- I : ensemble des informations (ses connaissances)
- C : ensemble des comportements (ses capacités d'action)
- P : ensemble des protocoles, des stratégies de communication (sa culture, au sens des définitions employées par Tranier ([Tranier, 2007]))
- O : ensemble d'objectifs (ses motivations)

L'ensemble I représente toute la connaissance de l'agent. L'ensemble C représente la liste de tous les comportements connus de l'agent. Chaque comportement peut être composé d'actions plus élémentaires ou faire référence à des plans plus élaborés avec des

enchainements d'actions conditionnés. L'ensemble P représente tous les protocoles et stratégies de communication de l'agent. On peut considérer que ceux-ci sont dépendants de la nature intrinsèques de l'agent et de ses capacités techniques de communication. L'ensemble O est la liste des objectifs (ou missions) que l'agent cherche à remplir.

Puis, en nous appuyant sur notre définition de la collaboration et en reprenant ce que nous avons abordé dans le chapitre précédent, c'est-à-dire notre volonté de lier les notions d'AGENTS, d'INFORMATIONS, de COMPORTEMENTS, de COMMUNICATIONS, d'OBJECTIFS et de RESULTATS, voici notre proposition de formalisation.

Proposition 2 : Formalisation de la collaboration

Nous définissons une collaboration telle que : $Collab = (A, I, C, P, O, R)$

- A : ensemble des agents acteurs de la collaboration
- I : ensemble des informations nécessaires à la compréhension et à la contextualisation de la collaboration
- C : ensemble des comportements à adopter pour exécuter cette collaboration
- P : ensemble des protocoles, des stratégies de communication
- O : ensemble d'objectifs à atteindre grâce à cette collaboration
- R : ensemble des résultats et récompenses attendus par les acteurs de cette collaboration

L'ensemble A représente les agents collaborants. Il est constitué au minimum de deux agents sinon la collaboration n'a pas lieu d'être. Bien évidemment, théoriquement, plus de deux agents peuvent collaborer ensemble. Il peut s'écrire : $A = \{a_1, a_2, \dots, a_n\}$.

L'ensemble I représente toutes les informations nécessaires et suffisantes. Il est constitué d'informations provenant des différentes agents collaborants. Il peut donc s'écrire de la manière suivante : $I = \bigcup_{i=1}^n I'_{a_i}$ où n est le nombre d'agents collaborants et $I'_{a_i} = \{i_{a_i^1}, i_{a_i^2}, \dots, i_{a_i^n}\}$ est un sous-ensemble contextualisé de l'ensemble des informations connues de l'agent a_i , $I'_{a_i} \subset I_{a_i}$.

L'ensemble C représente les comportements que les agents vont mettre en œuvre collectivement pour parvenir au terme de la collaboration (en s'appuyant sur leur bibliothèque de comportements individuels). Chaque élément est le résultat d'une communication préalable entre les acteurs pour déterminer le comportement idéal à adopter pour résoudre l'objectif défini. De plus, chaque comportement peut être également un ensemble d'actions ordonnancées selon les nécessités de la collaboration. En tenant compte de ces indications, cet ensemble peut donc s'écrire de la manière suivante : $C = \{c_1, c_2, \dots, c_n\}$ où $c_i = \{act_1, act_2, \dots, act_n\}$.

L'ensemble P représente les stratégies de communication à mettre en œuvre dans la réalisation de la collaboration. Par exemple, il peut s'agir d'une communication avec un envoi simple d'information ou d'une communication plus détaillée avec un jeu de questions / réponses. Ces stratégies de communication seront choisies en fonction des situations de collaboration (partielle, complète, altruiste, etc) ainsi que du contexte de réalisation (temps court ou temps long). Cet ensemble peut s'écrire : $P = \{p_1, p_2, \dots, p_n\}$. Ces stratégies font partie de la culture des agents à condition que ce soit des agents homogènes. Dans le cas contraire, ces stratégies ou protocoles de communication doivent être construites et acceptées par tous les agents collaborant ensemble.

L'ensemble O représente les objectifs à atteindre par les agents collaborant entre eux. Il s'agit de buts partagés, autrement dit, de buts communs aux entités. Il peut s'agir également d'objectifs individuels ou déterminés collectivement. Cet ensemble peut s'écrire : $O = \{o_1, o_2, \dots, o_n\}$ ou également $O = \bigcap_{i=1}^n O'_{a_i}$ où n est le nombre d'agents collaborants et $O'_{a_i} = \{o_{a_i^1}, o_{a_i^2}, \dots, o_{a_i^n}\}$ est un sous-ensemble contextualisé de l'ensemble des objectifs connus de l'agent a_i , $O'_{a_i} \subset O_{a_i}$. Par extension, nous pouvons aussi considérer des listes de sous-objectifs en fonction du découpage des tâches.

L'ensemble R représente les récompenses ou résultats que se partagent les entités. Cela peut être des mises à jour des informations de chaque agent (connaissances ou événements) et des modifications de certaines propriétés (comme la réputation par exemple). Cet ensemble peut s'écrire : $R = \{r_1, r_2, \dots, r_n\}$. Ces récompenses ou résultats de la collaboration peuvent s'intégrer dans l'ensemble des informations de chaque agent collaborant.

Théoriquement, il est possible de faire collaborer plusieurs agents autour d'un résultat commun. Cependant, il ne faut pas perdre de vue que la collaboration est un processus complexe engageant un partage d'information et une vision commune du résultat à atteindre. De plus, cela induit des processus de communication et de synchronisation des actions comme nous pourrions le voir dans la section 6.4. Il est possible que ces processus soient rédhibitoires en cas d'un trop grand nombre d'agents collaborant ensemble.

En reprenant les différentes questions abordées dans la section précédente, nous pouvons dire que l'ensemble A est le *qui* ?, l'ensemble I le *quoi* ?, les ensembles C et P le *comment* ? et l'ensemble O le *pourquoi* ?.

6.2/ GRADIENT DE COLLABORATION : CARACTÉRISATION FONCTIONNELLE

Notre définition de la collaboration étant posée et formalisée, nous pouvons maintenant lister les différentes collaborations sur une échelle de valeur.

Le gradient de collaboration caractérise le genre de relations établies entre les entités collaborant et le rôle joué par chacune d'elles. Cette classification représente les fondations de notre future architecture d'agent. Nous pouvons distinguer cinq catégories, de la plus complète et la plus engageante, à la plus indirecte et diffuse :

- Collaboration complète ;
- Collaboration partielle ;
- Collaboration altruiste ;
- Collaboration forcée ;
- Collaboration induite.

6.2.1/ COMPLÈTE

Il s'agit ici du processus standard de collaboration avec un partage du problème, un découpage en sous-tâches, une vision commune, une coordination d'action et une capitalisation des résultats. Il s'appuie sur une connaissance de l'organisation, un partage de l'information et des algorithmes de négociation. Le protocole d'échange sera simple ou

plus élaboré en fonction des caractéristiques des agents et leur degré d'homogénéité.

Les états mentaux des deux agents sont compatibles et leur état physique leur permet de continuer leur mission (niveau de batterie correct par exemple). Les dispositifs de communication nécessaires sont opérationnels et la situation géographique est favorable pour permettre un échange de messages potentiellement longs.

Par exemple, deux drones communiquent leur propre carte interne pour compléter leur vision réciproque de leur environnement. Puis ils échangent leurs objectifs afin d'optimiser l'utilisation de leurs ressources (batterie) ou d'utiliser leurs compétences spécifiques à bon escient. Dans le cadre de l'exemple suivant (cf. figure 6.1), l'objectif sera de partager leur mission et leur zone d'exploration pour ne pas couvrir le même territoire et ainsi construire un plan de patrouille

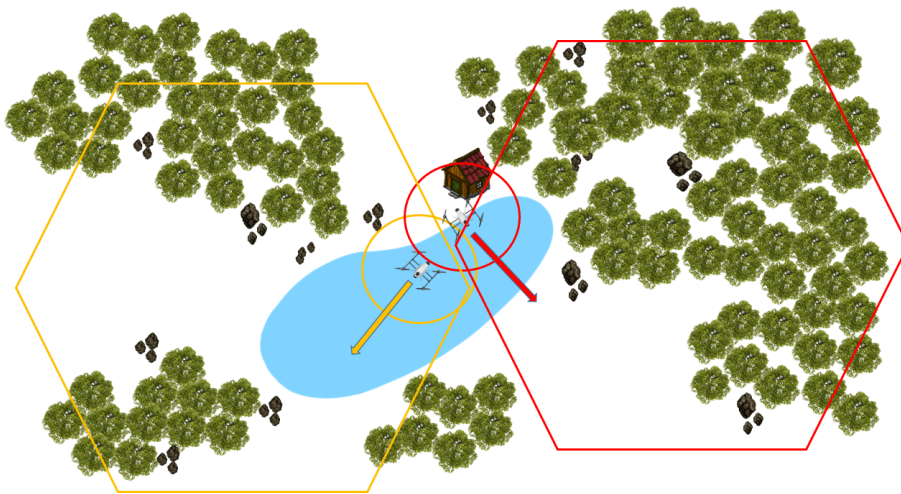


FIGURE 6.1 – Exemple de situation de collaboration complète

Dans ce cas, nous avons $Collab = \{A, I, C, P, O, R\}$ avec $A = \{a_1, a_2\}$, $I = I'_{a_1} \cup I'_{a_2}$ (informations sur les cartes mentales), $C = \{c_{a_1}, c_{a_2}\}$ où c_{a_1} est le comportement décidé pour l'agent a_1 et c_{a_2} est le comportement pour a_2 , P est le protocole de collaboration complète (dialogue), $O = O'_{a_1} \cap O'_{a_2}$ et $R = \{r_1, r_2, \dots, r_n\}$ contiendra des informations de réputation sur les agents collaborants et les zones à explorer.

6.2.2/ PARTIELLE

Il s'agit ici d'un échange partiel d'informations, le juste nécessaire pour que chaque entité puisse continuer à satisfaire ses objectifs et ses propres besoins dans l'environnement. Il n'y a pas besoin d'échanges complexes.

Les états mentaux des deux agents sont compatibles et leur état physique leur permet de continuer leur mission (niveau de batterie correct par exemple). Les dispositifs de communication nécessaires sont opérationnels mais les possibilités d'échange sont réduits car les deux drones ne sont que sur une trajectoire de rencontre fortuite. La situation ne permet qu'un échange partiel d'information. Compte tenu de leur vitesse respective, ils ne resteront pas dans la même zone très longtemps.

Dans le cadre de l'exemple suivant (cf. figure 6.2), l'objectif sera d'échanger rapidement quelques informations sur le territoire exploré afin de compléter la carte interne

des drones et remettre à jour leurs informations spatiales.

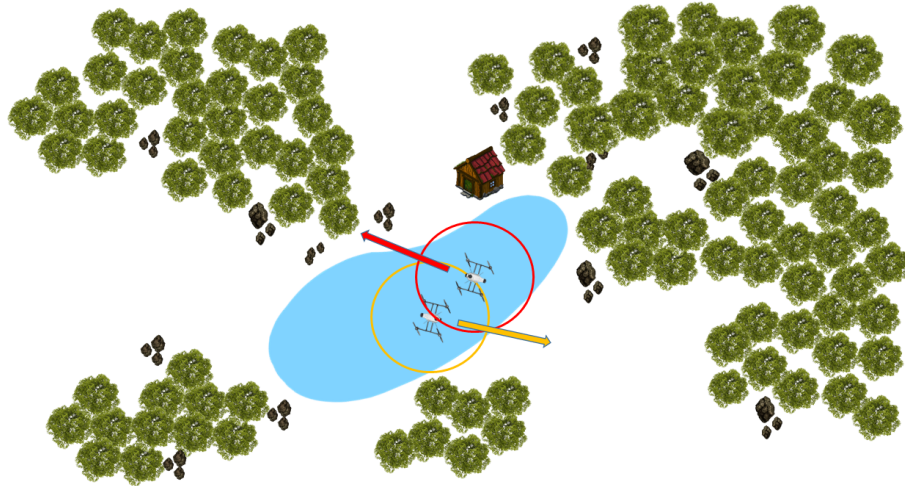


FIGURE 6.2 – Exemple de situation de collaboration partielle

Dans ce cas, nous avons $Collab = \{A, I, C, P, O, R\}$ avec $A = \{a_1, a_2\}$, $I = I'_{a_1} \cup I'_{a_2}$ (extraits des cartes mentales des deux agents), C n'est pas élaboré conjointement et les agents collaborant modifieront leur comportement en fonction des informations reçues, P est le protocole de collaboration partielle (échange simple), O correspond à l'objectif global du système, résultat des comportements des agents autour de la mission principale (et non des objectifs individuels locaux partagés) et $R = \{r_1, r_2, \dots, r_n\}$ contiendra des informations de réputation sur les agents collaborants et les informations de carte recueillies.

6.2.3/ ALTRUISTE

Dans ce cas, c'est une offre de service, une sorte d'entraide potentiellement sans contrepartie, facilitant la réalisation des objectifs sur le concept du bénévolat ou de la bienveillance [Goldman et al., 1994]. Cela ne peut être mis en œuvre que si l'entité offrant ses services peut reporter certains objectifs à atteindre (et se trouve donc dans une phase d'attente entre deux missions ou en attente d'informations complémentaires pour les poursuivre).

L'état mental de l'agent altruiste est compatible avec une offre de service et son état physique lui permet d'aider un autre agent (niveau de batterie correct par exemple). Les dispositifs de communication nécessaires sont opérationnels.

Par exemple, une entité sans affectation préalable d'une mission avec un niveau élevé de batterie et en possession d'un grand nombre d'informations utiles à transmettre, peut proposer ses services aux autres entités passant à proximité. Autre cas de figure, il peut servir de relais de communication pour des informations importantes.

Dans ce cas, nous avons $Collab = \{A, I, C, P, O, R\}$ avec $A = \{a_1, a_2\}$, $I = I'_{a_1}$ ou $I = I'_{a_2}$ (seulement les informations de l'agent requérant ou de l'agent altruiste), C n'est pas élaboré conjointement et les agents collaborant modifieront leur comportement en fonction des informations reçues, P est le protocole de collaboration altruiste (échange simple), $O = O'_{a_1}$ correspond aux seuls objectifs de l'agent aidé (c'est donc un transfert d'objectifs de l'agent aidé vers l'agent aidant que ce dernier est libre d'accepter ou

de refuser) et $R = \{r_1, r_2, \dots, r_n\}$ contiendra des informations de réputation sur les agents collaborants et les informations récupérées lors de l'échange.

6.2.4/ FORCÉE

Dans ce quatrième cas, l'une des deux entités n'est pas en mesure de refuser l'échange ou le partage de l'information car c'est une question de "vie ou de mort", au-delà des priorités propres ou des objectifs en cours de réalisation. Pour ce genre de collaboration, nous utilisons le concept de violation des règles et des comportements planifiés. Mais cela peut affecter le comportement général de l'agent car c'est une modification de ses objectifs et une altération possible de ses connaissances.

L'état mental de l'agent en danger reste compatible avec cette action (volonté de préserver l'information) mais son état physique est déclinant, ce qui l'empêche de continuer. Les dispositifs de communication nécessaires sont opérationnels.

Dans le cadre de l'exemple suivant (cf. figure 6.3), un drone n'a plus de batterie pour rentrer à la base mais il doit transmettre une information importante (comme l'emplacement d'un feu de forêt par exemple). Il pourrait ainsi forcer un autre drone à finir cet objectif pour transmettre l'information dès que possible.

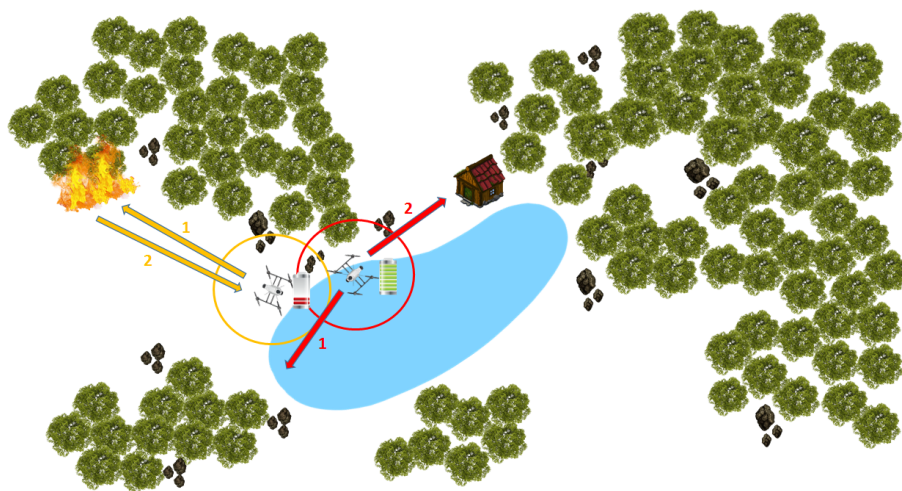


FIGURE 6.3 – Exemple de situation de collaboration forcée

Dans ce cas, nous avons $Collab = \{A, I, C, P, O, R\}$ avec $A = \{a_1, a_2\}$, $I = I'_{a_1}$ (seulement les informations de l'agent requérant), $C = c_{a_1}$ (comportement imposé), P est le protocole de collaboration forcée (communication simple en sens unique), $O = O'_{a_1}$ correspond aux seuls objectifs de l'agent en danger (c'est donc un transfert d'objectifs de l'agent *forçant* vers l'agent *forcé* que ce dernier ne peut pas refuser) et $R = \{r_1, r_2, \dots, r_n\}$ contiendra des informations de réputation sur les agents collaborants et les informations récupérées lors de l'échange.

6.2.5/ INDUITE

Nous mentionnons ce dernier cas de collaboration pour être complet. Il s'agit d'une caractéristique produite pour les échanges successifs d'information entre agents. Un

Etat mental \ Etat physique	Capable	Incapable
	Capable	Incapable
But précis	Collaboration complète (si but commun et temps matériel pour effectuer celle-ci), collaboration partielle sinon. Processus de négociation en fonction des cas	Collaboration forcée. Pas de négociation. Substitution de but
Pas de but précis	Collaboration altruiste. Pas de négociation. Offre de service	Sans objet (car incapacité totale)

TABLE 6.1 – Synthèse du gradient de collaboration selon l'état mental et physique

premier agent transmet des informations (tous types de collaboration confondu) à un deuxième agent qui va à son tour transmettre ces informations à un troisième. Nous pouvons dire qu'en définitive, le troisième agent collabore avec le premier de manière indirecte.

Par exemple, un premier drone (*A*) en danger force un deuxième drone (*B*) à transmettre une information importante (collaboration forcée). Ce drone *B* va rencontrer un troisième drone (*C*) pour diffuser l'information importante à propager plus des informations relatives à sa carte mentale (collaboration partielle). Nous pouvons donc affirmer que le drone *C* collabore de manière induite. Dit autrement, *A* collabore avec *B* puis *B* collabore avec *C*. Donc d'une certaine manière, *C* collabore avec *A* par l'intermédiaire de *B*.

6.2.6/ SYNTHÈSE

Nous venons de décrire le gradient de collaboration qui est conditionné par des aspects fonctionnels. La collaboration choisie dépend de critères mentaux et physiques des agents, eux-mêmes influencés par les buts recherchés. Le tableau 6.1 synthétise les collaborations envisageables en fonction de l'état mental et physique d'un agent. La méthode de négociation sera également choisie en fonction du gradient de collaboration.

Pour chaque collaboration, nous avons déterminé quelles valeurs peuvent avoir les différentes composantes (agents, informations, comportements, protocoles, objectifs, récompenses/résultats). Afin de résumer les différentes collaborations et leurs propriétés, nous proposons le tableau 6.2.

6.3/ TYPES DE COLLABORATION : CARACTÉRISATION TECHNIQUE

Après avoir formalisé ce qu'est la collaboration et exposé le gradient de possibilités, nous pouvons en définir plusieurs types (qui peuvent d'ailleurs être regroupés selon les usages). Nous allons nous aider des règles du théâtre classique^{1 2} pour illustrer notre

1. https://fr.wikipedia.org/wiki/R%C3%A8gles_du_th%C3%A9%C3%A2tre_classique

2. <https://www.etudes-litteraires.com/regle-trois-unites.php>

Collab. \ Propr.	Complète	Partielle	Altruiste	Forcée
Agents	$A = \{a_1, a_2\}$	$A = \{a_1, a_2\}$	$A = \{a_1, a_2\}$	$A = \{a_1, a_2\}$
Informations	$I = I'_{a_1} \cup I'_{a_2}$	$I = I'_{a_1} \cup I'_{a_2}$	$I = I'_{a_1}$ ou $I = I'_{a_2}$	$I = I'_{a_1}$
Comportements	$C = \{c_{a_1}, c_{a_2}\}$	C pas élaboré conjointement	C pas élaboré conjointement	$C = c_{a_1}$
Protocoles	P collab. complète (dialogue)	P collab. partielle (échange simple)	P collab. altruiste (échange simple)	P collab. forcée (communication en sens unique)
Objectifs	$O = O'_{a_1} \cap O'_{a_2}$	O objectif global du système	$O = O'_{a_1}$ (transfert d'objectifs proposé vers a_2)	$O = O'_{a_1}$ (transfert d'objectifs forcé vers a_2)
Récompenses	$R = \{r_1, r_2, \dots, r_n\}$	$R = \{r_1, r_2, \dots, r_n\}$	$R = \{r_1, r_2, \dots, r_n\}$	$R = \{r_1, r_2, \dots, r_n\}$

TABLE 6.2 – Synthèse du gradient de collaboration et les impacts sur les propriétés

propos :

- **Unité de lieu** : toute l'action doit se dérouler dans un même lieu ;
- **Unité de temps** : la durée totale de la collaboration coïncide avec la durée de l'action considérée ;
- **Unité d'action** : tous les événements doivent être liés et nécessaires à l'action considérée.

Ainsi nous pouvons lister les types de collaboration suivants :

- Locale ou distante géographiquement (**lieu**) ;
- Synchrone ou asynchrone (**temps**) ;
- Formelle ou informelle (**action**).

6.3.1/ LOCALE OU DISTANTE

Une collaboration locale est une collaboration entre deux entités (ou plus) dans un contexte d'exécution locale avec les ressources et les informations directement disponibles : **Unité de lieu**. La synchronisation est aisée car les entités sont proches spatialement. On peut penser que les processus de communication directement entre entités sont suffisants dans ce contexte local pour permettre les échanges d'information et la synchronisation des actions.

Au contraire, une collaboration distante est une collaboration entre deux entités (ou plus) n'évoluant pas dans les mêmes lieux. L'inconvénient majeur de ce type de processus est la synchronisation des actions et plus généralement la communication entre entités. Il faut pouvoir assurer une continuité des actions même si les moyens de communication ne sont pas disponibles, comme nous le verrons dans la section suivante.

Enfin, dans un contexte d'entités mobiles, une collaboration peut être initiée de manière

locale pour se transformer en collaboration distante et réciproquement. Il faut donc pouvoir mettre en œuvre les mécanismes de transfert d'information pour maintenir le contact entre les entités collaborant.

Prenons deux drones évoluant dans un même environnement et ayant pour mission de le parcourir afin de détecter des incendies de forêt.

Premier cas de figure (*collaboration locale*), au moment de leur interaction physique initiale et en fonction de leurs objectifs et motivations, une collaboration peut être initiée sur le plan local afin de se partager les zones à explorer pour couvrir un plus large périmètre en un minimum de temps.

Deuxième cas de figure (*collaboration distante*), les drones ne se croisent pas physiquement dans l'environnement mais sont à portée de communication. En fonction de leurs objectifs et motivation du moment, le processus de collaboration va être mis en œuvre. Enfin, une variante peut consister à mettre ces drones en relation via un relais de communication (drone tiers ou base). Une fois l'attribution des zones d'exploration, chaque drone va effectuer sa mission de reconnaissance. Il faut pouvoir, par la suite, assurer le suivi de cette collaboration notamment pour échanger leur carte réciproque ainsi que les incendies détectés. Si leur niveau de batterie le permet, elles peuvent définir un point de rendez-vous pour la synchronisation, soit celle-ci sera effectuée via des relais de communications qui peuvent être d'autres drones patrouillant dans l'environnement ou directement à la base lors d'une recharge de batterie.

6.3.2/ SYNCHRONE OU ASYNCHRONE

Dans un processus de collaboration synchrone, deux entités (ou plus) communiquent en temps réel afin de coordonner leurs actions : *Unité de temps*. Il n'est pas nécessaire qu'elles soient géographiquement situées au même endroit si les moyens de communication sont suffisants.

Au contraire, dans le cas d'un processus asynchrone, les entités passent par des relais de communication (tableau blanc, d'autres entités, boîte aux lettres, etc). Deux possibilités : soit les entités sont suffisamment autonomes pour engager les actions sans synchronisation constante et immédiate avec les autres entités collaborant ; soit les moyens de communication permettent cette synchronisation continue (ou au moins fréquente) : couverture réseau large, entité centralisatrice, utilisation de relais de communication, etc.

Enfin, dans un contexte d'entités mobiles, une collaboration peut être initiée de manière synchrone pour se transformer en collaboration asynchrone et réciproquement. Il faut donc pouvoir mettre en œuvre les mécanismes de transfert d'information pour maintenir le contact entre les entités collaborant.

Prenons toujours l'exemple de deux drones évoluant dans un même environnement et ayant pour mission de le parcourir afin de détecter des incendies de forêt.

Premier cas de figure (*collaboration synchrone*), au moment de leur interaction initiale, les échanges nécessaires à la mise en place de la collaboration sont réalisés au même moment. Par exemple, afin de déterminer les zones à explorer, les drones dialoguent pour déterminer les missions de chacun.

Deuxième cas de figure (*collaboration asynchrone*), pour la même mission de construction de plan de patrouille, une fois les zones attribuées, les drones partent explorer et

ne sont donc plus forcément en contact. Par contre, ils peuvent passer par des relais de communication pour échanger des informations de synchronisation si c'est nécessaire.

6.3.3/ FORMELLE OU INFORMELLE

Une collaboration formelle structure l'échange d'informations et la méthode de communication (découpage du problème, partage des résultats, capitalisation des résultats) : **Unité d'action**. Toutes les actions décidées et les messages échangés servent à la réalisation synchronisée de la collaboration dans un temps défini simultanément et sans interruption.

Au contraire, une collaboration informelle n'utilise que des connaissances partielles de l'autre sans la mise en commun de l'expertise et des ressources, en prenant juste le minimum pour répondre à un besoin individuel. Les actions ne seront donc pas forcément coordonnées précisément, chaque agent agissant selon ses propres besoins et objectifs mais dans le respect de l'objectif commun à l'origine de la collaboration.

Dans le cadre des drones de surveillance, une **collaboration formelle** peut être celle initiée dans le but de déterminer les meilleures zones d'exploration issues d'un dialogue entre deux drones. Elle sera suivie d'un contrôle régulier de l'exécution des tâches qui seront donc synchronisées, comme dans le cas d'une collaboration complète.

L'échange d'informations cartographiques est une forme de **collaboration informelle**. Seules les données pertinentes sont échangées dans le but d'offrir une meilleure chance aux drones collaborateurs de réussir plus efficacement leur mission. Il n'y a pas de synchronisation des actions qui suivent cet échange, comme par exemple dans le cas d'une collaboration partielle.

6.3.4/ SYNTHÈSE

Pour finir sur les processus de collaboration synchrone, asynchrone, locale et distante, le seul cas où la collaboration sera considérée comme directe sera dans le cas d'une collaboration synchrone et locale, assurant une unité de temps et de lieu. Tous les autres cas de figure sont donc des illustrations d'une collaboration indirecte. La table 6.3 présente une synthèse des types de collaboration technique en s'inspirant de travaux de Johansen ([Johansen, 1988]) puis d'Ellis ([Ellis et al., 1991]) sur les possibilités de travail en commun en utilisant une matrice espace-temps.

Grudin ([Grudin, 1994]) propose un raffinement supplémentaire. La table 6.4 présente une autre synthèse des collaborations techniques.

Les différents types de collaboration sont influencés par des critères physiques comme l'environnement dans lequel les agents évoluent et leur position dans celui-ci ainsi que par des aspects techniques comme les possibilités réelles de communication et les protocoles d'échanges entre agents.

6.4/ DESCRIPTION DU PROCESSUS

En utilisant la méthode des quadrants, nous allons reprendre les différentes étapes décrites au **chapitre 5** mais en nous intéressant maintenant plus précisément aux re-

	Temps identique	Temps différents
Lieu identique	Collaboration locale et synchrone (<i>aide à la réunion, interaction en face-à-face</i>)	Collaboration distante et asynchrone (<i>aide mémoire électronique, interaction asynchrone</i>)
Lieux différents	Collaboration distante et synchrone (<i>réunion virtuelle, interaction distribuée synchrone</i>)	Collaboration distante et asynchrone (<i>coordination permanente, interaction distribuée asynchrone</i>)

TABLE 6.3 – Synthèse des types de collaboration technique en s’inspirant des travaux de Johansen et d’Ellis (entre parenthèses, les termes utilisés originellement par *Johansen* [Johansen, 1988] et **Ellis** [Ellis et al., 1991])

	Temps identique	Temps différents et prévisibles	Temps différents et imprévisibles
Lieu identique	Collaboration locale et synchrone (<i>aide à la réunion</i>)	Collaboration distante et asynchrone, communication directe possible entre les 2 collaborateurs par prise de "rendez-vous" (<i>relais, enchainement, tour de parole</i>)	Collaboration distante et asynchrone, communication indirecte entre les 2 collaborateurs (<i>team rooms, post-it</i>)
Lieux différents et prévisibles	Collaboration distante et synchrone (<i>vidéo-conférences</i>)	Collaboration distante et asynchrone, communication possible par dépose de message (<i>courrier électronique</i>)	Collaboration distante et asynchrone, communication par relais (<i>écriture collaborative</i>)
Lieux différents et imprévisibles	Collaboration distante et synchrone communication par relais (<i>séminaires interactifs multi-points</i>)	Collaboration distante et asynchrone, communication par dépose de message (<i>tableaux d’affichage</i>)	Collaboration distante et asynchrone, communication par jeton (tour à tour), caractère aléatoire d’une rencontre (<i>workflow</i>)

TABLE 6.4 – Synthèse des types de collaboration technique en s’inspirant des travaux de Grudin (entre parenthèses, les termes utilisés originellement par *Grudin* [Grudin, 1994])

lations entre les quadrants.

En effet, comme nous l’avons indiqué, la collaboration est un processus global de la prise de décision motivant la recherche d’une collaboration (intention), à la mise en œuvre de moyens physiques (action), en passant par la mise en œuvre de moyens sociaux

avec la recherche d'interaction (organisation), en finissant par la collaboration effective (négociation, planification, exécution, porté par les protocoles de communication) et en s'appuyant sur une culture d'échange et de partage commune (interprétation). Evidemment toute action dans ce processus global a une influence sur tous les autres quadrants comme le montre bien la figure 6.4 avec ses principes rétroactifs.

