



HAL
open science

Architecture logique d'un système multi agents de suivi multi caméra distribué : exploitation du modèle de croyance transférable

Béthel Christian A.R.K. Atohoun

► To cite this version:

Béthel Christian A.R.K. Atohoun. Architecture logique d'un système multi agents de suivi multi caméra distribué : exploitation du modèle de croyance transférable. Vision par ordinateur et reconnaissance de formes [cs.CV]. Université du Littoral Côte d'Opale, 2013. Français. NNT : 2013DUNK0373 . tel-01279700

HAL Id: tel-01279700

<https://theses.hal.science/tel-01279700>

Submitted on 26 Feb 2016

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

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

REPUBLIQUE FRANCAISE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE



THESE

Présentée en vue d'obtenir le grade de

DOCTEUR

en

Spécialité : INFORMATIQUE

Par

Béthel Christian A.R.K. ATOHOUN

**Architecture logique d'un système multi agents de suivi multi-
caméra distribué : exploitation du Modèle de Croyance Transférable**

Soutenue le 03 décembre 2013 devant le jury d'examen :

Président	<i>François CABESTAING, Professeur</i>
Rapporteur	<i>El Mustapha MOUADDIB, Professeur</i>
Rapporteur	<i>Véronique BERGE-CHERFAOUI, Maître de Conférences-HDR</i>
Directeur de thèse	<i>Cina MOTAMED, Maître de Conférences-HDR</i>

Thèse préparée dans le Laboratoire d'Informatique Signal et Image de la Côte d'Opale

DEDICACE

À toi mon épouse et à vous nos adorables enfants

Trouvez en ceci l'excuse de tant de sacrifices

REMERCIEMENTS

Cette thèse ne saurait exister si le Laboratoire d'Analyse des Systèmes du Littoral (LASL), devenu Laboratoire d'Informatique Signal et Image de la Côte d'opale (LISIC) en 2010 suite à une fusion avec le Laboratoire d'Informatique du Littoral (LIL), ne m'a pas ouvert ses portes pour profiter du savoir et du savoir-faire de certains de ses chercheurs.

Je tiens donc à remercier toute l'équipe du LISIC et plus particulièrement Cina MOTAMED, mon Directeur de thèse, grâce à qui j'ai pu renouer avec la recherche et qui n'a ménagé aucun effort pour m'encadrer et également me soutenir aussi bien moralement que parfois financièrement dans des moments très difficiles de mon processus doctoral. Merci très sincèrement pour la patience et les conseils.

Je tiens également à témoigner toute ma reconnaissance à Véronique BERGE-CHERFAOUI, Maître de Conférences (HDR) à l'Université de Technologie de Compiègne, et El Mustapha MOUADDIB, Professeur à l'Université de Picardie Jules Verne, d'avoir accepté de rapporter ma thèse malgré leurs multiples occupations. Vos observations et critiques très pertinentes me permettront d'améliorer la qualité de ce travail.

Ce serait ingrat de ma part de ne pas remercier le personnel et le corps enseignant de l'ESGIS (Ecole Supérieure de Gestion, d'Informatique et des Sciences) où je travaille et plus particulièrement son Directeur Général Marcel AKAKPO pour sa compréhension en me permettant de sursoir de temps à autre à mes lourdes responsabilités administratives pour faire des aller-retour Cotonou-Calais-Cotonou dans le cadre de mes recherches doctorales. Qu'il me soit permis de faire un clin d'œil particulier au personnel du département Ingénierie Informatique et Réseaux dont je suis le responsable et à Pascal Hodonou DANNON qui n'ont cessé de me soutenir et de m'encourager à aller au bout de ce travail dans les conditions qu'ils connaissent.

Merci enfin à mes parents, mes amis et tous ceux qui de près ou de loin m'ont apporté leur soutien durant les années consacrées à ce travail.

RESUME

Cette thèse présente l'utilisation conjointe de la théorie de l'évidence et du suivi multi-hypothèses pour la modélisation et la gestion d'un système de suivi multi-caméras dans un environnement autoroutier. Le suivi est basé sur la ré-identification des objets (véhicules) sur la base d'information visio-temporelles. Une concrétisation de ces concepts se traduit par la conception et la mise en œuvre d'une architecture logicielle multi-agents de gestion du suivi multi-caméras. Après une présentation de l'état de l'art sur les cadres de gestion de l'incertain et celui relatif à fusion de l'information pour la mise en correspondance, et sur les systèmes multi-agents, notre apport dans ce travail se situe à trois niveaux.

Le premier a été une adaptation de la phase de décision du modèle de croyance transférable pour y intégrer l'utilisation du suivi multi-hypothèses comme outil de levée d'ambiguïté en cas d'indécision face à une situation de mise en correspondance.

Le second apport a été celui de proposer une architecture logicielle à base d'agents pour la gestion du système du suivi multi-caméras. Nous en avons proposé la modélisation globale ainsi que celle des agents et de leurs interactions en utilisant une démarche personnelle d'analyse mais toutefois inspirée de langages et outils de modélisation tels que Agent UML et MaSE pour ne citer que ceux-là, du fait qu'il n'existe pas réellement un standard normalisé à ce jour dans ce domaine.

Notre troisième apport a été de faire un début d'implémentation de notre architecture logicielle à base d'agent en nous basant sur la plateforme JADE (Java Agent DEvelopment Framework).

Quelques expérimentations et discussions des résultats sont présentées à la fin pour déboucher sur nos conclusions et perspectives.

Mots clés : vidéosurveillance, détection, mise en correspondance, suivi, théorie de l'évidence, agent, système multi-agents.

ABSTRACT

This thesis presents the joint use of the theory of evidence and multiple hypothesis tracking for modeling and managing a system for monitoring multiple cameras in a motorway. The tracking is based on the re-identification of objects (vehicles) on the basis of visual and time informations. A realization of these concepts results in the design and implementation of a software architecture for multiple agents management of multiple camera tracking system. After presenting the state of the art on the frameworks of uncertainty management and that on information fusion for the matching, and the multi-agent systems, our contribution in this work is on two or three levels.

The first was an adaptation of the decision phase of the transferable belief model to incorporate the use of multi-hypotheses tracking as a tool of ambiguity survey in case of indecision in matching situation.

The second contribution was a proposition of agent-based software architecture for management of a multiple cameras tracking system. We have proposed the global system modeling as well as agents and their interactions modeling using a personal analysis method but nevertheless inspired by modelisation languages and tools such as Agent UML, MaSE and others, because there is not yet a standard and normalized tool on the subject.

Our third contribution was to begin an implementation of our agent-based software architecture using JADE (Java Agent Development Framework).

Some experiments and discussions are presented at the end to lead to our conclusions and perspectives.

Keywords: video surveillance, detection, data association, tracking, theory of evidence, agent, multi-agent system.

TABLE DES MATIERES

DEDICACE	i
REMERCIEMENTS	ii
RESUME	iii
ABSTRACT	iv
TABLE DES MATIERES	v
Table des figures.....	ix
Liste des tableaux.....	xi
INTRODUCTION	1
CHAPITRE 1 : Etat de l'Art sur la Détection et le Suivi	4
1.1. Introduction	4
1.2. La détection d'objets en mouvement	6
1.3. Reconnaissance et suivi d'objets.....	10
1.3.1. Objets et Modèles d'Apparence	10
1.3.2. Mise en correspondance : Méthodes d'association de données.....	11
1.3.2.1. Le NN et le GNN.....	11
1.3.2.2. Le PDA et le JPDA.....	12
1.3.2.3. Le MHT.....	13
1.4. Conclusion	14
CHAPITRE 2 : Cadres de Gestion de l'Incertain	15
2.1. Introduction	15
2.2. La théorie des probabilités.....	16
2.3. Les réseaux bayésiens	17
2.4. La théorie des ensembles flous et des possibilités	19
2.5. La théorie de l'évidence	20
2.6. Bref aperçu de la théorie de l'évidence	21
2.6.1. La modélisation	21
2.6.1.1. Fonction de masse.....	22
2.6.1.2. Fonction de crédibilité ou de croyance	23
2.6.1.3. Fonction de plausibilité	23
2.6.1.4. Affaiblissement.....	24
2.6.1.5. Conditionnement.....	25
2.6.2. Estimation des fonctions vraisemblance.....	25
2.6.3. Combinaison.....	26
2.6.3.1. Loi de combinaison de Dempster	27
2.6.3.2. Monde ouvert et monde ouvert étendu	27

2.6.4.	Prise de décision.....	29
2.6.4.1.	Le maximum de plausibilité.....	29
2.6.4.2.	Le maximum de crédibilité.....	29
2.6.4.3.	Le maximum de crédibilité sans recouvrement des intervalles de confiance	29
2.6.4.4.	Le maximum de crédibilité avec rejet	29
2.6.4.5.	Le maximum de probabilité pignistique.....	30
2.7.	Le TBM comme cadre de gestion de l'incertain.....	30
2.7.1.	Niveau crédal.....	31
2.7.2.	Niveau pignistique.....	31
2.7.3.	Le MHT pour la levée d'ambigüité	32
2.8.	Conclusion	33
CHAPITRE 3 : Architecture Multi-Agents.....		34
3.1.	Introduction	34
3.2.	Concept d'Agent et de Système Multi-Agents	35
3.2.1.	Définitions d'Agents.....	35
3.2.2.	Propriétés ou caractéristiques d'un Agent.....	36
3.2.3.	Système Multi-Agents	37
3.2.3.1.	Quelques définitions	39
3.2.3.2.	Domaines d'application des SMA.....	40
3.2.4.	Catégorie ou classes d'agents	41
3.2.4.1.	Les agents réactifs	41
3.2.4.2.	Les agents cognitifs.....	41
3.2.5.	Architecture d'un agent et/ou d'un SMA.....	43
3.2.6.	Interactions et communication	50
3.2.6.1.	Interactions.....	51
3.2.6.2.	Communication	53
3.2.7.	Langages de communication.....	56
3.2.7.1.	Le KQML.....	58
3.2.7.2.	Le FIPA-ACL.....	60
3.2.8.	Ontologies	63
3.3.	Conclusion	66
CHAPITRE 4 : Approches Proposées pour le suivi		67
4.1.	Introduction	67
4.2.	Modélisation des caractéristiques visio-temporelles	68
4.2.1.	Modélisation de la couleur.....	68

4.2.2.	Modélisation de la forme	70
4.2.3.	Modélisation de la vitesse.....	71
4.3.	Application du TBM pour le suivi dans un espace avec caméras distants	72
4.3.1.	Cadres de discernement et référentiels.....	72
4.3.2.	Construction des fonctions de croyance.....	73
4.3.3.	Première combinaison des masses	74
4.3.3.1.	Niveau date.....	74
4.3.3.2.	Niveau Couleurs.....	75
4.3.4.	Deuxième combinaison des masses.....	76
4.3.5.	La Prise de décision	78
4.3.5.1.	Mise en œuvre du MHT dans la prise de décision	78
4.4.	Conclusion	81
CHAPITRE 5 : Architecture multi-agents proposée		82
5.1.	Introduction	82
5.2.	Architecture logique du système de suivi	83
5.2.1.	Structures de données utilisées	85
5.3.	Architecture du Système Multi-Agents	86
5.3.1.	Modélisation de système Multi-Agents	86
5.3.1.1.	Point des exigences du système en termes de fonctionnalités	88
5.3.1.2.	Les étapes de la phase d'analyse.....	89
5.3.1.3.	Description fonctionnelle du système.....	96
5.3.2.	Outils et environnements d'implémentation.....	100
5.3.2.1.	Etude comparatives de quelques outils et environnements.....	100
5.3.2.2.	Choix de JADE	101
5.3.2.3.	Présentation de l'architecture de JADE.....	103
5.3.2.4.	Les agents et leurs comportements dans JADE.....	105
5.3.2.5.	La Communication dans JADE	108
5.3.2.6.	Protocoles d'interaction dans JADE.....	109
5.3.3.	Conception détaillée du système multi-agents.....	114
5.3.3.1.	Raffinement et réorganisation des agents.....	114
5.3.3.2.	Recensement des interactions	116
5.3.3.3.	Les modèles de messages.....	117
5.3.3.4.	Enregistrement et recherche de services ou de fournisseur de services.....	117
5.3.3.5.	Interactions avec des ressources.....	118

5.3.3.6.	Interfaces utilisateurs	119
5.3.3.7.	Présentation des comportements internes des agents	119
5.3.3.8.	Approche de communication	122
5.4.	Conclusion	123
CHAPITRE 6 : Implémentations et Simulations		124
6.1.	Introduction	124
6.2.	Aspects techniques du système	124
6.2.1.	Matériel utilisé	124
6.2.2.	Calibration des caméras	125
6.2.3.	Framework et environnements de développement	128
6.2.4.	Données utilisées	128
6.3.	Expérimentations et évaluations	129
6.3.1.	Evaluation de nos descripteurs	129
6.3.1.1.	Descripteurs visuels	129
6.3.1.2.	Descripteurs temporels	132
6.3.1.3.	Récapitulatif des descripteurs	136
6.3.2.	Résultats de suivi et ré-identification de notre système	138
6.4.	Conclusion	141
Conclusion générale et perspectives		142

TABLE DES FIGURES

Figure 1.1 :	étapes classiques d'un processus de suivi	6
Figure 1.2 :	Classification des méthodes de détection de mouvement.....	7
Figure 2.1 :	Représentation des niveaux du TBM	31
Figure 3.1 :	Différents types d'agents d'après [124].....	42
Figure 3.2 :	Différentes fonctions d'une organisation.	46
Figure 3.3 :	Caractéristiques fonctionnelles, structurelles et de concrétisation	49
Figure 3.4 :	Caractéristiques de conception de l'architecture d'un agent ou d'un SMA.....	50
Figure 3.5 :	Système des activités de coopération.....	55
Figure 3.6 :	Système de communication d'après Claude Shannon.....	56
Figure 3.7 :	Décomposition d'un système de communication d'après [98]	57
Figure 4.1 :	Deux différentes images (a) et (b) ayant même histogramme	68
Figure 4.2 :	Deux images avec leurs projections verticales et horizontales	70
Figure 5.1:	Organisation logique de notre architecture de suivi	83
Figure 5.2:	Axes routiers représentant des zones de suivi multi-caméras	84
Figure 5.3:	Réseau de capteurs associé au schéma ci-dessus, sous forme de graphe orienté.....	84
Figure 5.4:	Etapas méthodologiques de conception et de réalisation	88
Figure 5.5:	Diagramme des buts du système	90
Figure 5.6:	Diagramme des rôles des agents du système	92
Figure 5.7:	Diagramme synthétisé de cas d'utilisation du système.....	93
Figure 5.8:	Modèle organisationnel des agents.....	94
Figure 5.9:	Diagramme d'agents	95
Figure 5.10:	Plan de détection	96
Figure 5.11:	Plan d'extraction et de gestion suivi local.....	97
Figure 5.12:	Plan de prototypage.....	97
Figure 5.13:	Plan de modélisation croyance	98
Figure 5.14:	Plan d'analyse probabilité/Association	98
Figure 5.15:	Diagramme de plan de déploiement MHT.....	99
Figure 5.16:	Architecture d'une plateforme JADE	103
Figure 5.17:	Plateforme multi-agents aux normes FIPA	104
Figure 5.18:	Modélisation UML de la classification hiérarchique des comportements	105
Figure 5.19:	Cycle de vie d'un agent et de ses comportements dans JADE.....	107
Figure 5.20:	Cycle de vie d'un agent tel que défini par FIPA.....	107
Figure 5.21:	Mécanisme de communication par envoi de messages	108
Figure 5.22:	Diagramme UML du protocole FIPA-Propose	111
Figure 5.23:	Diagramme UML du protocole FIPA-Query	112
Figure 5.24:	Diagramme UML du protocole FIPA-Request	112
Figure 5.25:	Diagramme UML du protocole FIPA-Subscribe.....	113
Figure 5.26:	Diagramme UML du protocole FIPA-ContractNet.....	113
Figure 5.27:	Enregistrement et recherche de service de traitement vidéo	118
Figure 5.28:	Enregistrement et recherche de service de fusion d'information et de re-identification.	118
Figure 6.1:	Passage de l'univers 3D au plan image 2D.....	125
Figure 6.2:	Effet de la profondeur sur la taille d'un objet.....	126