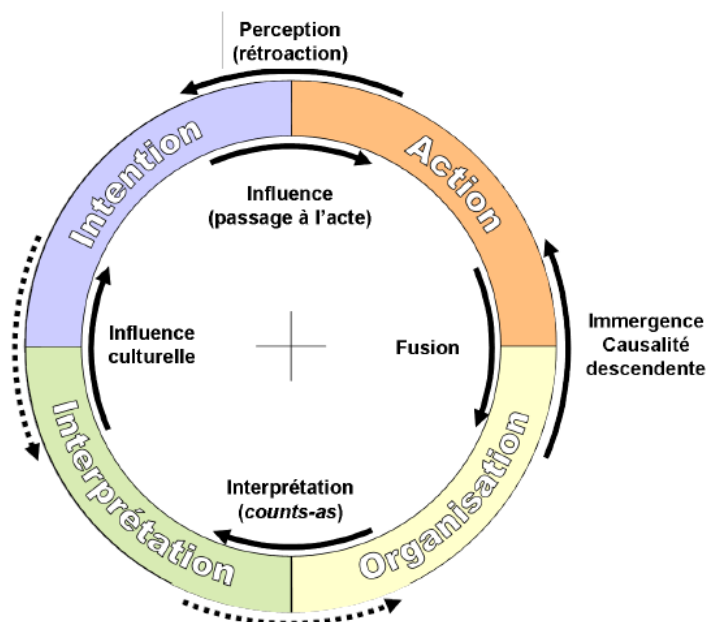


FIGURE 6.4 – Illustration des relations entre quadrants ([Tranier, 2007])

Nous nous intéressons ici plus spécifiquement au passage à l'action de collaboration qui couvre surtout les quadrants **action** et **organisation**. La figure 6.5 expose notre vision du déroulement d'un processus collaboratif.

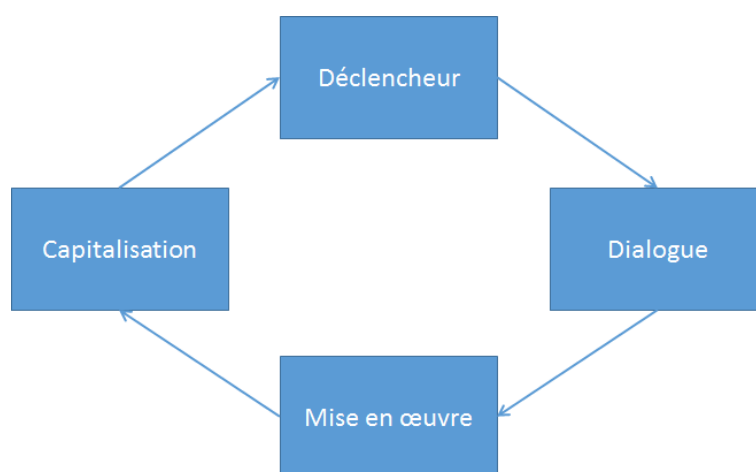


FIGURE 6.5 – Processus de collaboration

Nous découpons en 4 étapes principales : un déclencheur du processus, puis une phase de dialogue entre les agents, une phase de collaboration effective et enfin une phase de capitalisation.

6.4.1/ DÉCLENCHEUR

La figure 6.6 présente le synoptique de l'étape **Déclencheur**. Il s'agit de la prise de contact lorsque deux entités se rencontrent dans l'environnement. Ce contact initial peut être découpé en deux phases : une **interaction** et une **évaluation de l'entité rencontrée**.

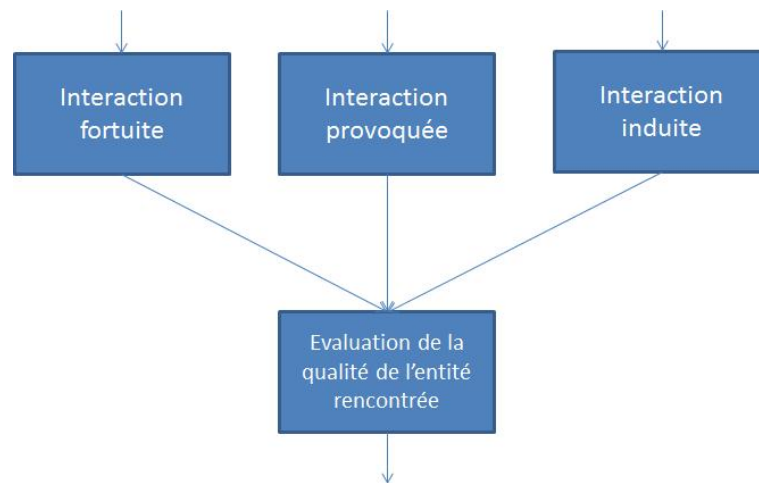


FIGURE 6.6 – Synoptique du déclencheur

Nous pouvons distinguer trois types d'interactions :

- Interaction fortuite : il s'agit d'une simple rencontre due au hasard dans l'environnement où évoluent les entités ;
- Interaction provoquée : une des deux entités cherche à entrer en contact car elle a identifié un problème qu'elle ne sait pas résoudre et se met en quête d'une autre entité capable de l'aider ;
- Interaction induite : l'entité rencontrée ou contactée à dessein ne peut pas répondre directement mais elle connaît une autre entité pouvant le faire. Elle se met en contact avec ou met en relation les deux autres entités. Selon le contexte, nous pouvons aussi considérer ce troisième type d'interaction comme une interaction provoquée par une entité tiers disposant de l'identité de l'agent et connaissant la manière de l'atteindre.

Remarque importante : une communication entrante peut aussi déclencher un processus de collaboration notamment dans le cas d'une collaboration distante.

Pour l'évaluation de l'entité rencontrée, différents aspects sont évalués : capacités cognitives, aptitude à communiquer, etc. Bien évidemment, si l'entité rencontrée n'est pas jugée capable, il y a interruption du cycle et reprise des investigations. Pour cette étape, nous considérons le cas le plus général possible où des entités hétérogènes sont susceptibles d'entrer en contact. Il est évident que le dialogue ne peut s'établir que si un vocabulaire commun et des moyens de communication appropriés sont disponibles.

6.4.2/ DIALOGUE

La figure 6.7 présente le synoptique de l'étape **Dialogue**. Après la prise de contact, il s'agit d'échanger toutes les informations nécessaires à une prise de décision quant à

la suite des événements : **échange des informations** de contexte, **détail des objectifs** et suite à ces échanges, **évaluation de la capacité à résoudre le problème**.

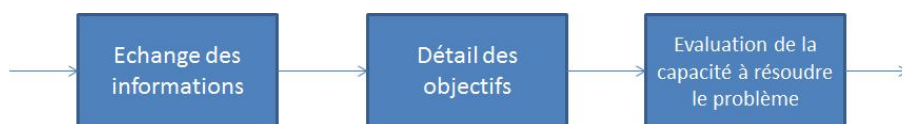


FIGURE 6.7 – Synoptique de dialogue

L'étape d'échange d'informations permet de transmettre le contexte du contact et le caractériser. Ce sont toutes les informations permettant à l'interlocuteur de comprendre le contexte général et ce qui va être demandé.

Puis les objectifs sont transmis afin de préparer la demande d'assistance. A cette étape, cela peut être des objectifs complets ou des sous-objectifs en fonction de ce que cherche à résoudre l'agent.

Enfin l'étape d'évaluation permet de confirmer ou d'infirmer la capacité de l'autre agent à aider à résoudre le problème posé. Il est clair que cela dépend beaucoup du contexte d'exécution et de la nature intrinsèque des agents. Encore une fois, il s'agit du cas le plus général, valable si ce sont des agents hétérogènes. En cas de modélisation d'agents homogènes, certaines étapes peuvent être contournées.

6.4.3/ MISE EN ŒUVRE

La figure 6.8 présente le synoptique de l'étape de **Mise en œuvre**. Nous rappelons ici que nous considérons la collaboration comme un processus global, résultant de la dynamique interne de l'agent jusqu'à sa bonne exécution. Ici il s'agit donc plus précisément de sa mise en œuvre effective. Une fois la communication établie et les premiers échanges effectués, confirmant l'intérêt à collaborer, il faut l'organiser et l'exécuter. Nous retrouvons les étapes suivantes : **organisation de la collaboration**, **exécution des tâches** puis **évaluation des résultats**; s'ils sont atteints, fin de la collaboration, sinon il faut évaluer la capacité à continuer le processus avec l'étape **poursuite du processus d'exécution/évaluation**.

La planification réelle de la collaboration est élaborée par les deux entités avec, d'une part, l'organisation de la collaboration qui comprend le découpage du problème, les affectations de tâches, les attentes réciproques et les processus de négociation et d'autre part, la distribution des rôles qui définit les objectifs de la collaboration pour chacun ainsi que les récompenses à attribuer (algorithme de coûts et bénéfices).

La partie principale de l'algorithme global est l'étape d'exécution. Chaque entité suit le plan de tâche assigné. L'évaluation des résultats, en les comparant au plan initial, permet de vérifier si les objectifs sont atteints. Puis un autre processus permet de décider de la poursuite de la collaboration en utilisant les stratégies d'engagement [Cohen et al., 1990b], utiles pour évaluer s'il est nécessaire, indispensable ou inutile de chercher plus avant. Premièrement, l'engagement aveugle (*blind commitment*) où l'intention de parvenir à une solution est plus forte que tout le reste au mépris des contraintes externes. Deuxièmement, l'engagement "monomaniaque" (*single-minded commitment*) où une entité continue à maintenir une intention jusqu'à ce qu'elle estime que celle-ci a été satisfaite ou qu'il n'est plus possible de poursuivre. Troisièmement, l'engagement

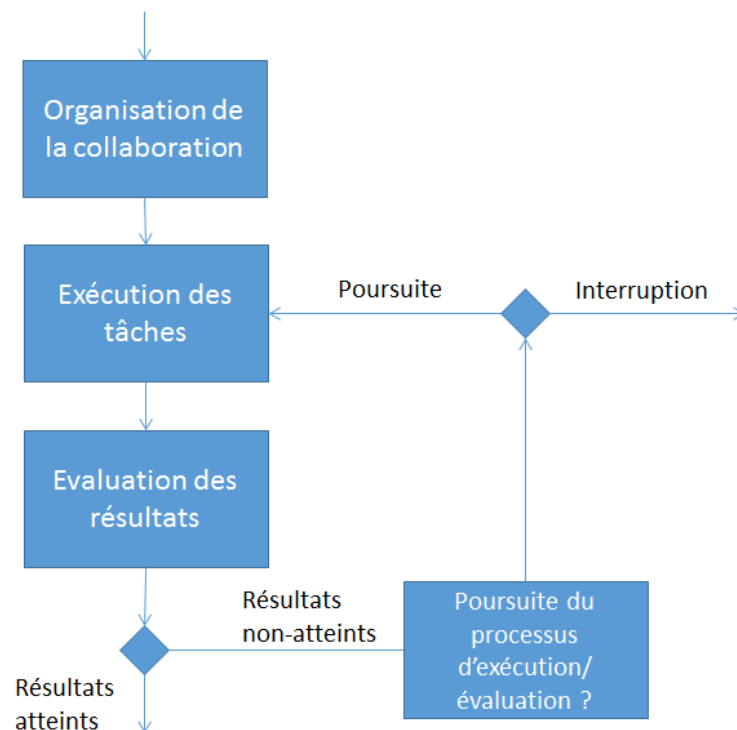


FIGURE 6.8 – Synoptique de mise en œuvre

"ouvert d'esprit" (*open-minded commitment*) lorsque l'entité continue aussi longtemps qu'elle pense qu'il est possible d'obtenir un résultat ou jusqu'à ce qu'elle change d'opinion si d'autres objectifs prennent le pas. Nous rappelons cependant que la collaboration est un processus intentionnel d'engagement. L'agent est tenu de réaliser sa part sauf s'il considère la chose comme impossible.

6.4.4/ CAPITALISATION

Pour finir, la figure 6.9 présente le synoptique de l'étape **Capitalisation**. Les trois dernières étapes sont : **bilan de la collaboration**, **mise à jour de la base de faits et des connaissances** et **historisation des échanges**.

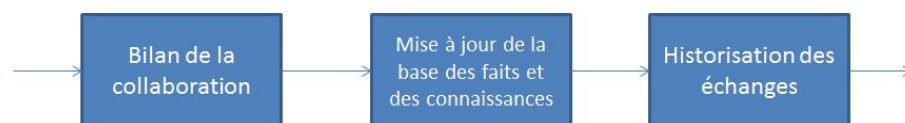


FIGURE 6.9 – Synoptique de capitalisation

A l'heure du bilan, les différents résultats sont examinés au regard des objectifs identifiés au début du processus. Ceci permet d'attribuer les récompenses aux différents intervenants (comme par exemple, un indice de satisfaction ou de réputation).

La base de connaissances est ainsi mise à jour avec les informations recueillies et ajoutées aux connaissances respectives des entités ayant participées au processus de collaboration et, afin de pouvoir capitaliser efficacement, nous ajoutons une historisation des échanges et des réalisations obtenues pour effectuer une synthèse des relations

Algo. \ Collab.	Complète	Partielle	Altruiste	Forcée
Evaluation de l'entité rencontrée (seulement si ce sont des entités hétérogènes)	Il faut que les 2 entités se comprennent parfaitement	Il faut a minima un vocabulaire commun	Il faut a minima un vocabulaire commun	L'entité contactée doit seulement pouvoir comprendre les ordres donnés
Echange d'informations	Toute information de contexte	Seulement les informations à échanger	L'agent altruiste reçoit les informations nécessaires à sa proposition d'assistance	Seulement les informations à transmettre
Détail des objectifs	Détermination d'objectifs communs	Seulement considération de l'objectif global	Seulement l'objectif de l'agent assisté	Seulement l'objectif de l'agent forçant la collaboration
Evaluation de la capacité à résoudre le problème (seulement si ce sont des entités hétérogènes)	Il faut que les 2 entités se comprennent parfaitement	Il faut a minima un vocabulaire commun	Il faut a minima un vocabulaire commun	L'entité contactée doit seulement pouvoir comprendre les ordres donnés

TABLE 6.5 – Synthèse de l'influence du gradient sur les algorithmes (partie 1)

entre entités lors de la collaboration. Nous retrouvons donc bien toutes les composantes de la collaboration telle que nous l'avons défini : AGENTS, INFORMATIONS, COMPORTEMENTS, PROTOCOLES, OBJECTIFS et RESULTATS.

6.5/ INFLUENCE DU GRADIENT SUR LES ALGORITHMES

En synthèse, nous proposons les tableaux suivants (cf. tables 6.5 et 6.6) reprenant les différentes phases d'une collaboration au regard du gradient.

Il convient de noter que, dans les phases préparatoires, deux étapes peuvent être complètement ignorées si la collaboration est mise en œuvre entre deux entités homogènes. En effet, disposant d'un vocabulaire commun et des comportements identiques, il n'est pas nécessaire de passer par l'évaluation de l'entité rencontrée ou de la capacité à résoudre un problème.

Les phases d'échanges d'information et d'objectifs reprennent les éléments déjà abordés dans les sections précédentes (cf. tableau de synthèse du gradient de collaboration 6.2).

Pour les autres phases du processus, selon le gradient choisi, il y a une simplification du fonctionnement pour la phase d'organisation de la collaboration car elle devient inutile s'il s'agit d'un comportement altruiste ou forcé par exemple. La décision de poursuivre ou non une collaboration est également modifiée car, par exemple, en cas de comportement forcé, l'agent n'a plus son libre-arbitre.

6.6/ CE QU'IL FAUT RETENIR

La figure 6.10 résume notre vision de la collaboration telle que nous l'imaginons. Il s'agit d'un processus intentionnel et cognitif avec un partage d'informations dans le cadre d'une vision commune. Cela concerne au moins deux agents éprouvant le besoin de collaborer pour sortir d'une situation difficile et simplement améliorer ses capacités par la recherche d'interactions ou d'opportunités.

Nous définissons également deux types de caractérisation du processus collaboratif : une caractérisation fonctionnelle influencée par l'état mental et physique de l'état mais aussi une caractérisation technique influencée par la position de l'agent dans l'environnement. Toute cette combinatoire conditionne le type d'échange réalisé entre les agents (notamment la qualité des informations et du contexte transmis) ainsi que les différents couches de programmes à exécuter. Nous décrivons dans le [chapitre 7](#) une méthode d'analyse de problème pour mettre en évidence les comportements collaboratifs et préciser quelle collaboration peut être conduite selon le niveau de réaction et d'engagement.

Comme nous l'avons vu précédemment, nous confirmons également ici que nous voyons la collaboration comme un tout qui va de l'intention à la réalisation de la collaboration. Il y a donc un lien indéniable entre les notions d'agent, d'information, de protocole d'échange, d'objectif et de résultat. Cela conduit à structurer et à contextualiser l'information comme nous le voyons dans le [chapitre 8](#).

La figure 6.11 reprend donc les grandes étapes du processus de collaboration tel que nous l'avons décrit précédemment. Celui-ci correspond à une collaboration "idéale" entre deux entités non forcément homogènes.

Algo. \ Collab.	Complète	Partielle	Altruiste	Forcée
Organisation de la collaboration	En fonction des objectifs communs par dialogue	Seulement prise en compte de l'objectif global, organisation dépendante des informations échangées	L'agent altruiste se plie à l'organisation proposée	Imposée
Exécution des tâches	Exécution synchronisée	Chaque agent a ses tâches (seules les informations les influencent)	Tâches proposées	Tâches imposées
Evaluation des résultats	Re-priorisation éventuelle des tâches	Sans objet	Pas obligatoire (car proposition de service)	Sans objet car tâches imposées
Poursuite du processus	Engagement fort, l'agent essaiera de maintenir la collaboration tant que résultats non-atteints	En fonction des attentes	Engagement faible, en fonction des attentes	Toujours vrai dans ce cas, l'agent forcé n'a plus son libre-arbitre
Bilan de la collaboration	Détermination des récompenses communes	Sans objet (sauf informations échangées)	Indice de réputation de l'agent altruiste en progression	Indice de réputation de l'agent forçant impacté (dégradation)
Mise à jour des connaissances	Avec les informations recueillies	Avec les informations échangées	Avec les informations reçues	Avec les informations imposées
Historisation des échanges	Enregistrement de l'historique de conversation (si mémorisation)	Enregistrement de l'historique de conversation (si mémorisation)	Enregistrement de l'historique de conversation (si mémorisation)	Enregistrement de l'historique de conversation (si mémorisation)

TABLE 6.6 – Synthèse de l'influence du gradient sur les algorithmes (partie 2)

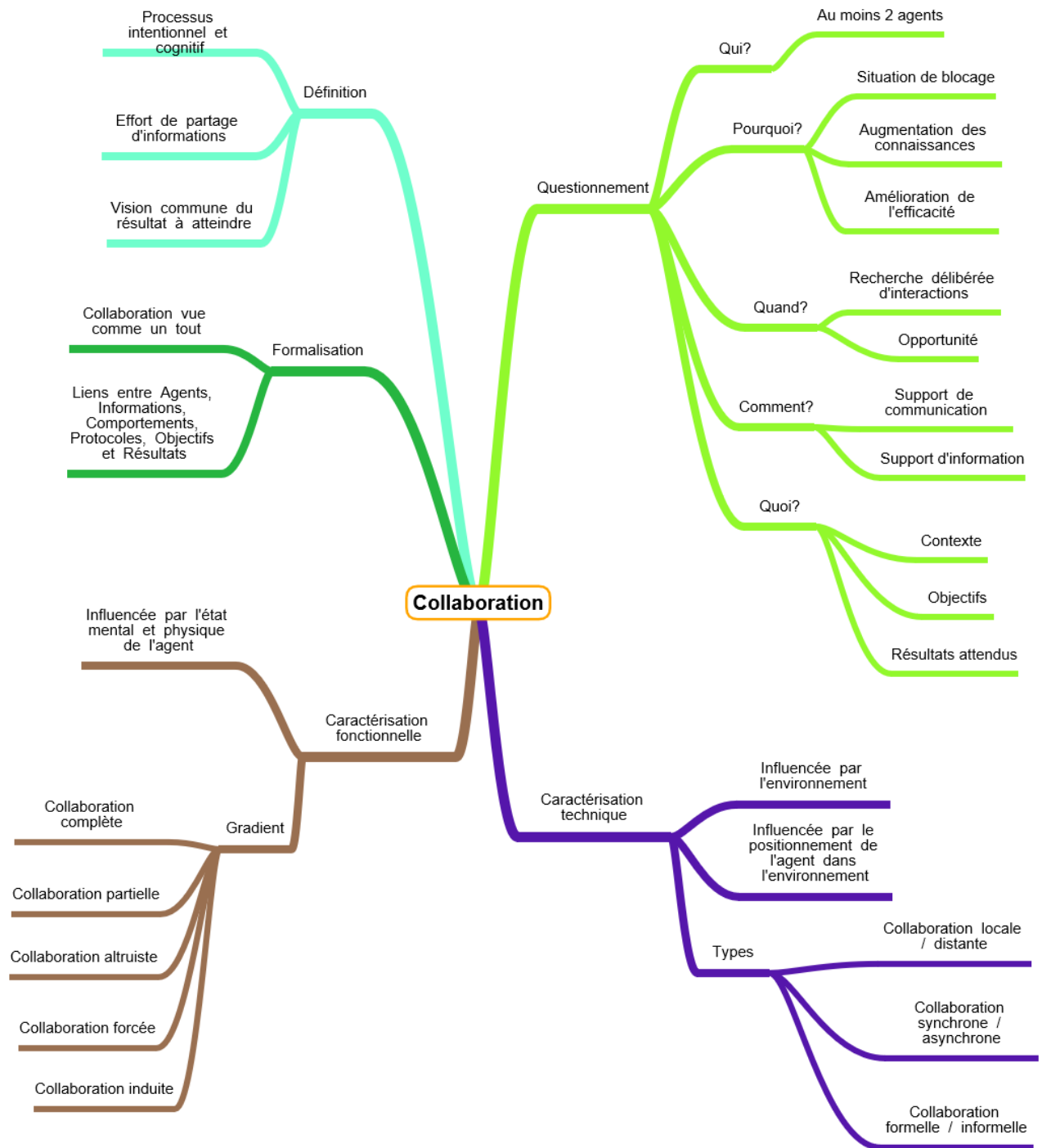


FIGURE 6.10 – Ce qu'il faut retenir : Vue synthétique de notre vision de la collaboration

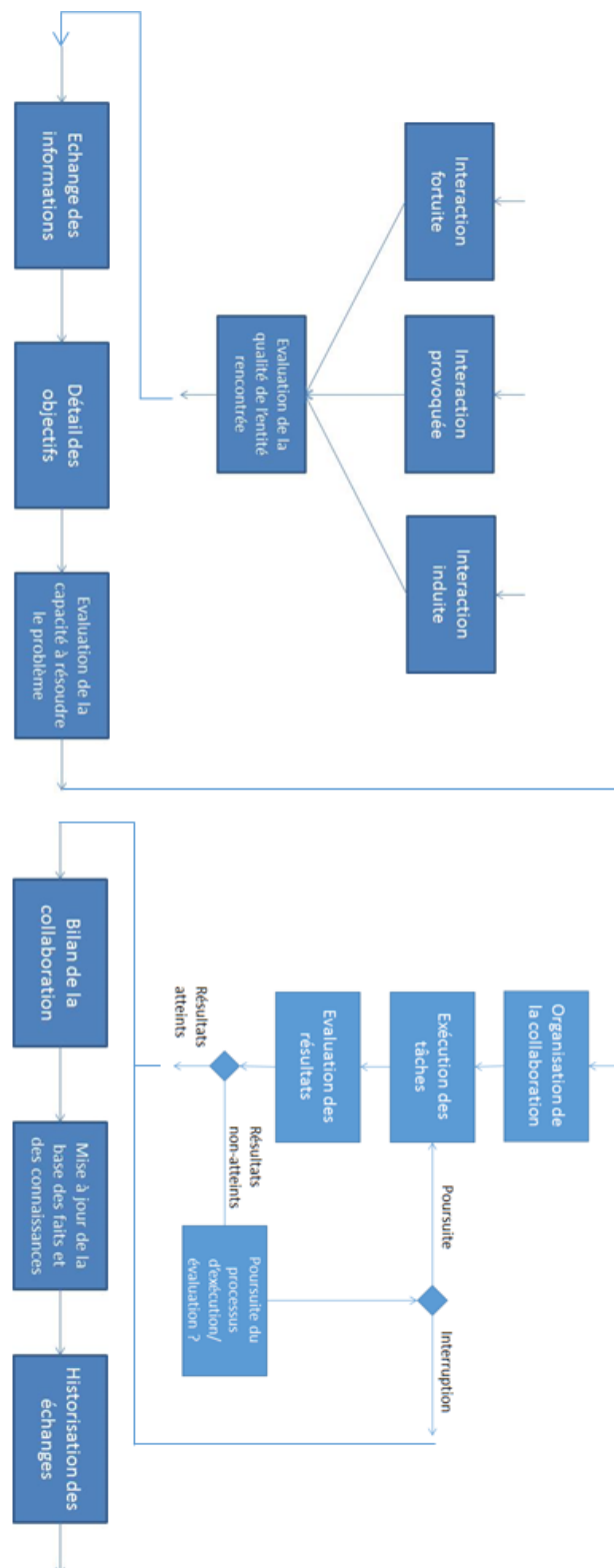


FIGURE 6.11 – Algorithme général d'une collaboration

MÉTHODE D'ANALYSE

Sommaire

7.1 Démarche	106
7.2 Présentation du cas d'application	107
7.3 Première itération : Analyse du problème et caractérisation	107
7.3.1 Données manipulées	107
7.3.2 Comportements	108
7.3.3 Interactions	108
7.3.4 Collaborations	109
7.3.5 Constat	109
7.4 Deuxième itération : Amélioration des comportements	109
7.4.1 Données manipulées	110
7.4.2 Comportements	110
7.4.3 Interactions	111
7.4.4 Collaborations	111
7.4.5 Constat	112
7.5 Troisième itération : Optimisation	114
7.5.1 Données manipulées	114
7.5.2 Comportements	114
7.5.3 Interactions	114
7.5.4 Collaborations	115
7.5.5 Constat	115
7.6 Formalisation	115
7.7 Ce qu'il faut retenir	117

Nous avons besoin de déterminer quels comportements peuvent faire l'objet d'optimisation par collaboration. Dans ce chapitre, nous proposons donc une méthode d'analyse de problèmes afin de décrire la structure d'un agent, ses fonctionnalités et de mettre en évidence les possibilités de collaboration (à l'aide de notre exemple fil rouge de drones de surveillance de forêt).

Tout d'abord, nous justifions notre démarche puis, après présentation du cas d'application, nous déroulons notre méthode en trois itérations successives sur les comportements afin de montrer les processus collaboratifs pouvant être mis en œuvre. Enfin, nous formalisons cette méthode pour pouvoir l'utiliser ultérieurement.

7.1/ DÉMARCHE

Globalement il faut, avant toute chose, poser le cas d'application et lister les caractéristiques intrinsèques de l'agent puis reprendre un à un les éléments le composant pour définir l'ensemble de ses comportements clés. La méthode d'analyse permettra ensuite de déterminer quels sont les réactions à affiner et les comportements à modifier en vue d'une meilleure adaptabilité de l'agent à son environnement et d'une amélioration de sa capacité à compléter ses objectifs.

Ensuite, il faut définir, dans le cadre des missions principales, quelles pourraient être les comportements pouvant être modifiés pour mettre en œuvre les différentes étapes de la collaboration et décrire les effets attendus en termes de résultats (notamment les gains capitalisés).

Il existe déjà de nombreuses méthodes. Nous pouvons citer par exemple ADELFE ([Bernon et al., 2002], [Picard et al., 2004]) et INGENIAS ([Pavón et al., 2005]) reposant sur le processus unifié (Unified Process, UP [Jacobson et al., 1999]) ou encore ASPECS ([Gaud, 2007]) reposant sur un processus itérable. La méthode INGENIAS est orientée buts, ASPECS plutôt orientée vers des systèmes holoniques. Mais la méthode ADELFE est dédiée aux agents coopératifs. Développée à partir des AMAS (Adaptative Multi Agent Systems) ([Gleizes et al., 1999], [Georgé et al., 2003]), elle fournit un méta-modèle de conception et de développement.

Cependant, dans le prolongement des [chapitres 5 et 6](#), nous allons continuer à nous inspirer de la méthode des quadrants notamment pour ce qui est relatif aux différentes lois (cf. figure 7.1) : dynamique interne (fonctionnement intime de l'agent, de son esprit, de ses processus de décision), lois de réaction/d'évolution (fonctionnement de son corps, transposition de son esprit dans l'action), lois d'interférences (interactions avec les autres agents) et règles constitutives/régulatrices (constituant la culture commune entre les agents et base des processus collaboratifs).

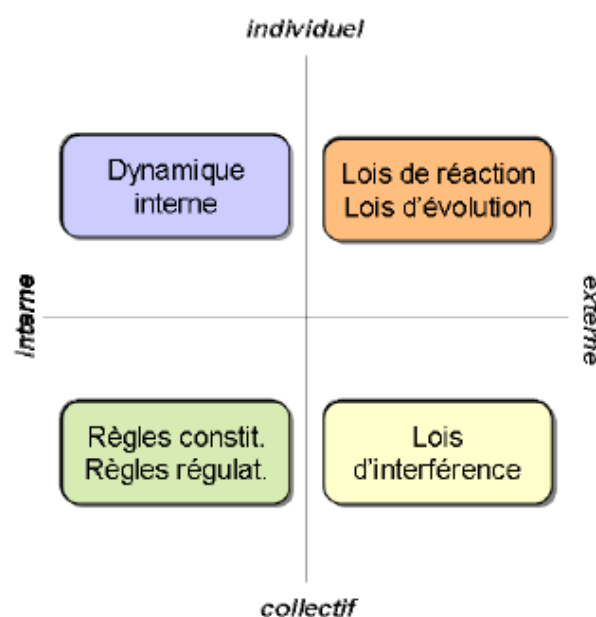


FIGURE 7.1 – Différentes lois dans les quadrants ([Tranier, 2007])

Comme dans la méthode Agile, nous allons fonctionner par itération. Une première itération permettra de fixer le comportement standard attendu de l'agent puis deux autres itérations successives permettront d'affiner les comportements et déterminer les améliorations souhaitables pour optimiser les missions de l'agent et donc son efficacité intrinsèque. Il convient de noter que chaque itération peut faire l'objet de simulations pour tester les comportements et faciliter le constat.

Cette méthode d'analyse ne se substitue pas à celle de Volker Grimm avec le protocole ODD par exemple ([Grimm et al., 2006] et [Grimm et al., 2010]) qui permet de décrire un modèle d'agents. Elle est complémentaire. En effet, nous voulons montrer, grâce aux quadrants, le cheminement intellectuel pour arriver à décrire et mettre en place des processus collaboratifs.