Figure 6.3:	Lignes routières de même taille mais à différentes profondeurs.....	127
Figure 6.4:	Images créées à partir de distances entre histogrammes d'images d'une base de données.	130
Figure 6.5:	Images obtenues après seuillage des images de la figure précédente.....	131
Figure 6.6:	Exemples de rectangles minimums englobant	131
Figure 6.7:	Comparaison de deux rectangles	132
Figure 6.8:	Parcours d'un tronçon (segment AB) par un véhicule : estimation du temps de parcours à partir de la vitesse, de la densité de véhicule sur le tronçon et de la longueur du tronçon.....	133
Figure 6.9:	Evolution du temps de parcours en fonction de la densité du trafic.....	137
Figure 6.10:	Evolution du temps de parcours en fonction de la densité du trafic et de la vitesse V_0 d'entrée	138
Figure 6.11:	Images de véhicules filmés à partir de caméras et de scènes différentes.....	140

LISTE DES TABLEAUX

Tableau 3.1 : Les caractéristiques d'un agent.....	37
Tableau 3.2 : Catégories d'agents suivant fonctionnement et représentation du monde	42
Tableau 3.3 : Présentation des rôles des composants ou agent d'une organisation	44
Tableau 3.4 : Différentes fonctions principales d'une organisation.....	45
Tableau 3.5 : Grille d'analyse fonctionnelle des organisations	47
Tableau 3.6 : Quelques performatifs de KQML	59
Tableau 3.7 : Quelques attributs utilisés dans les performatifs de KQML	59
Tableau 3.8 : Performatifs (Communication Acts) de FIPA-ACL	62
Tableau 3.9 : Liste des attributs utilisés dans les performatifs de FIPA-ACL	62
Tableau 3.10 : Paramètres des messages FIPA-ACL.....	62
Tableau 5.1 : Tableau comparatif d'outils de développement de SMA	100
Tableau 5.2 : Récapitulatif des agents et de leurs rôles d'après notre analyse initiale.....	114
Tableau 5.3 : Nouvelle liste des agents suite à la phase de conception.....	116
Tableau 5.4 : Les agents de notre système et leurs interactions	116
Tableau 5.5 : Les agents de notre système et leurs interactions	117
Tableau 5.6 : Tableau descriptif des rôles de nos agents et des comportements JADE associés	121
Tableau 6.1. Mesures de lignes dans le repère du monde	127
Tableau 6.2. Mesures de lignes dans le repère image.....	127
Tableau 6.3. Résultats d'associations de nos descripteurs visuels.....	137
Tableau 6.4. Résultats expérimentaux de ré-identification d'objets.....	140

INTRODUCTION

Contexte et Problématique

Les progrès technologiques ont entraîné la facilité d'augmenter la puissance de traitement des processeurs et rendu possible un certain nombre de traitement numérique de l'information aussi bien en différé qu'en temps réel. Un des domaines d'investigation ayant tiré beaucoup partie de ces avancées technologiques est celui de la vision par ordinateur où de nombreux systèmes de vision fixe ou mobile ont vu le jour dans plusieurs domaines pour la surveillance, le contrôle et/ou la gestion de grands espaces urbains dans le cadre de la surveillance de personnes et d'objets ou pour la biométrie ou l'imagerie médicale.

Pour des raisons de précision, de performance ou de couverture, on a de plus en plus de systèmes de vision ou de suivi multi-capteurs composés d'une variété de capteurs qui se complètent dans le cadre d'une mission bien déterminée. Ces capteurs peuvent assurer la couverture d'une même scène avec plusieurs angles de vue (points de vue) pour faire de la stéréovision ; Ils peuvent se chevaucher dans le but d'assurer la couverture totale d'une scène étendue dépassant le champ de vision d'un seul capteur ; Mais il est également possible d'avoir une configuration dans laquelle les capteurs sont répartis de façon à ne pas avoir des champs de vue adjacents. Cette dernière forme de répartition de capteurs avec des zones aveugles non couvertes, tout en étant économique par rapport aux autres configurations lorsqu'on veut couvrir de vastes zones, ajoute cependant une certaine complexité au problème de suivi de cible qui en lui-même n'est déjà pas d'une aisance en ce qui concerne l'association de données.

Plusieurs facteurs peuvent biaiser la précisions des mesures et donc affecter la mise en correspondance ou l'association des données qui est une phase critique dans un système de suivi, et particulièrement pour des traitements en temps réel pour une prise de décision.

Les méthodes classiques d'associations sont multiples et sont présentées en détail dans [25, 28, 29] et reposent essentiellement sur l'un ou la combinaison des principes suivants : le NN (Nearest Neighbor), le GNN qui est une approche globale du NN, le PDA (Probabilistic Data association), le JPDA qui est une approche plus robuste et de résolution de conflits du PDA, et le MHT (Multi Hypotheseis Tracking).

Ces différentes méthodes diffèrent par leur complexité, leur capacité à prendre en compte la création et la destruction de piste, ainsi que celle de leur aptitude à gérer les ambiguïtés dans certaines situations d'association.

La représentation et le traitement de l'incertitude dans les systèmes de suivi se basent sur des théories numériques ou non. Parmi les approches numériques on peut citer : la théorie des probabilités, la théorie des possibilités, la théorie de l'évidence, les réseaux bayésiens, la logique floue.

Notre travail s'inscrit dans un contexte de contribution à la sécurité routière et de personnes à travers la mise en place d'un système logiciel de gestion multi-capteurs pouvant comporter des capteurs co-localisés pour la précision d'un suivi local, mais surtout des capteurs distants avec des zones de couverture ne se recouvrant pas et coopérant dans un but commun de suivi global. Une telle architecture peut contribuer à la surveillance routière dans un contexte ouest africain et plus particulièrement au Bénin où il arrive fréquemment d'enregistrer dans le trafic routier des véhicules ayant des plaques d'immatriculation avec des caractères non lisibles ou camouflés, voire même inexistantes. Dans ce même contexte, la rareté des équipements appropriés de mesure et de contrôle automatique du respect de signalisations routières (limitations de vitesse, panneaux stop, feux tricolores, etc.) ainsi que l'inexistence d'une base de données centralisée faisant le lien entre véhicules et propriétaires ; augmentent significativement l'insécurité routière.

Notre objectif est de mettre en place une stratégie de gestion haut-niveau de capteurs dans un cadre de vidéosurveillance intégrant l'une et l'autre des approches ci-dessus citées pour la détection et le suivi de véhicules mais également et surtout d'arriver à implémenter un système logiciel multi-agents pour la gestion automatique des caméras et du suivi multi-caméras en se basant sur les caractéristiques visio-temporelles des véhicules.

Au-delà de modéliser et de faire fonctionner une telle architecture, notre objectif à travers cette thèse est d'apporter notre contribution à l'utilisation de la théorie de l'évidence et plus particulièrement de la théorie des croyances transférable comme cadre de gestion haut niveau d'un système de suivi multi-caméras pour la vidéosurveillance.

Structure de la thèse

Cette thèse est organisée en six (6) chapitres. Le chapitre 1 fait une présentation succincte de la détection et du suivi d'objets. Nous y abordons très rapidement quelques notions sur la détection et nous présentons quelques facteurs permettant de caractériser au mieux et visuellement un objet afin de servir à sa reconnaissance ultérieure dans un mécanisme de mise en correspondance ou d'association de données à des fins de suivi. Un aperçu des méthodes d'associations de données est fait pour aboutir au choix du MHT (Multiple Hypothesis Tracking) sur la base de ces performances malgré son risque d'explosion combinatoire qui est d'ailleurs le plus grand reproche qu'on formule à son encontre.

Dans le chapitre 2, par rapport au traitement numérique de l'information, nous présentons une liste de méthodes de représentation, de manipulation et fusion d'informations en vue de leur exploitation pour une prise de décision dans le cadre du suivi de véhicule en vidéosurveillance. Entre les formalismes symboliques reposant sur les systèmes de bases de connaissances ou la logique propositionnelle, et le formalisme numérique dont les outils les plus connus sont la théorie des probabilités, la théorie des possibilités, la logique floue, la théorie de l'évidence, nous montrons l'intérêt et l'apport de la théorie de l'évidence et plus particulièrement de la théorie des croyances transférables pour la représentation et la manipulation d'informations imparfaites (incertaines, imprécises, incomplètes) que

constituent celles que nous avons à manipuler dans le cadre de nos travaux. Il est présenté à la fin de ce chapitre, notre approche de prise de décision et notre adaptation du MHT dans la prise de décision en situations ambiguës.

Le chapitre 3, fait l'état de l'art des systèmes multi-agents qui représentent notre choix pour la mise en place de l'environnement de gestion du système multi-caméras pour la vidéosurveillance. Nous y abordons les différents types d'agents et de SMA (Systèmes Multi Agents) ainsi que les architectures de SMA et les contraintes liées à leur mise en place. Un accent est mis sur la gestion de la communication entre agents, les ontologies et les langages qui ont été développés à cet effet. L'objectif poursuivi dans ce chapitre est celui de mettre à disposition un ensemble d'informations pouvant aider le lecteur à comprendre dans le chapitre 5, nos choix d'architecture de SMA et de langage de communication pour la mise en place des interactions et la communication des agents dans notre architecture.

Le chapitre 4 est consacré à la présentation de nos différents choix par rapport au modèle de représentation ou de caractérisation de nos objets d'intérêt; par rapport à la méthode d'association de données et de fusion d'informations puis celle de prise de décision. Le MCT (Modèle de Croyances Transférable) ou TBM (Transferable Belief Model) en anglais, sera présenté. Pour chacun de ces aspects, nous exposons quelques détails de modélisation et de justificatifs au besoin de nos choix. Nous y évoquons également la particularité du fonctionnement de notre MHT pour nous rendre optimiste sur son efficacité dans la gestion de l'ambiguïté.

Dans le chapitre 5, nous présentons les détails de notre solution de SMA, son architecture logicielle, son mode de fonctionnement ainsi que la justification du choix de JADE comme environnement de son développement.

Le chapitre 6 présente nos expérimentations et les résultats obtenus à partir des données vidéo réelles provenant de captures de caméras, réalisées sur l'autoroute Cotonou Porto-Novo d'où proviennent la majorité des données de ce travail. Pour évaluer les performances de nos modèles et systèmes, nous allons comparer nos résultats à d'autres résultats issus d'autres approches. Une discussion est faite de ces résultats pour en tirer les leçons et présenter les perspectives de recherche vers lesquelles nous conduisent nos travaux.

Nous terminons cette rédaction par une conclusion pour faire le point de nos travaux.

CHAPITRE 1 : ETAT DE L'ART SUR LA DETECTION ET LE SUIVI

Sommaire

1.1.	Introduction	4
1.2.	La détection d'objets en mouvement	6
1.3.	Reconnaissance et suivi d'objets.....	10
1.3.1.	Objets et Modèles d'Apparence	10
1.3.2.	Mise en correspondance : Méthodes d'association de données.....	11
1.4.	Conclusion	14

1.1. INTRODUCTION

Nous vivons dans un monde où de plus en plus l'insécurité prend de l'ampleur. Pour répondre à ces problèmes d'insécurité, plusieurs solutions ont vu le jour depuis des décennies et plus encore aujourd'hui. Au nombre de ces solutions nous pouvons citer la vidéosurveillance qui bien qu'étant dans un premier temps une mesure dissuasive et préventive, devient progressivement un moyen répressif en permettant de confondre des auteurs d'actes répréhensifs. La vidéosurveillance connaît une forte croissance ces dernières années à cause des progrès technologiques qui ont permis d'augmenter ou de démultiplier la puissance de traitement des processeurs ainsi que l'amélioration de plusieurs algorithmes de traitement numérique de l'information dans le domaine du traitement d'image et en vision par ordinateur aussi bien en temps réel qu'en différé. Ces diverses évolutions ont entraîné une rapidité dans les traitements et ont favorisé la naissance de plusieurs domaines d'application de la vidéosurveillance. Ces domaines d'application vont de la surveillance simple de magasin ou de grands espaces urbains, à la surveillance d'installations militaires en passant par la surveillance médicale de personnes et la surveillance de trafics routiers pour ne citer que ceux-là.

Pour couvrir des espaces étendus, il n'est pas rare de trouver des dispositifs composés de plusieurs caméras réparties sur un périmètre donné. On parle de système multi-caméras de vidéosurveillance. Les champs de vision des caméras du système peuvent se chevaucher pour garantir une couverture totale de l'espace et assurer la précision des données du fait de la redondance des informations dans les zones de chevauchement. Elles peuvent également être totalement distantes sans aucune zone de chevauchement et donner lieu à des zones totalement aveugles. Cette seconde approche permet de couvrir de plus grandes étendues et de faire des économies en nombre de caméras utilisées mais ajoute de nouvelles complexités

du fait des zones aveugles dans lesquelles on ne peut prédire de façon exacte le comportement des objets suivis.

Un des défis majeurs dans un tel système est la reconnaissance (ré-identification) ou la mise en correspondance et le suivi d'objets déjà perçus par une des caméras du système. En effet, plusieurs facteurs peuvent influencer la qualité des données issues des acquisitions des différentes caméras et ainsi affecter l'association de données qui est une phase critique dans tout système de suivi. Au nombre de ces facteurs, on peut citer : les facteurs d'échelle, l'angle de vision ou d'acquisition, le facteur d'éclairage et de luminosité, les incertitudes et imprécisions des mesures, la différence entre les rendus des différentes caméras du système.

Dans cette partie de ce document, nous allons présenter brièvement les différents éléments qui entrent en ligne de compte dans la détection, l'identification et le suivi d'un objet ; puis présenter quelques conditions à satisfaire pour garantir une bonne exploitation de données provenant de plusieurs caméras (capteurs). Il est à noter que nous travaillons avec des caméras fixes qui observent des champs précis ayant des arrières plans bien déterminés et quasiment pour ne pas dire totalement statiques.

1.2. LA DETECTION D'OBJETS EN MOUVEMENT

Le cycle de suivi d'un objet dans un flux vidéo peut se présenter comme nous pouvons l'observer sur la figure ci-dessous. Dans cette figure, la première et la plus importante phase qui vient après la phase d'acquisition est celle de détection. De sa qualité dépend la fiabilité de tous les traitements qui vont suivre.

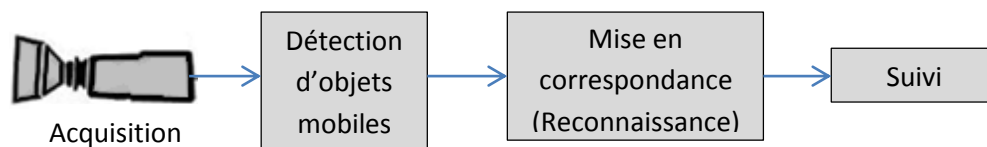


Figure 1.1 : étapes classiques d'un processus de suivi

La réussite de la détection de mouvement, et donc d'objets mobiles dépend de plusieurs facteurs dont la qualité de l'acquisition (conditions d'éclairage lors de l'acquisition, précision des capteurs), la qualité de l'arrière-plan (uniforme, bruité, clair, ou non), la distance entre les caméras et la scène observée, ainsi que de la vitesse et de la taille des objets à détecter. Plusieurs travaux ont été faits sur l'influence des conditions d'acquisition sur la qualité de la détection et un point de la littérature est présenté dans [1] et [2].