7.2/ PRÉSENTATION DU CAS D'APPLICATION

Comme nous l'avons présenté dans le [chapitre 5](#), notre exemple fil rouge est une flotte de drones de surveillance de forêt et de détection d'incendie. Pour rappel, il s'agit de petits engins de type quadricopteur ayant une autonomie limitée d'une quinzaine de minutes (obligation de revenir fréquemment à la base pour recharger) et pouvant s'élever à quelques centaines de mètres. Chaque engin dispose d'une caméra pouvant observer l'environnement à 360 degrés. Cependant, la taille et la résolution de celle-ci ne lui permet pas d'observer sur des kilomètres. La zone d'observation est donc limitée. Nous lui ajoutons aussi une capacité limitée de communication. Cela lui servira à transmettre des informations sans pour autant revenir systématiquement à sa base.

Les deux missions principales sont : explorer la forêt en vue de détecter des incendies et prévenir les secours. Cela peut se traduire en termes d'objectifs de la manière suivante : couvrir le maximum de surface à chaque cycle de batterie et alerter le plus vite possible les secours pour minimiser les surfaces incendiées et éviter une propagation trop rapide.

Deux indicateurs de suivi et de performance peuvent être utilisés : surface couverte (à maximiser), surface forêt incendiée (à minimiser).

Maintenant que le problème est posé, passons à l'analyse des comportements et des interactions et observons l'émergence des possibilités de collaboration.

7.3/ PREMIÈRE ITÉRATION : ANALYSE DU PROBLÈME ET CARACTÉRISATION

Pour cette première itération, nous allons poser le problème et décrire le comportement basique attendu de l'agent. Il s'agit ici de la description du contexte standard sans optimisation préalable.

7.3.1/ DONNÉES MANIPULÉES

Il s'agit de gérer principalement l'état de la batterie pour arbitrer la mission en cours : patrouiller pour éventuellement alerter en cas de détection d'incendie ou rentrer à la

base pour recharger.

Ce sont des entités mobiles, il convient donc également de gérer la localisation de celles-ci dans l'environnement à chaque instant. Cette information permettra de gérer le retour à la base en situant le drone par rapport à celle-ci.

Enfin, que faisons-nous du cap éventuel à suivre pour mener à bien la mission d'exploration ? Pour le moment, celui-ci est accessoire et nous pouvons imaginer qu'il est géré de manière aléatoire pour permettre une couverture satisfaisante de l'environnement.

A noter enfin que la vitesse du drone est considérée comme constante. Encore une fois, il s'agit de décrire le système le plus simple possible. Nous passerons aux raffinements dans les sections suivantes.

7.3.2/ COMPORTEMENTS

Comme nous sommes sur une modélisation d'un système simple où la poursuite de la mission n'est conditionnée que par le niveau de batterie résiduel, les deux premiers comportements qui viennent à l'esprit sont ceux qui vont gérer la batterie du drone : un comportement d'interrogation de son niveau (*Contrôle Etat*) et un comportement exécutant la fonction de rechargement (*Recharger*).

Puis nous pouvons décrire les comportements propres à la nature du drone : *Voler*, *Observer* et *Communiquer*.

Le drone ne peut voler que si le niveau de batterie est suffisant pour le faire. Le cap est déterminé de manière aléatoire.

Le drone peut observer à 360 degrés autour de lui sur quelques centaines de mètres avec précision. Dans sa zone d'observation, il est capable d'identifier un incendie. Si c'est le cas, il déclenche l'exécution du comportement *Alerter* avec l'objectif de transmettre la localisation de l'incendie le plus rapidement possible.

Le drone peut communiquer un ou plusieurs messages vers un destinataire si celui-ci se situe dans la zone de communication.

Sur ces comportements, nous pouvons en créer d'autres intégrant ces briques dans un ensemble plus élaboré : *Patrouiller* et *Alerter*.

Le comportement *Patrouiller* est l'utilisation combinée des comportements *Voler* et *Observer*.

Le comportement *Alerter* combine quant à lui les comportements *Voler* vers la base et *Communiquer* la localisation de l'incendie à la base pour l'envoi des secours.

7.3.3/ INTERACTIONS

Le drone a des interactions avec l'environnement : dans le processus de vol, d'observation et de détection. A noter que selon le choix du concepteur, on pourrait considérer une zone d'incendie comme un agent avec sa propre dynamique de propagation.

Cependant il n'y a pas d'interaction avec les autres drones constituant la flotte de surveillance. Ces comportements ne sont pas programmés et ne sont pas utiles dans ce cas. Nous pouvons considérer que la base en récupérant les informations de chaque

drone, centralise les décisions d'envoi des secours.

Devons-nous considérer la base comme un agent ou seulement un espace de dépose d'information ? A ce stade, il n'est pas utile d'aller plus en avant. Les missions sont remplies.

7.3.4/ COLLABORATIONS

Pour cette première passe, il n'y a pas collaboration détectable, le drone est programmé pour une tâche et l'effectue dans un mode action/réaction sans entrave. Il n'a pas de difficulté *a priori* à remplir sa mission.

7.3.5/ CONSTAT

Nous sommes donc clairement dans un contexte d'agent réactif. Il n'y a pas de processus de mémorisation des positions du drone et des drones, seulement une information volatile sur les coordonnées d'une zone d'incendie transmise immédiatement et consommée dès réception par la base.

Les deux missions principales (explorer la forêt en vue de détecter des incendies et prévenir les secours) sont effectivement remplies mais qu'en est-il de l'efficacité du drone ?

Rappelons les objectifs : couvrir le maximum de surface à chaque cycle de batterie et alerter le plus vite possible les secours pour minimiser les surfaces incendiées et éviter une propagation trop rapide. Il apparaît aisément que le fait de se déplacer aléatoirement dans l'environnement n'est pas la garantie de pouvoir couvrir le maximum de surface. Les drones peuvent en effet couvrir les mêmes zones plusieurs fois très rapidement. L'information en sera-t-elle plus pertinente ? De même, comment améliorer la transmission d'alerte si celle-ci ne peut se faire qu'au prix d'un retour rapide à la base (ou du moins, dans la zone de communication de celle-ci) ?

En résumé, nous sommes à un niveau opérationnel, d'action / réaction en fonction des observations et de l'état interne du drone (gestion de la batterie). La table 7.1 reprend les comportements à l'issue de cette première itération d'analyse.

7.4/ DEUXIÈME ITÉRATION : AMÉLIORATION DES COMPORTEMENTS

Fort des constats de la section précédente, nous allons chercher à améliorer les comportements en nous questionnant à chaque fois sur la meilleure façon d'agir, dans l'intérêt de la mission et en réponse aux objectifs posés.

Il faut améliorer la manière d'explorer en maximisant la surface couverte par cycle. Concrètement, cela veut dire mémoriser les lieux et les observations pour décider du meilleur cap. D'autre part, il faut modifier le système d'alerte pour le rendre plus efficace, en transmettant l'information à d'autres drones qui pourront, de proche en proche, transmettre plus rapidement les informations sur les zones d'incendie à la base.

Couche	Comportement	Variables en entrée	Action	Variables en sortie
Opérationnelle	Contrôle Etat		Gestion/Contrôle de la batterie	Niveau Batterie modifié à la baisse (consommation)
Opérationnelle	Voler		Voler vers base (besoin de recharger) ou Cap aléatoire	Localisation Drone modifiée
Opérationnelle	Observer		Observation dans la zone de vision	Localisation Incendie initialisée si détection
Opérationnelle	Communiquer	Message, Destinataire	Envoi du Message / Dialogue avec le Destinataire	
Opérationnelle	Patrouiller		Voler et Observer	
Opérationnelle	Alerter	Localisation Incendie	Envoi de l'information de localisation de l'incendie à la base le plus rapidement possible	Localisation Incendie supprimée (si communication réussie)
Opérationnelle	Recharger		Rechargement du drone à la base	Niveau Batterie modifié à la hausse (recharge)

TABLE 7.1 – Synthèse des comportements à la fin de la première itération

7.4.1/ DONNÉES MANIPULÉES

Nous ne reviendrons pas sur les informations déjà évoquées dans la section 7.3.1 comme la gestion de la batterie.

Par contre, le cap qui était choisi de manière aléatoire va devenir une données qui va évoluer dans le temps grâce à un processus de décision qui est explicité dans la section suivante.

Si le drone doit mémoriser son environnement, il faut utiliser une structure d'informations pouvant contenir chaque parcelle de terrain découverte et explorée ainsi que les zones d'incendie. Cela pourra être un graphe ou une grille selon le choix du développeur et le type de modeleur utilisé.

7.4.2/ COMPORTEMENTS

Le comportement clé de cette deuxième itération est donc : *Cartographier*.

L'objectif est d'enregistrer toutes les informations visibles pour constituer une *carte mentale* de l'environnement. Cela ne sera en rien une représentation absolue de l'environnement, seulement un instantané connu du drone et contenant également des informations temporelles afin d'en mesurer le degré de pertinence par la suite. En effet, si des informations sont trop anciennes, on peut douter légitimement de la valeur de celles-ci, notamment dans le cas de l'enregistrement de zones d'incendie qui, non-traitées, vont s'étendre de plus en plus.

Une fois que le drone possède ces informations, il va pouvoir s'en servir pour déterminer le meilleur cap possible (*Définir Cap*). Il va analyser sa *carte mentale* pour aller : là où il n'est pas encore allé, là où les informations sont les plus anciennes afin de les rafraîchir.

Cependant, même si la constitution individuelle et autonome d'une *carte mentale* va contribuer à améliorer le choix de la prochaine zone d'exploration, nous pouvons aller d'ores et déjà plus loin. Il est tout à fait envisageable de faire communiquer les drones entre eux (via le comportement *Communiquer*) afin d'échanger leur carte mentale pour contribuer à une meilleure vision d'ensemble. Ainsi, si des drones sont à proximité, il faut mettre en place des processus d'échange.

Par extension, cela va conduire à modifier le comportement *Observer* décrit dans la section 7.3.2. Il faut ajouter l'étape de cartographie dans le programme d'observation. Nous allons aussi modifier le comportement *Voler* pour intégrer la possibilité de suivre le cap défini par lecture de la *carte mentale*.

Maintenant que nous avons amélioré le système d'exploration, passons au système d'alerte (*Alerter*). En effet, nous allons utiliser le système de communication (*Communiquer*) pour transmettre la zone incendiée à la base le plus rapidement possible. Pour réaliser cette action, il ne faut plus seulement chercher à rentrer à la base le plus vite possible mais aussi propager l'information en forçant un drone devenir un relais de communication. Celui-ci sera peut-être plus en mesure d'accomplir rapidement lui-même la mission ou de transmettre l'information à d'autres drones dans son champ de perception.

7.4.3/ INTERACTIONS

Pour cette deuxième itération, en plus des interactions directes avec l'environnement physique, les interactions entre drones sont désormais manifestes mais restent encore aléatoires car dépendantes de la manière d'explorer l'environnement.

Toutefois, il apparaît nécessaire de construire un processus d'interaction un peu plus complexe qu'une simple détection pour échanger les informations nécessaires à une meilleure prise de décision individuelle ou pour transmettre un message contenant une zone incendiée à la base.

7.4.4/ COLLABORATIONS

Nous avons mis en évidence deux processus d'échange : pour cartographier et pour alerter. Questionnons-nous maintenant pour en déterminer les caractéristiques.

Dans le cas de la cartographie, deux agents drones participent au même moment à l'échange d'informations dans le but commun de se constituer plus rapidement une *carte mentale* précise de l'environnement et la plus à jour. Chaque échange fera l'objet d'un

Comportement	Informations	Collaboration fonctionnelle	Collaboration technique
Cartographe	Carte mentale	Collaboration partielle	Collaboration locale et synchrone
Alerter	Zone d'incendie	Collaboration forcée	Collaboration locale et synchrone
	Zone d'incendie (de proche en proche)	Collaboration induite	Collaboration distante et asynchrone

TABLE 7.2 – Synthèse des collaborations issues de la deuxième itération d'analyse

arbitrage individuel ou collectif lors de la mise à jour de la carte en ne considérant que l'information la plus fraîche. En reprenant les différents cas décrits dans le [chapitre 6](#), nous sommes en présence d'une collaboration partielle avec un échange limité d'information de contexte autour d'un objectif partagé. De plus, les aspects techniques soulignent qu'il s'agit d'une collaboration locale et synchrone.

Dans le cas de la transmission d'alerte, deux agents drones participent à l'échange mais la communication circule en sens unique car il s'agit d'apporter l'information le plus vite possible au destinataire final qu'est la base afin d'envoyer les secours. Même si l'objectif commun est partagé (transmission d'alerte), il n'y a pas réellement de négociation. Le contexte d'exécution suggère une collaboration forcée entre les deux drones. Celui qui a l'information prend le dessus sur celui qui la reçoit. Mais c'est aussi une collaboration induite dans le sens où même si le drone recevant l'information n'est pas à l'origine de la découverte de la zone d'incendie, en transmettant de proche en proche, il va faire collaborer entre eux tous les relais de communication au profit du drone à l'initiative de la découverte. Dans le premier cas, (collaboration forcée), c'est aussi techniquement une collaboration locale et synchrone alors que dans le deuxième cas (collaboration induite), c'est une collaboration distante et asynchrone.

La table 7.2 résume ces différentes collaborations à l'issue de cette deuxième itération d'analyse.

7.4.5/ CONSTAT

Nous avons mis en évidence l'amélioration : du processus d'exploration grâce à la mémorisation des lieux et des échanges inter-drones, du processus d'alerte grâce aux communications inter-drones.

Le drone n'est plus dans un seul processus d'action/réaction mais également dans un processus de décision plus élaboré. Les échanges et collaborations entre les drones, au départ individualistes, permettent d'introduire des comportements sociaux grâce aux échanges sur les données spatio-temporelles afin de co-construire leur *carte mentale*.

Même si les deux missions principales (explorer la forêt en vue de détecter des incendies et prévenir les secours) sont mieux remplies, sommes-nous arrivés au bout du raisonnement ? Par exemple, il n'y a pas encore de prise en compte plus large avec notamment la mise en place d'une coordination globale des drones par échange de plan d'exploration.

En résumé, nous sommes désormais à un niveau tactique avec des processus d'action

Couche	Comportement	Variables en entrée	Action	Variables en sortie
Opérationnelle	Voler	Cap	Voler vers base (besoin de recharger) ou Cap défini	Localisation Drone modifiée
Opérationnelle	Observer		Observation dans la zone de vision et Cartographie	Localisation Incendie initialisée si détection, Zone Observée (pour mémorisation)
Opérationnelle	Alerter	Localisation Incendie	Envoi de l'information de localisation de l'incendie à la base le plus rapidement possible, en passant éventuellement par l'intermédiaire d'un autre drone à proximité (collaboration forcée)	Localisation Incendie supprimée (si communication réussie)
Tactique	Cartographier	Zone Observée	Mise à jour de la carte mentale, Communication Carte Mentale si d'autres drones à proximité (collaboration partielle)	Carte Mentale modifiée avec l'ajout de Zone Observée
Tactique	Définir Cap	Carte Mentale	Analyse de la carte mentale et des zones déjà observées pour définir un cap, tenir compte également des temporalités	Nouveau Cap

TABLE 7.3 – Synthèse des comportements modifiés et ajoutés à la fin de la deuxième itération

et de décision plus efficace. La table 7.3 reprend les comportements ajoutés et modifiés à l'issue de cette deuxième itération d'analyse.

7.5/ TROISIÈME ITÉRATION : OPTIMISATION

Maintenant que nous avons décrit tous les comportements du drone et des améliorations possibles, par l'emploi de processus de mémorisation et d'échanges structurés, nous allons enfin voir s'il est possible de les optimiser une dernière fois et si oui, de quelle manière et dans quel but.

Nous avons conclu dans la section précédente qu'il était peut-être encore possible d'améliorer l'exécution de la mission de surveillance en laissant les drones s'auto-organiser afin de définir collectivement un plan de patrouille.

7.5.1/ DONNÉES MANIPULÉES

Il faut compléter les connaissances des agents drones en leur permettant de mémoriser des plans de vol complets et donc des zones à explorer. Cela peut se faire par ajout d'information sur la *carte mentale* et la création de variables *ad hoc*. Le choix sera laissé à la discrétion du développeur.

7.5.2/ COMPORTEMENTS

Nous pouvons créer un nouveau comportement pour permettre la création collective d'un plan de patrouille : *Construire Plan Patrouille*.

En cas de détection d'un drone à portée de communication, un dialogue peut s'établir. Il y a tout d'abord l'échange des informations de *carte mentale* par l'intermédiaire du comportement *Cartographier*. On peut ajouter un dialogue plus fourni entre les drones afin qu'ils se mettent d'accord sur une répartition spatiale du terrain à couvrir, au pire de grandes zones, au mieux un plan de vol complet en utilisant le comportement *Communiquer*. Bien évidemment, les drones sont libres d'accepter ou refuser une proposition. L'objectif est d'être collectivement d'accord pour que le plan de patrouille soit enregistré et suivi. Il peut y avoir une ou plusieurs fonctions d'utilité pour gérer les acception/refus (notamment si un précédent plan a été récemment élaboré avec un autre drone).

La création d'un plan de patrouille ou de zones à explorer a une influence sur le comportement *Définir Cap* qui doit désormais, en plus de l'analyse de la *carte mentale* et des temporalités, prendre en considération ces nouvelles zones pour définir le meilleur cap à suivre.

7.5.3/ INTERACTIONS

Dans ce cas de figure, nous sommes dans la situation la plus complexe où l'interaction ne se limite plus à un simple échange mais à la construction d'un dialogue jusqu'à la réussite ou l'échec de celui-ci.

Comportement	Informations	Collaboration fonctionnelle	Collaboration technique
Construire Plan Patrouille	Plan de vol, zones à explorer	Collaboration complète	Collaboration locale et synchrone

TABLE 7.4 – Synthèse des collaborations issues de la troisième itération d'analyse

7.5.4/ COLLABORATIONS

Pour ce dernier cas, nous sommes en face d'une collaboration complète telle que nous l'avons définie dans le chapitre 6. Deux drones collaborent pour optimiser leur déplacement et donc l'utilisation de leur batterie. Ce n'est pas forcément nécessaire, c'est un choix basé sur une opportunité.

Il y a une mise en commun des buts (zones à explorer initiales pour chaque drone), un partage d'informations (*carte mentale* déjà acquise), une définition d'un plan d'action commun (le futur plan de patrouille avec les zones à explorer pour chacun), un protocole d'échange simple ou complet avec introduction tactique de négociation, maximisation, récompenses (en dehors de la satisfaction des buts). D'ailleurs, comme nous l'avons vu au chapitre 3, nous pouvons nous appuyer sur des protocoles de type Contract-Net mais ce n'est pas exclusif. En effet, nous sommes avec des entités homogènes. Nous pouvons définir une culture commune composée de règles d'échange et d'ontologie pour faciliter les phases de dialogue.

La table 7.4 résume ces différentes collaborations à l'issue de cette troisième itération d'analyse.

7.5.5/ CONSTAT

Pour cette dernière itération d'analyse, nous avons mis en évidence les derniers raffinements pour optimiser les comportements en vue de maximiser la surface à explorer par cycle de batterie.

Il s'agit définitivement d'un niveau stratégique. Il n'est pas essentiel au bon fonctionnement de l'entité mais il porte les comportements les plus collaboratifs possibles. L'agent tire un bénéfice des interactions avec les autres agents. Grâce aux liens sociaux et aux échanges, il optimise son fonctionnement individuel. La table 7.5 résume les comportements ajoutés et modifiés à l'issue de cette troisième itération d'analyse.

7.6/ FORMALISATION

Maintenant que nous avons vu le déroulement du cheminement intellectuel grâce à notre exemple fil rouge, nous pouvons proposer une formalisation et une généralisation de la démarche pour les problèmes à base d'agents mobiles, intelligents et autonomes.

Première itération : Analyse et caractérisation du problème

L'important est de toujours partir du problème dans sa plus simple expression et avec le comportement le plus réactif possible. Il s'agit ici de reprendre les fondamentaux du

Couche	Comportement	Variables en entrée	Action	Variables en sortie
Tactique	Définir Cap	Carte Mentale	Analyse de la carte mentale (zones déjà observées) et des zones à explorer pour définir un cap, tenir compte également des temporalités	Nouveau Cap
Stratégique	Construire Plan Patrouille		Dialogue entre drones pour se répartir les zones à explorer (collaboration complète)	Zones à explorer

TABLE 7.5 – Synthèse des comportements modifiés et ajoutés à la fin de la troisième itération

problème et ne pas chercher à le caractériser finement dès la première itération.

Il faut poser le ou les types d'agent et décrire leur structure interne de manière intuitive. Les missions des agents ne doivent correspondre qu'à la première lecture du problème sans raffinement. Nous restons le plus possible dans un contexte réactif en ne gérant que les comportements basiques. La plupart du temps, nous restons sur des fonctions de type action/réaction avec un module de perception limité. C'est la couche *opérationnelle*.

Cette première itération constitue les fondations du futur modèle sur lesquelles nous allons construire les murs et le toit. En nous aidant de simulations, il est possible d'affiner un premier constat. Qu'est-ce qui fonctionne ? Qu'est-ce qui semble inadapté ? Qu'est-ce qui pourrait être amélioré ?

Deuxième itération : Amélioration des comportements

Pour cette itération, il faut questionner les objectifs afin de déterminer les voies d'amélioration et d'optimisation. Les différents constats mis en évidence dans l'étape précédente permettent de trouver quelques pistes.

Nous sommes ici dans la couche *tactique*. L'objectif est d'améliorer les comportements basiques en ajoutant des raffinements liés à la compréhension de l'environnement et de sa dynamique. Ces comportements peuvent être des modules d'analyse spécifiques de tel ou tel phénomène détecté par la couche opérationnelle. Cependant ces analyses ne doivent pas être gratuites mais contribuées au meilleur fonctionnement de l'agent dans le but de mener ses missions à son terme de manière efficace.

Pour cette étape, il convient d'étudier, en plus de la nature des phénomènes observés, les différentes interactions entre les agents pour déterminer si des échanges structurés sont à envisager pour améliorer les comportements nécessaires à la réussite des objectifs. Cela conduit à lister les informations susceptibles d'être transmises et pour quelles raisons.

Enfin, nous pouvons utiliser ce que nous avons décrit dans le [chapitre 6](#), pour déterminer le meilleur cas de collaboration possible dans le contexte du problème à modéliser (collaboration complète, partielle, altruiste ou forcée). Cela permet d'aider à concevoir les modalités d'échange et aussi à créer les bonnes structures d'enregistrement des données (carte mentale par exemple).

En nous aidant de simulations, il est encore possible d'en tirer un deuxième constat. Qu'est-ce qui fonctionne mieux ? Quel comportement pourrait encore être optimisé ?

Troisième itération : Optimisation

Cette dernière étape constitue la couche *stratégique*. En effet, après avoir caractérisé le problème et mis en évidence les comportements basiques, puis avoir amélioré les comportements en analysant l'environnement pour en souligner les phénomènes clés et ainsi proposer des structures adaptées d'enregistrement des informations, il s'agit ici de s'interroger sur la possibilité d'une optimisation globale de l'efficacité de l'agent sur le moyen/long terme. Quels comportements peuvent faire l'objet d'une analyse plus fine pour permettre à l'agent de parfaire ses missions ?

Encore une fois, cela passe par une analyse plus poussée des interactions entre agents en gardant en mémoire les objectifs et les indicateurs clés de performance. Cette étape sera l'occasion de mettre en œuvre des comportements plus collaboratifs encore en approfondissant les échanges pour établir, si le besoin est constaté, des communautés d'agents pouvant s'entraider à réaliser leurs objectifs par échanges constants et apprentissage social.

Cette dernière itération permettra d'optimiser non seulement le comportement d'un agent mais également d'optimiser le système tout entier.

7.7/ CE QU'IL FAUT RETENIR

La figure 7.2 reprend les principaux éléments de notre méthode d'analyse notamment de donner un cadre de réflexion pour poser le problème, définir les caractéristiques intrinsèques des agents et mettre en évidence les comportements collaboratifs.

Il s'agit d'un processus itératif où chaque étape permet de raffiner les comportements. Nous nous sommes inspirés de la logique des quadrants (dynamique interne, lois de réaction, d'évolution et d'interférences mais aussi règles constitutives et régulatrices) pour décrire les données manipulées, les comportements de l'agent, les interactions avec son environnement et les processus menant à la collaboration.

Avec notre exemple fil rouge, nous avons mis en évidence différents degrés d'action, de réflexion en fonction du contexte spatio-temporel. La communication entre les drones permet un support de la collaboration afin d'optimiser les comportements initiaux et le fonctionnement global du système. Nous observons donc le besoin de gérer plusieurs couches de traitement :

- Réflexe/Opérationnel : voler, observer et communiquer,
- Tactique : éviter la redondance d'information par mémorisation des observations et échange simple avec les autres agents drones, amélioration de l'utilisation de la batterie,
- Stratégique : optimisation de la batterie, augmentation de l'efficacité (prise en

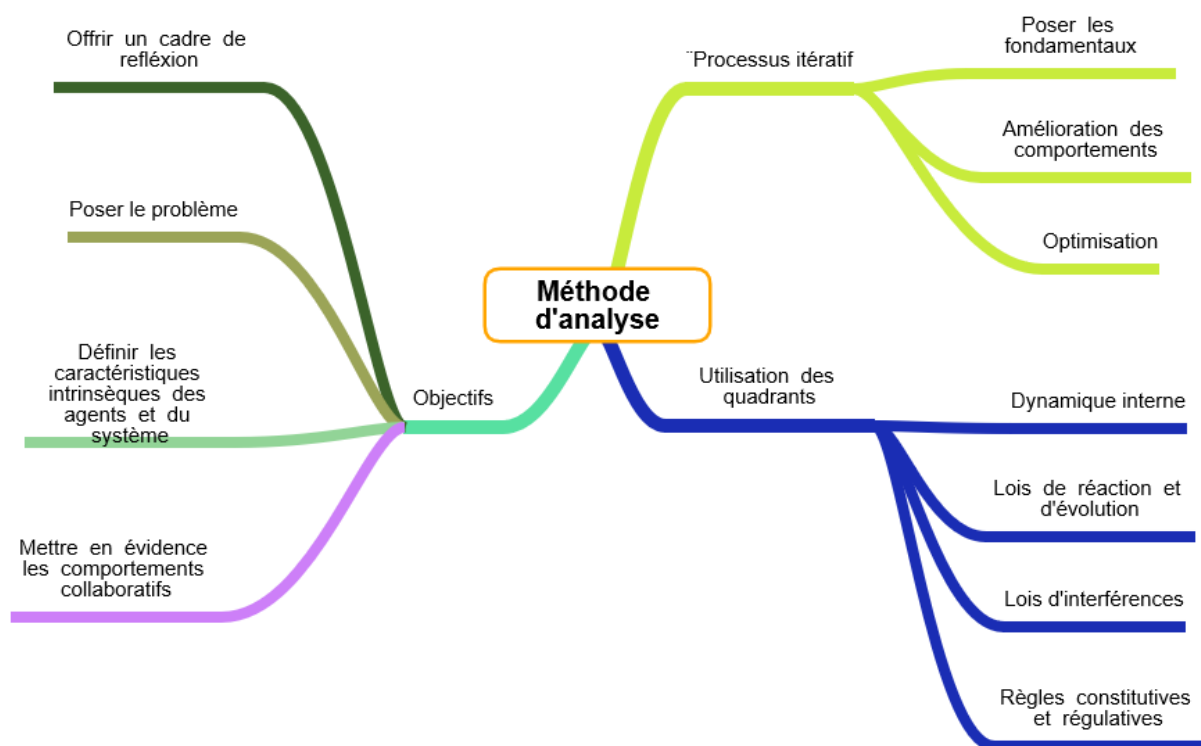


FIGURE 7.2 – Ce qu'il faut retenir : Méthode d'analyse de modèle pour mettre en évidence des comportements collaboratifs

compte des objectifs des autres drones, auto-organisation).

Cette méthode d'analyse nous a donc permis de mettre en lumière le besoin de trois couches de traitement, en fonction du degré de réaction et de réflexion ainsi qu'en fonction des événements. Nous allons reprendre ce concept dans le [chapitre 8](#) pour parler de notre proposition d'architecture. Puis nous réutiliserons cette méthode et sa généralisation dans le [chapitre 9](#) pour décrire l'instanciation du modèle urbain.

Sommaire

8.1 Architecture générale	119
8.1.1 Constat et justification	120
8.1.2 Modèle conceptuel	120
8.1.3 Principes de fonctionnement	121
8.2 Description détaillée	122
8.2.1 Modules de perception et d'action	123
8.2.2 Module d'enregistrement des données	123
8.2.3 Trois couches de traitement	123
8.2.4 Règles de comportement	124
8.2.5 Connaissances de l'agent	126
8.2.6 Communications	128
8.3 Cinématique et exemple d'application	129
8.4 Ce qu'il faut retenir	132

Ce présent chapitre décrit l'une de nos contributions principales : notre proposition d'architecture pour un agent communicant collaboratif dans un contexte d'agent mobile intelligent et autonome.

Tout d'abord nous revenons sur les différents constats évoqués dans les chapitres précédents pour justifier nos choix. Puis une description détaillée de l'architecture HACCA (Hybrid Architecture for Collaborative Communicating Agent) est proposée de manière statique. Enfin, utilisant l'exemple fil rouge, nous montrons la cinématique d'utilisation de cette architecture et notamment la gestion des connaissances et les interactions avec les comportements afin de constituer une carte mentale.

8.1/ ARCHITECTURE GÉNÉRALE

Dans cette section, nous listons les différents points de vue développés précédemment pour appuyer notre vision d'architecture. Nous présentons ensuite le schéma général de celle-ci, d'abord de manière statique puis de manière dynamique.

8.1.1/ CONSTAT ET JUSTIFICATION

Dans notre contexte d'étude (les entités mobiles intelligentes et autonomes), nous avons mis en évidence le besoin d'utiliser plusieurs niveaux de réaction/réflexion en fonction du contexte temporel et physique des agents.

En effet, dans le chapitre 5, nous avons évoqué l'architecture "temps réel" de Michon ([Michon, 1985]) avec ces 3 niveaux (de contrôle, de manœuvre et de stratégie) en fonction du temps de réaction souhaitable.

Puis, comme nous l'avons précisé dans le chapitre 6, nous considérons la collaboration comme un tout (Agents, Informations, Comportements, Protocoles de communication, Objectifs, Résultats). Les informations enregistrées et communiquées doivent être contextualisées pour faciliter la compréhension de l'objectif et du contexte. Nous nous appuyons également sur le concept de carte mentale évoqué dans le chapitre 5 avec l'exemple du projet SafeSpot.

Enfin, dans le chapitre 7, nous avons mis en évidence le besoin de plusieurs niveaux de réaction/réflexion (opérationnel, tactique et stratégique) en fonction également du contexte spatio-temporel mais aussi des besoins d'optimisation par processus collaboratif et par raffinement des comportements basiques des agents.

Il nous a fallu alors imaginer une architecture d'agent hybride à plusieurs niveaux avec des modules de traitements spécifiques afin de répondre à ces constats. Nous allons la décrire dans la section suivante.