Dans une image ou une vidéo, il existe deux catégories d'objets. Les objets qui font partie du décor ou de l'environnement, et les objets d'intérêt. La première catégorie d'objets, composée d'objets qu'on peut qualifier de passifs, doit être confondue à l'arrière-plan. Par contre, la seconde catégorie d'objets est celle des objets d'intérêt ou réellement actifs qu'on doit prendre en compte dans un processus de détection. La plupart des méthodes de détection de mouvement ou d'objets s'articulent autour de la prise en compte ou non de la gestion de l'arrière-plan dans les algorithmes de détection. Il en découle une répartition des méthodes en trois grandes catégories. Une première catégorie qui s'appuie sur la modélisation de l'arrière-plan ; une seconde catégorie qui n'utilise pas la modélisation de l'arrière-plan ; puis une dernière catégorie qui utilise une approche mixte. Un bon état de l'art sur la problématique de détection est présenté par [3] et [4] dans leurs thèses respectives. Chacune de ces catégories de détection a ses avantages et ses inconvénients et est mieux adaptée à certaines situations.

Dans le cas de capteur (caméra) fixe, les méthodes qui modélisent l'arrière-plan utilisent une image de référence à laquelle sont comparées les images issues de l'acquisition. L'image de référence qui peut devenir dans certaines conditions obsolète avec le temps, peut, dans certaines approches, être mise à jour de temps en temps pour prendre en compte les éventuelles modifications que peut subir l'arrière-plan et éviter ainsi certaines fausses détections.

Il existe très peu de méthodes qui ne s'appuient pas sur la modélisation de l'arrière-plan de la scène observée lors du processus de détection d'objets en mouvement. Ces méthodes se

basent dans plusieurs cas sur le calcul de maximums d'entropie spatio-temporelle pour identifier les zones en mouvement. Des travaux illustrant ces méthodes sont disponibles dans [5] et [6]. Une autre approche basée sur l'analyse en composantes principales est présentée dans [7].

Dans sa thèse [3], N. Verbeke dresse une liste exhaustive des méthodes de détection de mouvement et en fait une classification que nous présentons sur la figure 1.2 ci-dessous.

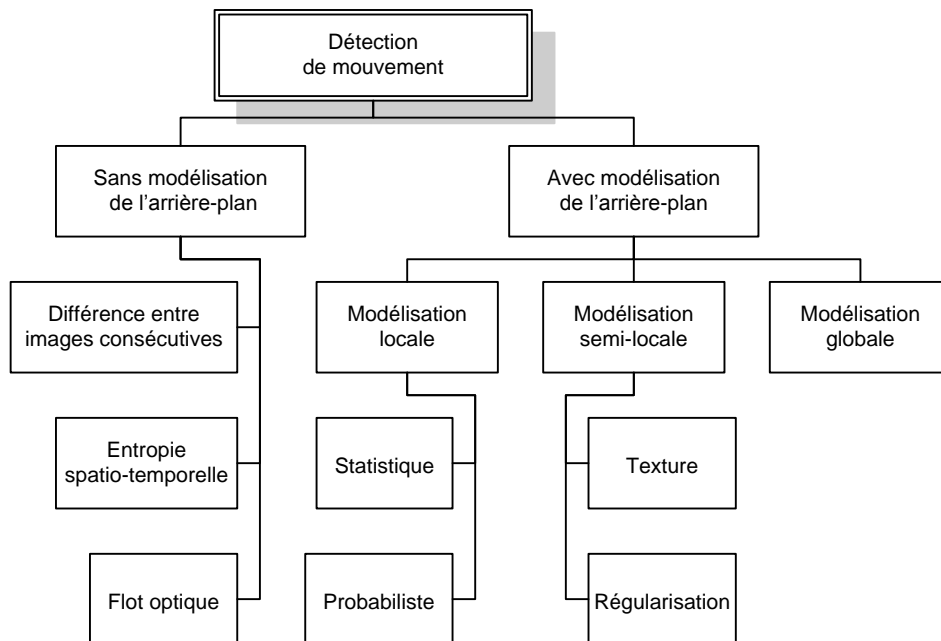


Figure 1.2 : Classification des méthodes de détection de mouvement

Nous donnons ici une brève présentation de certaines de ces méthodes mais pour plus de détail, [3] reste une bonne référence.

Ainsi, au nombre des méthodes sans arrière-plan on a comme présenté sur la figure 1.2 ci-dessus :

- la méthode du calcul de dérivées temporelles pour évaluer le changement d'apparence des pixels entre deux trames consécutives. Méthode peu robuste face à des mouvements lents ou saccadés, des arrêts brefs d'un objet en mouvement ou à la présence de trames redondantes dans certaines séquences vidéo. Une amélioration se fait par lissage temporel de la séquence. Mais ce lissage n'est pas vraiment efficace et ses résultats sont fonction de la valeur du facteur de pondération des mesures passées par rapport à la mesure courante.
- la méthode du calcul, sur une certaine durée, de l'entropie spatio-temporelle en chaque point pour mesurer la variabilité de l'intensité lumineuse (ou de la couleur, ou du gradient). Dans [5], les auteurs proposent une méthode de calcul de l'entropie et sa quantification en 256 niveaux pour en faire une représentation sous la forme d'une image appelée STEI (Spatial Temporal Entropy Image) sur laquelle sont mises en évidence les points sujets à de

fortes variations d'intensité lumineuse dans le temps mais également ceux sujets de forts gradients (contours) qui sont des faux positifs. Une variante de cette approche a été proposée par [6] pour corriger ces faux positifs. Dans cette approche, en lieu et place des séquences successives, on utilise en entrée plutôt la différence entre trames successives.

- La méthode du calcul du flot optique pour obtenir et expliquer les variations dans une séquence vidéo (suite d'images consécutives) en terme de déplacement de points images ; sous l'hypothèse que la couleur d'un point image devrait normalement être indépendante du temps. Cette méthode est assez gourmande en temps de calcul à moins de sous échantillonner les trames et d'utiliser un algorithme très rapide comme celui présenté dans [10].

Au nombre des méthodes modélisant l'arrière-plan, nous avons :

- Celles qui se basent sur l'utilisation d'une image statique de référence. Bien que cette méthode présente de multiples avantages tels que la détection aussi bien de mouvements lents que rapides ou même d'objets momentanément non mobiles, elle présente quelques limitations dans sa mise en œuvre pratique surtout en milieu extérieur. En effet, sans compter qu'il n'est pas facile d'avoir l'image de référence dépourvu de tout objet, il faut noter que plusieurs facteurs peuvent rendre cette image rapidement inexploitable. Au nombre de ceux-ci on peut citer les conditions d'éclairage, des bruits causés par des aléas climatiques, des objets qui s'immobilisent dans la scène sur une assez longue durée. Pour cette raison, il existe des variantes qui proposent une mise à jour régulière de l'arrière-plan (background maintenance) comme c'est le cas dans [11] et [19].
- Celles privilégiant une approche de modélisation statistique qui se base sur un ensemble de fonctions de densité de probabilité par pixel de l'image. Ces fonctions sont construites par apprentissage et les mesures dont les probabilités d'observation sont élevées correspondent aux pixels appartenant à l'arrière-plan. Certains auteurs proposent d'utiliser une distribution gaussienne ou un mélange de distributions gaussiennes pour modéliser les intensités des points de l'arrière-plan. Un calcul récursif de moyenne et d'écart-type permet à tout moment de décider si une nouvelle observation fait partie de l'arrière-plan ou non selon qu'il est proche ou non de la moyenne courante sur la base de la valeur de l'écart-type. Cette approche reste efficace en milieu intérieur mais montre ses limites en milieu extérieur. Une autre approche, assez semblable mais basée sur une modélisation prédictive permet de résoudre les problèmes liés aux arrière-plans non statiques et à certains bruits cycliques. Cette approche utilise un filtre de Kalman ou de Wiener pour prédire la valeur de la prochaine observation en chaque point et utilise l'écart entre la valeur prédite et celle observée pour estimer l'amplitude du mouvement.
- Celles préférant la prise en compte d'un ensemble de pixels d'un voisinage (une région) en lieu et place d'un seul pixel (point) pour amoindrir les risques d'erreurs liés à des imprécisions des outils de capture et à la nature progressive des mouvements observés dans les flux vidéo analysé. A cet effet, certains auteurs [12] proposent de construire une matrice carré 3x3 non recouvrant de chaque point et mettre en place un modèle de Markov

caché à trois états pour décider si la zone associée à cette matrice fait partie ou non de l'arrière-plan ou si elle est plutôt un élément d'un objet mobile ou d'une ombre. Différentes distributions de probabilités (uniformes, gaussiennes) sont utilisées suivant l'état du système parmi les trois états possibles. Pour décider si une région fait partie de l'arrière-plan, d'un objet mobile ou d'une ombre, on s'en tient à la probabilité la plus élevée.

Une variante de cette approche est celle définissant pour tout point, un vecteur représentant la texture de son voisinage. Un espace de représentation des données de l'image est construit à partir de l'ensemble de ces vecteurs.

- Celles construisant k modèles de l'arrière-plan et les utilisant tous pour des détections locales pour finalement retenir celui détectant le moins de pixels mobiles comme l'arrière-plan utilisé pour la détection globale. Les k modèles de l'arrière-plan sont construits par apprentissage. Cette approche peut être utile dans des environnements à arrière-plan dynamique.
- Celles modélisant les pixels comme des dimensions d'un espace de représentation, et les images qui se suivent comme des individus dans cet espace [3].

En se référant aux travaux très récents de [9] et les conclusions auxquelles il est parvenu après comparaisons de plusieurs méthodes de modélisation d'arrière-plan suite à des implémentations, il ressort que deux méthodes se démarquent du lot par la qualité de leurs résultats. Il s'agit de la méthode de la mixture (mélange) de gaussiennes présentée dans [13] et celle utilisant un dictionnaire codebook [14]. Dans sa thèse [9], l'auteur nous propose une modification de ces méthodes, particulièrement celle du codebook pour pallier les quelques insuffisances qu'on y relève dans certaines situations (courte durée d'apprentissage, nombre très élevé d'objets en mouvement, couleur des objets très proche de celle de l'arrière-plan). Il propose la prise en compte de la fréquence d'apparition des objets, pour modifier la politique d'ajout ou de suppression d'un codeword dans le codebook. Les résultats obtenus après segmentation ont été très concluant comparativement à la solution native et à plusieurs autres solutions. Rappelons que ces solutions sont adaptées aux situations de prise de données depuis une caméra fixe.

Conclusion

Ce bref aperçu de différentes approches de détection de mouvement nous a permis de nous faire une idée sur comment cette question est abordée dans la littérature et les approches de solutions proposées ces dernières années suivant les types d'application et la configuration des caméras (fixes ou mobiles).

Notre environnement de travail étant composé de caméras fixes, les solutions qui ont attiré le plus notre attention restent celles modélisant l'arrière-plan et particulièrement celles utilisant un mélange de gaussienne et celle s'appuyant sur le codebook. Prenant en compte la pertinence des résultats obtenus par [9], nous avons décidé d'opter dans notre phase de détection pour l'utilisation d'une méthode inspirée de sa solution de modification de la

méthode de codebook. Nous avons essayé d'ailleurs par nous-même d'implémenter cette solution pour nous convaincre de son efficacité comparativement à d'autres méthodes.

Toutefois, une fois les objets en mouvement détectés, l'étape suivante est celle devant permettre la détection de véhicules parmi les objets en mouvement. Dans cette phase, nous avons décidé de ne pas chercher à comparer les différentes méthodes qui existent mais de juste opté pour l'utilisation de la méthode de Viola et Jones du fait de son efficacité malgré les quelques critiques formulées à son endroit par rapport au choix des images et leur quantité pour la phase d'apprentissage, à la durée de la phase d'apprentissage, et les limites de l'apprentissage.

1.3. RECONNAISSANCE ET SUIVI D'OBJETS

La reconnaissance est une phase aussi complexe qu'intéressante dans un processus de suivi. C'est pour nous une phase cruciale pour la réussite du suivi. En effet, pour suivre une cible dans un flux vidéo, il faut arriver à la localiser et la ré-identifier. Nous entendons donc ici par reconnaissance d'objet cette nouvelle identification de l'objet qui est donc supposé connu par le passé et dont on a une description ou représentation. La reconnaissance d'objet est une tâche très délicate et sa complexité est fonction de la complexité de la représentation des facteurs valorisant la singularité (facteurs discriminants) de cet objet par rapport à d'autres objets de la même famille ou de la même classe afin d'éviter ou de limiter au maximum les risques de confusion. Pour une caractérisation robuste d'un objet, il n'est souvent pas rare d'avoir recours à plusieurs catégories voire plusieurs sources d'information avec des niveaux de fiabilité différents. Ces informations sont susceptibles d'être incomplètes, imprécises, incertaines, et parfois même pas totalement cohérentes. Leur traitement nécessite le recours à des outils qui prennent en compte la modélisation de l'incertain pour rendre leur exploitation plus ou moins réaliste.

Notre travail portant sur le suivi multi-caméras de véhicules, la seconde partie de cette session est relative aux facteurs permettant de caractériser au mieux les objets que nous comptons suivre. Il est question de descripteurs visuels et de modèles d'apparence pouvant nous permettre de définir une signature de nos objets. Dans une troisième partie, nous allons présenter quelques notions de distances permettant de mesurer la similarité entre deux objets et en retenir celles que nous jugeons pertinentes dans le cadre de notre étude. La quatrième partie de cette session portera sur la fusion d'informations et l'association de données (la mise en correspondance) en vue de prendre des décisions dans le cadre du suivi d'objets. Nous y présentons quelques outils et méthodes d'association de données ainsi que quelques outils d'aide à la prise de décision dans l'incertain. Nous terminons cette session par la présentation de nos choix et une conclusion.

1.3.1. OBJETS ET MODELES D'APPARENCE

Les notions que nous allons aborder ici se limiteront à celles qui nous intéressent particulièrement dans le cadre de suivi de véhicules et éventuellement de motos en mouvement. Une bibliographie complète sur la modélisation de l'apparence des objets reste disponible via les références suivantes [16] [79].

Les caractéristiques auxquelles nous allons nous intéresser ici sont celles visuelles (couleur, forme) et celle relatives aux contraintes spatio-temporelles (vitesse ou modèle d'accélération, densité du trafic pour prédire éventuellement les variations de vitesse) des objets. Ces caractéristiques doivent nous permettre de mettre en place une signature pour identifier chaque objet suivi.

La couleur, la forme, la vitesse (ou éventuellement le modèle d'accélération) constitueront les primitives sur lesquelles vont se baser nos modèles.

1.3.2. MISE EN CORRESPONDANCE : METHODES D'ASSOCIATION DE DONNEES

Les méthodes classiques d'association de données sont multiples et sont présentées en détail dans [25] et reposent essentiellement sur l'un ou la combinaison des principes suivants : le NN (Nearest Neighbor), le GNN qui est une approche globale du NN, le PDA (Probabilistic Data association), le JPDA qui est une approche plus robuste et de résolution de conflits du PDA, et le MHT (Multi Hypotheses Tracking).

Ces différentes méthodes diffèrent par leur complexité, leur capacité à prendre en compte la création et la destruction de piste, ainsi que leur capacité à gérer les ambiguïtés dans certaines situations d'association.

La représentation et le traitement de l'incertitude dans les systèmes de suivi se base sur des théories numériques ou non. Parmi les approches numériques on peut citer : la théorie des probabilités, la théorie des possibilités, la théorie de l'évidence, les réseaux bayésiens, la logique floue.

Dans cette session, nous allons présenter brièvement chacune de ces approches puis proposer l'utilisation de la théorie de l'évidence et plus précisément la théorie des croyances transférables, tout en justifiant notre choix.

1.3.2.1. Le NN et le GNN

La méthode du plus proche voisin ou NN (Nearest Neighbor) associe l'objet qu'on veut identifier à l'objet qui lui est plus proche parmi un ensemble d'objets connus. Un seul objet de cet ensemble est associable à l'objet à identifier. On calcule la distance entre l'objet à identifier et chacun des objets candidats, et on retient l'objet qui minimise cette distance. La distance qui est courante dans cette méthode est la distance de Mahalanobis qui est basée sur la corrélation de la série de données traitées et qui permet la prise en compte de certaines

erreurs telles que celles relatives aux prédictions et celles liées aux observations à l'instant courant. Dans un suivi multipistes, les problèmes que pose cette approche sont relatifs au choix de l'ordre de traitement des pistes ainsi que celui de la bonne distance discriminant correctement deux objets, pour ne pas faire de fausses associations.

Une solution à ce problème serait de calculer dans un premier temps les distances entre chacune des pistes et tous les objets candidats à l'association, puis chercher dans un second temps l'association globale minimisant par exemple la somme des distances d'association. D'autres distances sont utilisables. Notons, qu'il est à proscrire la choix d'une distance minimisant le produit des distances individuelles (locales) en raison du fait que zéro est absorbant pour le produit et une distante nulle d'une seule association locale engendrerait une valeur zéro pour la distance globale même si plusieurs distances locales sont très grandes (imprécision d'association); ce qui peut fausser le coût global d'une association. Dans [29], les auteurs en ont proposé une déclinaison très intéressante. Cette nouvelle approche est développée sous le nom de GNN ou Global Nearest Neighbor.

Un inconvénient majeur des méthodes NN et GNN est qu'elles nécessitent la connaissance du nombre de cibles à suivre et ce nombre doit être statique dans le temps ; Ce qui enlève toutes possibilités de gestion d'apparition ou la disparition d'une cible.

1.3.2.2. Le PDA et le JPDA

Certaines situations peuvent justifier la nécessiter de faire des associations entre une piste et plusieurs objets ; Ce que le NN et le GNN n'envisagent pas. En effet, pour des raisons d'occultation, de multiples détections ou de problèmes d'angle de vue, il peut exister plusieurs facettes ou observations d'un même objet. Dans de pareilles situations, on fait recours à une approche probabiliste d'association de données qui est le PDA ou Probabilistic Data Association. Cette approche a été développée par Bar-Shalom et Tsé dans [24] et repose sur la prise en compte de l'ensemble des observations candidates à une association pour la mise à jour d'une piste. Pour cela, une région de validité est définie à l'aide d'une matrice d'innovation (qui est une matrice de covariance) et les candidats à l'association sont les observations qui sont dans cette région de validité. Un estimateur global est construit à partir d'estimateurs locaux associés à chaque hypothèse d'association. Cet estimateur global est une somme pondérée des estimateurs locaux.

Pour le suivi multi-cibles, une évolution du PDA a été développée dans [28], [30] et [31]. Cette approche notée JPDA (Join Probabilistic Data Association), se distingue du PDA par le fait qu'elle considère plusieurs pistes et définit tous les événements joints possibles pour le nombre de pistes ou d'objets connus. Chaque événement joint est donc un ensemble d'associations entre pistes et objets connus. La probabilité de chaque événement joint est alors évaluée et l'état des cibles en cours de suivi est alors mis à jour avec la famille d'objets dont la somme pondérée de la mesure de l'événement joint. L'un des inconvénients du JPDA est qu'il utilise seulement les données ou observations de la capture courante. L'autre inconvénient est la possibilité de combinatoire explosive des hypothèses jointes dans le cas où plusieurs observations coïncideraient avec plusieurs objets connus.

Des détails sur les éléments du PDA et du JPDA sont présentés par [25] dans son Habilitation à diriger une thèse, et par [27] dans sa thèse.

1.3.2.3. Le MHT

Le suivi multi-hypothèses ou MHT (Multiple Hypothesis Tracking) a été développé par Reid [32] en 1979 et est une méthode de suivi de plusieurs cibles sous plusieurs hypothèses. Il est orienté observations et génère trois (3) hypothèses (objet déjà connu, apparition d'un nouvel objet, fausse alarme) à partir des observations. Il est basé sur la génération, à chaque étape du suivi, des différentes possibilités induites par la situation, y compris l'initiation de nouvelles pistes ainsi que la terminaison de pistes. Le MHT génère des hypothèses de manière explicite et toutes les possibilités sont étudiées; la poursuite d'une trajectoire existante, l'éventualité d'une occultation, l'apparition ou la disparition de nouvelles cibles ou trajectoires, ou bien la présence de fausses alarmes due à des données erronées. On calcule la probabilité a posteriori de chaque hypothèse en tenant compte de toutes les possibilités. A partir de cela et en tenant compte de la contrainte d'unicité d'association, on génère des hypothèses globales, qui englobent l'ensemble des associations entre les trajectoires et les points depuis le début de la séquence. Ce processus est répété sur plusieurs images et forme un arbre d'hypothèses globales. A chaque instant, l'algorithme du MHT cherche à maximiser la probabilité des enchaînements d'associations de toutes les détections présentes depuis leur apparition. Chaque itération se base sur l'ensemble des hypothèses établies à l'itération précédente, et chaque hypothèse fournit une interprétation de toutes les mesures effectuées depuis le début de l'algorithme.

Le sommet (racine) de l'arbre du MHT n'est associé à aucune hypothèse (l'hypothèse nulle) étant donné qu'il n'y a encore aucune détection au début. A chaque fois qu'il y a une nouvelle détection, un nouveau palier ou niveau (profondeur) est créé dans l'arbre. Ce niveau contient, pour chaque nœud parent, l'ensemble des possibilités d'affectation de la nouvelle détection qu'on peut lui relier. Chaque arc (segment parent-enfant) est pondéré par la probabilité de vérité de l'enfant (nouvelle association ou affectation), sachant l'historique des hypothèses représentées par le parent.

L'arbre MHT peut conduire à une combinatoire explosive mais ce qui fait l'intérêt du suivi multi-hypothèses est justement cette possibilité qu'il offre de pouvoir différer une décision dans le temps en maintenant les hypothèses jusqu'à avoir la confirmation d'une hypothèse et permettre ainsi de lever davantage d'ambiguïtés d'association. Il existe plusieurs approches d'optimisation du MHT pour élaguer l'arbre d'hypothèses et éviter une combinatoire explosive qui est le principal frein de son application temps réel.

La recherche de la meilleure interprétation de l'ensemble des détections traitées à un instant donné revient à rechercher dans l'arbre un flot maximum entre la racine et l'une des feuilles.

1.4. CONCLUSION

Des approches de détections de mouvement, celles utilisant une image d'arrière-plan semblent efficaces et étant toutefois les unes plus adaptées que les autres suivant le contexte et le type d'application. Au nombre de celles-ci, une association de celle utilisant un mélange de gaussienne et celle basée sur la méthode de codebook a retenu notre attention. Une mise en commun de ce choix avec celui de l'utilisation de la méthode de Viola et Jones devra nous permettre d'avoir de bon résultats dans la détection des véhicules

Les différentes méthodes d'association présentées ci-dessus ont chacune leurs avantages et leurs inconvénients. Plusieurs d'entre elles ont la capacité de prendre en compte la gestion de fausses alarmes et de bonnes détections. Cependant, en dehors du MHT, aucune de ces méthodes ne prend en compte aussi bien la gestion des situations ambiguës, que celle de la gestion de l'initialisation et de la terminaison de pistes qui sont deux notions cruciales dans un système de suivi de véhicule dans un flux vidéo. En effet, le nombre de véhicules en mouvement dans champ de vue d'une caméra de vidéosurveillance évolue constamment. Tandis que certains vont sortir de son champs de vue, d'autres pourront y entrer à tout instant et on ne peut prédire ni leur nombre ni leur modèle de mouvement. De même plusieurs véhicules se ressemblent et même si ce n'est pas le cas, des situations d'occultation peuvent se présenter et rendre difficiles les associations. Bien que le MHT présente comme inconvénient sa lenteur du fait de sa combinatoire explosive, c'est la solution que nous avons retenue dans notre travail pour la levée d'ambigüité en cas difficulté d'association.

Nous présentons plus loin dans une autre session (chapitre 2, sous-session 2.7.3.) les adaptations que nous proposons pour rendre l'utilisation du MHT réaliste et plus ou moins efficace.

CHAPITRE 2 : CADRES DE GESTION DE L'INCERTAIN

Sommaire

2.1.	Introduction	15
2.2.	La théorie des probabilités.....	16
2.3.	Les réseaux bayésiens	17
2.4.	La théorie des ensembles flous et des possibilités	19
2.5.	La théorie de l'évidence	20
2.6.	Bref aperçu de la théorie de l'évidence	21
2.6.1.	La modélisation	21
2.6.2.	Estimation des fonctions vraisemblance.....	25
2.6.3.	Combinaison.....	26
2.6.4.	Prise de décision.....	29
2.7.	Le TBM comme cadre de gestion de l'incertain.....	30
2.7.1.	Niveau crédal.....	31
2.7.2.	Niveau pignistique.....	31
2.7.3.	Le MHT pour la levée d'ambiguïté	32
2.8.	Conclusion	33

2.1. INTRODUCTION

Dans le suivi, les informations manipulées doivent être modélisées de façon à en faire ressortir leurs différentes caractéristiques ainsi que les perturbations dont elles peuvent faire l'objet.

De façon générale les informations issues de capteurs (caméras) sont très complexes en fonction de la forme des objets qu'ils contiennent, de leur nombre, de leur dynamique, du type d'informations traitées, de la diversité des sources qui sont très souvent imprécises mais peuvent par ailleurs être hétérogènes. Cette complexité des informations les rend sujettes à de l'imprécision, de l'incertitude, de l'incomplétude et parfois même conflictuelles en cas de diversité de sources. Pour la manipulation ou le traitement de ces informations, on a besoin de recourir à un formalisme qui peut être numérique et s'appuyer sur la théorie des probabilités, la théorie des possibilités, la logique floue, la théorie de l'évidence, la théorie de la résonance adaptative ; ou un formalisme symbolique reposant sur les systèmes de bases de connaissances ou la logique des propositions.

Cette partie de ce document présente assez brièvement les différentes solutions ci-dessus citées pour finalement choisir et justifier la solution qui retient notre attention et que nous allons utiliser plus loin dans nos modèles.

En effet, face au problème d'imprécision de l'information, trois approches sont alors proposées. Une première consiste à réduire le plus possible les imperfections en améliorant la qualité des sources d'information (capteurs) et en les multipliant (sources redondantes) pour renforcer leur précision. Cette approche n'est pas toujours efficace parce dans un environnement réel il est très difficile de garantir la perfection des sources de données surtout lorsqu'il s'agit de capteurs. En outre, la fusion d'informations provenant de plusieurs sources, bien que permettant de préciser ou de renforcer la qualité des informations, peut être source de conflit. Ne pas prendre en compte cette éventualité biaise l'information et par conséquent les résultats.

Une seconde approche consisterait à accepter l'imprécision et recourir à des descripteurs et des algorithmes robustes permettant d'en minimiser les effets. Cette approche très intéressante est largement utilisée dans la littérature mais le fait que les causes des imperfections sont multiples ne facilite pas la mise en place de ces descripteurs et algorithmes, robustes à la fois à plusieurs types de variations.

La troisième possibilité est d'intégrer la prise en compte des imperfections dans la modélisation de l'information, en considérant ces imperfections comme une catégorie de connaissance. Cette approche est beaucoup plus proche de la réalité et permet d'utiliser des connaissances sur les imperfections possibles pour approximer le raisonnement et nuancer les décisions. Cependant, très peu de théories permettent de prendre en compte de façon réaliste le caractère imparfait de l'information ou de la donnée dans sa modélisation et dans le mode de raisonnement en vue d'une prise de décision.

2.2. LA THEORIE DES PROBABILITES

Elle est l'approche la plus ancienne et la plus utilisée. On distingue trois façons de la concevoir :

- Une combinatoire, basée sur l'équipossibilité. On trouve tous les cas probables dans le référentiel et on dénombre les cas favorables à l'événement observé ;
- Une objective ou fréquentiste basée sur une étude statistique d'un phénomène pour en déduire une distribution de probabilité d'une variable aléatoire sur un référentiel ;
- Une subjective privilégiant le codage de l'état des connaissances sur un phénomène et la confiance dans l'apparition d'un événement pour établir une répartition de probabilités sur le même référentiel, reflet de l'état des connaissances.

L'approche probabiliste permet de prendre en compte de manière rigoureuse des incertitudes aléatoires ou stochastiques. Mais il est assez difficile voire impossible de prendre en compte d'autres formes d'imperfections aussi bien formellement que sémantiquement. Plusieurs états de connaissance peuvent donner la même probabilité subjective.

La représentation de l'ignorance se fait de façon très simple en utilisant la loi de probabilité uniforme et en répartissant équitablement la probabilité d'un sous ensemble d'événement ou d'hypothèses sur chacun de ses événements ou hypothèses. La fusion d'information se base sur la loi de Bayes sans aucune possibilité de modéliser le conflit qui est interprété comme un doute ou une ignorance. Il en résulte souvent une représentation infidèle et parfois paradoxale de l'information disponible. Cette incomplétude de la théorie des probabilités dans la modélisation de l'information conduit à la nécessité de recourir à d'autres formalismes plus généralistes.

2.3. LES RESEAUX BAYESIENS

Les réseaux bayésiens reposent sur un formalisme basé sur les théories des probabilités et des graphes. Un réseau bayésien est un graphe causal auquel on a associé une représentation probabiliste sous-jacente pour rendre quantitatifs les raisonnements sur les causalités que l'on peut faire à l'intérieur du graphe. Il permet de faire de l'inférence à partir de calculs de probabilités conditionnelles d'événements reliés les uns aux autres par des relations de cause à effet.

Les modèles bayésiens permettent de faire de la modélisation (représentation des connaissances, de l'estimation, de la combinaison et de la prise de décisions. L'objet principal est de mener, grâce à l'observation d'un phénomène aléatoire, une inférence sur la distribution probabiliste à l'origine de ce phénomène ; donc fournir une analyse (ou une description) d'un phénomène passé, ou une prédiction d'un phénomène à venir de nature similaire.

La modélisation se base sur les probabilités conditionnelles qui sont des distributions de probabilités sur des hypothèses exclusives d'un monde fermé donc exhaustif. Ces distributions sont le plus souvent estimées pour des événements discrets, à l'aide de la formule de Bayes et de fréquences à partir d'apprentissage. En présence d'événements continus, elles sont approchées par des distributions connues ou en utilisant les k plus proches voisins de la réponse.

Ces distributions peuvent être combinées dans le cadre d'une fusion d'information pour une prise de décision. La combinaison peut s'appuyer sur la loi de Bayes ou se faire à travers la combinaison des probabilités à l'aide de plusieurs types d'opérateurs.

Avec la loi de Bayes, on détermine les probabilités conditionnelles par :

$$p(d_i | S_1; S_2 ; \dots ; S_m) = \frac{p(S_1; S_2 ; \dots ; S_m | d_i) \cdot p(d_i)}{p(S_1; S_2 ; \dots ; S_m)} \quad (1-7)$$

Où : d_i représente la décision i parmi n décisions

S_j représente la source j .

Avec la combinaison des probabilités on définit les vecteurs probabilités p_j qu'on considère comme étant chacun le résultat de chaque source S_j .

$$p_j(x) = (p(d_1|S_j); p(d_2|S_j); \dots; p(d_n|S_j)) \quad (1-8)$$

La probabilité globale est obtenue après combinaison de ces vecteurs probabilités p_j en utilisant l'un des multiples opérateurs disponibles (maximum, minimum, moyenne, etc.).

La prise de décision se fait généralement à partir des probabilités a postériori et on choisit la décision p_k qui maximise les p_j . Il existe cependant plusieurs autres approches dont le maximum d'entropie, l'espérance maximale.

L'approche bayésienne, au-delà de ses autres multiples domaines d'application, est très efficace en fusion d'images, en fusion de classifieurs et en fusion pour la segmentation d'images satellitaires. Mais son efficacité est subordonnée à la qualité de son étape d'apprentissage qui a lieu lors de la phase d'estimation et qui impose des contraintes très strictes au niveau des mesures du fait des exigences mathématiques des axiomes de base des probabilités. Aussi, dans les cas de fusion d'images, la qualité des images influe beaucoup sur la qualité des décisions. La qualité des résultats de l'approche bayésienne est donc très vite dégradée par une mauvaise qualité de l'apprentissage et/ou celle des images.