8.1.2/ MODÈLE CONCEPTUEL

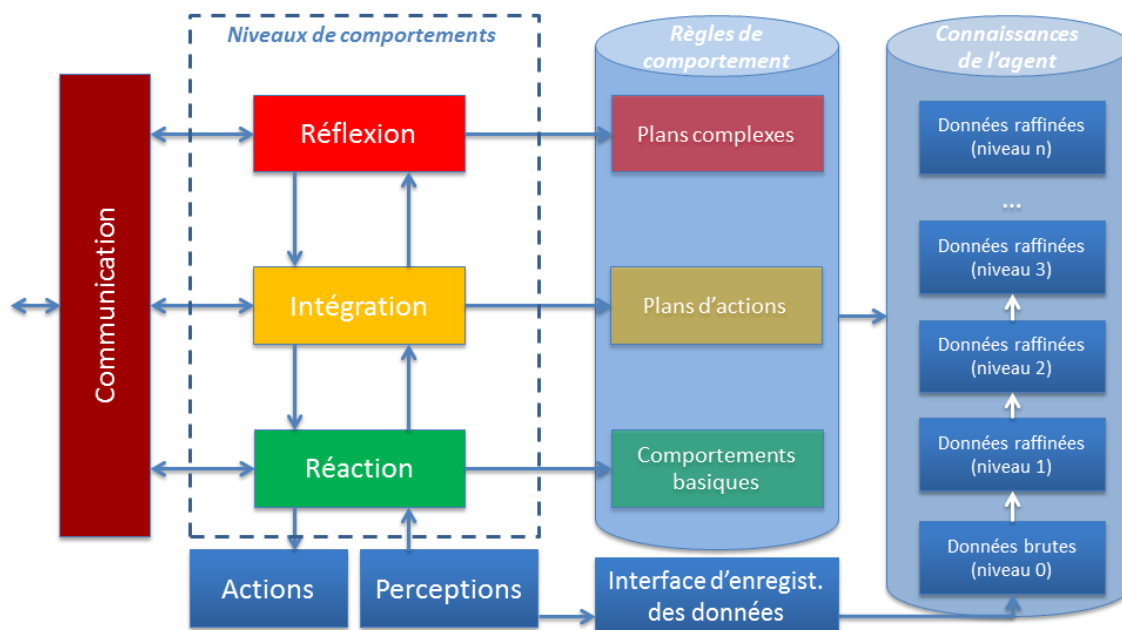


FIGURE 8.1 – Modèle conceptuel HACCA (Hybrid Architecture for Collaborative Communicating Agent)

La figure 8.1 présente notre modèle d'architecture avec les grands principes retenus dans

son élaboration ([Lucien et al., 2016a]).

Conformément à ce que nous avons décrit précédemment, il s'agit d'une architecture d'agent hybride basée sur trois couches de traitement. Chaque couche est dédiée à un niveau spécifique de réaction/réflexion comme nous le décrivons dans les sections suivantes. Ces couches s'appuient sur un ensemble de règles de comportement organisées en trois groupes (comportements basiques, plans d'actions et plans élaborés conjointement) ainsi qu'une base de connaissances hiérarchisées.

Les données brutes sont recueillies au moyen du module de perception et sont enregistrées par l'interface d'enregistrement des données. Puis elles sont raffinées progressivement au fur et à mesure des traitements de chaque couche qui en assure l'analyse puis le stockage.

L'architecture comprend également un module de communication implémentant les communications directes inter-agents pour une question de flexibilité et d'efficacité. Après cette présentation synthétique, une description plus précise est proposée dans les paragraphes suivants.

8.1.3/ PRINCIPES DE FONCTIONNEMENT

Nous reprenons notre exemple fil rouge avec la patrouille de drone de surveillance de forêt pour l'explicitier. Un drone évolue dans l'environnement et l'observe. Soudainement, il détecte une zone d'incendie et déclenche l'alerte en envoyant un message au drone le plus proche et en profite pour échanger des éléments cartographiques pour mettre à jour sa *carte mentale*. Pour cet exemple, nous supposons que le drone ayant détecté la zone d'incendie cherche à former un groupe d'observation avec le drone rencontré pour optimiser les recherches.

La figure 8.2 décrit le processus d'enchaînement des tâches au sein de l'architecture.

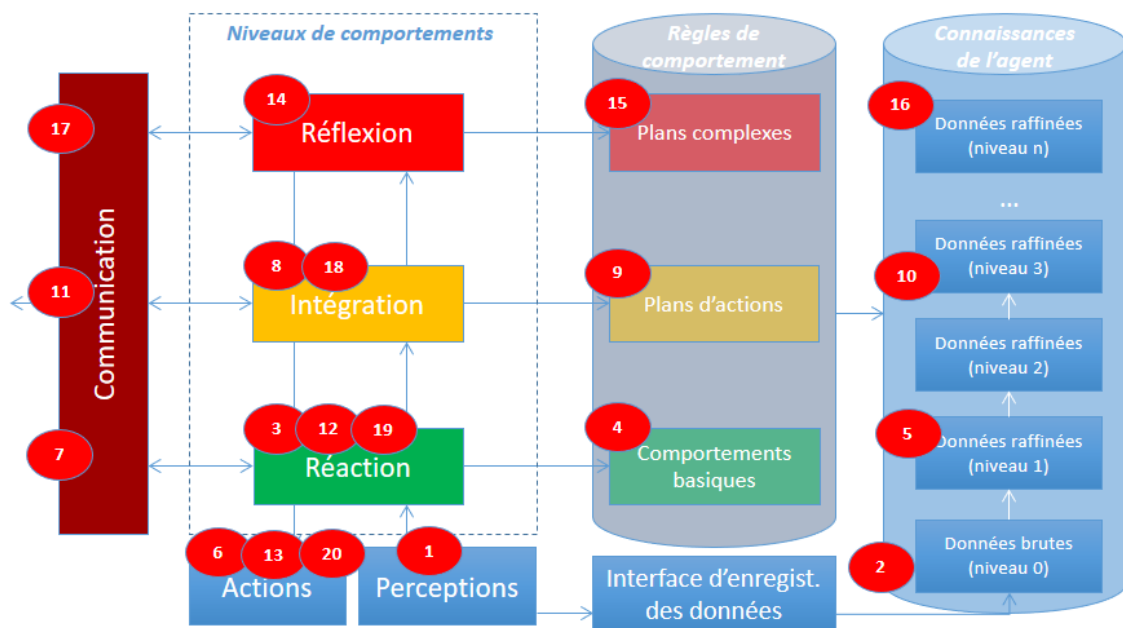


FIGURE 8.2 – Principes de fonctionnement HACCA

1. Réception des données des capteurs (observation de la zone)
2. Mise à jour des données brutes (avec les éléments détectés, notamment une zone d'incendie)
3. Déclenchement Comportements réactifs
4. Réactions s'appuyant sur la base de modèles (*Alerter* la base en cherchant à transmettre le message le plus rapidement possible)
5. Réactions utilisant les données brutes ou raffinées (Coordonnées de la zone d'incendie)
6. Réactions déclenchant une ou plusieurs actions (Voler vers la base)
7. Communication vers l'extérieur en fonction du comportement (*Communiquer* l'information au drone le plus proche à défaut de la base)
8. Transmission selon les besoins à la couche d'intégration pour traitement plus élaboré des données
9. Couche d'intégration s'appuyant sur des modèles de comportement plus développés (*Cartographier*)
10. Couche d'intégration utilisant des données brutes ou raffinées selon la précision demandée (*carte mentale*)
11. Communication vers l'extérieur selon le plan d'action choisi (*Communiquer* les éléments cartographiques et recevoir d'autres informations pour complément de mise à jour)
12. Déclenchement des réactions appropriées selon le plan d'actions (rester sur zone le temps de l'échange)
13. Réactions déclenchant les actions adaptées
14. Transmission selon les besoins à la couche de réflexion (traitements complexes)
15. Réflexion s'appuyant sur des modèles de comportement complexes (voire élaborés avec d'autres, ici *Construire Plan Patrouille*)
16. Réflexion utilisant potentiellement toutes les données à disposition (brutes ou raffinées selon les besoins. Utilisation des données sur l'environnement et l'historique des rencontres)
17. Communication vers l'extérieur selon le plan d'action choisi (Dialogue avec l'autre drone pour construire le plan de patrouille)
18. Déclenchement des plans d'actions appropriés
19. Déclenchement des réactions appropriées selon le plan d'actions
20. Réactions déclenchant les actions adaptées

Ceci est un exemple de conditionnement des actions par les interactions et les événements détectés. Il ne faut pas oublier également la résolution d'objectifs et la conduite des missions qui apportent une dynamique de la couche la plus haute vers la plus basse comme il sera présenté dans la section 8.2.4.

8.2/ DESCRIPTION DÉTAILLÉE

Après une première description de l'architecture et ses principes fondamentaux, nous allons maintenant décrire les différents modules la composant ainsi que leur rôle respectif et leurs interactions.

8.2.1/ MODULES DE PERCEPTION ET D'ACTION

Le niveau inférieur de l'architecture fait l'interface avec l'environnement. Il est composé de deux modules : le module de perception et le module d'action.

Le module de perception collecte les informations de l'environnement et les envoie dans la base de connaissances en passant par l'intermédiaire de l'interface d'enregistrement des données. Le module d'action est l'interface avec les effecteurs de l'agent. Ils sont programmés en fonction de la nature de celui-ci.

Par exemple, dans le cadre du fil rouge, le module de perception est l'ensemble des capteurs disponibles pour le drone de surveillance de forêt. On peut y retrouver une ou plusieurs caméras vidéos mais aussi des capteurs thermiques. Le module d'action pour notre drone agit directement sur les hélices, le régime moteur, etc. Il est piloté par la première couche de réaction.

8.2.2/ MODULE D'ENREGISTREMENT DES DONNÉES

Dans le cadre de ce module d'enregistrement des données, nous pouvons faire un parallèle avec le fonctionnement du cerveau d'un être humain. Comme Burton le décrit dans [Burton et al., 2011], le cerveau doit traiter des milliers d'informations toutes les secondes. Premièrement, il est impossible de tout enregistrer délibérément et deuxièmement, il y aurait énormément de "déchets" car de nombreuses informations ne sont en fait pas ou peu pertinentes.

D'autre part, comme le décrit Burnett dans [Burnett, 2017], il y a plusieurs mécanismes d'enregistrement et d'organisation de la mémoire. Il y a l'hippocampe qui centralise les informations à traiter. Puis différents mécanismes électro-chimiques permettent de catégoriser les informations selon le type de traitement *a posteriori*. Il y a donc plusieurs types de mémoire : à court terme ou à plus long terme. L'hippocampe ici sert de tampon avant un traitement plus poussé dans les lobes du cerveau concernés.

A l'instar de l'hippocampe dans le cerveau humain, cette interface enregistre des données brutes dans la base de connaissances de l'agent. Mais ce n'est pas un programme intelligent. Les données ne sont pas filtrées, ni classées à ce niveau. Le but de ce module est de faciliter l'enregistrement des données brutes sans la nécessité d'un programme spécifique décrit dans une couche supérieure. Comme nous l'avons dit au début de cette section, comme le cerveau, il est impossible de tout détecter pour tout interpréter. Ce module sera programmé en fonction de la mission première de l'agent dans son contexte d'exécution et dans la limite physique de ses capteurs. Par la suite, les différents traitements effectués sur les données permettent le stockage intelligent et contextualisé comme cela se décrit plus en avant (cf. section 8.2.5).

8.2.3/ TROIS COUCHES DE TRAITEMENT

Dans la continuité de l'enregistrement des données brutes, un message est envoyé au niveau réaction pour déclencher le comportement approprié. Nous allons voir maintenant comment cela fonctionne.

Comme nous l'avons vu dans la section 8.1.2, HACCA est constitué de trois couches

de traitement en fonction du contexte temporel et du besoin de réaction de l'agent : la couche de réaction, celle d'intégration et enfin celle de réflexion. Nous pouvons aussi faire un parallèle avec le cerveau. On pourrait donc parler de cerveau reptilien pour les aspects de réaction, de cerveau limbique pour la partie intégration et de néo-cortex pour la partie réflexion. Enfin, de manière plus pragmatique et dans un contexte d'agent mobile intelligent autonome, nous avons à faire avec une couche opérationnelle, une couche tactique et une couche stratégique.

La première couche (réaction) gère ainsi tous les comportements réactifs. C'est le module opérationnel où toutes les réactions basiques sont traitées. La deuxième couche (intégration) apporte quelques comportements plus raffinés avec un approfondissement du processus de décision. C'est le module tactique où des plans d'action avec des contraintes particulières sont mis en œuvre. La troisième couche (réflexion) permet d'intégrer des comportements plus complexes basés sur les processus de décision des couches précédentes. C'est le module stratégique avec des vues à long terme où des missions complexes à accomplir sont mises en œuvre. Les différentes couches sont sollicitées par message, par événement ou par intention (cela peut varier en fonction de la mise en œuvre ou de la plateforme multi-agents utilisée).

Par exemple, la couche de réaction est déclenchée par les données recueillies par le module de perception. En fonction des comportements basiques programmés, la couche d'intégration peut être appelée à exécuter des sous-programmes qui aident à la réalisation de la tâche complète. De même, la couche de réflexion peut être appelée par le programme de la couche d'intégration si nécessaire. Toutes les communications entre les couches peuvent être associées à des priorités de traitement (qui sont enregistrées dans les règles de comportement en tant que pré-conditions).

Cette manière d'opérer semble être la plus appropriée pour gérer le comportement d'un drone intelligent qui doit parfois répondre rapidement à des événements de l'environnement (voler, observer) tout en étant capable d'exécuter des tâches plus complexes nécessitant de passer par la base de connaissances et d'établir des plans d'action (cartographier, alerter rapidement). Les trois couches utilisent donc la bibliothèque de règles de comportement ainsi que la base de connaissances de l'agent.

8.2.4/ RÈGLES DE COMPORTEMENT

La bibliothèque de règles de comportement est un objet hiérarchisé. Les briques atomiques sont les comportements basiques des agents, leurs réactions aux perceptions de l'environnement. Chaque comportement basique a ses propres conditions de déclenchement, ses contraintes, ses entrées et ses sorties. Ensuite, nous trouvons les plans d'action qui peuvent être composés de comportements basiques. Enfin il y a les plans complexes pouvant être composés de plans d'action. Pour toutes les actions ou plans, il y a également des conditions de déclenchement, des contraintes, des entrées et des sorties.

Tous ces comportements basiques, ces plans d'action et ces plans complexes sont reliés entre eux pour former un arbre d'actions où chaque feuille est un comportement basique et chaque nœud est un plan d'actions ou un plan complexe. Chaque comportement basique, plan d'actions ou plan complexe est appelé par une couche spécifique qui est une sorte de point d'entrée dans l'arborescence des actions. Cela sous-entend que l'exécution d'une tâche peut être conditionnée par la couche de réaction et peut être pour-

suivie par le déclenchement d'un plan plus vaste dans une couche supérieure. Ceci peut s'avérer très utile dans le processus de modélisation d'agents très mobiles comme des drones communicants qui se croisent à grande vitesse. Le processus de détection (conditionné par le module de perception) exécute le comportement basique. Cependant, si l'information à échanger est pertinente pour un autre drone, une requête complémentaire utilisant un plan d'actions structuré peut être appelée. Cette action structurée ou ce plan est une partie de l'arbre global des actions.

Comme il a été mentionné dans la section 8.1.3, il y a deux types de fonctionnement : par les objectifs (dynamique Top-Down) et les événements (dynamique Bottom-Up).

Par exemple, pour notre fil rouge, nous pouvons reprendre les comportements identifiés dans le [chapitre 7](#) pour construire cet objet hiérarchisé. La figure 8.3 présente l'ensemble des comportements pour un drone, couche par couche et leur mise en relation.

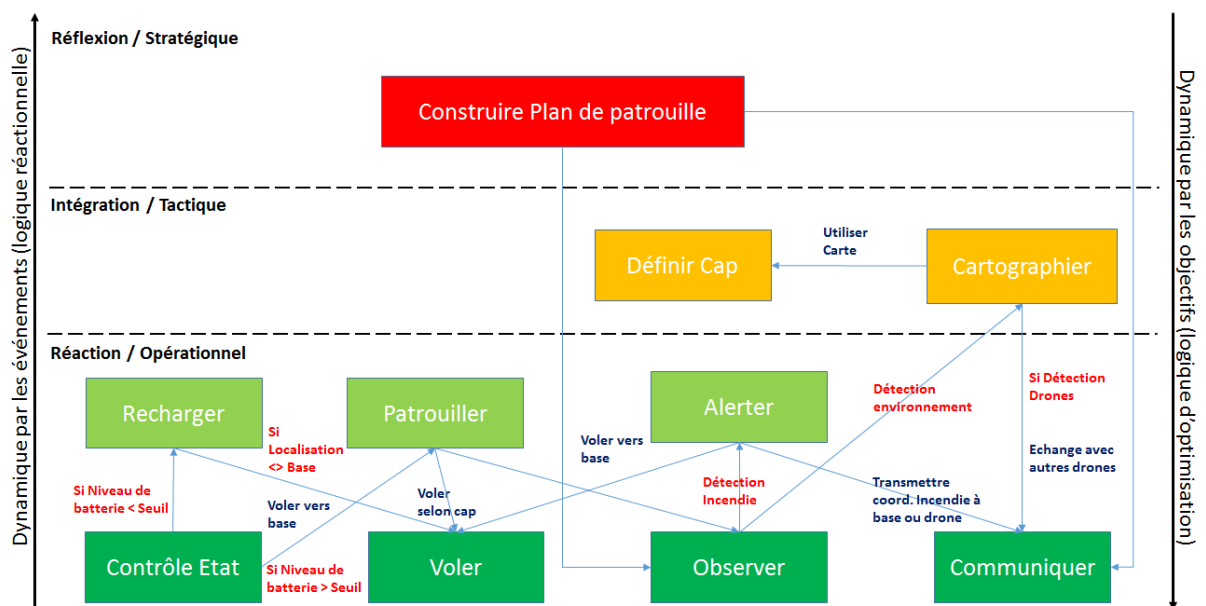


FIGURE 8.3 – Cartographie des comportements pour un drone

Nous retrouvons les comportements "clés" du drone : le *Contrôle d'état* (ici principalement la gestion de la batterie), *Voler*, *Observer* et *Communiquer*. Sur ces comportements basiques, nous construisons des comportements un peu plus élaborés comme : *Recharger*, *Patrouiller* et *Alerter*. Les comportements suivants permettent d'améliorer l'efficacité du drone : *Définir Cap* et *cartographier*, s'appuyant également sur les comportements basiques. Enfin nous avons un comportement plus complexe, s'appuyant aussi sur les comportements précédemment cités : *construire plan de patrouille*, combinant les briques d'observation et de communication autour de l'optimisation des déplacements pour maximiser la surface couverte par cycle de batterie en provoquant des échanges structurés entre drones.

Les comportements *Alerter*, *Cartographier* et *Construire Plan de patrouille* sont collaboratifs (un par couche) et s'appuient sur le comportement *Communiquer* qui fait le lien avec la brique technique transversale de communication.

8.2.5/ CONNAISSANCES DE L'AGENT

Contexte

L'objectif principal de la base de connaissances de l'agent est de structurer correctement les données pour pouvoir y accéder rapidement, les gérer efficacement et les communiquer vers l'extérieur à destination d'autres agents. Par exemple, des capteurs recueillent des informations brutes qui sont stockées dans la base de connaissances. Ensuite, ces informations brutes sont analysées et traitées progressivement par des actions attachées aux différentes couches. Nous pouvons y ajouter d'autres informations pour compléter et raffiner les informations brutes. Après exécution des différents traitements, on obtient donc un arbre de connaissances avec des informations brutes basiques à la racine et toutes les informations raffinées dans les branches. Ces branches peuvent être atteintes par les différentes couches selon les exigences du traitement à effectuer ([Lucien et al., 2016b]).

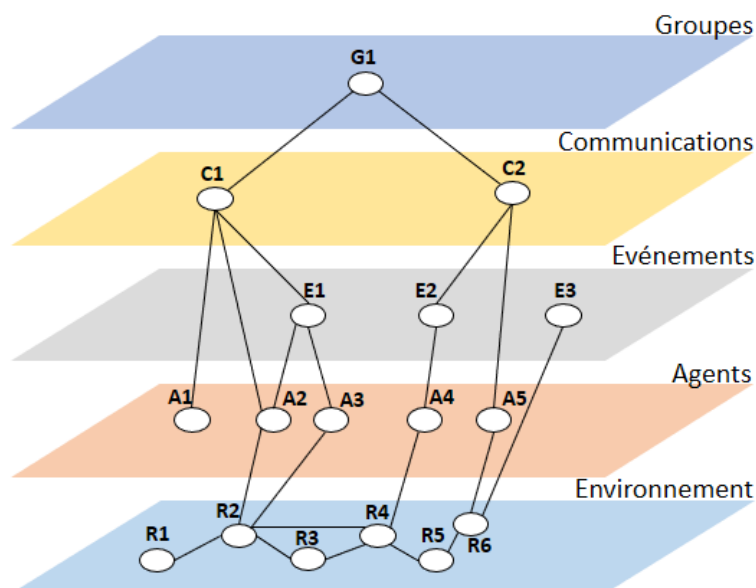


FIGURE 8.4 – Représentation de la connaissance sous forme d'un multi-graphe

Proposition

Nous utilisons un multigraphe (cf. figure 8.4) pour matérialiser les liens entre les données, les agents, les interactions, etc. Cette structure permet de gérer toutes les données pertinentes pour une action donnée. Par conséquent, il existe un lien entre l'action demandée, le niveau d'information que nous voulons communiquer (informations basique ou des informations plus ou moins raffinées), l'agent contacté et les résultats de l'échange. Ces éléments forment un tout cohérent. Les nœuds du multigraphe deviennent des points d'entrée sur un ensemble de données structurées. Cette structure de données est utile pour gérer une énorme quantité d'informations. Elle permet également de ne pas perdre de temps lors des échanges car on accède directement au bon niveau d'information (en choisissant le bon nœud de l'arbre). Cependant toutes les données restent accessibles si la situation l'exige. Chaque changement de contexte (couche de traitement) ajoute de nouveaux liens et de nouveaux nœuds à la structure originale. Comme évoqué plus haut, toutes ces informations ou ensemble d'informations vont former des grappes de données permettant d'échanger correctement le contexte d'une demande d'aide ou de traitement

collaboratif lors de la prise de contact. En effet, une collaboration est avant tout un travail autour d'objectifs partagés et d'une vision commune du résultat à atteindre. Les données échangées sont donc systématiquement contextualisées.

Perspective

Dans le contexte d'un agent mobile évoluant dans un environnement, il y a principalement deux types d'éléments : des éléments statiques immuables (l'environnement, par exemple une carte d'une zone géographique ou un réseau routier) ainsi que des éléments dynamiques ou mobiles (comme d'autres drones volant dans l'environnement ou des véhicules évoluant sur le réseau routier).

Tous les agents vont interagir continuellement tout au long de la simulation. Comme nous sommes dans un contexte d'agents autonomes devant posséder un certain niveau de réflexion, il est nécessaire d'enregistrer les événements auxquels ils sont confrontés. Conformément à l'architecture que nous avons décrit dans la section précédente, l'agent dispose de plusieurs niveaux d'interprétation (réaction ou opérationnel, intégration ou tactique, réflexion ou stratégique). Les informations perçues dans l'environnement doivent être enregistrées dans une base de connaissance afin de pouvoir être réexploitées par les différentes couches avec leurs comportements associés. Cela veut dire qu'au cours de la simulation, les informations enregistrées vont progressivement être raffinées, modifiées ou supprimées en fonction des traitements.

De plus, si l'agent se retrouve dans une impasse (pas de solution immédiate face à une situation particulière de danger par exemple), il doit pouvoir communiquer rapidement l'information vers les autres agents. Comme nous sommes dans un contexte collaboratif et en accord avec la définition que nous avons donnée dans le [chapitre 6](#), chaque information doit pouvoir (si le besoin est avéré) être contextualisée. Cela sous-entend qu'il est important de la transmettre ainsi que toutes les données nécessaires à la compréhension globale, pour que les autres agents puissent correctement interpréter l'information qu'ils reçoivent.

Description

La première couche du graphe représente l'environnement ou globalement les objets statiques connus ou rencontrés par l'agent. La deuxième couche sera dédiée aux objets dynamiques (agents) qu'il pourra observer et avec lesquels il va interagir. La troisième couche représente les événements que l'agent va percevoir ou déduire de ses observations. La quatrième couche permet le suivi des interactions avec les autres agents en enregistrant les communications en cours. La cinquième couche symbolise la notion de groupe et les actions de collaboration. Tous les objets représentés constituent les nœuds des différentes couches de graphe. A chaque phase d'observation, l'agent identifie des éléments de l'environnement (statiques ou dynamiques) et les enregistre dans le multigraphe puis établit des liens entre les éléments. Les différentes actions ou comportements viennent modifier ou enrichir progressivement le multigraphe.

Chaque objet dispose d'un certain nombre de propriétés communes : un identifiant unique, une désignation (ou nature de l'élément, par exemple : un incendie, un pylône électrique, une route, un drone), une localisation, une date d'observation et un statut (ou état de l'élément à l'instant de l'observation). Cependant, en fonction de la nature de chaque objet, des propriétés supplémentaires peuvent être enregistrées. Par exemple, pour un objet drone, on trouve le chemin parcouru ou le temps de déplacement entre un point A et un point B ou encore la vitesse moyenne sur le parcours. Autre exemple : pour

un objet communication, on trouve la zone géographique concernée (diffusion locale, diffusion sur toute une zone, etc), le degré d'importance du message ou un temps d'attente limite entre deux messages.

Il est important de noter que toutes les données contiennent une date et une heure d'observation. En effet, il paraît évident qu'une situation n'est pas immuable et que le contexte évolue en fonction de la dynamique de l'environnement et de l'action des agents le composant. Des données peuvent donc être jugées de moins en moins pertinentes à mesure que le temps passe sans un rafraîchissement de l'information via une observation directe ou une interaction avec un agent possédant des données plus à jour. Cela suppose que les agents disposent d'une horloge synchronisée.

8.2.6/ COMMUNICATIONS

Rappelons-le, nous pouvons définir des actions et des communications de collaboration à différents niveaux. Le module de communication est la partie la plus importante de cette architecture. Afin de garantir un haut niveau de réactivité, ce module est transversal. Le module de communication peut ainsi être appelé par toutes les couches de traitement. Il sélectionne le bon protocole en fonction de la distance géographique et la tâche assignée (diffusion, collaboration partielle avec un échange restreint d'information, collaboration complète avec un partage du contexte et du résultat à atteindre), entretient des liens entre les deux agents communicants tant que la mission le nécessite et donc permet de passer d'un protocole à l'autre selon les besoins.

Par exemple, l'agent détecte une anomalie dans son environnement. Il exécute quelques comportements basiques (de la couche de réaction) notamment la diffusion de l'information à tous les agents autour de lui. Il utilise donc le module de communication pour la diffusion des informations critiques. Pendant ce temps, cette action déclenche un plan d'action plus élaboré dans la couche supérieure afin de comprendre ce qu'est l'anomalie. Les agents à proximité recevront le message qu'un événement critique a été détecté. Certains agents (en fonction de leurs objectifs) retourneront un message au premier agent pour lui demander plus d'informations. Le message est reçu par le module de communication mais il est acheminé vers une couche supérieure, celle qui porte précisément sur le processus d'analyse de l'anomalie. Ce bloc permet de surveiller le processus de communication entre les agents. Au départ, il y a une communication générique avec un processus de diffusion qui va se transformer en communication spécifique entre deux agents. La base de connaissances de l'agent est un multigraphe, le processus de communication utilise donc cette logique afin de lier l'information échangée aux agents communicants pour enrichir le graphe préexistant. Ce module permet la construction d'un échange structuré sur la base de données hiérarchisées.

Dans le cadre de notre exemple fil rouge et en accord avec ce que nous avons décrit dans le [chapitre 7](#), nous avons identifié trois types de communication : alerter les secours (couche opérationnelle), échanger des informations cartographiques (couche tactique) et construire un plan de patrouille (couche stratégique). La table 8.1 résume quelques caractéristiques et notamment les collaborations qui sont mises en œuvre.

Couche	Fonction	Données communiquées	Collaboration
Opérationnelle	Alerter les secours	Zone d'incendie	Forcée
Tactique	Echanger des infos cartographiques	Carte mentale (couches Environnement, Agents et Evénements)	Partielle
Stratégique	Construire un plan de patrouille	Carte mentale et Objectifs	Complète

TABLE 8.1 – Synthèse des échanges entre agents par couche

8.3/ CINÉMATIQUE ET EXEMPLE D'APPLICATION

Nous allons maintenant utiliser notre exemple de patrouille de surveillance de forêt pour montrer la construction du multi-graphe et ainsi décrire concrètement comment la carte mentale, véritable colonne vertébrale de la connaissance de l'agent, est créée.

En reprenant la figure présentée dans la section 8.2.4, nous allons illustrer la dynamique d'exécution des différents comportements qui sont appelés en fonction de l'état interne de l'agent, des objectifs, des observations et des rencontres.

Le drone avec une batterie chargée (comportement Contrôle Etat) commence à patrouiller, en volant et en observant. Au début de l'expérience, le cap est défini aléatoirement, le drone n'ayant aucune référence cartographique. Au fur et à mesure des observations, la cartographie va être réalisée.

Comme nous l'avons précisé précédemment la couche 1 représente l'environnement du drone (cf. figure 8.5). Les différents nœuds sont des points d'intérêts (POI) détectés par le processus d'observation (pylône, groupe d'arbres, rivière, etc). Les liens entre les nœuds sont les chemins pour y parvenir. Si ces arêtes doivent aussi enregistrer de l'information le cas échéant, il faudrait donc les changer en nœud. Nous pouvons aussi prendre une grille comme représentation de l'environnement ce qui peut être plus pratique selon le problème et l'environnement à modéliser. La grille serait donc interprétée comme un graphe. Les informations enregistrées vont maintenant être employées pour définir un cap en tenant compte des zones déjà explorées pour maximiser la couverture du territoire à surveiller. Le comportement Voler exécuté via le comportement Patrouiller va désormais prendre en considération le nouveau cap calculé.

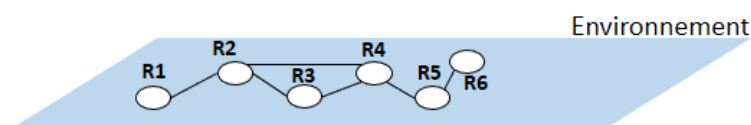


FIGURE 8.5 – Construction de la carte mentale étape 1 : Environnement

Au cours des déplacements dans l'environnement, le drone va également détecter d'autres agents drones qu'il va enregistrer en les positionnant sur la carte qu'il connaît déjà sur la couche 2 (cf. figure 8.6). Des liens sont créés pour relier les POI aux agents détectés. Chaque nœud porte également des données temporelles ce qui permettra des mises à jour si les drones se croisent. Si les données sont trop anciennes, elles sont jugées non-pertinentes et les nœuds sont supprimés.