En général si les approches probabilistes représentent bien l'incertain qui entache l'information, elles ne donnent la possibilité de modéliser correctement l'imprécision qu'elles tendent à confondre avec l'incertitude. Leur avantage vient du cadre mathématique cohérent et puissant sur lequel elles s'appuient, de la justification axiomatique dont leurs outils font l'objet, et de la richesse de ces outils statistiques et combinatoires mis à disposition. Cependant, leur contrainte d'additivité peut s'avérer trop forte dans certaines situations et domaines comme il est montré dans [36]. La notion de monde ouvert n'y est pas non plus envisageable car l'exhaustivité est de rigueur. La théorie des probabilités et celle bayésienne qui en découle combinent les informations en procédant à des produits de probabilités conditionnelles ; choses qui agissent fortement sur les probabilités des événements qui sont issus de longues chaînes de production, en amenuisant leur valeur au point de les rendre parfois inexploitable. Une utilisation efficace des méthodes bayésiennes nécessite généralement d'avoir à disposition beaucoup de connaissances ou d'informations sur le problème ; ce qui n'est pas toujours évident en situation réelle.

Ces différentes contraintes et limitations justifient la nécessité d'explorer d'autres approches non probabilistes, plus généralistes et permettant de modéliser plus fidèlement les problèmes en se basant sur la sémantique de l'information disponible et en rapprochant de la réalité.

2.4. LA THEORIE DES ENSEMBLES FLOUS ET DES POSSIBILITES

La théorie des ensembles flous a été développée par Zadeh [33, 34] et permet une modélisation très souple des informations imprécises. Le même auteur a introduit quelques années plus tard [37] la théorie des possibilités qui en est une dérivée et qui sera reprise par plusieurs chercheurs dont Dubois et Prade [35, 38].

La théorie des ensembles flous propose une fonction d'appartenance qui établit une notion d'appartenance graduelle d'un élément à un ensemble qui sera appelé de ce fait un ensemble flou et qui sera complètement décrit par la fonction d'appartenance. La fonction d'appartenance traduit le degré d'appartenance de chaque élément d'un ensemble flou à ce dernier.

$$\begin{aligned} \mu_F : \mathcal{S} &\rightarrow [0, 1] \\ x &\rightarrow \mu_F(x) \end{aligned} \quad (1-9)$$

où :

- \mathcal{S} est un ensemble classique (dit net) composé d'éléments qui satisfont à des propriétés précises.
- F est un sous-ensemble flou de \mathcal{S}
- μ_F est une fonction d'appartenance

La théorie des possibilités repose quant à elle sur les mesures de possibilité et de nécessité qui permettent de décrire l'incertitude d'un événement par le degré de possibilité et le degré de possibilité de l'évènement contraire qui s'interprète comme un degré de nécessité.

Une mesure de possibilité est une fonction Π de \mathcal{S} dans $[0, 1]$ telle que :

$$\begin{aligned} \Pi(\emptyset) &= 0, \\ \Pi(\mathcal{S}) &= 1, \end{aligned} \quad (1-10)$$

$$\forall I \subset \mathbb{N}, \forall A_i \subseteq \mathcal{S} (i \in I), \Pi(\bigcup_{i \in I} A_i) = \sup_{i \in I} \Pi(A_i)$$

Dans les situations d'ignorance totale sur la réalisation d'un événement, sa mesure de possibilité et la mesure de possibilité de son complément peuvent toutes prendre 1 comme valeur.

$$\Pi(A) = \Pi(A^c) = 1 \quad (1-11)$$

La notion de mesure de nécessité a été alors introduite pour lever cette indétermination.

Une mesure de nécessité est définie comme une fonction N de \mathcal{C} dans $[0, 1]$ telle que :

$$\forall A \subseteq \mathcal{S}, \quad N(A) = 1 - \Pi(A^c) \quad (1-12)$$

La mesure de nécessité d'un événement permet de mesurer l'impossibilité de son événement contraire et il découle de cette dualité entre nécessité et possibilité que si un événement est nécessaire, son contraire est impossible.

Contrairement aux mesures de probabilité, les mesures de possibilité et de nécessité sont deux mesures non additives.

La fusion floue et possibiliste [34, 37, 38, 39], permet de prendre en compte l'imprécision et l'incertitude et met à disposition plusieurs opérateurs de combinaison avec une grande souplesse dans les modes de combinaison. Elle permet de représenter des informations qualitatives, imprécises et vagues. L'inférence se fait par des règles logiques (ou leur équivalent sous forme numérique), réalisant essentiellement un raisonnement de type déductif, pouvant être qualitatif.

Cependant, la souplesse dans le mode de combinaison de la multitude d'opérateurs que ces théories mettent à disposition met en difficulté l'utilisateur dans la définition des fonctions à utiliser. Même si la modélisation de l'imprécision est bien gérée avec les fonctions d'appartenance, elles n'arrivent pas à modéliser correctement l'incertitude. La théorie des possibilités permet une bonne représentation aussi bien de l'imprécision que de l'incertitude. La véritable difficulté de ces deux approches de modélisation réside dans l'estimation et le choix des fonctions d'appartenances ou des distributions de possibilités. Plusieurs critères [40], [41] peuvent intervenir dans leur choix en fonction du domaine d'application.

2.5. LA THEORIE DE L'EVIDENCE

La théorie de l'évidence ou théorie des fonctions de croyance est née suite aux travaux du statisticien Arthur P. Dempster [42] sur les bornes inférieures et supérieures d'une famille de distributions de probabilités. Elle a pris toute son importance grâce aux travaux complémentaires du mathématicien Glenn Shafer [43] qui en a proposé un formalisme mathématique plus abouti en se basant sur les travaux de Dempster. En référence aux travaux de ses deux initiateurs et plus grands contributeurs, la théorie de l'évidence portera également le nom de théorie de Dempster-Shafer.

La théorie de l'évidence est vue comme une généralisation de l'inférence bayésienne au traitement de l'incertain et un de ses atouts essentiels est qu'elle manipule des sous-ensembles plutôt que des singletons comme c'est le cas dans les approches probabilistes et bayésiennes. Cette distinction lui confère une grande souplesse de modélisation et la prise en compte de multiples situations. Elle sait représenter parfaitement ce qui est certain tout en offrant la possibilité de prendre en compte dans sa modélisation, ce qui est inconnu (l'ignorance), ce qui est incertain et ce qui est imprécis. Elle permet de ce fait, la modélisation et la manipulation de connaissances imparfaites à partir de plusieurs fonctions. Grâce à ses multiples outils, développés au fil du temps par plusieurs auteurs, elle permet de prendre en compte le conflit qui peut résulter de mise en commun (fusion) d'informations provenant de plusieurs sources. Au-delà de la prise en compte d'hypothèses exhaustives d'un monde fermé, il ouvre la voix, à partir des travaux de certains chercheurs, à la prise en compte d'un monde ouvert ou de contradictions d'observations.

Elle semble répondre le mieux à nos attentes pour la représentation et la manipulation des données pour le suivi et la ré-identification de véhicules dans un environnement multi-capteurs. C'est pour cette raison que nous avons décidé de l'aborder un peu plus en détail dans la sous session qui va suivre pour en présenter les concepts de base ainsi que quelques outils pour la fusion d'information.

L'utilité de ces fonctions, comme une alternative aux probabilités subjectives, a été démontrée plus tard de manière axiomatique par P. Smets [67, 68] au travers du modèle des croyances transférables. Ce modèle fournit une interprétation non probabiliste cohérente et permet de clarifier le concept sous-jacent à la théorie. Nous finirons par une brève présentation du TBM qui a été introduite par Smets.

2.6. BREF APERÇU DE LA THEORIE DE L'EVIDENCE

La théorie de l'évidence est décrite comme une généralisation de l'inférence bayésienne au traitement de l'incertain. Elle permet la représentation des informations d'une situation, d'émettre un ensemble d'hypothèses disjoints et exhaustifs sur un problème qu'on cherche à résoudre (on verra plus loin le cas de non exhaustivité des hypothèses), et d'évaluer à partir des degrés de croyance, la véracité de propositions, aussi bien élémentaires (hypothèses singleton) que composition de propositions élémentaires (disjonction d'hypothèses singletons).

Cette évaluation, très souvent subjective, s'établit selon notre état de connaissance et est essentiellement conditionnée par un ensemble d'informations issues d'une ou de plusieurs sources (un ou plusieurs experts, un ou plusieurs capteurs). Lorsque ces informations proviennent de plusieurs sources, elles sont fusionnées en vue de renforcer les croyances qu'elles véhiculent et les utiliser dans le cadre d'une prise de décision. On verra également, plus loin que la non-idempotence de certains opérateurs peut produire l'effet contraire de celui visé par la fusion en générant des données traduisant un conflit.

Les différentes étapes de base de la fusion d'information en théorie de l'évidence sont : la modélisation, l'estimation, la combinaison et la prise de décision.

2.6.1. LA MODELISATION

La phase de modélisation est une phase très importante qui permet d'associer une représentation numérique ou symbolique à chacune des différentes connaissances et observations, de choisir les fonctions de croyances les plus appropriées pour exprimer la vraisemblance de chacune des propositions. Elle commence par la définition de l'ensemble des hypothèses solutions du problème observé. Cet ensemble, noté Ω , est appelé cadre de discernement ou corps évidentiel. Il contient des éléments deux à deux disjoints et est considéré comme étant exhaustif par Shafer [43] qui introduit ainsi la notion de monde fermé

ou clos (Closed world) qui sous-entend l'unicité de la solution et son appartenance obligatoire au cadre de discernement.

L'ensemble des parties ou des sous-ensembles de Ω est noté 2^Ω et est appelé référentiel de définition. Cet ensemble contient toutes les disjonctions possibles des éléments de Ω et correspond à l'ensemble de toutes les propositions pouvant être rencontrées dans le monde associé au problème observé. Une proposition est une hypothèse singleton ou une disjonction d'hypothèses.

On a les notations suivantes :

$$\begin{aligned} \Omega &= \{H_1, H_2, \dots, H_n\} \\ 2^\Omega &= \{A / A \subseteq \Omega\} = \{\phi, \{H_1\}, \{H_2\}, \dots, \{H_n\}, \{H_1, H_2\}, \dots, \Omega\} \end{aligned} \quad (1-13)$$

où :

- $H_i, i \in [1, n]$: est une hypothèse du cadre de discernement Ω
- L'hypothèse vide ϕ correspond à la proposition impossible ;
- L'hypothèse Ω correspond à l'ignorance totale.

La théorie des fonctions de croyance fait partie des théories de l'incertain. Un certain nombre de fonctions sont mis à disposition pour caractériser la vraisemblance ou évaluer la véracité de chacune des propositions (hypothèses) du référentiel vis-à-vis d'une information issue d'un capteur, d'un agent ou d'un expert et traduisant leur opinion sur l'état du monde.

La prochaine étape de la modélisation est celle de définition des fonctions de vraisemblance qui représentent des outils mathématiques de manipulation des connaissances.

2.6.1.1. Fonction de masse

Elle est définie par une fonction m qui, à toute proposition du référentiel de définition, lui associe la partie du degré de croyance (ou confiance) placée exactement en cette proposition, sans pouvoir la répartir sur les hypothèses singletons qui la compose s'il y en a. Elle présente une grande analogie avec la notion de distribution de probabilité à la différence que la répartition de la masse unité est faite, non seulement sur l'ensemble des hypothèses singletons, mais également sur les disjonctions de celles-ci.

$$\begin{aligned} m : 2^\Omega &\rightarrow [0,1] \\ A &\mapsto m(A) \end{aligned} \quad (1-14)$$

La fonction de masse, vérifie la propriété suivante :

$$\begin{aligned} \sum_{A \subseteq \Omega} m(A) &= 1 \\ m(\phi) &= 0 \end{aligned} \quad (1-15)$$

La deuxième composante de cette équation voulant dire que la proposition ϕ ne peut se réaliser dans le cas d'un monde clos ; donc pour un cadre de discernement exhaustif.

Nous verrons plus tard que dans la réalité le cadre de discernement n'est pas souvent exhaustif et que dans ce cas il faudra donner une autre signification à $m(\phi)$ qui n'est plus nulle.

On désigne par éléments focaux, les propositions dont la masse de croyance est strictement supérieure à zéro. L'ensemble des éléments focaux est appelé noyau.

Nota :

- Lorsque les seuls éléments focaux sont les hypothèses singletons du cadre de discernement, la fonction de masse est confondue à une fonction de probabilité.
- L'ignorance totale d'une source se traduit par : $m(\Omega) = 1$.
- L'imprécision d'une source se traduit par : $m(A) = 1$, $A \subset \Omega$ et $A \neq H_{i \in \{1, \dots, n\}}$.
- La connaissance précise se traduit par : $\exists i \in \{1, \dots, n\} / m(H_i) = 1$.

2.6.1.2. Fonction de crédibilité ou de croyance

Elle est définie par la fonction Cr , appelée crédibilité, qui, à toute proposition du référentiel de définition, associe exactement la totalité de la croyance qu'on lui accorde, et non seulement la partie du degré de croyance. Elle permet de mesurer à quel point les informations données par une source sont en faveur d'une proposition, et s'interprète donc comme la croyance minimum en cette proposition.

$$\begin{aligned} Cr : 2^\Omega &\rightarrow [0,1] \\ A &\mapsto Cr(A) = \sum_{B \subseteq A, B \neq \phi} m(B) \end{aligned} \tag{1-16}$$

On en déduit que :

$$m(A) = \sum_{B \subseteq A, B \neq \phi} (-1)^{|A-B|} Cr(B) \tag{1-17}$$

Elle vérifie les propriétés suivantes :

$$\begin{aligned} Cr(\phi) &= 0 && \text{et} && Cr(\Omega) &= 1 \\ \forall A \in 2^\Omega, & Cr(A) + Cr(A^c) &\leq 1 \\ Cr\left(\bigcup_{i=1}^n A_i\right) &\geq \sum_{I \subseteq \{1,2,\dots,n\}, I \neq \phi} (-1)^{|I|+1} Cr\left(\bigcap_{i \in I} A_i\right) \end{aligned} \tag{1-18}$$

2.6.1.3. Fonction de plausibilité

Elle est définie par la fonction Pl , appelée plausibilité, qui, à toute proposition du référentiel de définition, associe l'intensité avec laquelle on ne doute pas de cette proposition. Elle mesure à quel point les informations données par une source ne contredisent pas cette proposition, et s'interprète donc comme la croyance maximale qu'on accorde à cette proposition.

$$Pl : 2^\Omega \rightarrow [0,1]$$

$$A \mapsto Pl(A) = \sum_{B \cap A \neq \emptyset} m(B) \quad (1-19)$$

On peut aussi écrire :

$$Pl(A) = Cr(\Omega) - Cr(A^c) \quad (1-20)$$

Ce qui donne en monde fermé, la définition de Shafer :

$$Pl(A) = 1 - Cr(A^c) \quad (1-21)$$

Elle vérifie les propriétés suivantes :

$$Pl(\emptyset) = 0 \quad \text{et} \quad Pl(\Omega) = 1$$

$$\forall A \in 2^\Omega, \quad Pl(A) \geq Cr(A)$$

$$\forall A \in 2^\Omega, \quad Pl(A) + Pl(A^c) \geq 1$$

$$\forall A \in 2^\Omega, \quad Cr(A) + Cr(A^c) = 1 \Leftrightarrow Cr(A) = Pl(A)$$

$$\text{On a également : } Cr(A) \leq P(A) \leq Pl(A), \quad \forall A \in 2^\Omega \quad (1-23)$$

Et l'intervalle $[Cr(A), P(A)]$ représente l'intervalle de confiance en la proposition A et sa longueur correspond à la mesure de toute l'ignorance relative à A et son complémentaire A^c .

La plausibilité et la crédibilité sont deux mesures duales qui sont toutes en bijection avec la fonction de masse de croyance.

Il existe d'autres fonctions de croyances telles que la fonction de communalité et la fonction d'implicabilité qui sont très peu utilisées et que nous n'allons pas présenter dans ce document.

2.6.1.4. Affaiblissement

Chacune des fonctions de croyance permet de caractériser la vraisemblance ou d'évaluer la véracité de chacune des propositions (hypothèses) du référentiel vis-à-vis d'une information issue d'un capteur, d'un agent ou d'un expert et traduisant leur opinion sur l'état du monde. Mais il est très courant que la source d'information ne soit totalement fiable. Ne pas tenir compte de cette imperfection des sources, biaiserait les données et éloignerait de la réalité des observations.

Shafer introduit donc la notion d'affaiblissement des sources pour modéliser la fiabilité des sources à partir d'un coefficient d'affaiblissement. Une nouvelle fonction de masse de croyance est alors définie à partir de la fonction de masse initiale et de ce coefficient d'affaiblissement de la manière suivante :

$$\forall A \in 2^\Omega, \quad \begin{cases} m'(A) = \alpha m(A) \\ m'(\Omega) = 1 - \alpha(1 - m(\Omega)) \end{cases} \quad (1-24)$$

$\alpha \in [0,1]$ et plus la source est fiable, plus α sera proche de 1. Inversement, plus la source sera non fiable, plus la valeur de α sera proche de 0. Lorsque $\alpha = 0$, on a affaire à une source totalement non fiable et on se retrouve dans une situation d'ignorance totale.

L'affaiblissement de la fonction de masse entraîne un affaiblissement proportionnel des fonctions de crédibilité et de plausibilité. Elle permet d'augmenter l'intervalle de confiance de la source et réduit le conflit lors des combinaisons.

2.6.1.5. Conditionnement

Le conditionnement permet de modifier sa connaissance a priori venant d'une source en prenant en compte une nouvelle information totalement certaine ou véridique. Par exemple, lorsque pour une source définie initialement sur Ω , on apprend par la suite que la proposition solution se trouve dans une partie A de Ω , alors on recalcule les différentes grandeurs utilisées mais cette fois-ci conditionnées par la proposition A .

Pour ce faire, on transforme la fonction de masse quelconque m sur Ω en une fonction de masse $m(\cdot | A)$ dont tous les éléments focaux sont inclus dans A , par transfert de la masse de chaque élément focal de m sur sa partie incluse dans A , et par nouvelle normalisation par rapport à la masse qui a ainsi pu être affectée :

$$m(B|A) = \frac{\sum_{C \cap A = B} m(C)}{\sum_{C \cap A \neq \emptyset} m(C)} \quad (1-25)$$

L'opération inverse du conditionnement est le déconditionnement. Elle part de la connaissance de la fonction de masse de croyance sur un sous ensemble A de Ω pour arriver à reconstruire une fonction de masse de croyance complète sur tout l'espace de discernement Ω .

2.6.2. ESTIMATION DES FONCTIONS DE VRAISEMBLANCE

L'un des problèmes majeurs en théorie de l'évidence est l'estimation des masses de croyance. En effet, nous avons vu qu'une fois le cadre de discernement retenu, tout le reste du travail repose sur la manipulation des fonctions de croyance et la qualité des résultats dépend fortement de celle de l'estimation de ces fonctions qui permettent de caractériser la vraisemblance ou d'évaluer la véracité de chacune des propositions du référentiel. Il n'existe pas de solution miracle pour estimer les fonctions de croyance. Plusieurs travaux ont été

réalisés sur la question et les approches de solution sont variées et souvent fonction du domaine et du type d'informations manipulées.

La situation est d'autant plus difficile qu'il est souvent question de masses qui sont des données dynamiques et instantanées qui évoluent avec le temps et les observations. Il est donc quasi impossible de faire recours à de données réelles produites par un expert. Utiliser également des hypothèses simplificatrices comme c'est le cas en théorie des probabilités enlèverait aussi considérablement à la théorie des fonctions de croyance tout son intérêt.

Plusieurs travaux de recherche ont permis de mettre en place un certain nombre d'approches pour le calcul des masses d'évidence ; chaque méthode étant adaptée à des domaines d'application bien définis. Parmi les approches qu'on trouve dans la littérature, il y a celles qui s'appuient sur des traitements niveau pixels ; celles qui établissent les attributs des objets en s'appuyant sur leurs modèles géométriques ou visuels [44, 46]; et celles plus abstraites et symboliques qui s'appuient sur des masses données par des experts.

On peut citer les travaux de Shafer [43], de Denoeux [45, 54], de Vannoorenberghe et Denoeux [49], de Appriou [50, 51], de Yager [53], de Lefevre [52]. Certains de ces approches sont présentées dans [47, 48, 57, 58]. La majorité des méthodes fournissent des masses d'évidence pour les singletons (éventuellement pour leur complémentaires) et pour le cadre de discernement.

De façon globale, notre approche va se baser sur les travaux de Denoeux [45, 54] et ceux de Permal et al. [47] pour la définition de nos masses de croyance. Elle sera présentée plus loin dans le troisième chapitre de notre travail.

2.6.3. COMBINAISON

Les informations viennent généralement de plus d'une source. Elles sont en outre généralement imparfaites (incertaines, imprécises, incomplètes). Il est très souvent nécessaire de combiner ces informations pour en renforcer la qualité et les rendre plus fiable à travers leur redondance ou leur complémentarité. La complémentarité permet de rendre les connaissances plus complètes et la redondance sert à les rendre plus précises et plus certaines. Plusieurs opérateurs ont été développés dans le cadre de la théorie de l'évidence pour faire principalement deux types de combinaisons : la combinaison conjonctive et la combinaison disjonctive. La fusion d'information permet de construire une nouvelle fonction de masse élémentaire à partir des fonctions de masses issues des données provenant de plusieurs sources. Une des exigences de la fusion d'information est la nécessité d'avoir un référentiel commun.

Dans cette partie de notre travail nous allons présenter certaines déclinaisons de ces opérateurs et présenter également certaines opérations préalables visant à revenir à un référentiel commun au cas où cette condition préalable ne serait pas satisfaite.

2.6.3.1. Loi de combinaison de Dempster

Elle permet de combiner les jeux de masses m_j de plusieurs sources S_j indépendantes à travers la somme orthogonale \oplus de Dempster, pour obtenir un jeu de masse unique m définie sur le même cadre de discernement Ω que les n masses m_j ; $n \geq 2$.

$$m = m_1 \oplus m_2 \oplus \dots \oplus m_n$$

$$m(A) = \frac{1}{1-K} \sum_{B_1 \cap B_2 \cap \dots \cap B_n = A} \prod_{j=1}^n m_j(B_j), \quad \forall A \subseteq \Omega, A \neq \phi \quad (1-26)$$

Où

$$K = \sum_{B_1 \cap B_2 \cap \dots \cap B_n = \phi} \prod_{j=1}^n m_j(B_j), \quad \text{et } K \neq 1 \quad (1-27)$$

Représente la mesure du niveau de conflit entre les sources. $(1-K)$ est un facteur permettant d'ignorer le conflit en répartissant la masse conflictuelle sur les autres propositions pour rester conforme à l'hypothèse du monde fermé de Shafer.

L'indépendance des sources est importante du fait de la non idempotence de l'opérateur \oplus de Dempster. Cela permet d'éviter d'atténuer les croyances sur les propositions recevables en les redistribuant, plutôt que de les renforcer [128].

En présence de deux sources, cette loi de combinaison de Dempster s'écrit :

$$m(A) = m_1 \oplus m_2(A) = \frac{1}{1-K} \sum_{B \cap C = A} m_1(B) \cdot m_2(C) \quad (1-28)$$

$$K = \sum_{B \cap C = \phi} m_1(B) \cdot m_2(C) \quad (1-29)$$

Notons en passant que cette loi de combinaison de Dempster est commutative et associative.

2.6.3.2. Monde ouvert et monde ouvert étendu

Plusieurs critiques ont été portées à l'endroit de la normalisation de la loi de Dempster pour répondre aux exigences du monde fermé de Shafer. En effet, en considérant l'exhaustivité des hypothèses du cadre de discernement, la seule possibilité qu'on laisse à l'explication d'un éventuel conflit, reste celle d'un dysfonctionnement d'un ou de plusieurs sources. Or, dans la réalité un conflit peut avoir plusieurs origines dont celle de la non-exhaustivité des hypothèses.

Il est effectivement difficile, voire impossible dans plusieurs cas, de mettre en place un cadre de discernement exhaustif. Cette situation va amener Smets [55, 59] à proposer un monde

ouvert (open World) contenant toutes les hypothèses connues mais non forcément exhaustives. Ainsi, on autorisera et on comprendra l'affectation de masse à l'ensemble vide pour exprimer un conflit résultant soit de la non-exhaustivité du cadre de discernement, soit du dysfonctionnement des sources. Dans cette approche, il ne serait donc plus nécessaire de normaliser la masse de croyance résultant de la combinaison de sources comme le proposait Shafer. Mais dans ce cas, il faudra s'assurer également que toutes les sources sont fiables pour appliquer la règle de combinaison conjonctive ou règle de Dempster non normalisée définie comme suit :

$$m(A) = m_1 \circledast \dots \circledast m_n(A) = \sum_{B_1 \cap B_2 \cap \dots \cap B_n = A} \prod_{j=1}^n m_j(B_j), \quad \forall A \subseteq \Omega, A \neq \phi$$

$$m(\phi) = \sum_{B_1 \cap B_2 \cap \dots \cap B_n = \phi} \prod_{j=1}^n m_j(B_j)$$
(1-30)

Si les sources sont totalement fiables alors la seule cause du conflit serait la non-exhaustivité. Par contre, si on considère que les sources ne sont pas fiables et qu'elles peuvent avoir des défaillances, il devient impossible de situer exactement la cause d'un conflit lorsque cela surviendrait.

Pour pallier cet état de choses, Rombaut [56] propose un monde ouvert étendu qui contient non seulement toutes les hypothèses disjointes connues, mais également une hypothèse singleton (notée *) qui représente toutes les hypothèses éventuellement inconnues ou non prises en compte lors de l'élaboration du cadre de discernement. Cette hypothèse est alors ajoutée au cadre de discernement pour obtenir un nouveau cadre de discernement qui sera cette fois-ci exhaustif et autorisera l'application de la loi de combinaison de Dempster telle que proposée par Shafer.

$$\Omega = \{H_1, H_2, \dots, H_n, *\}$$
(1-31)

L'autre avantage de cette représentation est qu'il est désormais possible de dissocier le conflit dû à l'imperfection des sources de celui dû à l'éventualité de nouvelles hypothèses.

Il existe d'autres approches de combinaison telle que la combinaison disjonctive dont la masse est obtenue en considérant les unions des masses des sources combinées. Avec cette fonction de masse le conflit n'existe pas car la masse allouée à l'ensemble vide est toujours nulle. Cependant, cette fonction est très peu utilisée car plutôt que de renforcer les croyances, elle la rend plus imprécise en l'élargissant.

$$m(A) = \sum_{B_1 \cup B_2 \cup \dots \cup B_n = A} \prod_{j=1}^n m_j(B_j), \quad \forall A \subseteq \Omega$$
(1-32)

De toutes ces approches, celle de Rombaut nous semble plus représentative pour la combinaison des informations qui sont manipulées dans le cadre de nos travaux où nous

manipulons des données vidéo et où le cadre de discernement est autant dynamique que complexe.

2.6.4. PRISE DE DECISION

Il existe plusieurs outils de prise de décision dans le cadre de la théorie de l'évidence. Au nombre de ces outils figurent dans un premier temps ceux qui se basent sur la crédibilité ou la plausibilité qui permettent de représenter respectivement la croyance minimum et celle maximum ; puis celui se basant sur la probabilité pignistique qui est un compromis proposé par Smets [60] et qui est une mesure permettant d'approcher le couple (crédibilité, plausibilité).

Ainsi, pour une observation x et un ensemble de décisions singleton ω_i possibles, chacune des méthodes de prise de décision, ci-dessous citée, va proposer de choisir la décision ω_k qui maximise une des fonction de croyance bien précise, et parfois sous certaines contraintes.

On a donc :

2.6.4.1. Le maximum de plausibilité

$$x \in \omega_k \text{ si } Pl(\omega_k)(x) = \max\{Pl(\omega_i)(x), 1 \leq i \leq n\} \quad (1-33)$$

Il présente comme inconvénient principal son manque de pouvoir discriminant du fait de son caractère optimiste.

2.6.4.2. Le maximum de crédibilité

$$x \in \omega_k \text{ si } Cr(\omega_k)(x) = \max\{Cr(\omega_i)(x), 1 \leq i \leq n\} \quad (1-34)$$

Il constitue le critère le plus pessimiste car il détermine la décision qui donne le minimum de chance à chacune des disjonctions. Il est identique au précédent dans le cas où la combinaison ne porte que sur les singletons.

2.6.4.3. Le maximum de crédibilité sans recouvrement des intervalles de confiance

$$x \in \omega_k \text{ si } Cr(\omega_k)(x) \geq \max\{Pl(\omega_i)(x), 1 \leq i \leq n, i \neq k\} \quad (1-35)$$

Il est encore plus strict et plus sélectif que le critère précédent et est très peu utilisé car peu réaliste et peut engendrer l'impossibilité de prendre des décisions sur les singletons.

2.6.4.4. Le maximum de crédibilité avec rejet

$$x \in \omega_k \text{ si } Cr(\omega_k)(x) = \max\{Cr(\omega_i)(x), 1 \leq i \leq n\} \quad (1-36)$$

$$et Cr(\omega_k)(x) \geq Cr(\omega_k^c)(x)$$

Il s'agit d'une restriction du maximum de crédibilité à l'assurance de ne prendre la décision que si on lui accorde plus de crédit qu'à son contraire. Ici également, il y a la possibilité de se retrouver dans une incapacité de prendre de décision sur les singletons.

2.6.4.5. Le maximum de probabilité pignistique

$$BetP(\omega_i) = \frac{1}{1 - m(\emptyset)} \sum_{\omega_i \in A} \frac{m(A)}{|A|}, \quad \forall \omega_i \in \Omega \quad (1-37)$$

Où $|A|$ représente le cardinal de A

Ce compromis proposé par Smets permet de répartir de façon équitable (équiprobable) sur les singletons dont une proposition est la disjonction, les masses de cette proposition. Ceci permet de répartir la masse unité sur les hypothèses singleton et de se retrouver dans la situation d'une loi de probabilité. C'est un critère prudent permettant de lever certaines ambiguïtés.

Il existe d'autres règles de décision que nous n'allons pas aborder dans le cadre de ce travail.

2.7. LE TBM COMME CADRE DE GESTION DE L'INCERTAIN

Le TBM (Transferable Belief Model) ou Modèle de Croyance Transférable, est un modèle développé par P. Smets depuis [59, 60, 61]. C'est un cadre formel générique basé sur des interprétations subjectives et non probabilistes pour la représentation et la combinaison des connaissances incertaines, imparfaites ou imprécises qui s'appuie sur les fonctions de croyances de la théorie de l'évidence de Dempster-Shafer dont il est une extension. La singularité de l'approche du TBM vient de la séparation qu'il fait entre la représentation et la combinaison des données (niveau crédal) d'une part, et la prise de décision (niveau pignistique) d'autre part, pour en faire deux processus successifs dans sa démarche.

Il sera utilisé dans le cadre de ce travail pour la représentation de connaissances issues d'observations provenant de caméras d'un système de suivi multi-caméras, la fusion d'information multi-capteur pour une meilleure caractérisation d'objets issus de suivi local pour en faire un suivi global, et enfin dans le cadre de la prise de décision dans un processus de mise en correspondance (association de données) pour une ré-identification d'objets.