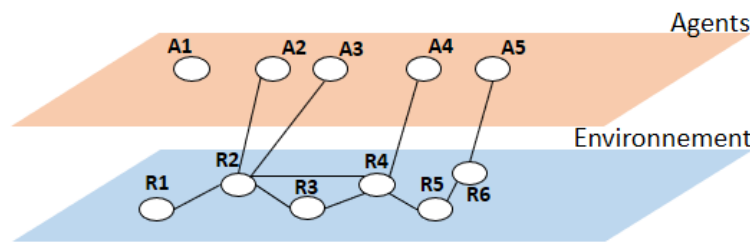


FIGURE 8.6 – Construction de la carte mentale étape 2 : Agents

Au cours de l'exécution de sa mission, le drone va découvrir des zones d'incendie qui vont constituer des événements importants à signaler (communication avec la base ou d'autres drones) sur la couche 3. Ces nœuds événements sont donc enregistrés et contextualisés (cf. figure 8.7). L'événement E3 par exemple, identifie une zone d'incendie observé au point R6. L'événement E2 a été communiqué suite à une rencontre avec A4 et relie la zone à R4.

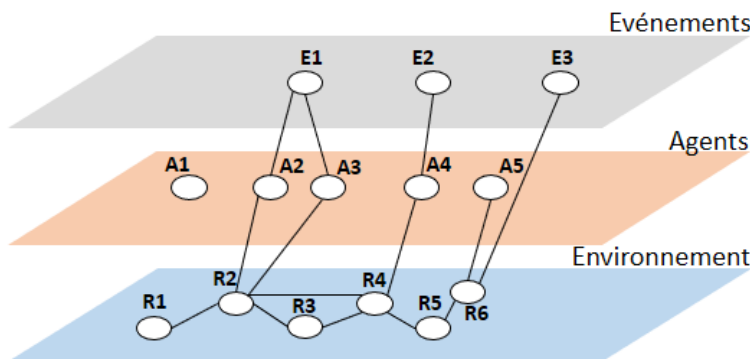


FIGURE 8.7 – Construction de la carte mentale étape 3 : Evénements

La détection d'incendie va exécuter le comportement *Alerter* qui va lancer un comportement *Voler* vers la base et un comportement *Communiquer* si le drone est dans la zone de réception de la base. Cela va donc provoquer la création d'un nœud événement et d'un nœud communication. En cas de détection d'un autre drone sur le chemin du retour et si la base est encore hors de portée, le comportement *Communiquer* sera exécuté pour transmettre les coordonnées de la zone d'incendie. Un nœud agent est créé.

Pour enregistrer les communications en cours, le drone matérialise ces liens sur la couche 4 (cf. figure 8.8). Par exemple, le nœud C2 indique une communication en cours entre le drone et l'agent drone A5 au sujet de l'événement E2 observé au POI R4.

Par extension, dans le cadre de l'exécution du plan complexe *Construire Plan de patrouille*, il peut être nécessaire de constituer des groupes coordonnés. La couche 5 permet l'enregistrement de ces liens (cf. figure 8.9). Par exemple, le drone a permis la création du groupe G1 constitué des drones A1, A2 et A5. Cela permettra de transférer toutes les informations nécessaires à l'ensemble des membres du groupe, tant que les communications sont en cours. Il est aussi possible de relier directement le nœud G1 aux agents concernés quand la communication sera coupée pour conserver les liens sociaux.

Au fur et à mesure du déroulement et des interactions successives, ce multigraphe évolue. Certains liens sont créés, d'autres sont déplacés ou encore supprimés. De la même manière, les nœuds peuvent être modifiés ou supprimés si le contexte a changé.

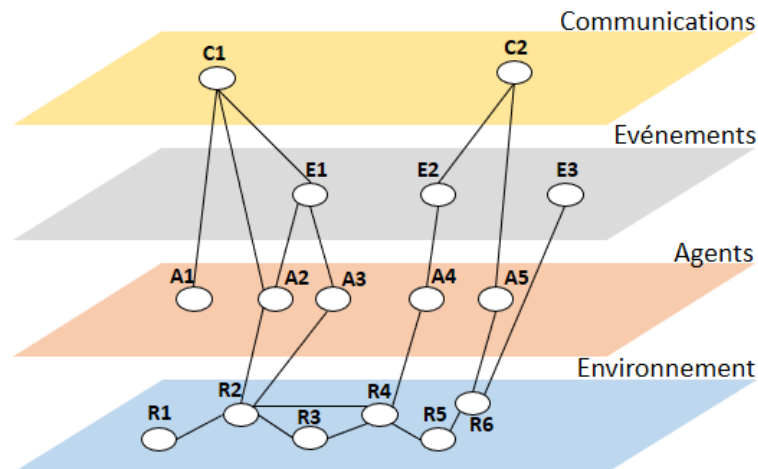


FIGURE 8.8 – Construction de la carte mentale étape 4 : Communications

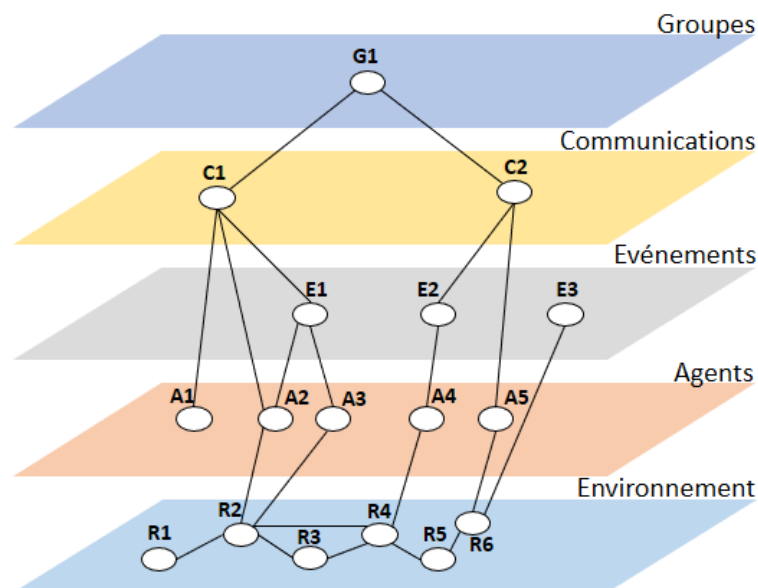


FIGURE 8.9 – Construction de la carte mentale étape 5 : Groupes

La figure 8.10 en illustre un exemple. La communication entre le drone et l'agent drone A5 est terminée. Cela se traduit dans la structure par la suppression du nœud communication C2 qui matérialisait celle-ci. En conséquence, le lien reliant le nœud C2 au nœud A5 est également supprimé ainsi que le lien vers l'événement qui a suscité la communication (entre le nœud C2 et E2). Autre modification, l'agent drone A5 se déplace dans une autre zone de l'environnement (de R6 à R4). Le lien qui le reliait au nœud R6 est donc supprimé. Enfin, l'événement E3 symbolisant une zone d'incendie n'est plus d'actualité. Le nœud E3 est supprimé ainsi que les liens avec la zone d'environnement (nœud R6).

Il faut être conscient qu'il s'agit de la représentation symbolique du monde tel que l'agent le perçoit avec les informations qu'il a collectées ou qu'il a échangées. Cela n'est donc pas une vision temps réel de la situation. Certaines données peuvent donc être erronées si la date d'observation est trop ancienne. D'où l'importance de la propriété *date d'observation* enregistrée dans chaque nœud d'information. Certains comportements peuvent incorporer des instructions de nettoyage de la structure. Par exemple, si la différence

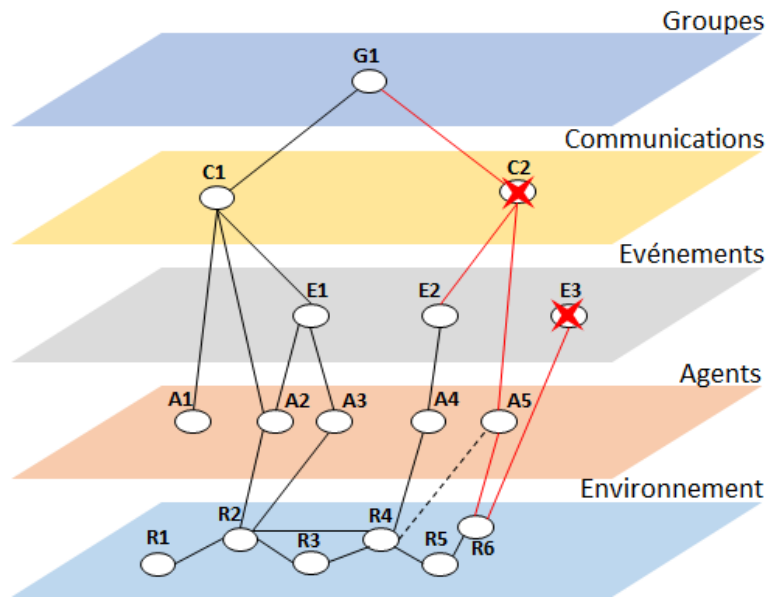


FIGURE 8.10 – Modification de la carte mentale

entre la date actuelle du système et la date d'observation dépasse une certaine durée. De plus, ces dates seront forcément évaluées par les agents recevant les informations. En fonction de la date, les données pourront être tout simplement ignorées.

En résumé, le multigraphe peut être modifié par perception de l'environnement, échange d'information, exécution de comportements, nettoyage périodique ou événementiel en fonction des comportements retenus par l'agent pour accomplir ses objectifs.

En cas de défaillance de batterie, le comportement *Recharger* est déclenché pour faire rentrer le drone immédiatement à la base. Et ainsi de suite jusqu'à la fin de la simulation.

8.4/ CE QU'IL FAUT RETENIR

La figure 8.11 reprend les principaux éléments de notre proposition d'architecture HACCA pour les entités mobiles intelligentes et communicantes. C'est une structure d'agent hybride construit sur une couche réactive avec des comportements basiques et des couches plus intelligentes exécutant des comportements plus élaborés.

Les comportements sont hiérarchisés : du plus basique au plus complexe. Les connaissances sont structurées et contextualisées autour d'un multigraphe (plusieurs niveaux d'informations). Le module de communication transversal permet d'être appelé par tout type de comportement quelque soit le niveau. Les informations transmises sont des extractions de la structure multigraphe.

Les concepts collaboratifs font partie intégrante de l'architecture et se retrouvent dans la contextualisation des informations enregistrées et le module transversal de communication. Ils s'appuient enfin directement sur les comportements eux-mêmes (un par couche de traitement, en fonction des missions), programmés pour interagir de manière structurée avec d'autres entités. Dans le chapitre 9, nous allons mettre en pratique les éléments d'architecture évoqués ici en instanciant HACCA dans un contexte urbain.

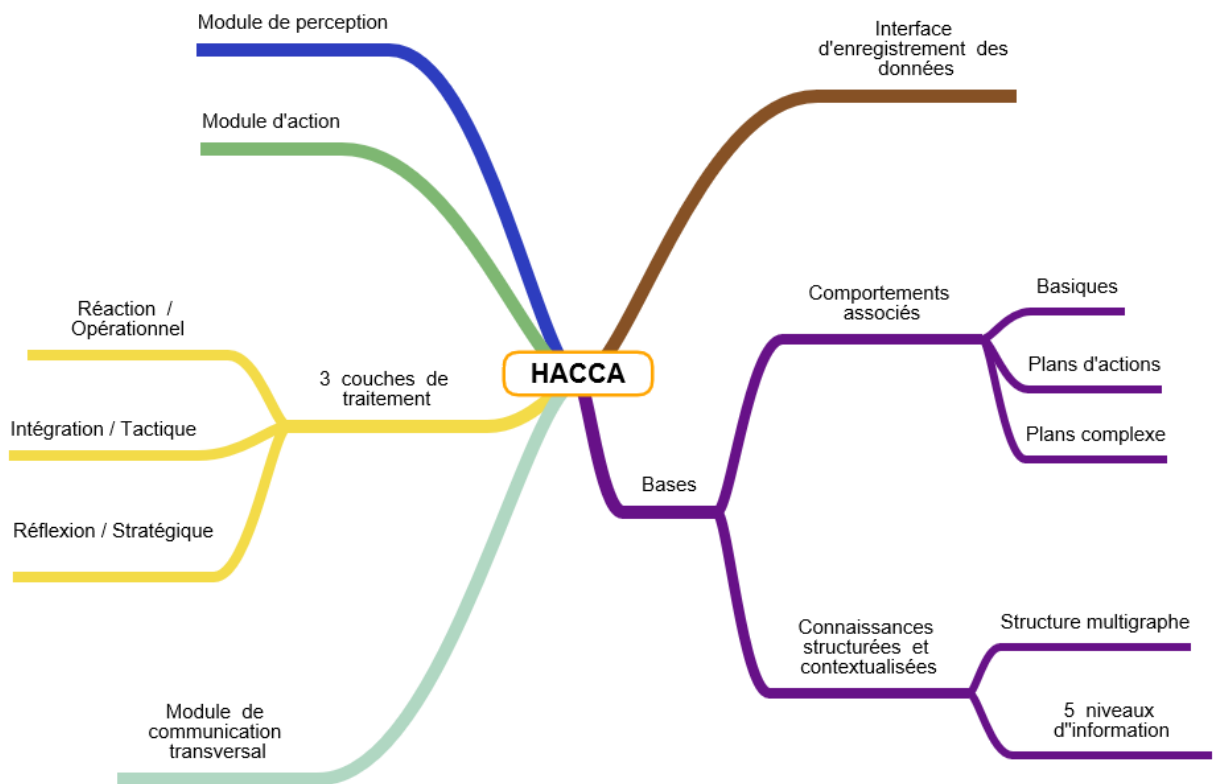


FIGURE 8.11 – Ce qu'il faut retenir : Architecture HACCA (fondamentaux)



INSTANCIATION, IMPLÉMENTATION ET EXPÉRIMENTATIONS

INSTANCIATION DU MODÈLE URBAIN

Sommaire

9.1	Scenario	137
9.1.1	Discussion préalable	137
9.1.2	Description	138
9.2	Méthode d'analyse	139
9.2.1	Analyse du problème et caractérisation	139
9.2.2	Amélioration des comportements	140
9.2.3	Optimisation	143
9.3	Synthèse	144
9.3.1	Comportements	144
9.3.2	Connaissances	145
9.4	Ce qu'il faut retenir	146

Ce chapitre décrit l'instanciation du modèle HACCA pour le véhicule connecté. C'est la traduction du modèle théorique dans le domaine d'application que nous avons choisi. En effet, étant collaborateur du groupe PSA depuis 20 ans, il nous semble naturel de tester notre architecture dans ce contexte.

Dans un premier temps, nous explicitons le scénario choisi puis nous utilisons la méthode d'analyse décrite dans le [chapitre 7](#) pour distinguer ce qui caractérise fondamentalement un véhicule connecté et les comportements à mettre en œuvre.

9.1/ SCENARIO

Cette étude de cas a pour objectif de tester et d'évaluer notre architecture d'agent collaboratif. Nous avons donc choisi un cas réel : la modélisation de voitures intelligentes se déplaçant dans une ville. Ces véhicules doivent circuler du domicile de leur propriétaire à son lieu de travail aussi vite que possible et éviter les embouteillages.

9.1.1/ DISCUSSION PRÉALABLE

Il est nécessaire d'effectuer une mise au point préalable sur le vocabulaire utilisé dans le contexte des véhicules intelligents car on parle souvent de véhicules intelligents, communicants et connectés, parfois les trois à la fois. En fait, il y a souvent une confusion entre les différents adjectifs.

Nous considérons ici qu'un véhicule est intelligent s'il a des capacités d'analyse de son environnement. Il est communicant s'il possède des moyens de communication (émission et réception). Et il est connecté s'il peut établir une communication structurée avec d'autres entités communicantes et notamment des infrastructures centralisées ou des systèmes de transport intelligent.

9.1.2/ DESCRIPTION

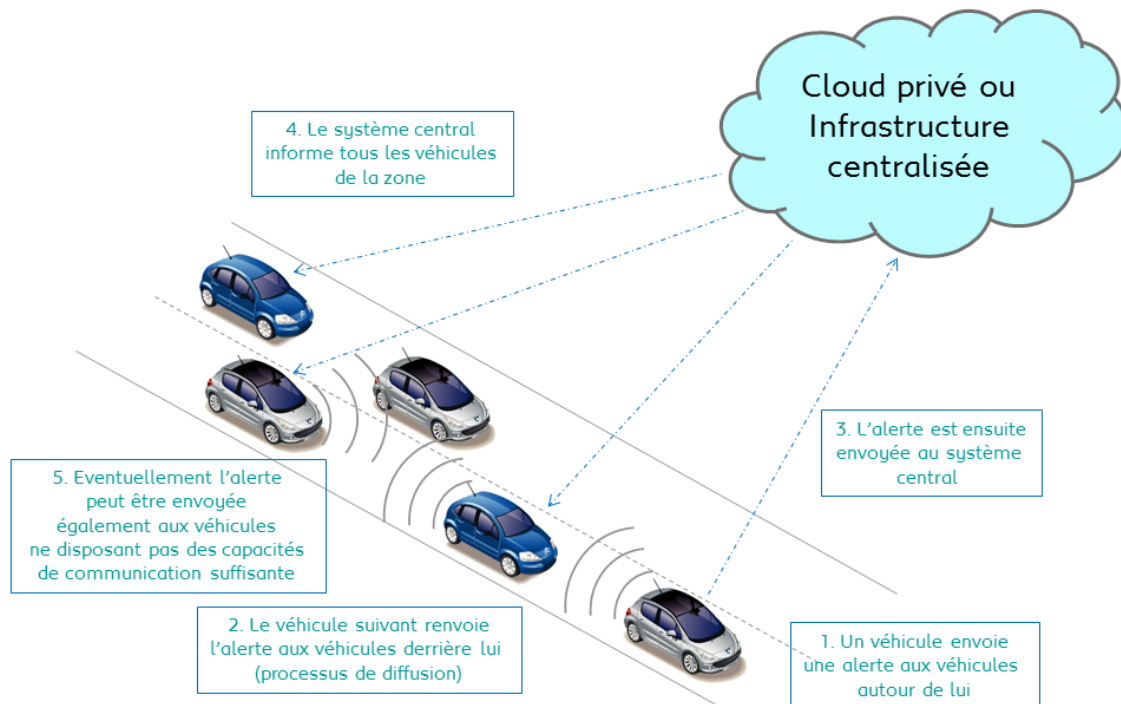


FIGURE 9.1 – Description du processus de collaboration entre véhicules

Nous sommes dans un contexte urbain où des événements peuvent survenir comme des accidents, des pannes ou des véhicules immobilisés sur la route. Il faut donc mettre en place des processus de détection d'événements et d'interprétation. Différentes catégories d'agents *Véhicule* sont définies (en fonction de leurs propriétés) : des véhicules standards, des véhicules intelligents (avec une capacité d'observation et d'analyse du monde), des véhicules communicants (avec une capacité de communication locale) et des véhicules connectés à une infrastructure centralisée comme un Cloud.

La figure 9.1 présente le scénario choisi. Un véhicule détecte une anomalie (accident ou voitures arrêtées) et envoie une alerte tout autour de lui. Les véhicules suivants reçoivent l'alerte, mettent à jour leurs connaissances et envoient les informations aux véhicules suivants par processus de diffusion. Dans le même temps, le premier véhicule envoie l'alerte au système central. Cette infrastructure mutualisée reçoit toutes les informations de chaque agent et regroupe les connaissances liées au statut des différentes routes et à l'état global du réseau routier. Note : ce n'est pas un processus en temps réel parce que ce système peut ne pas être connecté tout le temps (au moment où le véhicule transmet les informations, en fonction de la couverture réseau, l'infrastructure peut être inaccessible). Ainsi, après la consolidation, l'infrastructure envoie une mise à jour avec

les nouvelles informations à tous les véhicules de l'environnement à intervalles réguliers (quelques minutes entre chaque cycle d'envoi). Une communication supplémentaire peut être envisagée entre un véhicule complètement connecté (intelligent, communicant et connecté) et un véhicule simplement communicant (non connecté) pour envoyer les informations pertinentes de sécurité principalement.

Les différentes études sur les véhicules communicants intelligents [Silberg et al., 2012, Silberg et al., 2013] font apparaître que ce double système de communication (véhicule à véhicule et de véhicule à infrastructure) est efficace pour améliorer et assurer une diffusion correcte de l'information critique, même si une partie du système est déconnectée. Sur la base de l'information qu'il reçoit, l'agent véhicule modifie sa trajectoire et communique les changements d'itinéraire si les conditions de circulation ont changé.

9.2/ MÉTHODE D'ANALYSE

Nous reprenons les différentes étapes de la méthode d'analyse décrite dans le **chapitre 7** : analyse du problème et caractérisation, amélioration des comportements et optimisation.

La première itération permet de définir les caractéristiques intrinsèques des agents et de s'intéresser à la couche réactive (opérationnelle). La deuxième itération va préciser les comportements afin de les améliorer en mettant en évidence les premières collaborations. Il s'agit de la couche d'intégration (tactique) où l'environnement va être analysé. Enfin la troisième itération permet d'aller plus loin avec la couche de réflexion (stratégique) pour mettre en lumière les voies d'optimisation.

9.2.1/ ANALYSE DU PROBLÈME ET CARACTÉRISATION

Nous sommes au niveau opérationnel et nous considérons les entités comme des agents réactifs (véhicule standard sans perfectionnement). Son objectif est de se déplacer sans encombre et sans ralentissement d'un point A (domicile) à un point B (travail), en parcourant donc la plus petite distance entre A et B en un minimum de temps.

Nous ne considérons ici que des agents de type *véhicule*. Les informations importantes à gérer concernent principalement son statut (en fonctionnement, en panne ou accidenté), sa position dans l'environnement, sa vitesse, les localisations (domicile et travail) et son itinéraire (celui-ci est choisi lors de la phase d'initialisation).

Basiquement, les deux seuls comportements identifiés sont : déterminer régulièrement son statut (*Contrôler Etat*) et se déplacer dans l'environnement du point A au point B (*Rouler*) si son état le permet.

Il n'y a pas d'interaction directe avec l'environnement en dehors du suivi de l'itinéraire sur le réseau routier. Il n'y a pas de contact avec les autres véhicules. A ce stade, aucune collaboration n'est détectable. Il s'agit d'un simple déplacement du point A au point B en suivant un chemin prédéfini.

Nous pouvons constater que l'objectif de déplacement est rempli. Cependant, s'il y a un problème sur la route, il n'y a pas de possibilité de contournement. Il n'y a pas de compréhension de l'environnement, aucune communication avec les autres entités et

Couche	Comportement	Variables en entrée	Action	Variables en sortie
Opérationnelle	Contrôle Etat		Gestion/Contrôle du statut du véhicule	Statut modifié
Opérationnelle	Rouler	Itinéraire	Circuler du point A au point B en suivant l'itinéraire choisi	Position du véhicule modifiée

TABLE 9.1 – Synthèse des comportements à la fin de la première itération

donc pas d'anticipation possible. La seconde itération permettra de proposer une modification des comportements existants ainsi que de nouveaux comportements pour prendre en compte ces besoins.

La table 9.1 reprend les comportements à l'issue de cette première itération d'analyse.

9.2.2/ AMÉLIORATION DES COMPORTEMENTS

Nous sommes maintenant au niveau tactique et nous allons considérer un véhicule plus intelligent, capable d'observer et de communiquer des informations autour de lui. Plus largement, nous utilisons un véhicule connecté. Cette deuxième itération est consacrée à la compréhension de celui-ci ainsi qu'à la mise en place des premières collaborations entre agents afin d'améliorer les comportements intrinsèques des véhicules.

Nous reprenons les agents de type *véhicule* mais nous ajoutons de nouvelles informations telles que la carte exhaustive de l'environnement sous la forme d'un réseau routier. Dans la section précédente, nous pouvions nous passer d'une telle représentation (l'itinéraire était considéré comme pré-calculé sans modification possible). Une carte globale pouvait cependant être utilisée pour gérer le suivi de l'itinéraire. Comme nous considérons des agents intelligents et autonomes, chacun dispose d'une représentation personnalisée du réseau. Il peut ainsi contenir l'ensemble des informations sur l'état des routes (utilisables ou bloquées).

Pour nous conformer à l'exemple que nous avons présenté dans la section 9.1.2, nous allons ajouter un agent de type *cloud*. Il sert d'agent de centralisation et de liaison avec les agents *véhicules*. Il dispose également de sa propre représentation du réseau routier et enregistre les informations concernant son état. Il n'a qu'un comportement *Communiquer* pour recevoir des données et les retransmettre à intervalles réguliers (car il peut ne pas être en ligne de manière continue).

Pour les agents de type *véhicule*, nous ajoutons le comportement *Observer* qui va regarder l'environnement (trafic routier). Il s'agit d'une modification de la couche opérationnelle. En effet, si un événement important est détecté, il faut pouvoir réagir immédiatement (comme un véhicule arrêté en plein milieu du chemin). Le comportement *Communiquer* est également ajouté afin de diffuser rapidement ce genre d'information et en récupérer du *Cloud*.

Au niveau tactique, il faut pouvoir analyser l'environnement et comprendre les observations brutes (comportement *Analyser*). Plusieurs types d'événement sont détectables :

identifier un véhicule accidenté / en panne (si les capteurs embarqués sont assez précis ou les possibilités de communication sont suffisantes avec le véhicule arrêté ou les infrastructures environnantes) et identifier un embouteillage par analyse des données observées (si plusieurs véhicules sont immobilisés pendant un temps donné).

En fonction de ses capacités de communication (communiquant et/ou connecté), l'agent *véhicule* doit pouvoir recevoir ou diffuser les informations pertinentes sur les événements détectés autour de lui (en fonction également des capacités de communication des différentes entités). Il doit communiquer aussi au *cloud*, l'infrastructure de traitement centralisé.

Enfin, compte tenu des informations observées, analysées et reçues, l'agent *véhicule* réévalue un nouvel itinéraire (comportement *Evaluer Chemin*). L'objectif est toujours d'aller le plus rapidement et sans trop d'encombres du point A au point B.

Nous pouvons constater que les interactions sont désormais potentiellement nombreuses : avec l'environnement (par le biais de l'observation directe), avec les autres agents *véhicules* s'ils disposent de capacités de communication et enfin avec le *cloud*. Nous pouvons identifier des processus de collaboration partielle par échange d'informations (données cartographiques, événements de type *accident*, *embouteillage*, *route bloquée*) mais également de collaboration induite entre agents (par diffusion des informations via des intermédiaires).

Afin d'illustrer les comportements des entités, prenons le cas présenté dans la figure 9.2. Il s'agit d'une portion d'un réseau routier où huit véhicules circulent. Un accident s'est produit avec le véhicule 1 qui entraîne la création d'un embouteillage. Les autres véhicules vont observer, analyser et prendre des décisions. Les cercles représentent les limites de perception des véhicules.

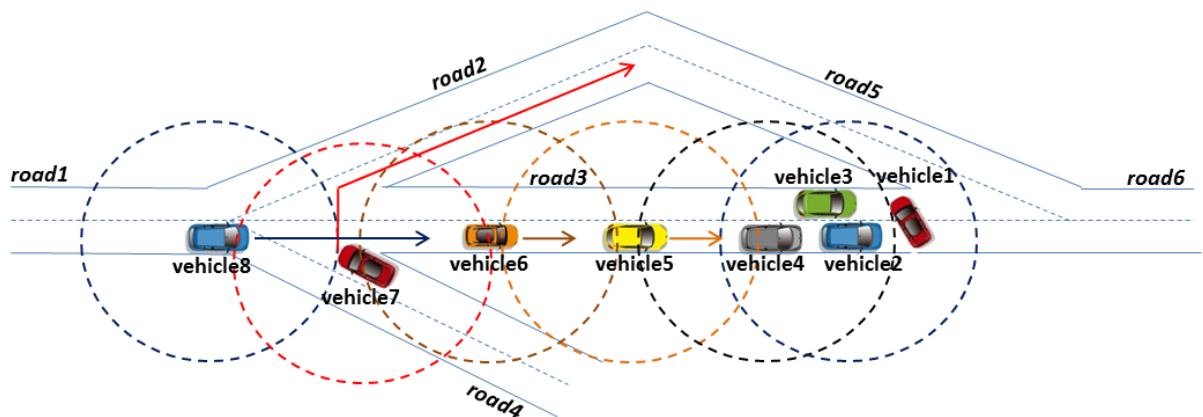


FIGURE 9.2 – Exemple d'un réseau de véhicules

La table 9.2 décrit une partie des interactions entre les véhicules. Nous retrouvons pour 4 véhicules du réseau de la figure 9.2 : ce qui est vu (par les capteurs du véhicule), ce qui est enregistré (*Observer*), ce qui est interprété (*Analyser*), ce qui est communiqué (*Communiquer*), ce qui est reçu et ce qui est fait.

Nous avons maintenant précisé les comportements relatifs à la compréhension de l'environnement et les premières collaborations sont proposées pour permettre les échanges d'information entre véhicules. Il convient de noter également la communication entre les véhicules et le *cloud* afin de consolider une *carte mentale* centralisée. Il y a un début d'an-

	vehicle2	vehicle4	vehicle7	vehicle8
Statut	arrêté	arrêté	en mouvement	en mouvement
Ce qu'il voit	<i>vehicle1, vehicle3, vehicle4</i>	<i>vehicle2, vehicle3, vehicle5</i>	<i>vehicle6, vehicle8</i>	<i>vehicle7</i>
Ce qu'il enregistre	<i>vehicle1, vehicle3, vehicle4</i> sont arrêtés sur <i>road3</i>	<i>vehicle2, vehicle3</i> sont arrêtés et <i>vehicle5</i> est en mouvement (tous sur <i>road3</i>)	<i>vehicle6</i> est en mouvement sur <i>road3</i> et <i>vehicle8</i> est en mouvement sur <i>road1</i>	<i>vehicle7</i> est en mouvement sur <i>road4</i>
Ce qu'il interprète	<i>vehicle1</i> a un <i>accident</i> , <i>road3</i> est bloquée	un <i>embouteillage</i> est détecté, <i>road3</i> est bloquée		
Ce qu'il communique	transmission de 2 messages : événement <i>accident</i> (seulement l'info prioritaire), événement <i>route bloquée</i> (avec le contexte <i>événement, route, véhicules</i>)	transmission de 2 messages : événement <i>embouteillage</i> (avec contexte), événement <i>route bloquée</i> (avec contexte)		
Ce qu'il reçoit		de <i>vehicle2</i> : événement <i>accident</i> (haute priorité, à propager), événement <i>route bloquée</i> (déjà connu, renforce les connaissances)	de <i>vehicle6</i> : événement <i>embouteillage</i> , événement <i>route bloquée</i>	de <i>vehicle7</i> : événement <i>embouteillage</i> , événement <i>route bloquée</i>
Finalement ce qu'il fait		il propage l'événement <i>accident</i>	il n'est pas concerné (<i>road3</i> n'est pas son chemin) mais il propage les messages	il recherche un nouveau chemin en communiquant avec <i>vehicle7</i> et en échangeant des informations

TABLE 9.2 – Description partielle des interactions entre les véhicules

tipication des problèmes avant l'arrivée sur zone grâce à la diffusion des informations. La table 9.3 reprend les comportements ajoutés et modifiés à l'issue de cette deuxième itération d'analyse.