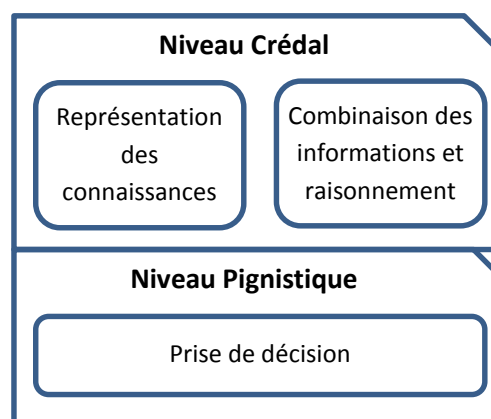


Figure 2.1 : Représentation des niveaux du TBM

2.7.1. NIVEAU CREDAL

Dans la partie crédale, la représentation des connaissances se fait à l'aide d'une distribution de masses de croyance (fonctions de masse de croyance) ou une combinaison de ces fonctions de masses de croyance sur la base d'un certain nombre d'opérateurs suivant que les sources soient fiables, concordantes, indépendantes, complémentaires ou non. Ces fonctions de croyance permettent de définir les parts de croyances accordées à différentes hypothèses ou groupes d'hypothèses (sans pouvoir en faire une répartition stricte sur les différentes composantes) d'un cadre de discernement. L'intérêt de l'approche est la possibilité de représentation du doute, de l'ignorance et même du conflit à travers l'affectation de masse de croyance respectivement à un sous-ensemble d'hypothèses, au cadre de discernement tout entier, à l'ensemble vide.

Toute fonction de masse de croyance notée m est en correspondance biunivoque (bijective) avec deux autres fonctions de croyance : celle de crédibilité notée bel et celle de plausibilité noté pl . Ces trois fonctions sont des représentations équivalentes d'un même élément d'information.

Plusieurs fonctions de masses de croyance peuvent être combinées sur un même univers (cadre de discernement) pour permettre de renforcer les croyances sur des hypothèses et apporter des précisions sur l'analyse qui en découlera dans la phase association de données. La théorie de l'évidence et particulièrement le TBM, mettent à disposition plusieurs outils et opérateurs de combinaison (somme conjointe, somme conjointe normalisée, somme disjointe, moyenne, etc.) suivant la catégorie des sources et les objectifs poursuivis.

La fusion d'information ne pouvant être faite que dans un référentiel commun, on fait souvent recours au principe du minimum d'information qui est un principe de précaution et qui consiste à choisir une fonction de croyance la moins informative (lorsqu'elle existe). Ce principe est utilisé à travers l'extension vide pour créer un espace de discernement commun par raffinement des différents espaces de discernement en faisant le transfert des masses vers les sous-ensembles les plus larges ou plus fins possibles. Une fois qu'on dispose d'un cadre de discernement commun, la combinaison des masses de croyances ne pose plus de problème.

2.7.2. NIVEAU PIGNISTIQUE

Dans la phase de la prise de décision, Smets propose dans le TBM, l'utilisation d'une distribution de probabilités en lieu et place des structures du niveau crédal (croyance, plausibilité, etc.). Il existe plusieurs approches mais la plus utilisée est celle reposant sur le calcul de probabilité pignistique. Il définit pour cela la probabilité pignistique **BetP** qui se présente comme suit :

$$BetP(\omega_i) = \frac{1}{1 - m(\emptyset)} \sum_{A \ni \omega_i} \frac{m(A)}{|A|} \quad \forall \omega_i \in \Omega \quad (1-38)$$

Une fois le **BetP** calculé, la décision est finalement prise en choisissant l'élément ω_k possédant la plus grande probabilité pignistique.

Cette solution ne lève pas cependant tous les cas d'ambiguïté. En effet, selon qu'on aura privilégié les objets perçus ou les objets connus pour définir l'univers, il arrive des fois qu'on aboutisse à des conclusions contradictoires. Au-delà de ces considérations, les situations d'occultation sont également source d'ambiguïté.

2.7.3. LE MHT POUR LA LEVEE D'AMBIGÜITE

Pour améliorer les mises en correspondance et lever certaines ambiguïtés, un suivi temporel orienté observation peut être utilisé pour retarder (différer) la prise de décision sur une certaine durée, le temps de consolider les informations sur certaines hypothèses et de lever les ambiguïtés. Parmi les méthodes de suivi multi-Hypothèses, le MHT (Reid 1979) qui est orienté observation, dans un suivi temporel, reste la méthode la plus utilisée et la plus appropriée à notre cadre car gérant tous les cas particuliers de trajectoires (apparitions, disparitions, fausses alarmes, trajectoires partielles). Notre objectif est d'implémenter une version personnelle du MHT et de l'utiliser dans la phase de prise de décision en cas d'ambiguïté.

Le principal reproche fait au MHT étant celui de sa combinatoire explosive, nous allons utiliser les optimisations de son algorithme pour réduire les hypothèses à plusieurs niveaux. Emmanuel Etievent présente dans sa thèse [69] trois grands axes d'optimisation que sont :

- 1- un partitionnement spatial pour diminuer la taille des problèmes d'association en scindant les trajectoires en sous-groupes indépendants composés de trajectoires en compétition entre elles mais pas avec les trajectoires d'autres groupes
- 2- élagage en largeur de l'arbre des hypothèses en ne gardant que les **n** meilleures hypothèses (**n** à définir) pour rendre la complexité du MHT plutôt polynomiale qu'exponentielle. On utilise pour cela un algorithme classique de détermination des **n** meilleures hypothèses. De ces hypothèses on pourra encore éliminer les moins probables

- 3- élagage en profondeur de l'arbre en jouant sur la dimension temporelle pour limiter la durée en termes de nombre d'images à explorer (3 à 4), et au-delà duquel les avantages du parcours en profondeur de l'arbre deviennent très faibles.

La combinaison du MHT avec les probabilités pignistiques permet de choisir la branche pour laquelle la somme des probabilités pignistiques des feuilles pour le suivi global est plus grande.

2.8. CONCLUSION

On peut retenir de ce chapitre que les théories basées sur les probabilités ont montré leurs limites du fait de leur incomplétude, justifiée par leur incapacité de modéliser aussi bien l'ignorance que le conflit. Les approches basées sur la théorie des ensembles flous et/ou celle des possibilités, semblent être une bonne solution en raison de leur prise en compte de la modélisation des informations imparfaites et de leur richesse en opérateurs de combinaison. Cependant la véritable difficulté avec ces deux méthodes reste comment identifier les critères devant garantir une estimation et le choix judicieux des fonctions d'appartenances ou des distributions de possibilités à utiliser. La théorie de l'évidence semble plus complète et plus adaptée à nos besoins pour peu qu'on arrive à bien définir le cadre de discernement et les fonctions de masse de croyance.

Le TBM (Transferable Belief Model) de Smets [59, 60, 61] qui s'appuie sur les fonctions de croyances de la théorie de l'évidence de Dempster-Shafer nous a semblé plus intéressant d'autant plus que dans sa démarche il fait la séparation entre la phase de représentation et la combinaison des données, et celle de la prise de décision. En associant au TBM le MHT, nous comptons mettre en place un couplet gagnant pour notre système de suivi.

CHAPITRE 3 : ARCHITECTURE MULTI-AGENTS

Sommaire

3.1.	Introduction	34
3.2.	Concept d'Agent et de Système Multi-Agents	35
3.2.1.	Définitions d'Agents	35
3.2.2.	Propriétés ou caractéristiques d'un Agent.....	36
3.2.3.	Système Multi-Agents	37
3.2.4.	Catégorie ou classes d'agents	41
3.2.5.	Architecture d'un agent et/ou d'un SMA.....	43
3.2.6.	Interactions et communication	50
3.2.7.	Langages de communication.....	56
3.2.8.	Ontologies	63
3.3.	Conclusion	66

3.1. INTRODUCTION

La modélisation orientée objets et la programmation orientée objets qui en a découlée a été une grande révolution dans le monde informatique. Mais s'il y a quelque chose qui manque aux objets c'est l'autonomie alors que dans la réalité de tous les jours plusieurs entités sont dotées de cette aptitude à décider de leur comportement malgré les sollicitations et les influences de l'environnement. Depuis quelques années les SMA (Systèmes Multi-Agents) ont fait leur apparition pour pallier en cas de besoin cette incomplétude de l'approche orientée objets. Les SMA trouvent leur application dans plusieurs domaines à commencer par celui de l'intelligence artificielle pour s'étendre à presque tous les domaines (systèmes distribués, robotique, vision par ordinateur, médecine, vie artificielle) où on a besoin de résoudre des problèmes complexes qui nécessitent une intelligence collective et une coopération.

Ce chapitre va nous permettre de faire le point sur les Agents, les SMA ainsi que leur architecture, et de comprendre comment ils arrivent à coopérer et à coordonner leurs actions pour atteindre les objectifs assignés à leur organisation ou à leur société tout en restant autonomes. Il est abordé également les différents modes de communication entre les agents et les langages de communication les plus connus dans ce domaine. Ce qui va nous permettre de proposer plus loin au chapitre 5 une architecture logicielle multi-agents pour notre système de suivi.

3.2. CONCEPT D'AGENT ET DE SYSTEME MULTI-AGENTS

3.2.1. DEFINITIONS D'AGENTS

La notion d'agent est tout un concept qui est né depuis des décennies. Dans la littérature, plusieurs travaux ont porté sur les agents et les systèmes multi-agents. Les définitions varient d'un auteur à un autre et d'un domaine d'application à un autre. Toutefois une certaine constante demeure. Ainsi, parmi les définitions les plus consensuelles que nous avons eu à trouver et qui reviennent le plus souvent, on peut citer celles de :

[82][76] Un agent est une entité qui exhibe les caractéristiques suivantes : autonomie, réactivité, proactivité et qui possède une certaine habilité sociale. Définition qui sera reprise par [80].

[75] en a donné une définition générale assez détaillée qui se décline en 9 points à la page 13 de son ouvrage avec une explication précise des différents thèmes et concepts qui rentrent dans le cadre de cette définition. Il poursuivra d'ailleurs en définissant à la page 15 de ce même ouvrage ce qu'il entend par agent purement communicant (ou agent logiciel) en le distinguant de l'agent au sens général, par le fait qu'il ne possède pas de perception des autres agents, que ces tendances prennent l'aspect d'objectifs (ou intentions), qu'il n'agit pas dans un environnement et que son contexte d'évolution est naturellement celui des réseaux informatiques. Cette distinction reste cependant discutable à plusieurs égards par rapport à la notion d'environnement et à celle de perception si on s'en tient déjà à la définition que lui-même il en donne.

Tout ceci montre la dimension et la diversité du vocabulaire des chercheurs suivant leur compréhension et leur interprétation de la notion d'agent. Il faut noter qu'il n'existe pas une contradiction entre les différentes définitions mais plutôt une certaine complémentarité issue de la multitude de domaines d'application. Cette complémentarité peut venir même du même chercheur [75][83].

Pour avoir une synthèse et plus de détail sur les thèmes et définitions on pourra se référer également aux travaux de [77] et à la thèse de Sana Moujahed [78].

Pour compléter ce tableau déjà très exhaustif mais qui n'épuise pas la question relative aux agents, et tenter de donner une définition à nous de ce qu'est un agent, nous dirons que :

Venant du mot latin *agere* qui signifie agir, un agent est une entité autonome, réelle ou abstraite, située dans un environnement ouvert qu'il perçoit de par ses capteurs et dans lequel il peut agir et communiquer de façon rationnelle sur la base d'un certain nombre d'objectifs en vue d'atteindre une satisfaction de façon optimale. Il a la possibilité d'interagir avec d'autres agents au besoin dans le cadre d'une utilité.

[79] nous dira que la rationalité d'un agent lui impose d'être en mesure d'exécuter l'action qui maximise sa mesure de performance en fonction de sa perception de l'environnement, de ses connaissances.

3.2.2. PROPRIETES OU CARACTERISTIQUES D'UN AGENT

Il ressort des différentes définitions d'un agent, qu'un agent doit avoir, suivant les tâches qui lui sont assignées, un certain nombre de propriétés et caractéristiques dont les plus importantes sont :

Autonomie : un agent ne doit pas être dirigé par des commandes venant de l'utilisateur (ou d'un autre agent), mais par un ensemble de tendances qui peuvent prendre la forme de buts individuels à satisfaire ou de fonctions de satisfaction ou de survie que l'agent cherche à optimiser [75]. L'agent est son propre moteur et son autonomie va au-delà d'une autonomie comportementale mais s'étant à ses ressources qui le rendent à la fois dépendant de son environnement mais dans le même temps indépendant vis-à-vis de lui-même.

Les agents encapsulent leur état interne (inaccessible pour les autres agents) et prennent des décisions basées sur cet état interne [82].

Perception : c'est la vision que l'agent a du monde extérieur. C'est sa représentation de son environnement, son entourage. C'est son point de vue. C'est la description qu'elle se fait du monde qu'il ne voit très souvent que partiellement à travers ses capteurs.

Action : les agents sont capables d'agir, et non pas seulement de raisonner comme dans les systèmes d'intelligence artificielle classique. L'action, qui est un concept fondamental pour les systèmes multi-agents, repose sur le fait que les agents accomplissent des actions qui vont modifier leur environnement et donc leurs prises de décision futures. Les agents en générale agissent dans l'environnement, sauf, pour les agents purement communicants pour lesquels toutes les actions se résument à des communications.

Communication : c'est par elle que les agents agrandissent leurs capacités perceptives en leur permettant de bénéficier des informations et du savoir-faire des autres agents. Les communications sont indispensables à la coopération et il est difficile de concevoir un système d'agents coopérants s'il n'existe pas un système permettant aux agents d'échanger des informations ou de véhiculer des requêtes. C'est un vecteur sur lequel les agents peuvent s'appuyer pour coordonner leurs actions, de coopérer à un but commun et de s'intégrer à un groupe d'agents.

Réactivité : aptitude de l'agent à répondre rapidement à de nouvelles informations perçues de son environnement.

Les agents sont situés dans un environnement (monde physique, internet, ...) et sont capables de percevoir cet environnement. De plus ils peuvent évoluer en fonction de changements qui interviennent dans cet environnement [82].

Proactivité : d'après certains auteurs dont [82][81], les agents ne doivent pas se limiter à agir ou réagir en réponse à leur environnement. Ils doivent être capable d'explorer des

opportunités, avoir des comportements orientés but/utilité, et prendre des initiatives là où c'est approprié.

Social : il s'agit d'une propriété centrale des systèmes multi-agents. Les agents doivent être en mesure de communiquer et de coopérer avec d'autres agents et les humains pour compléter la résolution de leur propre problème et d'aider les autres dans leurs activités.

Au-delà de ces propriétés ci-dessus citées d'autres propriétés secondaires telles l'adaptabilité, la mobilité, l'aptitude à apprendre (l'apprentissage) et l'intelligence.

Le tableau 3.1 ci-dessous présente une synthèse de certaines caractéristiques d'un agent.

Caractéristique	Signification
Actif	Capable d'agir et pas seulement de raisonner
Autonome	Exerce un contrôle interne sur ses propres actions Indépendance agents et ressources (énergie, CPU, mémoire, infos...) Différence avec objet, module et processus
Situé dans un environnement	Capable de percevoir l'environnement (souvent partiellement) et d'agir dessus
Interactif, communiquant	Interagit (communique) avec les autres agents (humains ou artificiels)
Continu dans le temps	C'est un processus qui est continuellement en exécution
Réactif	Répond dans un temps déterminé à des changements de l'environnement
Orienté but (pro-actif)	Ne se contente pas de réagir à des changements de l'environnement ; il anticipe les actions pour atteindre un but précis.
Coopératif	Il coopère avec d'autres agents pour accomplir des tâches
Flexible, reconfigurable	Les actions ne sont pas écrites à l'avance
Adaptatif, Apprenant	Change son comportement en se basant sur ses expériences passées
Mobile	Capable de se transporter d'une machine à une autre.
Intelligent	Capable d'initier des inférences et des raisonnements sur ces connaissances propres et celles des autres
Caractère	A une « personnalité » et un « état émotionnel »

Tableau 3.1 : Les caractéristiques d'un agent

3.2.3. SYSTEME MULTI-AGENTS

Ayant fait le tour de la notion d'agent pour en cerner les contours, on peut retenir qu'un agent tout en étant autonome est destiné à accomplir une tâche ou une mission pour le compte d'un système et dans un environnement donné. La notion de SMA remonte aux années 80 et

se situe à l'intersection de celle de l'intelligence artificielle distribuées (IAD) et de celle de la vie artificielle, à partir de l'enrichissement du concept d'échange d'informations par l'élaboration de l'idée du tableau noir avec le projet HERSAY II [Erman et al. 1980]. Ce concept a évolué au fil des années et aujourd'hui, de par son efficacité, trouve son application dans presque tous les domaines en partant des systèmes mono-agent aux systèmes multi-agents. Un système qui est composé d'un seul agent sera appelé système mono-agent. A l'opposé de cela, tout système composé de plus d'un agent est appelé système multi-agents.