Couche	Comportement	Variables en entrée	Action	Variables en sortie
Opérationnelle	Observer		Observation dans la zone de vision et Cartographie	Mise à jour de la carte du réseau routier
Opérationnelle	Communiquer	Message, Destinataire	Envoi du Message / Dialogue avec le Destinataire (selon contexte)	
Tactique	Analyser		Réflexion sur observation, déduction nouveaux événements, diffusion autres agents (collaboration partielle)	<i>Carte mentale</i> modifiée avec nouveaux événements détectés
Tactique	Evaluer Chemin	Itinéraire	Evaluation d'un nouvel itinéraire par lecture de la carte mentale et en tenir compte des événements	Itinéraire modifié

TABLE 9.3 – Synthèse des comportements modifiés et ajoutés à la fin de la deuxième itération

9.2.3/ OPTIMISATION

Pour cette dernière itération, nous nous intéressons à la dernière couche (niveau stratégique). Ici nous cherchons à définir le meilleur chemin possible pour les agents *véhicules*. Cela passe par une prise en compte des expériences (capitalisation) ainsi qu'une prise en compte en temps réel du trafic (grâce à la diffusion d'informations et aux communications directes). L'objectif ultime est une auto-organisation progressive du trafic par échanges de données.

Nous proposons le comportement *Construire Itinéraire* qui est un dialogue entre agents afin d'échanger des informations issues des *cartes mentales* et des données capitalisées (itinéraire modifié légèrement à chaque cycle d'exécution).

Les interactions sont potentiellement très nombreuses et les échanges compliqués à mettre en œuvre compte tenu de la quantité d'informations à transmettre (entre les véhicules et avec le *cloud*). Nous sommes sur un processus de collaboration complète avec partage d'objectifs et de moyens avec le but d'anticiper les évolutions du trafic et de l'auto-réguler. Par exemple, deux véhicules peuvent échanger des données de cartographie et d'itinéraire pour créer conjointement un itinéraire commun sur une partie de leur itinéraire personnel. Il faut dialoguer pour arriver à un compromis. Au départ, nous sommes dans une situation à deux véhicules mais les groupes vont progressivement s'auto-organiser entre eux. Le nombre d'échanges et la complexité pour arriver à trouver

Couche	Comportement	Variables en entrée	Action	Variables en sortie
Stratégique	Construire Itinéraire		Dialogue entre véhicules pour construire les itinéraires (collaboration complète)	Itinéraire créé conjointement

TABLE 9.4 – Synthèse des comportements modifiés et ajoutés à la fin de la troisième itération

un itinéraire commun sans encombrer les routes du parcours seront croissants.

La table 9.4 montre le comportement ajouté à l'issue de cette troisième itération d'analyse.

9.3/ SYNTHÈSE

9.3.1/ COMPORTEMENTS

La figure 9.3 présente l'ensemble des comportements pour un véhicule, couche par couche et leur mise en relation.

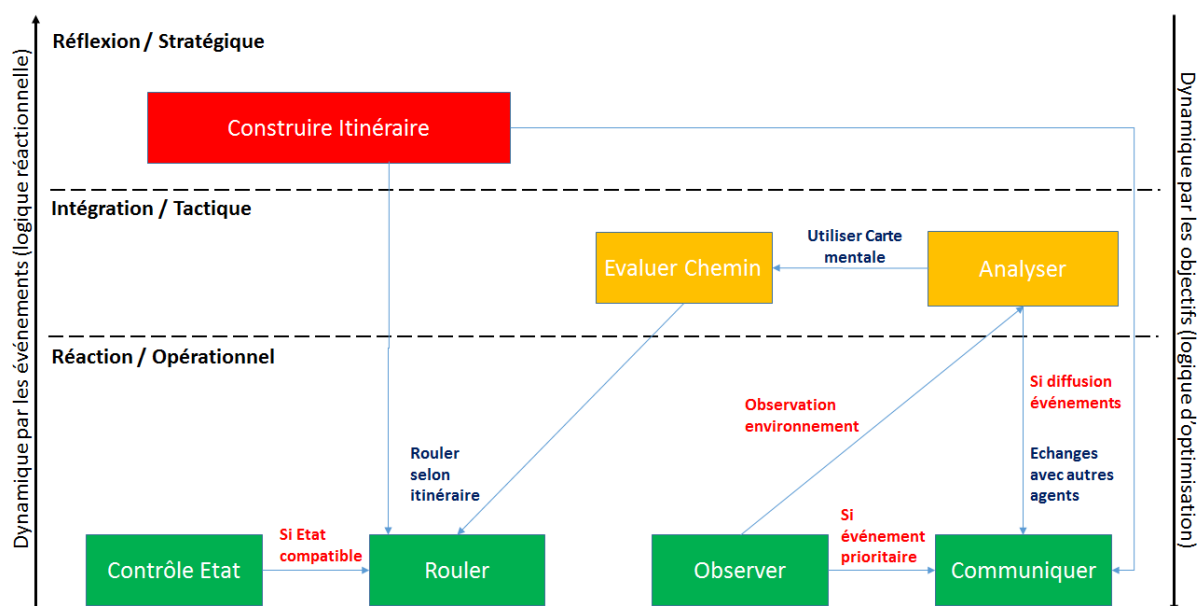


FIGURE 9.3 – Cartographie des comportements pour un véhicule

Nous retrouvons les comportements "clés" du véhicule : le *Contrôle Etat*, *Rouler*, *Observer* et *Communiquer*. Les comportements suivants permettent d'améliorer l'efficacité du véhicule : *Evaluer Chemin* et *Analyser*, s'appuyant également sur les comportements basiques. Enfin nous avons un comportement plus complexe, s'appuyant aussi sur les

comportements précédemment cités : *Construire Itinéraire*, utilisant la *carte mentale* et la brique de communication pour optimiser des déplacements (minimiser la distance parcourue et le temps de déplacement tout en évitant les secteurs problématiques) en provoquant des échanges structurés entre véhicules.

9.3.2/ CONNAISSANCES

Si nous reprenons la structure de *carte mentale* décrite dans le chapitre 8, nous pouvons préciser, couche par couche, ce que nous gérons pour chaque véhicule.

La couche *Environnement* est constituée par le réseau routier (les informations sur l'état des portions de routes sont également enregistrées). Sur la couche *Agents*, nous trouvons les différents agents en leur situation géographique à l'instant de leur rencontre avec le véhicule courant. A noter que dans cette configuration, l'agent *cloud* n'apparaît pas ici car nous sous-entendons qu'il s'agit d'un élément commun, certes dynamique mais immuable pour tous les agents Véhicule. La couche *Événements* regroupe les différents résultats issus de la phase d'analyse (*accident*, *embouteillage*). La couche *Communications* matérialise les contacts en cours entre agents. Et la couche *Groupes* permet d'enregistrer les différents groupes formés pour les phases d'optimisation.

Prenons le cas de l'agent *vehicle4* (cf. figure 9.4), extrait de notre exemple de la section 9.2.2.

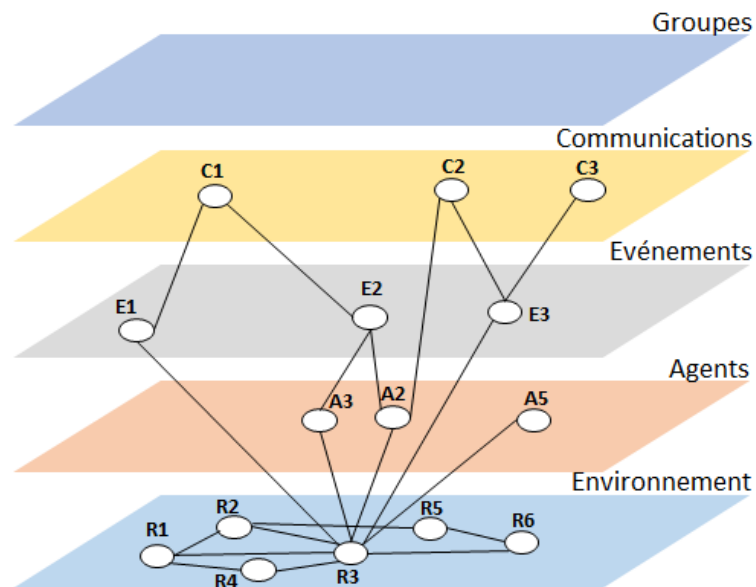


FIGURE 9.4 – Exemple de Carte mentale pour l'agent *vehicle4*

Le réseau routier comprend les routes R1 (*road1*) à R6 (*road6*). *Vehicle4* observe les agents A2 (*vehicle2*), A3 (*vehicle3*) et A5 (*vehicle5*) situés sur R3. *Vehicle4* observe que R3 est bloquée et crée E1. Il déduit un embouteillage (E2) impliquant A2 et A3 sur R3. Il diffuse alors E1 et E2 (communication C1). Il reçoit un message (communication C2) de A2 l'informant d'un accident (E3) sur R3. Il propage enfin E3 (communication C3). Toutes les informations sont correctement contextualisées.

9.4/ CE QU'IL FAUT RETENIR

Nous avons présenté dans ce chapitre une instanciation du modèle HACCA en utilisant la méthode d'analyse décrite au [chapitre 7](#) ainsi que les concepts présentés dans le [chapitre 8](#).

La figure 9.5 résume les comportements et les concepts. L'instanciation porte sur des agents de type *véhicule* et *cloud*. Nous retrouvons les objectifs du modèle (trajets domicile/travail) avec la minimisation de la distance parcourue et du temps de déplacement. Les différents comportements sont repris couche par couche (opérationnelle, tactique et stratégique). Les éléments de la *carte mentale* sont décrits ainsi que les événements reconnus (*accident*, *embouteillage*). Enfin nous précisons à nouveau les trois types de collaboration mentionnés (partielle, induite et complète).

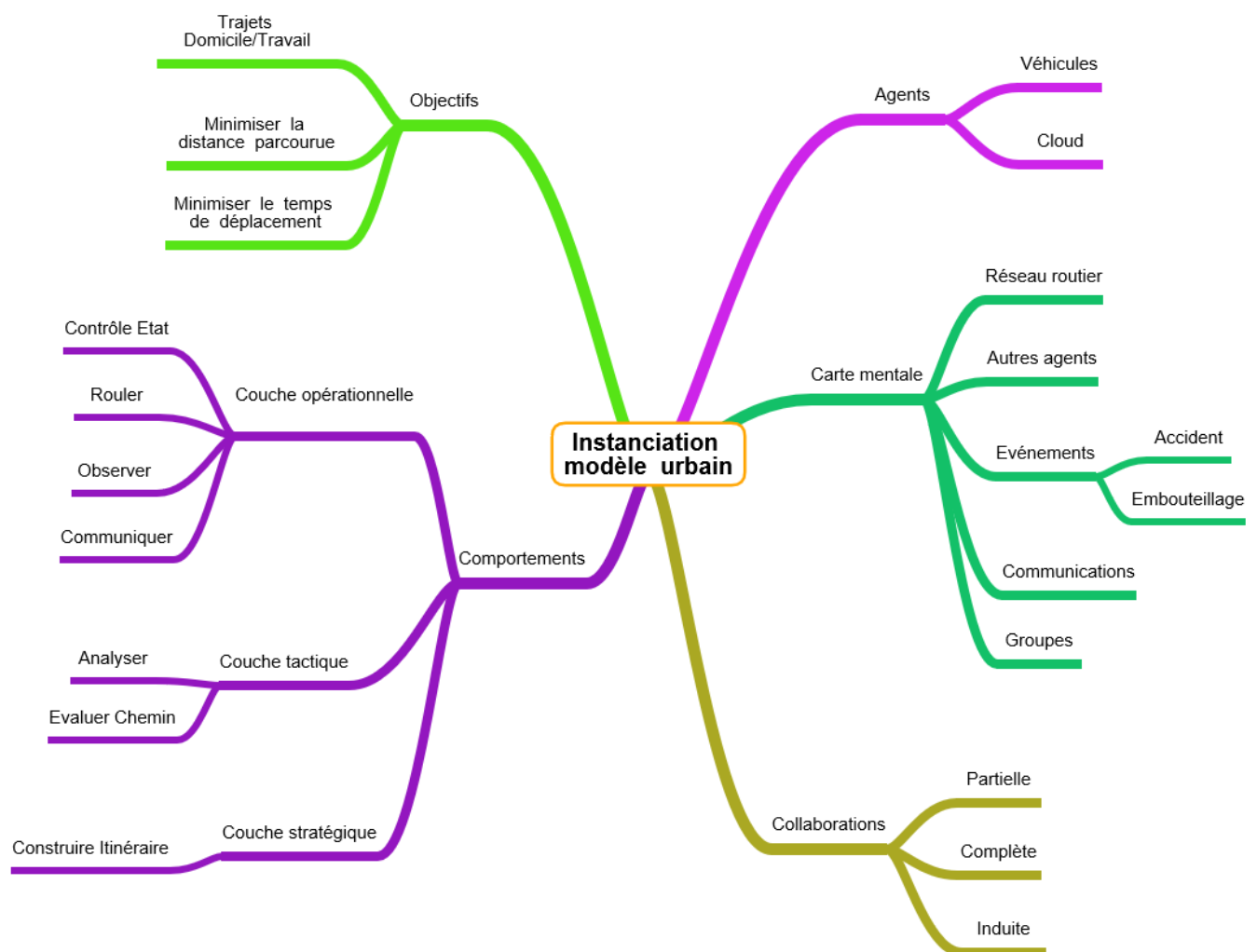


FIGURE 9.5 – Ce qu'il faut retenir : Instanciation du modèle urbain

Le [chapitre 10](#) décrit l'implémentation de ce modèle et présente les résultats d'expérience.

IMPLÉMENTATION ET EXPÉRIMENTATIONS DU MODÈLE URBAIN

Sommaire

10.1 Modeleur GAMA	147
10.1.1 Justification du choix	148
10.1.2 Présentation générale	148
10.1.3 Concepts retenus	149
10.2 Implémentation	149
10.2.1 But de la modélisation	150
10.2.2 Entités, Variables d'état, Echelles	150
10.2.3 Revue des processus	150
10.2.4 Concepts d'implémentation	151
10.3 Expérimentations	153
10.4 Difficultés	153
10.5 Ce qu'il faut retenir	154

Dans ce chapitre, nous implémentons l'instance du modèle urbain décrit au [chapitre 9](#) en utilisant le modeleur GAMA.

Après une justification de l'utilisation de GAMA, nous décrivons rapidement l'interface de ce modeleur. Puis nous proposons une implémentation du modèle en nous appuyant sur le protocole de description ODD ([Grimm et al., 2006] et [Grimm et al., 2010]). Enfin nous présentons les résultats de l'expérimentation.

10.1/ MODELEUR GAMA

GAMA ([Grignard et al., 2013]) est une plate-forme libre de modélisation et de simulation à base d'agents née en 2007 et en développement depuis. Cette plateforme permet de construire des simulations à temps discrets (temps segmenté en un ensemble de pas de simulation ayant tous la même durée). A chaque cycle de simulation, tous les agents sont activés et peuvent agir (modification de leur état ou de l'état du monde). La version 1.6 de la plateforme utilisée dans le cadre de cette étude est disponible depuis octobre 2013.

GAMA se veut multi-domaines (écologie, agriculture, mobilité urbaine, etc) mais avec une forte composante géographique ([Taillandier et al., 2012] et [Drogoul et al., 2013]).

Dans les sections suivantes, nous justifions le choix de cette plateforme et nous la présentons rapidement. Enfin nous décrivons les concepts que nous retenons pour cette implémentation.

10.1.1/ JUSTIFICATION DU CHOIX

Dans le cadre de cette étude, le choix du modeleur est apparu évident, la facilité de conception étant l'enjeu principal.

GAMA se prête bien à l'étude de phénomènes spatialisés et permet justement une conception assez aisée grâce à son langage de haut-niveau et son environnement complet de développement. De plus, il dispose d'une bibliothèque importante de fonctions et d'opérateurs spécialisés.

Enfin, Nicolas Marilleau est un expert de cette plateforme et nous avons donc un accès facile au support.

10.1.2/ PRÉSENTATION GÉNÉRALE

GAMA fournit un environnement de développement intégré complet avec un langage de modélisation (GAML), un éditeur de code, des outils d'aide à la programmation (compilation, détection d'erreurs, etc). Le langage GAML a été conçu pour être utilisé par tout type de personne (du débutant au plus expérimenté) qu'il soit informaticien ou non. A l'instar de langages comme NetLogo, il emploie une grammaire simple et dispose de nombreuses fonctions pré-programmées. GAML est aussi un langage de programmation orienté agent (tout est objet dans GAMA). Afin de créer un modèle en GAML, il faut définir trois types de bloc : le bloc global, les espèces et les expérimentations.

Le bloc global représente le modèle du monde et décrit les caractéristiques et les fonctions qui peuvent concerner tous les agents. Le lancement d'une simulation correspond à la création du monde qui peut à son tour créer d'autres agents.

Une espèce (ou classe en programmation orientée objet) est un archétype d'agent qui définit pour un type d'agents quels sont ses caractéristiques, ses comportements et ses représentations visuelles.

Les caractéristiques d'une espèce d'agents sont définies au travers de variables qui définissent l'état interne des agents. Les variables en GAML peuvent être statiques ou dynamiques (par exemple, recalculées à chaque pas de simulation) et sont typées (entier, réel, point, liste, graphe, etc).

Le comportement d'une espèce d'agents est défini au travers de réflexes et d'actions. Un réflexe est un bloc d'instructions qui est exécuté à chaque pas de simulation lorsque sa condition d'exécution est vérifiée. Une action représente une capacité que l'agent possède et à laquelle il pourra faire appel. Il s'agit d'un bloc d'instructions qui est exécuté lorsque l'action est appelée. Une action correspond à une méthode en programmation orientée objet.

La représentation visuelle d'un agent est définie au travers d'aspects (manière de

représenter un agent). Un agent peut disposer de plusieurs aspects (selon le contexte, en 2D ou en 3D).

Le bloc de type expérimentation fournit un cadre d'exécution à la simulation. En particulier, on peut définir les entrées (paramètres) et les sorties (fenêtre d'affichage, moniteurs).

10.1.3/ CONCEPTS RETENUS

Afin d'implémenter une première représentation d'HACCA dans GAMA, nous devons choisir ce qui est le plus pratique pour transcrire les concepts fondamentaux de l'architecture notamment les trois couches de traitement (opérationnelle, tactique et stratégique).

Nous allons utiliser les réflexes et les actions. En effet, les réflexes permettent de modéliser les comportements de la couche réactive ou opérationnelle car devant réagir immédiatement en fonction des stimuli et des observations de l'environnement. Les actions permettent de coder les deux autres couches de traitement (tactique et stratégique) et sont appelées directement par l'intermédiaire des réflexes de la première couche.

Il existe une autre possibilité pour gérer les couches de traitement. GAMA propose une implémentation de l'architecture BDI (Belief, Desire, Intention). Nous pourrions utiliser également celle-ci afin de constituer des plans d'actions exécutés via des intentions de l'agent ([Taillandier et al., 2016]). Cependant, la conception est moins aisée et cette architecture nécessite l'exécution de la version de développement de GAMA.

Nous implémentons l'architecture HACCA dans un contexte urbain comme nous l'avons décrit dans le chapitre 9. GAMA met à disposition un certain nombre de compétences ou *skill* facilitant la mise en œuvre d'agents. Par exemple, le *skill advanced driving* permet de créer rapidement un agent mobile dans un environnement de type graphe ([Taillandier, 2014]). Le gros avantage est que ce *skill* gère automatiquement les paramètres routiers tels que les probabilités de respect du code de la route, de changement de voies de circulation, les facteurs d'accélération, etc. Cette compétence offre des réflexes et des actions spécifiques comme la gestion des déplacements en s'appuyant sur un réseau routier de type graphe, qui peut être initialisé grâce à la lecture de fichiers *shape* issus de systèmes géographiques. Ces fichiers contiennent des informations sur la taille des axes routiers, les vitesses théoriques, le nombre de voies, le sens de circulation, etc. Le graphe résultant est donc orienté mais aussi pondéré.

La *carte mentale* peut être implémentée en utilisant des variables de type graphe ou une série de variables de type liste selon la difficulté de programmation.

Les communications seront simulées en implémentant une boîte aux lettres d'envoi et de réception de messages. Elle sera pilotée par un réflexe. Les principes d'implémentation sont les mêmes pour les agents de type *véhicule* et l'agent *cloud*.

10.2/ IMPLÉMENTATION

Dans cette section, nous détaillons l'implémentation du modèle urbain en utilisant le protocole ODD ([Grimm et al., 2006] et [Grimm et al., 2010]). Nous reprenons les items proposés : le but de la modélisation, la description des entités et des variables, la revue des processus, la description des concepts d'implémentation.

10.2.1/ BUT DE LA MODÉLISATION

Nous voulons implémenter l'exemple d'instanciation décrit au chapitre précédent (chapitre 9). Nous n'allons pas reproduire ici toute l'analyse déjà effectuée par ailleurs.

Pour résumer, il s'agit de décrire un système où des agents véhicules circulent dans un espace urbain pour se rendre d'un point A (domicile) à un point B (travail). Des échanges d'information entre les véhicules et avec une infrastructure centralisée permettent à chacun de choisir l'itinéraire le plus adapté et le plus rapide.

Nous allons tester les différents types de véhicules que nous avons identifiés : standard, intelligent (disposant de capacités d'analyse), communicant (capacités de communication avec son entourage direct) et connecté (communication longue distance avec une infrastructure centralisée). Afin d'évaluer leur efficacité, nous avons choisi deux indicateurs : la vitesse moyenne des véhicules dans l'environnement et le nombre de véhicules à l'arrêt.

10.2.2/ ENTITÉS, VARIABLES D'ÉTAT, ECHELLES

Nous utilisons principalement deux types d'agent : véhicule et cloud.

L'espèce (ou *species*) *vehicle* implémente l'agent véhicule. Les principales variables d'états sont :

- Les variables définissant le type de véhicule : *smart*, *communicating*, *connected* de type booléen ;
- Les variables définissant l'état du véhicule : *alive*, *stopped*, *breakdown* de type booléen ;
- Les variables de description du réseau : *local_road_network* de type graphe et *blocked_road* de type liste pour enregistrer les différentes portions de routes bloquées ;
- Les variables contenant les nœuds domicile et travail : *home_node*, *work_node* ;
- La variable de gestion de la boîte aux lettres : *mailbox* de type liste.

La *species cloud* implémente l'agent cloud. Nous retrouvons principalement deux séries de variables, celles permettant la gestion du réseau routier et celles gérant la boîte aux lettres.

L'échelle de simulation est la seconde. Cela permet un meilleur suivi microscopique des interactions et des échanges de messages.

10.2.3/ REVUE DES PROCESSUS

Nous avons décrit dans le chapitre précédent l'ensemble des comportements souhaités. La figure 10.1 transcrit ces comportements en réflexes et actions GAMA.

Le réflexe *get_status* est le comportement *Contrôle Etat*. Le réflexe *move* correspond au comportement *Rouler*. Le réflexe *observe* est la transcription du comportement *Observer*. Un véhicule arrêté ne constitue pas un embouteillage. Par contre, si plus de 5 véhicules sont arrêtés pendant plus de 5 cycles, un embouteillage est signalé.

L'action *evaluate* correspond au comportement *Evaluer Chemin*. L'action *find_path*

exécute une fonction du *skill advanced_driving* qui recalcule un itinéraire par lecture de la copie individuelle du réseau routier (mise à jour à chaque échange ou observation). L'action *analyze_traffic* est la transcription du comportement *Analyser*.

Les autres actions sont des sous-routines permettant de fragmenter le code principal en fonctions plus petites. A noter que l'action *get_messages* gère la boîte aux lettres. Les actions *diffuse_all_around* et *diffuse_cloud* gèrent les envois de messages vers les autres entités (véhicule ou cloud).

Pour certains réflexes et actions, nous explicitons les conditions d'exécution comme par exemple, de lancer le réflexe *get_status* si le véhicule est actif, le réflexe *move* si le véhicule n'est pas accidenté ou le réflexe d'observation si c'est un véhicule intelligent.

Pour cette première implémentation et compte tenu de sa complexité, le comportement *Construire Itinéraire* n'a pas été repris dans GAMA (cf. section 10.4 sur les difficultés de mise en œuvre).

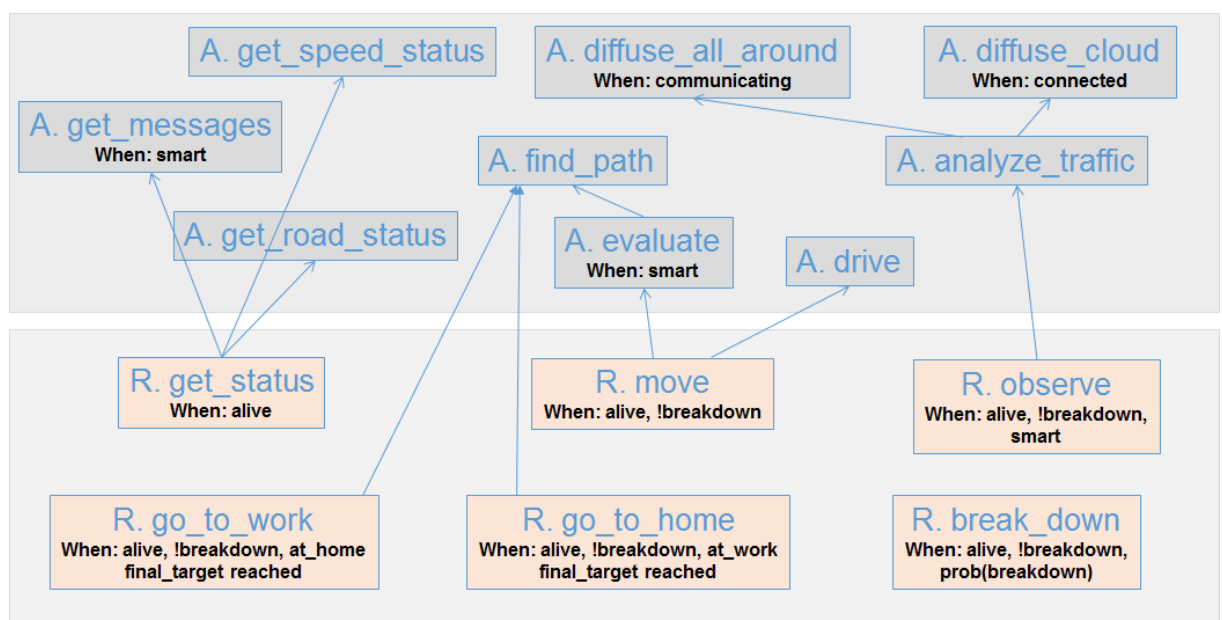


FIGURE 10.1 – Cartographie des fonctions sous GAMA (R pour *reflex* et A pour *action*)

10.2.4/ CONCEPTS D'IMPLÉMENTATION

L'objectif de cette section n'est pas de décrire l'implémentation dans les moindres détails du code mais de fournir les indications nécessaires à une compréhension globale. Nous reprenons les items du protocole ODD en les regroupant : principes fondamentaux, phase d'initialisation, interactions, apprentissage, adaptation, émergence et observations.

Principes fondamentaux et Objectifs

Pour chaque véhicule, il s'agit d'un déplacement domicile/travail. Nous utilisons un réseau routier qui est un graphe orienté et nous utilisons des listes pour enregistrer les routes bloquées pour chaque agent. Chaque véhicule conserve une copie du réseau et son propre historique des événements. Il dispose également d'une boîte aux lettres pour envoyer et recevoir des messages (en provenance d'autres agents ou de l'infrastructure

centralisée). Nous utilisons les fonctions implémentées dans le *skill advanced driving* pour gérer les déplacements sur les arcs de réseau.

Globalement, chaque véhicule doit aller du point A au point B en minimisant le temps et la distance parcourue tout en évitant les zones d'embouteillage.

Initialisation et Données d'entrée

La phase d'initialisation comprend la lecture des fichiers *shape*, la création du réseau routier (graphe orienté), la création des agents véhicules et cloud.

Les données d'entrée correspondent aux informations de graphe routier contenues dans les fichiers *shape* avec la description des routes et des carrefours.

Détection et Interactions

Nous simulons une observation à 360 degrés autour du véhicule. L'utilisation du *skill advanced driving*, combiné aux opérateurs GAMA, permet de déterminer la position des véhicules (devant, derrière, sur la file d'à côté, etc) et de mettre à jour sa *carte mentale*.

Il y a plusieurs interactions possibles : entre les véhicules (de manière instantanée si les possibilités de communication sont actives) et avec le cloud (si les possibilités techniques le permettent avec une temporisation forcée à 3 minutes pour simuler les problèmes de communication et la congestion des messages).

Apprentissage et Prédictions

L'apprentissage s'effectue de deux manières : observation directe de l'environnement et échange d'informations (collaboration). Il y a un apprentissage individuel et un apprentissage collectif ou social.

Au fur et à mesure des expériences et des échanges, le véhicule choisit le meilleur itinéraire possible compte tenu des aléas. Il améliore son processus de décision (actions *evaluate*, *find_path*) par analyse de sa copie individuelle du graphe de réseau routier.

Stochasticité et Adaptation

Des événements aléatoires de type accident ou bouchon sont générés par le réflexe *break_down*. Chaque véhicule a une probabilité non nulle de tomber en panne et une probabilité de "réparation".

Les véhicules s'adaptent progressivement au trafic en échangeant leurs informations sur l'état du réseau routier et en évaluant le nouvel itinéraire en fonction des données reçues.

Emergence et Collectifs

Les différents échanges entre les véhicules permettent progressivement de fluidifier leur déplacement et plus généralement le trafic urbain. Il s'agit d'un phénomène émergent.

Cependant, même s'il y a un échange d'informations, il n'y a pas de collectif défini.

Observations

Afin de générer les courbes de résultats, des fichiers CSV sont écrits à intervalles réguliers (toutes les 20 secondes). Nous verrons les résultats dans la section suivante.

10.3/ EXPÉRIMENTATIONS

Nous avons choisi d'effectuer nos expérimentations en utilisant un extrait du réseau routier du centre ville de Dijon. Pendant la simulation, nous générons des événements aléatoires (comme un accident ou un véhicule en panne) pour créer des embouteillages. Comme nous l'avons déjà mentionné, nous utilisons deux indicateurs : la vitesse moyenne des véhicules dans l'environnement et le nombre de véhicules à l'arrêt.

La figure 10.2 illustre l'écran principal de la plateforme GAMA. En haut à droite, nous avons un zoom sur le réseau routier. Les axes en bleu représentent les routes principales où les vitesses théoriques de circulation sont les plus grandes. Les véhicules sont représentés sous la forme de triangles de couleur. Les carrés rouges sont les zones d'embouteillage détectées par les véhicules en circulation. En bas à droite, nous apercevons une partie du code. La partie gauche est la console affichant les différents messages générés lors de la simulation. Nous pouvons y voir une détection d'embouteillage (Traffic Jam) et une diffusion de l'information de proche en proche ainsi qu'au cloud.

Nous avons testé les déplacements urbains de 700 véhicules dans cette zone assez restreinte pour évaluer tour à tour les différents types de véhicules et leur impact sur le trafic. Nous avons donc effectué 10 simulations avec 700 véhicules standards, 10 simulations avec 700 véhicules intelligents (analyse de l'environnement), 10 simulations avec 700 véhicules communicants (analyse de l'environnement et communications de proche en proche) et 10 simulations avec 700 véhicules connectés (analyse, communications de proche en proche avec les autres véhicules et communications avec l'infrastructure centralisée). Les résultats de la simulation sont présentés sur la figure 10.3 et la figure 10.4. Il s'agit de courbes moyennes et pour une meilleure comparaison, tous les types de véhicules apparaissent en même temps sur les 2 figures. Sur l'axe des abscisses, nous trouvons le temps de simulation en secondes. Sur l'axe des ordonnées, nous avons la vitesse des véhicules en mètres par seconde pour la figure 10.3 (évolution des vitesses moyennes des 700 véhicules simulés, une courbe par type de véhicule) et nous avons le nombre de véhicules arrêtés pour la figure 10.4 (évolution du nombre de véhicules arrêtés, une courbe par type).

Nous pouvons noter que la vitesse moyenne est significativement plus élevée lorsque les véhicules commencent à échanger des informations sur les tendances du trafic pour permettre un changement dynamique de la trajectoire. Alors que l'on observe une dégradation de la vitesse moyenne pour les véhicules standards, les communications entre véhicules permettent de maintenir une vitesse moyenne à peu près constante sur cet échantillon. Nous constatons également que les véhicules entièrement connectés (avec échange d'informations supplémentaires avec une entité centralisée) maintiennent une bien meilleure vitesse moyenne. Par corrélation, le nombre de véhicules à l'arrêt a tendance à rester stable lorsque les véhicules communiquent entre eux. Encore une fois, les véhicules entièrement connectés sont plus fréquemment en mouvement.

10.4/ DIFFICULTÉS

Nous avons rencontré de nombreuses difficultés d'implémentation, de mise en œuvre et d'exécution.

En effet, l'architecture HACCA préconise l'utilisation d'une *carte mentale* sous la forme d'un multigraphe comme nous l'avons décrit dans le chapitre 8. Or il n'est pas facile d'implémenter cette structure avec les types disponibles nativement dans GAMA. C'est pour cette raison que nous avons utilisé un graphe pour le réseau routier et une liste pour enregistrer les routes bloquées sans les relier formellement.

De plus, pour ce premier essai, il n'a pas été possible de programmer le comportement *Construire Itinéraire* car nous avons constaté un nombre très important de messages échangés entre les véhicules lors des phases d'analyse et de diffusion des informations de trafic. Cela a ralenti considérablement les simulations.

Enfin, nous avons également observé une fuite mémoire critique récurrente lors de l'utilisation du *skill advanced_driving*. Dans le code de la fonction *find_path*, nous utilisons la routine *compute_path*. Il semble qu'elle soit à l'origine des dysfonctionnements. Ces ralentissements importants et l'utilisation massive de la mémoire nous ont contraint à limiter les simulations sur des durées de 10 à 12 minutes.

10.5/ CE QU'IL FAUT RETENIR

Nous avons pu transcrire une partie des concepts fondamentaux de l'architecture HACCA en utilisant la plateforme GAMA notamment les réflexes et les actions pour simuler les couches opérationnelles et tactiques.

Nous avons utilisé le protocole ODD pour décrire les concepts de développement. Sans rentrer dans le détail du code, il a permis de mieux appréhender le modèle.

Même si elle est incomplète, cette première implémentation est assez prometteuse. Il faut maintenant lever les difficultés techniques liées à l'emploi du *skill advanced_driving*. La *carte mentale* devrait pouvoir être codée en modifiant ou en ajoutant de nouvelles structures de données. Heureusement, l'utilisation du graphe orienté pour le réseau routier a facilité les premiers développements.

Les premiers résultats montrent l'intérêt des échanges collaboratifs sans pour autant aller dans la programmation intégrale des structures évoquées dans le chapitre 6. Les véhicules ayant le plus de capacités de communication sont les plus aptes à se déplacer rapidement et sans encombre en milieu urbain.

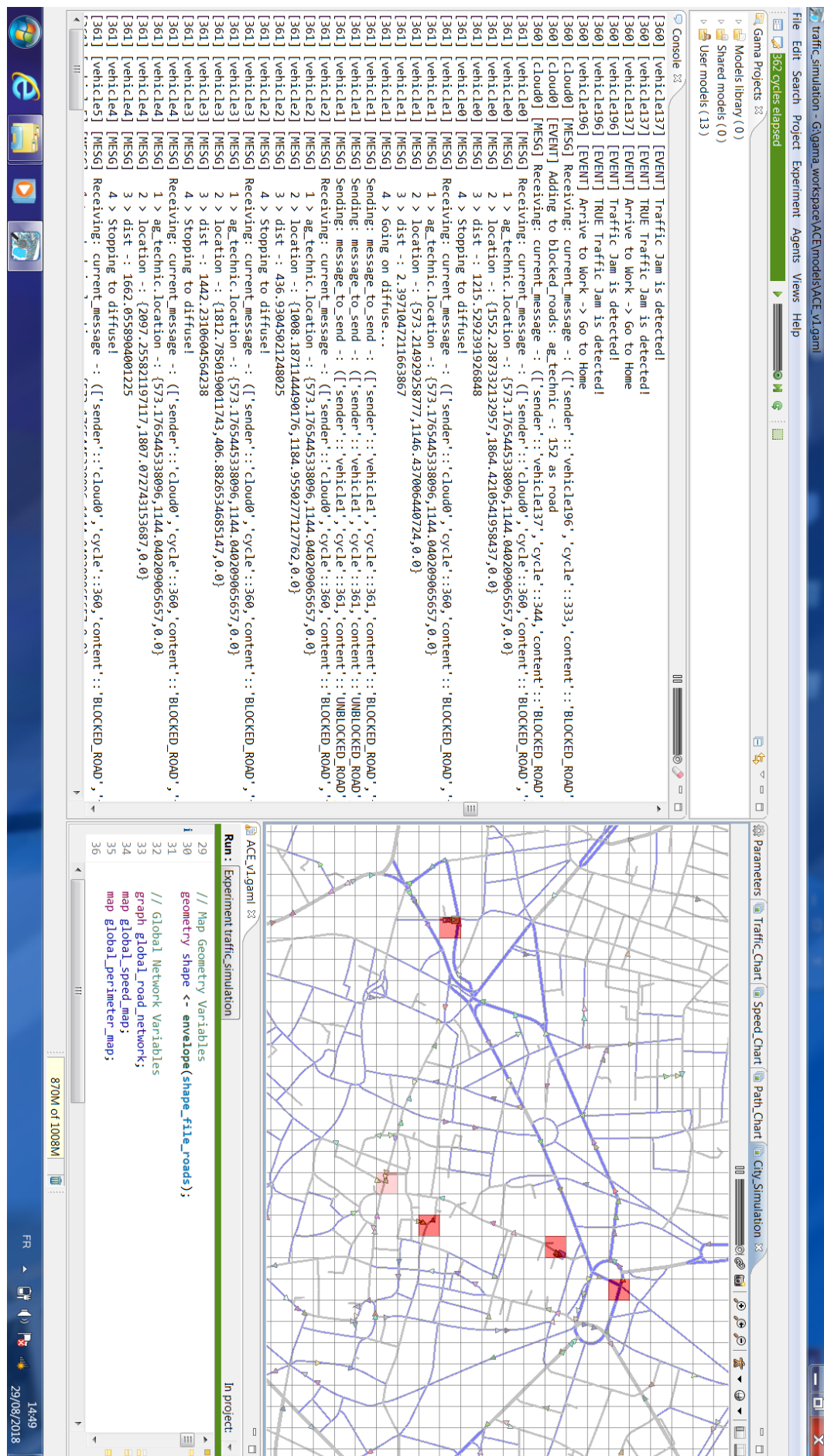


FIGURE 10.2 – Exemple HACCA dans GAMA

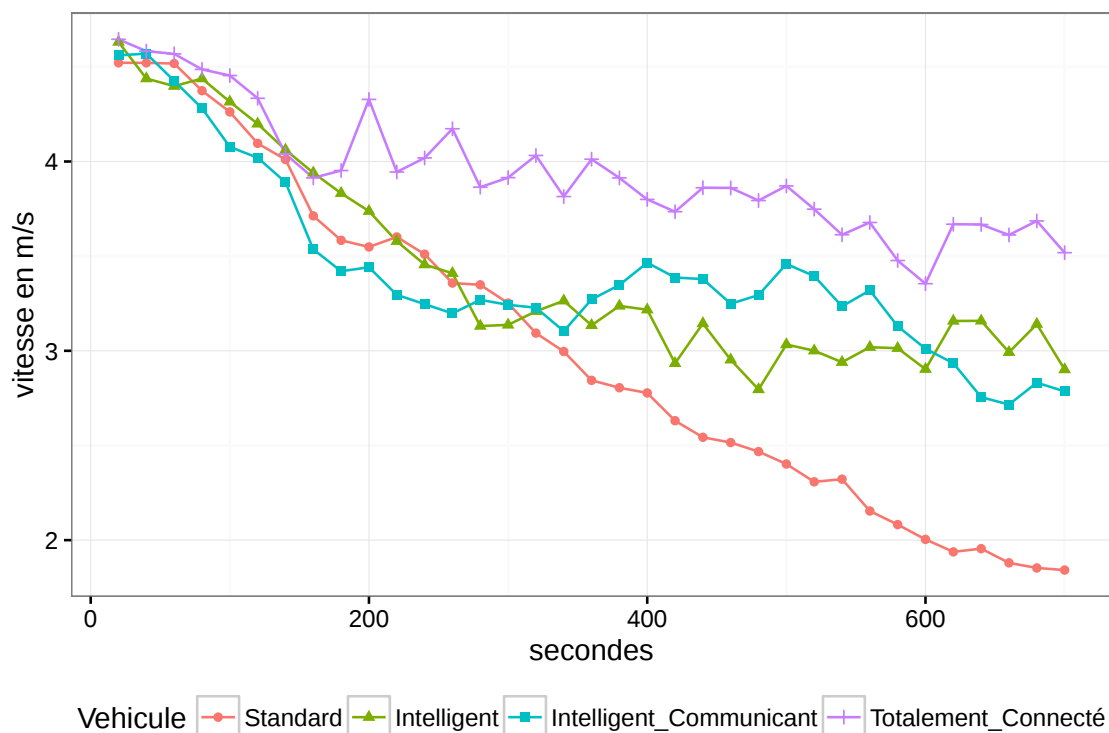


FIGURE 10.3 – Simulation avec 700 agents (vitesse moyenne)

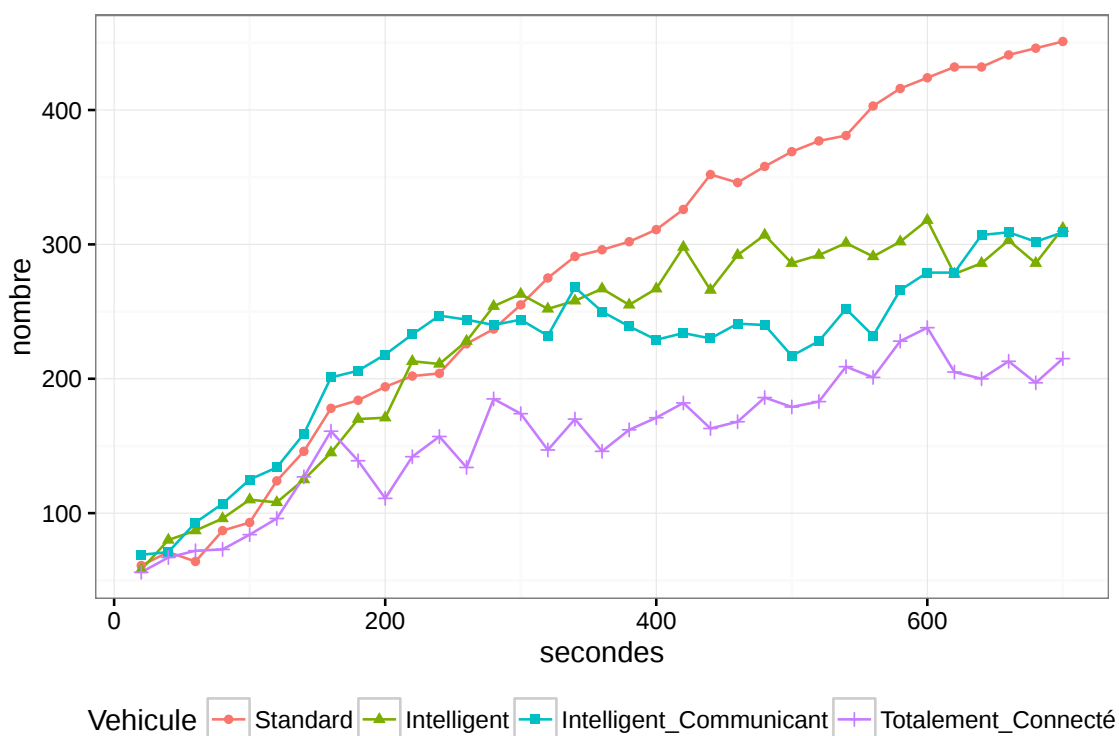


FIGURE 10.4 – Simulation avec 700 agents (nombre de véhicules arrêtés)

CONCLUSION ET PERSPECTIVES

Sommaire

11.1 Conclusion générale	157
11.2 Perspectives	158
11.2.1 Dans le domaine des systèmes multi-agents	158
11.2.2 Dans le monde réel	159

Au terme de cette étude, nous pouvons maintenant conclure en reprenant les différentes questions posées et les contributions proposées. Dans une dernière section, nous évoquerons les perspectives possibles tant dans le domaine des systèmes multi-agents que dans le monde réel.

11.1/ CONCLUSION GÉNÉRALE

Nous avons concentré cette étude sur les systèmes multi-agents. Aussi il a été nécessaire de considérer tout d'abord les freins et les enjeux de la mise en application du concept de collaboration dans ce domaine. Nous avons effectué un état de l'art non-exhaustif car il était quasiment impossible d'en présenter un complet tant la collaboration couvre de sous-domaines d'étude.

Par la suite, nous avons détaillé ce qu'était la collaboration en proposant notre définition et en caractérisant le mieux possible ce que ce concept engendrait. Nous avons présenté également les caractéristiques fonctionnelles et techniques en les regroupant par type et en définissant le gradient de collaboration. Nous avons considéré que la collaboration était un processus intentionnel avec un partage d'informations et l'objectif de réaliser une œuvre commune mais également qu'elle devait être appréhendée d'une manière globale (agents, informations, comportements, protocoles de communication, objectifs et résultats). En utilisant la méthode des quadrants, nous avons montré qu'il s'agissait d'un processus englobant totalement tous les aspects de l'agent (de l'esprit au corps, de l'espace brut à la culture) et nous avons proposé une série d'algorithmes permettant la réflexion sur l'utilité ou l'opportunité de collaborer ainsi que la mise en œuvre de celle-ci.

Notre exemple fil rouge (drones de surveillance de forêts) nous a permis de définir une méthode d'analyse de problème pour mettre en évidence et décrire des comportements collaboratifs pour des entités mobiles intelligentes. Nous avons pu montrer l'importance d'un processus itératif permettant de raffiner les comportements et mettre le doigt sur l'intérêt de mettre en place une collaboration entre agents afin de résoudre certains

problèmes plus efficacement, de répondre aux objectifs ou de permettre une augmentation des capacités de l'agent grâce à l'échange d'informations (apprentissage social).

Au-delà des aspects purement théoriques, nous avons également proposé l'architecture HACCA (Hybrid Architecture for Collaborative Communicating Agent) répondant aux différents constats présentés dans le cas d'agents mobiles autonomes : besoin de prendre en considération le contexte spatio-temporel mais aussi d'avoir des informations contextualisées et pouvant être communiquées efficacement en fonction des niveaux de réaction/réflexion (opérationnel, tactique et stratégique). Ainsi nous avons développé cette architecture autour de trois couches de traitement (réaction, intégration et réflexion), de règles de comportements hiérarchisés (actions basiques, plans d'actions et plans complexes) et d'une base de connaissance sous la forme d'un multigraphe permettant de relier toutes les connaissances en fonction du contexte spatio-temporel. Un module de communication transversal complète cette architecture.

Afin de montrer la faisabilité technique, nous avons choisi d'implémenter une instanciation de cette architecture en modélisant un milieu urbain avec des agents véhicules intelligents et communicants. Nous avons choisi une implémentation sur la plateforme de modélisation et de simulation de systèmes multi-agents GAMA. Les différentes couches de traitement ont été programmées notamment autour des concepts de réflexe et d'action.

L'objectif principal était de démontrer que cela fonctionnait car nous ne pouvions pas établir de comparaison avec d'autres systèmes. Les résultats obtenus montrent effectivement tout l'intérêt qu'ont les agents à collaborer. Dans ce contexte de circulation dans un centre urbain, la collaboration entre les véhicules a permis d'optimiser leur vitesse moyenne et d'éviter d'avoir un trop grand nombre de voitures arrêtées pour cause d'aléas de circulation (accident ou bouchon). Ainsi, plus globalement, cela a permis au système d'être plus fluide.

Cette première proposition n'est certes pas complète mais elle pose les bases d'une future implémentation plus exhaustive avec la gestion plus fine des concepts collaboratifs, de la gestion de la connaissance ainsi que des aspects de communication.

11.2/ PERSPECTIVES

Afin d'ouvrir quelques pistes de réflexion, nous proposons quelques perspectives tout d'abord dans le domaine des systèmes multi-agents et plus généralement dans le monde réel en transposant l'architecture HACCA et ses concepts.

11.2.1/ DANS LE DOMAINE DES SYSTÈMES MULTI-AGENTS

Au niveau méthodologique, une perspective serait d'intégrer le questionnement explicite sur les processus collaboratifs dans le protocole ODD. En effet, même si l'on trouve des sections dans les concepts d'implémentation permettant de poser des éléments de comportements collaboratifs (apprentissage, prédictions, interactions ou collectifs), aucune ne permet totalement de les mettre en avant. Il s'agit d'un instantané d'implémentation d'un modèle, éloigné de la démarche intellectuelle permettant d'arriver à la conclusion du besoin de collaboration et des caractéristiques de celle-ci à intégrer dans les comporte-

ments de l'agent modélisé.

Concernant le modèle HACCA dans GAMA, comme nous l'avons déjà indiqué, il est encore assez incomplet. Tout d'abord, on pourrait implémenter directement une bibliothèque permettant la gestion du multi-graphe portant la connaissance. Puis une autre bibliothèque pourrait être écrite pour faciliter la gestion des comportements avec une vue externe pour une implémentation dynamique par fichiers de configuration (au format JSON ou XML) externes à la plateforme, soit sous la forme d'un graphe d'exécution de comportements basiques (un peu comme un réseau de Pétri), soit sous la forme de pseudo-programmes externes. Enfin, il serait à terme envisageable d'écrire un *skill* GAMA reprenant l'ensemble des fondamentaux de l'architecture HACCA en renommant éventuellement les concepts de *reflex*, *action* et *plan*.

11.2.2/ DANS LE MONDE RÉEL

Plus généralement, nous pourrions utiliser l'architecture HACCA dans un contexte d'intelligence ambiante ou pour les réseaux de capteurs. Ainsi chaque entité, porteuse de connaissance, pourrait mettre à disposition ses capacités de traitement. La collaboration permettrait alors de structurer les échanges pour accomplir des missions d'analyse et de traitement. Elle pourrait également être porteuse d'organisation (auto-organisation par réalisation de mission).

Dans un système plus ouvert avec des agents hétérogènes, cela pourrait contribuer à plus d'échanges d'informations et de capacités entre agents et aller ainsi plus dans le sens d'une collaboration dynamique ([Mermet, 2002]) pour s'affranchir de comportements programmés.

Par extension, tous ces principes d'échanges structurés autour d'un but commun peuvent faire sens dans un monde ultra-connecté où de plus en plus d'objets communiquent (de l'IoT montre bracelet au véhicule connecté). Cela permettrait de créer des essaims de capteurs et de programmes pouvant travailler ensemble pour fournir des résultats globaux dépassant leurs capacités intrinsèques initiales.

Enfin, plus spécifiquement, la figure 11.1 traduit comment pourrait être implémenté le modèle HACCA au sein d'une architecture mécanique, électronique et informatique comme un véhicule connecté, communicant et qui a pour vocation à devenir autonome.

Nous retrouvons ainsi le module de perception (ADAS/Capteurs) combiné au module d'enregistrement des données brutes (Filtrage/Fusion/Agrégation) servant de socle à l'intelligence artificielle embarquée. Celle-ci a pour objectifs de :

- Superviser et monitorer le comportement du véhicule en interprétant les données capteurs,
- Prédire des comportements ou des aléas routiers,
- Décider de la meilleure réponse face à un problème donné ou déduit.

Les trois couches de traitement de HACCA sont ici reprises :

- Couche réactive (ou opérationnel) pour les urgences et les réactions rapides,
- Couche d'intégration (ou tactique) pour les mécanismes de compréhension globale de l'environnement physique,
- Couche de réflexion (ou stratégique) pour les opérations d'optimisation (consommation carburant ou minimisation des déplacements urbains, etc).

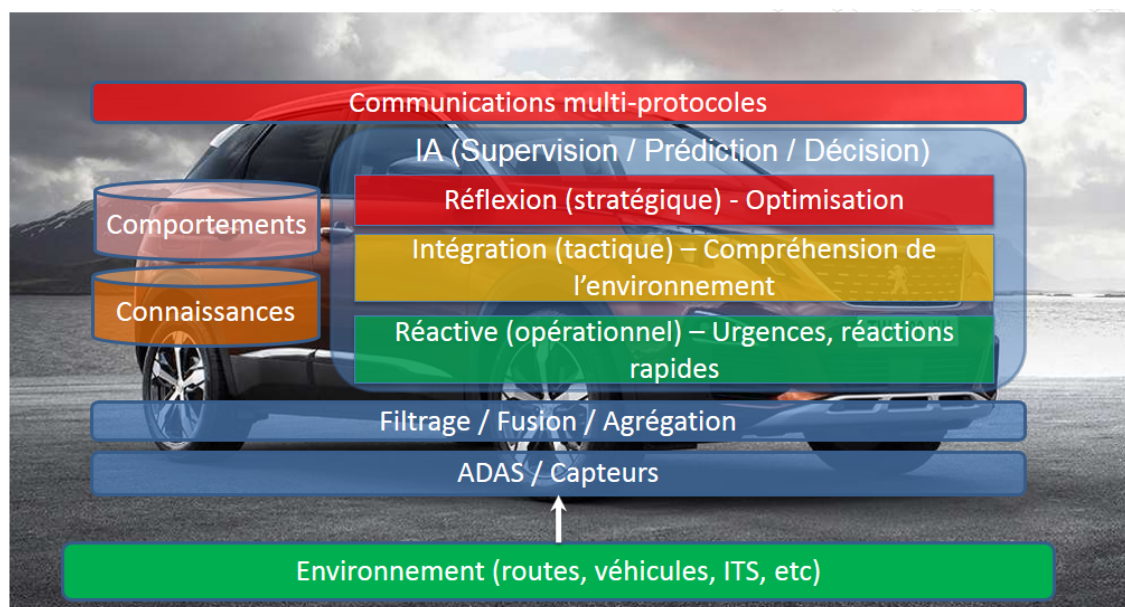


FIGURE 11.1 – Transposition du modèle HACCA dans le contexte d'un véhicule connecté et communicant

Cette intelligence artificielle s'appuie évidemment sur des bases de comportements et de connaissance qui sont régulièrement mises à jour au fil des expériences accumulées, des échanges inter-véhicules ou inter-infrastructures mais aussi des mises à niveau générales effectuées par les constructeurs.

Depuis quelques années, des travaux de normalisation des échanges Car2X sont menés par les constructeurs automobiles. Des travaux récents permettant la communication en direct entre véhicules par l'intermédiaire de technologie 5G ont été présentés à la presse¹.

En conclusion, voici ce que pourra être l'avenir. Le véhicule de demain :

- Sera au coeur d'un écosystème complexe et massivement connecté,
- Sera un objet intelligent et communicant,
- Apprendra de ses propres expériences,
- Tirera profit des expériences collectives,
- Participera à la mutation économique, culturelle et sociétale de nos environnements urbains et péri-urbains (notamment dans notre rapport au véhicule en particulier, aux moyens de transport en général et à notre manière d'envisager la mobilité au sens large).

1. <https://www.01net.com/actualites/decouvrez-comment-les-voitures-communiceront-entre-elles-grace-a-la-5g-1396486.html>

BIBLIOGRAPHIE

- [Adam, 2000] Adam, E. (2000). **Modèle d'organisation multi-agent pour l'aide au travail coopératif dans les processus d'entreprise : application aux systèmes administratifs complexes**. PhD thesis, Université de Valenciennes et du Hainaut-Cambrésis.
- [Adam et al., 2002] Adam, E., Mandiau, R., et Kolski, C. (2002). **Une methode de modelisation et de conception d'organisations multi-agents holoniques**. *Hermes, Paris*.
- [Agre et al., 1987] Agre, P. E., et Chapman, D. (1987). **Pengi : An implementation of a theory of activity**. Dans *AAAI*, volume 87, pages 286–272.
- [Aknine et al., 2004] Aknine, S., Pinson, S., et Shakun, M. (2004). **An Extended Multi-Agent Negotiation Protocol**. *Autonomous Agents and Multi-Agent Systems*, 8(1) :5–45.
- [Aknine et al., 1998] Aknine, S., Pinson, S., et Zacklad, M. (1998). **Un système d'agents récurifs pour l'aide au travail coopératif**. *Actes des 6ièmes Journées Franco-phones d'Intelligence Artificielle Distribuée et Systèmes Multi-Agents*.
- [Amouroux et al., 2007] Amouroux, E., Chu, T.-Q., Boucher, A., et Drogoul, A. (2007). **Gama : an environment for implementing and running spatially explicit multi-agent simulations**. Dans *Pacific Rim International Conference on Multi-Agents*, pages 359–371. Springer.
- [Austin, 1962] Austin, J. L. (1962). **Le langage de la perception**.
- [Báez-Barranco et al., 2006] Báez-Barranco, J.-A., Stratulat, T., et Ferber, J. (2006). **A unified model for physical and social environments**. Dans *International Workshop on Environments for Multi-Agent Systems*, pages 41–50. Springer.
- [Banos et al., 2015] Banos, A., Lang, C., et Marilleau, N. (2015). **Simulation spatiale à base d'agents avec NetLogo 1 : introduction et bases**.
- [Bernon et al., 2002] Bernon, C., Gleizes, M.-P., Peyruqueou, S., et Picard, G. (2002). **Adelfe : a methodology for adaptive multi-agent systems engineering**. Dans *International Workshop on Engineering Societies in the Agents World*, pages 156–169. Springer.
- [Berryman, 2008] Berryman, M. (2008). **Review of software platforms for agent based models**.
- [Blanquet, 2009] Blanquet, M.-F. (2009). **Web collaboratif, web coopératif, web 2.0. : quelles interrogations pour l'enseignant documentaliste**. *Formation des personnes ressources en documentation*.
- [Bonabeau et al., 1999] Bonabeau, E., Dorigo, M., et Theraulaz, G. (1999). **Swarm intelligence : from natural to artificial systems**. Numéro 1. Oxford university press.
- [Bordini et al., 2006] Bordini, R. H., Braubach, L., Dastani, M., Seghrouchni, A. E. F., Gomez-Sanz, J. J., Leite, J., O'Hare, G., Pokahr, A., et Ricci, A. (2006). **A survey**

- of programming languages and platforms for multi-agent systems. *Informatica*, 30(1).
- [Bouron, 1992] Bouron, T. (1992). **Structures de communication et d'organisation pour la coopération dans un univers multi-agents**.
- [Bousquet et al., 1998] Bousquet, F., Bakam, I., Proton, H., et Le Page, C. (1998). **Cor-mas : common-pool resources and multi-agent systems**. Dans *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 826–837. Springer.
- [Bratman, 1987] Bratman, M. (1987). **What is Intention ?** Numéro n 27 ; n 69 dans Report (Center for the Study of Language and Information (U.S.)). Stanford University.
- [Briot et al., 2001] Briot, J.-P., Demazeau, Y., et others (2001). **Principes et architecture des systèmes multi-agents**, volume 217. Hermès Science Publications.
- [Brooks, 1991a] Brooks, R. A. (1991a). **Intelligence without reason**. Dans Myopoulos, John ; Reiter, R., éditeur, *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, pages 569–595, Sydney, Australia. Morgan Kaufmann.
- [Brooks, 1991b] Brooks, R. A. (1991b). **Intelligence without representation**. *Artificial intelligence*, 47(1-3) :139–159.
- [Burnett, 2017] Burnett, D. (2017). **Le cerveau, cet imbécile**. Terra Nova. Terra Nova.
- [Burton et al., 2011] Burton, K., et Ready, R. (2011). **PNL - La Programmation neuro-linguistique Pour les Nuls**. Pour les Nuls Psychologie / Développement Personnel. edi8.
- [Bussmann et al., 1992] Bussmann, S., et Muller, J. (1992). **A negotiation framework for co-operating agents**. *Proceedings of CKBS-SIG*, pages 1–17.
- [Chaib-draa, 1996] Chaib-draa, B. (1996). **Interaction between agents in routine, familiar and unfamiliar situation**. *International journal of cooperative information systems*, 5(01) :1–25.
- [Chaumette et al., 2011] Chaumette, S., Laplace, R., Mazel, C., Mirault, R., Dunand, A., Lecoutre, Y., et Perbet, J. (2011). **Carus, an operational retasking application for a swarm of autonomous uavs : First return on experience**. Dans *MILITARY COMMUNICATIONS CONFERENCE, 2011-MILCOM 2011*, pages 2003–2010. IEEE.
- [Cohen et al., 1990a] Cohen, P. R., et Levesque, H. J. (1990a). **Intention Is Choice with Commitment**.
- [Cohen et al., 1990b] Cohen, P. R., et Levesque, H. J. (1990b). **Intention is Choice with Commitment**. *Artif. Intell.*, 42(2-3) :213–261.
- [Cohen et al., 1995] Cohen, P. R., et Levesque, H. J. (1995). **Communicative actions for artificial agents**. Dans *ICMAS*, volume 95, pages 65–72.
- [Collier, 2003] Collier, N. (2003). **Repast : An extensible framework for agent simulation**. *The University of Chicago's Social Science Research*, 36 :2003.
- [Conte et al., 1992] Conte, R., et Castelfranchi, C. (1992). **Mind is not enough : Pre-cognitive bases of social interaction**. Dans *Proceedings of 1992 Symposium on Simulating Societies*, pages 93–110.
- [Conte et al., 1991] Conte, R., Miceli, M., et Castelfranchi, C. (1991). **Limits and levels of cooperation : Disentangling various types of prosocial interaction**. *Decentralized AI-2*, pages 147–157.

- [Davis et al., 1983] Davis, R., et Smith, R. G. (1983). **Negotiation as a metaphor for distributed problem solving**. *Artificial intelligence*, 20(1) :63–109.
- [Decker et al., 1997] Decker, K., Sycara, K., et Williamson, M. (1997). **Middle-agents for the internet**. Dans *IJCAI (1)*, pages 578–583.
- [Dignum et al., 2000] Dignum, F., Morley, D., Sonenberg, E. A., et Cavedon, L. (2000). **Towards socially sophisticated bdi agents**. Dans *MultiAgent Systems, 2000. Proceedings. Fourth International Conference on*, pages 111–118. IEEE.
- [Drogoul, 1993] Drogoul, A. (1993). **De la Simulation multi-agent à la résolution collective de problèmes**. PhD thesis, Université Paris VI.
- [Drogoul et al., 2013] Drogoul, A., Amouroux, E., Caillou, P., Gaudou, B., Grignard, A., Marilleau, N., Taillandier, P., Vavasseur, M., Vo, D.-A., et Zucker, J.-D. (2013). **Gama : multi-level and complex environment for agent-based models and simulations**. Dans *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 1361–1362. International Foundation for Autonomous Agents and Multiagent Systems.
- [Durfee, 2012] Durfee, E. H. (2012). **Coordination of distributed problem solvers**, volume 55. Springer Science & Business Media.
- [Durfee et al., 1989] Durfee, E. H., Lesser, V. R., et Corkill, D. D. (1989). **Trends in cooperative distributed problem solving**. *IEEE Transactions on knowledge and data Engineering*, 1(1) :63–83.
- [Ellis et al., 1991] Ellis, C. A., Gibbs, S. J., et Rein, G. (1991). **Groupware : some issues and experiences**. *Communications of the ACM*, 34(1) :39–58.
- [Eskandarian, 2012] Eskandarian, A. (2012). **Handbook of Intelligent Vehicles**. Handbook of Intelligent Vehicles. Springer London.
- [Ferber, 1994] Ferber, J. (1994). **Coopération réactive et émergence**. *Intellectica*, 2(19) :19–52.
- [Ferber, 1995] Ferber, J. (1995). **Les systèmes multi-agents : vers une intelligence collective**. Informatique, Intelligence Artificielle. InterÉditions.
- [Ferber et al., 1998] Ferber, J., et Gutknecht, O. (1998). **A meta-model for the analysis and design of organizations in multi-agent systems**. Dans *Multi Agent Systems, 1998. Proceedings. International Conference on*, pages 128–135. IEEE.
- [Ferber et al., 2003] Ferber, J., Gutknecht, O., et Michel, F. (2003). **From agents to organizations : An organizational view of multi-agent systems**. Dans *International Workshop on Agent-Oriented Software Engineering*, pages 214–230. Springer.
- [Ferber et al., 2004] Ferber, J., Michel, F., et Báez-Barranco, J.-A. (2004). **Agre : Integrating environments with organizations**. *E4MAS*, 4 :48–56.
- [Ferber et al., 2009] Ferber, J., Stratulat, T., et Tranier, J. (2009). **Towards an Integral Approach of Organizations in Multi-Agent Systems**. Dans Dignum, V., éditeur, *Handbook of Research on Multi-Agent Systems : Semantics and Dynamics of Organizational Models*, pages 51–75. IGI Global.
- [Ferguson, 1992] Ferguson, I. A. (1992). **Touring Machines : Autonomous Agents with Attitudes**. *Computer*, 25(5) :51–55.
- [Finin et al., 1994] Finin, T., Fritzson, R., McKay, D., et McEntire, R. (1994). **KQML as an agent communication language**. Dans *Proceedings of the third international conference on Information and knowledge management*, pages 456–463. ACM.

- [Flores-Mendez, 1999] Flores-Mendez, R. A. (1999). **Towards a standardization of multi-agent system framework**. *Crossroads*, 5(4) :18–24.
- [Flury-Hérard et al., 2015] Flury-Hérard, B., et De Tréglodé, H. (2015). **Les véhicules communicants nécessitent-ils de nouvelles réglementations, 2015**.
- [Forster et al., 2013] Forster, C., Pizzoli, M., et Scaramuzza, D. (2013). **Air-ground localization and map augmentation using monocular dense reconstruction**. Dans *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 3971–3978. IEEE.
- [Galland et al., 2016] Galland, S., Balbo, F., Picard, G., Boissier, O., Gaud, N., et Rodriguez, S. (2016). **Environnement multidimensionnel pour contextualiser les interactions des agents**. *Revue d'intelligence artificielle–no*, 1 :25.
- [Galland et al., 2014] Galland, S., Gaud, N., Rodriguez, S., Balbo, F., Picard, G., et Boissier, O. (2014). **Contextualiser l'interaction entre agents en combinant dimensions sociale et physique au sein de l'environnement**. Dans *JFSMA*, pages 65–74.
- [Galland et al., 2010] Galland, S., Gaud, N., Rodriguez, S., et Hilaire, V. (2010). **Janus : Another yet general-purpose multiagent platform**. Dans *Seventh AOSE Technical Forum, Paris*.
- [Galliers, 1991] Galliers, J. R. (1991). **Modelling autonomous belief revision in dialogue**. *Decentralized AI*, 2 :231–43.
- [Gaud, 2007] Gaud, N. (2007). **Systèmes multi-agents holoniques : de l'analyse à l'implantation**. Université de Belfort-Montbéliard. Laboratoire Systèmes et Transport.
- [Gechter et al., 2006] Gechter, F., Chevrier, V., et Charpillet, F. (2006). **A reactive agent-based problem-solving model : Application to localization and tracking**. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 1(2) :189–222.
- [Gechter, 2003] Gechter, F. J. A. (2003). **Modèle multi-agent réactif pour la fusion de percepts : application à la localisation et au suivi en robotique mobile**. PhD thesis, Nancy 1.
- [Georgé et al., 2003] Georgé, J.-P., Gleizes, M. P., et Glize, P. (2003). **Conception de systèmes adaptatifs à fonctionnalité émergente : la théorie amas**. *Revue d'intelligence artificielle*, 17(4) :591–626.
- [Gerber et al., 1999] Gerber, C., Siekmann, J., et Vierke, G. (1999). **Holonic multi-agent systems**. DFKI.
- [Gilbert, 2011] Gilbert, M. A. (2011). **The kisceral : Reason and intuition in argumentation**. *Argumentation*, 25(2) :163–170.
- [Gleizes et al., 1999] Gleizes, M.-P., Camps, V., et Glize, P. (1999). **A theory of emergent computation based on cooperative self-organization for adaptive artificial systems**. Dans *Fourth European Congress of Systems Science*, pages 20–24.
- [Goldman et al., 1994] Goldman, C. V., et Rosenschein, J. S. (1994). **Emergent coordination through the use of cooperative state-changing rules**. Dans *AAAI*, pages 408–413.
- [Grignard et al., 2013] Grignard, A., Taillandier, P., Gaudou, B., Vo, D. A., Huynh, N. Q., et Drogoul, A. (2013). **Gama 1.6 : Advancing the art of complex agent-based modeling and simulation**. Dans Boella, G., Elkind, E., Savarimuthu, B. T. R., Dignum, F., et Purvis, M. K., éditeurs, *PRIMA 2013 : Principles and Practice of Multi-Agent Systems*, pages 117–131, Berlin, Heidelberg. Springer Berlin Heidelberg.

- [Grimm et al., 2006] Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., Goss-Custard, J., Grand, T., Heinz, S. K., Huse, G., et others (2006). **A standard protocol for describing individual-based and agent-based models**. *Ecological modelling*, 198(1-2) :115–126.
- [Grimm et al., 2010] Grimm, V., Berger, U., DeAngelis, D. L., Polhill, J. G., Giske, J., et Railsback, S. F. (2010). **The odd protocol : a review and first update**. *Ecological modelling*, 221(23) :2760–2768.
- [Grocholsky et al., 2006] Grocholsky, B., Keller, J., Kumar, R. V., et Pappas, G. J. (2006). **Cooperative air and ground surveillance**.
- [Grudin, 1994] Grudin, J. (1994). **Computer-supported cooperative work : History and focus**. *Computer*, 27(5) :19–26.
- [Guessoum, 1996] Guessoum, Z. (1996). **Un environnement opérationnel de conception et de réalisation de systèmes multi-agents**. PhD thesis, Paris 6, Grenoble. Th. : informatique.
- [Gutknecht et al., 2000] Gutknecht, O., et Ferber, J. (2000). **Madkit : a generic multi-agent platform**. Dans *Proceedings of the fourth international conference on Autonomous agents*, pages 78–79. ACM.
- [Hayes-Roth, 1985] Hayes-Roth, B. (1985). **A blackboard architecture for control**. *Artificial intelligence*, 26(3) :251–321.
- [Hayes-Roth et al., 1979] Hayes-Roth, B., et Hayes-Roth, F. (1979). **A cognitive model of planning**. *Cognitive science*, 3(4) :275–310.
- [Henri et al., 2001] Henri, F., et Lundgren-Cayrol, K. (2001). **Apprentissage collaboratif à distance : pour comprendre et concevoir les environnements d'apprentissage virtuels**. Puq.
- [Hsieh et al., 2007] Hsieh, M. A., Cowley, A., Keller, J. F., Chaimowicz, L., Grocholsky, B., Kumar, V., Taylor, C. J., Endo, Y., Arkin, R. C., Jung, B., et others (2007). **Adaptive teams of autonomous aerial and ground robots for situational awareness**. *Journal of Field Robotics*, 24(11-12) :991–1014.
- [Huang, 2007] Huang, H.-M. (2007). **Autonomy levels for unmanned systems (alfus) framework : safety and application issues**. Dans *Proceedings of the 2007 Workshop on Performance Metrics for Intelligent Systems*, pages 48–53. ACM.
- [Hudda et al., 2013] Hudda, R., Kelly, C., Long, G., Luo, J., Pandit, A., Phillips, D., Sheet, L., et Sidhu, I. (2013). **Self driving cars**. *College of Engineering University of California, Berkeley, Berkeley : College of Engineering University of California*, 2013.
- [Ibañez-Guzman et al., 2012] Ibañez-Guzman, J., Laugier, C., Yoder, J.-D., et Thrun, S. (2012). **Autonomous driving : Context and state-of-the-art**. Dans *Handbook of Intelligent Vehicles*, pages 1271–1310. Springer.
- [Jacobson et al., 1999] Jacobson, I., Booch, G., et Rumbaugh, J. E. (1999). **The unified software development process-the complete guide to the unified process from the original designers**.
- [Jennings, 1996] Jennings, N. R. (1996). **Coordination techniques for distributed artificial intelligence**. *Foundations of distributed artificial intelligence*, pages 187–210.
- [Jennings, 2000] Jennings, N. R. (2000). **On agent-based software engineering**. *Artificial intelligence*, 117(2) :277–296.

- [Jennings et al., 2001] Jennings, N. R., Faratin, P., Lomuscio, A. R., Parsons, S., Wooldridge, M. J., et Sierra, C. (2001). **Automated negotiation : prospects, methods and challenges**. *Group Decision and Negotiation*, 10(2) :199–215.
- [Jennings et al., 1998a] Jennings, N. R., Sycara, K., et Wooldridge, M. (1998a). **A roadmap of agent research and development**. *Autonomous agents and multi-agent systems*, 1(1) :7–38.
- [Jennings et al., 1998b] Jennings, N. R., et Wooldridge, M. J. (1998b). **Agent technology : foundations, applications, and markets**. Springer Science & Business Media.
- [Johansen, 1988] Johansen, R. (1988). **Groupware : computer support for business teams**. Series in communication technology and society. Free Press.
- [Koestler, 1968] Koestler, A. (1968). **The ghost in the machine**.
- [Kraus et al., 1998] Kraus, S., Sycara, K., et Evenchik, A. (1998). **Reaching agreements through argumentation : a logical model and implementation**. *Artificial Intelligence*, 104(1-2) :1–69.
- [Labidi et al., 1993] Labidi, S., et Lejouad, W. (1993). **De l'intelligence artificielle distribuée aux systèmes multi-agents**.
- [Langlois et al., 2013] Langlois, P., Blanpain, B., et Daudé, E. (2013). **Magéo, une plateforme de simulation multi-agents pour tous**. Dans *SimTools 2013*.
- [Lenat, 1975] Lenat, D. B. (1975). **BEINGS : knowledge as interacting experts**. Dans *Advance Papers of the Fourth International Joint Conference on Artificial Intelligence, Tbilisi, Georgia, USSR, September 3-8, 1975*, pages 126–133.
- [Lucien et al., 2016a] Lucien, L., Lang, C., Marilleau, N., et Philippe, L. (2016a). **Multiagent hybrid architecture for collaborative exchanges between communicating vehicles in an urban context**. *Procedia Computer Science*, 83 :695–699.
- [Lucien et al., 2016b] Lucien, L., Lang, C., Marilleau, N., et Philippe, L. (2016b). **A proposition of data organization and exchanges to collaborate in an autonomous agent context**. Dans *Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES), 2016 IEEE Intl Conference on*, pages 561–568. IEEE.
- [Luke et al., 2004] Luke, S., Cioffi-Revilla, C., Panait, L., et Sullivan, K. (2004). **Mason : A new multi-agent simulation toolkit**. Dans *Proceedings of the 2004 swarmfest workshop*, volume 8, pages 316–327. Department of Computer Science and Center for Social Complexity, George Mason University Fairfax, VA.
- [Mandiau et al., 2002] Mandiau, R., Grislin-Le Strugeon, E., et Péninou, A. (2002). **Organisation et applications des sma**.
- [Mandiau et al., 1999] Mandiau, R., Le Strugeon, E., et Agimont, G. (1999). **Study of the influence of organizational structure on the efficiency of a multi-agent system**. *Networking and Information Systems Journal*, 2(2) :153–179.
- [Mansour et al., 2007] Mansour, S., et Ferber, J. (2007). **Agent groupe rôle et service : Un modèle organisationnel pour les systèmes multi-agents ouverts**.
- [Maslow, 1943] Maslow, A. H. (1943). **A theory of human motivation**. *Psychological review*, 50(4) :370.
- [McCaffrey, 2006] McCaffrey, C. (2006). **Starlogo tng : the convergence of graphical programming and text processing**.

- [Mermet, 2002] Mermet, B. (2002). **Collaboration dynamique dans un sma**. LIH, Université du Havre.
- [Michael et al., 2014] Michael, N., Shen, S., Mohta, K., Kumar, V., Nagatani, K., Okada, Y., Kiribayashi, S., Otake, K., Yoshida, K., Ohno, K., et others (2014). **Collaborative mapping of an earthquake damaged building via ground and aerial robots**. Dans *Field and Service Robotics*, pages 33–47. Springer.
- [Michon, 1985] Michon, J. A. (1985). **A critical view of driver behavior models : What do we know, what should we do**. *Human behavior and traffic safety*, pages 485–520.
- [Minsky, 1988] Minsky, M. (1988). **Society of mind**. Simon and Schuster.
- [Müller et al., 1993] Müller, J. P., et Pischel, M. (1993). **The Agent Architecture InteR-RaP : Concept and Application**. Rapport technique.
- [Morin et al., 1990] Morin, E., Morin, E., Morin, E., et Morin, E. (1990). **Introduction à la pensée complexe**, volume 96. Esf Paris.
- [Müller, 2003] Müller, J.-P. (2003). **Emergence of collective behaviour and problem solving**. Dans *International Workshop on Engineering Societies in the Agents World*, pages 1–20. Springer.
- [North et al., 2007] North, M. J., Tatara, E., Collier, N. T., Ozik, J., et others (2007). **Visual agent-based model development with repast simphony**. Rapport technique, Tech. rep., Argonne National Laboratory.
- [Nwana, 1994] Nwana, H. S. (1994). **Negotiation strategies : an overview**. BT Labs.
- [Odell et al., 2002] Odell, J. J., Parunak, H. V. D., Fleischer, M., et Brueckner, S. (2002). **Modeling agents and their environment**. Dans *International Workshop on Agent-Oriented Software Engineering*, pages 16–31. Springer.
- [Panait et al., 2005] Panait, L., et Luke, S. (2005). **Cooperative multi-agent learning : The state of the art**. *Autonomous agents and multi-agent systems*, 11(3) :387–434.
- [Parunak, 1997] Parunak, H. V. D. (1997). **” go to the ant” : Engineering principles from natural multi-agent systems**. *Annals of Operations Research*, 75 :69–101.
- [Pavón et al., 2005] Pavón, J., Gómez-Sanz, J. J., et Fuentes, R. (2005). **The ingenias methodology and tools**. Dans *Agent-oriented methodologies*, pages 236–276. IGI Global.
- [Picard et al., 2004] Picard, G., et Gleizes, M.-P. (2004). **The adelfe methodology**. Dans *Methodologies and Software Engineering for Agent Systems*, pages 157–175. Springer.
- [Poslad et al., 2000] Poslad, S., Buckle, P., et Hadingham, R. (2000). **The FIPA-OS agent platform : Open source for open standards**. Dans *Proceedings of the 5th International Conference and Exhibition on the Practical Application of Intelligent Agents and Multi-Agents*, volume 355, page 368.
- [Pruitt, 2013] Pruitt, D. G. (2013). **Negotiation behavior**. Academic Press.
- [Rao et al., 1995] Rao, A. S., et Georgeff, M. P. (1995). **BDI Agents : From Theory to Practice**. Dans *IN PROCEEDINGS OF THE FIRST INTERNATIONAL CONFERENCE ON MULTI-AGENT SYSTEMS (ICMAS-95)*, pages 312–319.
- [Rao et al., 1992] Rao, A. S., Georgeff, M. P., et Sonenberg, E. A. (1992). **Social plans : A preliminary report**. *Decentralized AI*, 3 :57–76.

- [Ricci et al., 2005] Ricci, A., Viroli, M., et Omicini, A. (2005). **Programming mas with artifacts**. Dans *PROMAS*, volume 3862, pages 206–221. Springer.
- [Rodriguez et al., 2014] Rodriguez, S., Gaud, N., et Galland, S. (2014). **Sarl : a general-purpose agent-oriented programming language**. Dans *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014 IEEE/WIC/ACM International Joint Conferences on*, volume 3, pages 103–110. IEEE.
- [Rosenschein et al., 1994] Rosenschein, J. S., et Zlotkin, G. (1994). **Rules of encounter : designing conventions for automated negotiation among computers**. MIT press.
- [Rousset et al., 2016] Rousset, A., Herrmann, B., Lang, C., Philippe, L., et Bride, H. (2016). **Using nested graphs to distribute parallel and distributed multi-agent systems**. Dans *Parallel, Distributed, and Network-Based Processing (PDP), 2016 24th Euromicro International Conference on*, pages 710–717. IEEE.
- [Russell et al., 1995] Russell, S., Norvig, P., et Intelligence, A. (1995). **A modern approach**. *Artificial Intelligence*. Prentice-Hall, Egnlewood Cliffs, 25 :27.
- [Saunier et al., 2006] Saunier, J., Balbo, F., et Badeig, F. (2006). **Environment as active support of interaction**. Dans *International Workshop on Environments for Multi-Agent Systems*, pages 87–105. Springer.
- [Searle, 1969] Searle, J. R. (1969). **Speech acts : An essay in the philosophy of language**, volume 626. Cambridge university press.
- [Shoham, 1993] Shoham, Y. (1993). **Agent-oriented programming**. *Artificial intelligence*, 60(1) :51–92.
- [Sichman, 1995] Sichman, J. S. (1995). **Du raisonnement social chez les agents : une approche fondée sur la théorie de la dépendance**. PhD thesis, Institut National Polytechnique de Grenoble-INPG.
- [Silberg et al., 2013] Silberg, G., Manassa, M., Everhart, K., Subramanian, D., Corley, M., Fraser, H., et Sinha, V. (2013). **Self-driving cars : Are we ready**. *KPMG. October 2013*.
- [Silberg et al., 2012] Silberg, G., Wallace, R., Matuszak, G., Plessers, J., Brower, C., et Subramanian, D. (2012). **Self-driving cars : The next revolution**. *White paper, KPMG LLP & Center of Automotive Research, 2012*.
- [Smith, 1980] Smith, R. G. (1980). **The contract net protocol : High-level communication and control in a distributed problem solver**. *IEEE Transactions on computers*, (12) :1104–1113.
- [Stentz et al., 2002] Stentz, T., Kelly, A., Rander, P., et others (2002). **Integrated air/ground vehicle system for semi-autonomous off-road navigation**.
- [Sycara, 1989] Sycara, K. (1989). **Multi-agent compromise via negotiation**. *Distributed artificial intelligence*, 2(1) :119–139.
- [Sycara, 1988] Sycara, K. P. (1988). **Resolving goal conflicts via negotiation**. Dans *AAAI*, volume 88, pages 245–250.
- [Sycara, 1998] Sycara, K. P. (1998). **Multiagent systems**. *AI magazine*, 19(2) :79.
- [Taillandier, 2014] Taillandier, P. (2014). **Traffic simulation with the gama platform**. Dans *International Workshop on Agents in Traffic and Transportation*, pages 8–p.
- [Taillandier et al., 2016] Taillandier, P., Bourgaïs, M., Caillou, P., Adam, C., et Gaudou, B. (2016). **A bdi agent architecture for the gama modeling and simulation platform**. Dans *International Workshop on Multi-Agent Systems and Agent-Based Simulation*, pages 3–23. Springer.

- [Taillandier et al., 2012] Taillandier, P., Vo, D.-A., Amouroux, E., et Drogoul, A. (2012). **Gama : a simulation platform that integrates geographical information data, agent-based modeling and multi-scale control.** Dans *Principles and Practice of Multi-Agent Systems*, pages 242–258. Springer Berlin Heidelberg.
- [Tisue et al., 2004] Tisue, S., et Wilensky, U. (2004). **Netlogo : A simple environment for modeling complexity.** Dans *International conference on complex systems*, volume 21, pages 16–21. Boston, MA.
- [Tranier, 2007] Tranier, J. (2007). **Vers une vision intégrale des systèmes multi-agents : Contribution à l'intégration des concepts d'agent, d'environnement, d'organisation et d'institution.** PhD thesis, Université Montpellier II-Sciences et Techniques du Languedoc.
- [Weiss, 1999] Weiss, G. (1999). **Multiagent systems : a modern approach to distributed artificial intelligence.** MIT press.
- [Weyns, 2007] Weyns, D. (2007). **An architecture-centric approach for software engineering with situated multi-agent systems.**
- [Weyns et al., 2004] Weyns, D., Parunak, H. V. D., Michel, F., Holvoet, T., et Ferber, J. (2004). **Environments for multiagent systems state-of-the-art and research challenges.** *E4MAS*, 3374 :1–47.
- [Wilensky, 1999] Wilensky, U. (1999). **Netlogo.**
- [Wooldridge, 2009] Wooldridge, M. (2009). **An introduction to multiagent systems.** John Wiley & Sons.
- [Wooldridge et al., 1994] Wooldridge, M., et Jennings, N. R. (1994). **Towards a theory of cooperative problem solving.** Dans *European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, pages 40–53. Springer.
- [Wooldridge et al., 1995] Wooldridge, M., et Jennings, N. R. (1995). **Intelligent Agents : Theory and Practice.** *Knowledge Engineering Review*, 10 :115–152.

TABLE DES FIGURES

1	Organisation des chapitres (entre parenthèses, le numéro du chapitre) . . .	6
1.1	Représentation AGR Agent Groupe Rôle ([Ferber et al., 1998])	16
1.2	Approche MASQ par quadrants ([Tranier, 2007])	18
1.3	Système holonique ([Gaud, 2007])	19
1.4	Ce qu'il faut retenir : Systèmes multi-agents	24
1.5	Ce qu'il faut retenir : Agents	25
2.1	Pyramide des besoins de Maslow (illustration communément admise par la communauté)	28
2.2	Cycle de la collaboration proposé par l'AIIM (http ://www.aiim.org/What-is-Collaboration)	33
2.3	Ce qu'il faut retenir : Collaboration	35
3.1	TouringMachines (Fergusson)	50
3.2	InteRRaP (Müller)	51
3.3	Ce qu'il faut retenir : Collaboration dans les SMA	52
4.1	Drone militaire General Atomics MQ9 Reaper	54
4.2	Drone civil Delta H	55
4.3	Niveaux d'autonomie, norme SAE (https ://iq.intel.fr/voitures-autonomes-la-voie-de-lavenir/)	58
4.4	Cas d'application pour les véhicules intelligents en fonction du degré d'autonomie et du degré de coopération ([Silberg et al., 2012])	59
4.5	Evolution des architectures automobiles ([Ibañez-Guzman et al., 2012]) . .	60
4.6	Les 4 piliers de l'impact de la voiture autonome (Cazenave, Mobilis 2014) .	62
4.7	Ce qu'il faut retenir : Drones	65
4.8	Ce qu'il faut retenir : Véhicules connectés	66
5.1	Réflexion sur la collaboration à l'aide des quadrants	71
5.2	Architecture proposée par Michon ([Michon, 1985])	74
5.3	Exemple de carte dynamique du projet SafeSpot (http ://www.safespot-eu.org)	75

5.4	Drone SDIS pour la surveillance de la forêt des Landes	77
5.5	Modèle jouet NetLogo : Exemple d'interface principale (zones d'observation)	78
5.6	Modèle jouet NetLogo : Exemple d'interface principale (zones de communication et lien entre drones)	78
5.7	Modèle jouet NetLogo : Cartes internes des dix drones	79
5.8	Modèle jouet NetLogo : Fusion des cartes internes	79
5.9	Ce qu'il faut retenir : Synthèse du processus complet pour un agent	80
5.10	Ce qu'il faut retenir : Interactions entre 2 agents (utilisation des quadrants)	81
5.11	Ce qu'il faut retenir : Vue synthétique de notre analyse et des verrous liés au concept de collaboration	82
6.1	Exemple de situation de collaboration complète	88
6.2	Exemple de situation de collaboration partielle	89
6.3	Exemple de situation de collaboration forcée	90
6.4	Illustration des relations entre quadrants ([Tranier, 2007])	96
6.5	Processus de collaboration	96
6.6	Synoptique du déclencheur	97
6.7	Synoptique de dialogue	98
6.8	Synoptique de mise en œuvre	99
6.9	Synoptique de capitalisation	99
6.10	Ce qu'il faut retenir : Vue synthétique de notre vision de la collaboration	103
6.11	Algorithme général d'une collaboration	104
7.1	Différentes lois dans les quadrants ([Tranier, 2007])	106
7.2	Ce qu'il faut retenir : Méthode d'analyse de modèle pour mettre en évidence des comportements collaboratifs	118
8.1	Modèle conceptuel HACCA (Hybrid Architecture for Collaborative Communicating Agent)	120
8.2	Principes de fonctionnement HACCA	121
8.3	Cartographie des comportements pour un drone	125
8.4	Représentation de la connaissance sous forme d'un multi-graphe	126
8.5	Construction de la carte mentale étape 1 : Environnement	129
8.6	Construction de la carte mentale étape 2 : Agents	130
8.7	Construction de la carte mentale étape 3 : Événements	130
8.8	Construction de la carte mentale étape 4 : Communications	131
8.9	Construction de la carte mentale étape 5 : Groupes	131

8.10	Modification de la carte mentale	132
8.11	Ce qu'il faut retenir : Architecture HACCA (fondamentaux)	133
9.1	Description du processus de collaboration entre véhicules	138
9.2	Exemple d'un réseau de véhicules	141
9.3	Cartographie des comportements pour un véhicule	144
9.4	Exemple de Carte mentale pour l'agent <i>vehicle4</i>	145
9.5	Ce qu'il faut retenir : Instanciation du modèle urbain	146
10.1	Cartographie des fonctions sous GAMA (R pour <i>reflex</i> et A pour <i>action</i>) . .	151
10.2	Exemple HACCA dans GAMA	155
10.3	Simulation avec 700 agents (vitesse moyenne)	156
10.4	Simulation avec 700 agents (nombre de véhicules arrêtés)	156
11.1	Transposition du modèle HACCA dans le contexte d'un véhicule connecté et communicant	160

LISTE DES TABLES

1.1	Agents réactifs contre agents cognitifs [Ferber, 1995]	21
1.2	Classification de Boissier des modèles et plateformes agents selon leur niveau de cognition et de sociabilité (les architectures de type BDI sont en italique) [Briot et al., 2001]	22
2.1	Synthèse et comparaison sélective des apprentissages coopératifs et collaboratifs ([Henri et al., 2001])	32
3.1	Classification des situations d'interactions ([Ferber, 1995])	42
3.2	Structure d'un message FIPA/ACL	44
6.1	Synthèse du gradient de collaboration selon l'état mental et physique . . .	91
6.2	Synthèse du gradient de collaboration et les impacts sur les propriétés . . .	92
6.3	Synthèse des types de collaboration technique en s'inspirant des travaux de Johansen et d'Ellis (entre parenthèses, les termes utilisés originellement par <i>Johansen</i> [Johansen, 1988] et Ellis [Ellis et al., 1991])	95
6.4	Synthèse des types de collaboration technique en s'inspirant des travaux de Grudin (entre parenthèses, les termes utilisés originellement par <i>Grudin</i> [Grudin, 1994]	95
6.5	Synthèse de l'influence du gradient sur les algorithmes (partie 1)	100
6.6	Synthèse de l'influence du gradient sur les algorithmes (partie 2)	102
7.1	Synthèse des comportements à la fin de la première itération	110
7.2	Synthèse des collaborations issues de la deuxième itération d'analyse . . .	112
7.3	Synthèse des comportements modifiés et ajoutés à la fin de la deuxième itération	113
7.4	Synthèse des collaborations issues de la troisième itération d'analyse . . .	115
7.5	Synthèse des comportements modifiés et ajoutés à la fin de la troisième itération	116
8.1	Synthèse des échanges entre agents par couche	129
9.1	Synthèse des comportements à la fin de la première itération	140
9.2	Description partielle des interactions entre les véhicules	142

9.3	Synthèse des comportements modifiés et ajoutés à la fin de la deuxième itération	143
9.4	Synthèse des comportements modifiés et ajoutés à la fin de la troisième itération	144

LISTE DES DÉFINITIONS

1	Définition : Qu'est-ce que la complexité ? par EDGAR MORIN	9
2	Définition : Système multi-agents vu par JACQUES FERBER	12
3	Définition : Système multi-agents vu par EDMUND H. DURFEE	13
4	Définition : Système multi-agents vu par NICHOLAS R. JENNINGS	13
5	Définition : Agent vu par JACQUES FERBER	14
6	Définition : Agent vu par MICHAEL WOOLDRIDGE et par NICOLAS JENNINGS .	14
7	Définition : Agent vu par ROBERTO FLORES-MENDEZ	15
8	Définition : Collaboration et Coopération, en français	28
9	Définition : Collaboration et Coopération, en latin	29
10	Définition : Collaboration et Coopération, en anglais	29
11	Définition : Collaboration et Coopération vu par MARIE-FRANCE BLANQUET .	30
12	Définition : Collaboration et Coopération vu par EDUTECH	31
13	Définition : Drone	54
14	Définition : Collaboration	84