Toutefois, suffirait-il d'avoir plusieurs agents dans un système pour parler d'un système multi-agents, surtout du point de vue cohérence, efficacité et optimalité ? A quoi cela servirait-il d'avoir plusieurs agents dans un système si c'est pour avoir une indépendance comportementale des uns et des autres ou des uns envers les autres ? Un essai de réponse à ces questions nous aidera à donner une première définition de ce qu'on peut réellement entendre par système multi-agents. Cela nous permettra également de faire ressortir, dans cette partie de notre rédaction, les aspects fondamentaux permettant de distinguer un ensemble de systèmes mono-agents d'un système multi-agent.

Essayons de considérer un système qui est formé de plusieurs agents totalement indépendants les uns des autres du point de vue comportement. Autrement, un système dans lequel les actions des uns n'influencent pas sur celles des autres et ne sont pas coordonnées pour aboutir à un résultat commun. Alors, pourquoi c'est un système ? En effet, dans le dictionnaire Larousse on peut trouver plusieurs définitions du mot système. En voici quelques-unes.

« Ensemble organisé de principes coordonnés de façon à former un tout scientifique ou un corps de doctrine. »

« Ensemble d'éléments considérés dans leurs relations à l'intérieur d'un tout, fonctionnant de manière unitaire. »

« Ensemble de procédés, de pratiques organisées, destinés à assurer une fonction définie. »

« Ensemble de structures cohérentes dont les éléments constitutifs, même détachés de leur contexte, en conservent la référence et restent indispensables à son bon fonctionnement, à son développement, ainsi qu'à son décodage »

En revenant à notre hypothèse de départ, on se rend compte qu'un problème se pose puisque si on s'en tient aux différentes définitions ci-dessus, on voit bien qu'un minimum de relations ou de coordination s'imposent aux différents éléments du système pour en assurer son unité. Si nous devons caractériser notre supposé système, il ne s'agit rien d'autre que d'un ensemble d'agent ou si on veut d'un ensemble de systèmes mono-agent.

3.2.3.1. Quelques définitions

Définition

Un Système Multi-Agents (SMA) à notre sens est tout au moins un système formé d'agents de même type ou non, qui coopèrent et interagissent de façon coordonnée dans le souci d'atteindre un but ou un objectif commun dans un environnement dont ils ont la même perception ou point de vue ou des perceptions ou points de vues complémentaires.

Plusieurs définitions plus élaborées d'un système multi-agents existent dans la littérature.

Définition [84][85][86]

Un système multi-agent est un système distribué composé d'un ensemble d'agents. Contrairement aux systèmes d'IA, qui simulent dans une certaine mesure les capacités du raisonnement humain, les SMA sont conçus et implantés idéalement comme un ensemble d'agents interagissant, le plus souvent, selon des modes de coopération, de concurrence ou de coexistence.

Définition [75]

On appelle système multi-agent (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 ($A \subseteq O$), lesquels représentent les entités actives du système.
4. Un ensemble de relations R qui unissent des objets (et donc des agents) entre eux.
5. Un ensemble d'opérations Op permettant aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O .
6. Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, que l'on appellera les lois de l'univers

Définition [87]

Un système multi-agents est un ensemble d'agents qui agissent (et interagissent) dans un environnement commun.

Il peut être :

- **ouvert** : les agents y entrent et en sortent librement ;
- **fermé** : l'ensemble d'agents reste le même ;
- **homogène** : tous les agents sont construits sur le même modèle ;
- **hétérogène** : des agents de modèles différents, de granularité différentes.

Ayant maintenant une idée de ce qu'on peut entendre par un Système Multi-Agents que nous désignons désormais par SMA, il serait intéressant de voir leurs différents domaines d'application pour aboutir à la relation éventuelle qui existe entre les SMA et notre travail.

3.2.3.2. Domaines d'application des SMA

Si l'on se base déjà sur les diverses définitions des SMA, on se rend compte que les SMA permettent d'avoir des points de vue multiples et donc une multi-expertise mise au service de la résolution d'un problème qui est soit décomposé en plusieurs sous problèmes répartis entre plusieurs agents complémentaires, ou soit résolu par chaque agent séparément et de façon concurrente pour une mise en commun des solutions et une synthèse en vue d'apporter une solution optimale. Un SMA peut permettre de distribuer aussi physiquement/géographiquement la résolution d'un problème que d'en faire une distribution fonctionnelle et de tirer profit de chacune de ces approches ou d'une approche mixte selon le modèle mis en place. Ils conviennent aux systèmes ouverts, généralement mal structurés où les approches classiques ne conviennent pas. Ils répondent aux besoins des environnements hautement distribués et dynamiques, à la résolution de problèmes faisant intervenir coopération, interaction et négociation ; ils sont adaptés aux problèmes de simulation et de formation de personnel via l'apprentissage.

Toutes ces différentes aptitudes qu'ont les SMA avec toute la flexibilité qui entoure leur conception et leur mise en œuvre expliquent le fait que leurs domaines d'applications soient multiples et variés. Ils peuvent s'appliquer pratiquement dans presque tous les domaines pour l'assistance, la résolution de problèmes, l'aide à la décision, la simulation.

Nous donnons ci-dessous une liste non exhaustive des domaines d'application des SMA.

- Télécommunications, systèmes de transports, réseaux
- Travail collaboratif assisté par ordinateur,
- Médecine,
- Robotique,
- Télématicque (Internet),
- Simulation de systèmes complexes,
- Systèmes multi-capteurs,
- Simulation.

Les SMA peuvent donc permettre la modélisation et même la simulation de systèmes multi-capteurs. C'est à ce titre que nous portons notre intérêt sur eux dans le cadre de nos travaux qui portent sur la gestion haut niveau de systèmes multi-agents pour le suivi à base de caméras co-localités et de caméras distants. Un outil pour tester l'efficacité de nos modèles de suivi serait la mise en place d'un simulateur à base d'agents hétérogènes pour reproduire un système de suivi alimenté par des données vidéo réelles du terrain.

Mettre en place un tel système demande beaucoup de rigueur dans la démarche et nécessite un minimum de connaissance en modélisation et simulation de systèmes multi-agent. Cette problématique n'étant pas l'objet principal de ce travail, nous nous contentons dans les lignes qui vont suivre de donner seulement un bref aperçu des différentes composantes d'un agent ou d'un SMA, leurs mécanismes de communication et d'interaction, puis quelques impératifs

pour la bonne conduite d'un tel projet. Nous reviendrons plus tard, dans nos choix, sur les aspects de modélisation et de simulation.

3.2.4. CATEGORIE OU CLASSES D'AGENTS

On distingue plusieurs types d'agents qui se différencient les uns des autres de par leurs caractéristiques, leur mode de communication et d'action.

Même s'il existe une multitude de types d'agents, on peut cependant les regrouper en deux grandes catégories [75], les autres catégories n'étant que des sous catégories ou des déclinaisons d'une de ces deux catégories. Ces deux catégories fondamentales sont constituées des agents réactifs et des agents cognitifs.

3.2.4.1. Les agents réactifs

Les agents réactifs sont les plus sommaires. Ils ont un comportement du type « stimulus – réponse ». L'agent réactif ne possède pas une représentation complète de son environnement et n'est pas capable de tenir compte de ses actions passées.

Les systèmes multi-agents constitués uniquement d'agents réactifs possèdent un grand nombre d'agents. La convergence du comportement de l'ensemble des agents vers un état décisionnel stable n'est pas forcément assurée, et si un état stable est atteint, il n'est pas toujours évident qu'il s'agisse de la solution optimale.

3.2.4.2. Les agents cognitifs

Les agents cognitifs sont plus évolués. Ils sont le résultat direct des recherches menées dans le domaine de l'intelligence artificielle. Les agents cognitifs ont une représentation globale de leur environnement et des autres agents avec lesquels ils communiquent. Ils savent tenir compte de leur passé et s'organisent autour d'un mode social d'organisation.

Les systèmes multi-agents constitués uniquement d'agents cognitifs sont constitués d'un nombre assez faible d'agents. Ils réclament des ressources plus importantes que les systèmes d'agents réactifs. La convergence du système vers un état décisionnel stable n'est pas non plus assurée par l'utilisation de ce type d'agents, mais ils permettent de résoudre des problèmes plus complexes et nécessitant une plus grande abstraction.

Nous présentons ci-dessous sur la figure 3.1, une liste non certainement exhaustive de noms d'agents qu'on rencontre dans la littérature mais il ne nous sera pas possible d'en faire une classification sur la base des deux catégories préalablement citées.

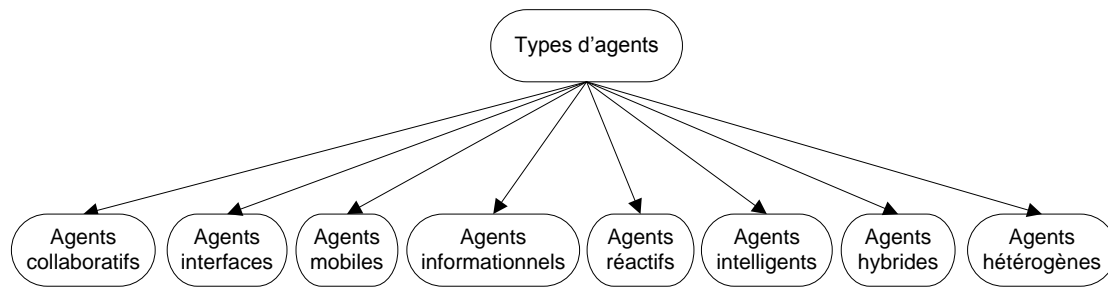


Figure 3.1 : Différents types d'agents d'après [124]

En effet, on se rend bien compte, à la lecture de cette figure que plusieurs agents peuvent être aussi bien cognitifs que réactifs. Alors, trouvant cette catégorisation cognitif/réactif parfois trop simpliste, [75] va proposer (pages 20 et 21) un affinement sur la base de deux axes ou critères de comportement. Les agents, quel que soit leurs catégories, sont par définition mus par des tendances. Mais, certaines de ces tendances peuvent provenir de l'environnement, alors que d'autres sont explicitement exprimées dans les agents. On parlera de comportements "réflexes" dans le premier cas et de comportements "téléonomiques" dans le second cas. Ces deux critères de comportement (téléonomique et réflexe) peuvent donc aider à mettre en place un second niveau de dissociation des agents.

Un croisement de ces deux axes de catégorisation des agents va conduire à l'obtention de quatre nouvelles catégories d'agents en fonction de leurs capacités de représentation du monde et de leurs modes de conduite.

Relation au monde / Conduites	Agent Cognitif	Agent Réactif
Téléonomique	Agent intentionnel	Agent pulsionnel
Réflexe	Agent « module »	Agent tropique

Tableau 3.2 : Catégories d'agents suivant fonctionnement et représentation du monde

A partir de cette nouvelle décomposition, il devrait être possible de catégoriser plus facilement chaque type d'agent. Surtout qu'on en aura besoin dans la définition des fonctionnalités de chacune des couches de l'architecture de l'agent. L'architecture d'un agent étant la caractéristique de sa structure interne, représente le principe d'organisation qui soutient l'agencement de ses différents composants.

Il existe cependant, toujours dans la littérature, une autre forme de regroupement toujours en 4 types ou catégories à savoir :

- agent simple réflexe ;
- agent réflexe avec états internes ;
- agent basé sur buts ;
- agent basé sur l'utilité.

Pour notre part, après exploitation des explications des différents courants et différentes écoles, nous pouvons prétendre qu'il s'agit de la même chose sous différentes appellations.

3.2.5. ARCHITECTURE D'UN AGENT ET/OU D'UN SMA

En se référant aux diverses définitions du concept d'agent et faisant une synthèse, on en vient à dire qu'un agent est caractérisé par ce qu'il sait faire et comment il sait le faire. Est-ce par réaction ou après réflexion ? En d'autres termes, ce qui va permettre de caractériser un agent, c'est l'ensemble de ces actions et la manière dont il est conçu pour arriver à agir comme il sait le faire. On parlera alors de son comportement et de son organisation interne ou architecture.

L'architecture d'un agent est l'émanation de son concepteur et représente les différentes parties ou composants de l'agent ainsi que leur agencement pour lui permettre d'agir dans le sens de la satisfaction d'un objectif donné. Il s'agit donc de la structure interne de l'agent et elle est fonction des choix d'implémentation de son concepteur. De ces choix, dépendra la catégorie de l'agent. L'architecture d'un agent est une composante fondamentale dans sa conception puisque c'est elle qui représente le centre nerveux de l'agent.

Mettre en place l'architecture d'un agent n'est donc pas chose facile. Toutefois, en partant d'un certain nombre de constats de base on peut en arriver à cerner les contours et aboutir à des propositions d'architectures.

La notion d'architecture fait référence à une notion d'organisation interne et dans une organisation, malgré l'autonomie et les objectifs personnels de chaque élément, il y a une discipline de groupe. Chacun des membres est donc astreint à jouer sa partition de façon solidaire avec les autres membres pour atteindre un but dans une activité convergente. Chaque composant d'une organisation est donc un acteur qui doit jouer un rôle. Il y a généralement plusieurs rôles à répartir aux différents acteurs mais il n'est pas exclu que certains acteurs soient programmés pour jouer plusieurs rôles en fonction du contexte.

Pour Edgar Morin [96] « Une organisation peut être définie comme un agencement de relations entre composants ou individus qui produit une unité, ou système, dotée de qualités inconnues au niveau des composants ou individus. L'organisation lie de façon interrelationnelle des éléments ou des événements ou individus divers qui dès lors deviennent des composants d'un tout. Elle assure solidarité et solidité relative, donc assure au système une certaine possibilité de durée en dépit de perturbations aléatoires. »

[125, 126] définira une architecture d'agent comme « une méthodologie particulière pour la construction d'agents » qui « spécifie comment l'agent peut-être décomposé, dans la construction, en un ensemble de modules et comment ces modules doivent être faits pour agir réciproquement. L'ensemble total de modules et leurs interactions doit fournir une réponse à la question de comment les données et l'état interne actuel de l'agent déterminent

l'action et l'état interne futur de l'agent. Une architecture globale des techniques et les algorithmes qui soutiennent cette théorie.

La notion d'organisation est valable pour comprendre aussi bien l'élaboration de l'architecture d'un agent (micro-organisation) que celle d'un système multi-agent qui peut être vu comme un macro-agent (macro-organisation). En matière d'organisation au sein d'un agent ou d'un SMA, plusieurs travaux ont été réalisés par des chercheurs, pour déterminer les différents rôles des agents ou composants dans l'architecture d'une organisation ainsi de les différentes fonctions nécessaires à une organisation pour qu'elle puisse s'adapter à son environnement et répondre à ces finalités.

Deux grandes pensées s'affrontent pour retenir les fondamentaux de l'architecture d'un agent. Il s'agit une fois encore de celle des chercheurs de l'école d'agents cognitifs et celle des chercheurs de l'école d'agents réactifs. Les premiers pensent que les agents d'un SMA n'ont pas besoin d'avoir une intelligence individuelle pour former un système cohérent avec un comportement d'ensemble intelligent leur permettant de résoudre des problèmes complexes. Les seconds par contre estiment que c'est la qualité cognitive des agents d'un système qui lui confère son intelligence et son aptitude à résoudre de façon optimale un problème en n'utilisant généralement très peu d'agent et de types différents.

De par ces différentes réflexions et constats, Ferber [75] va proposer à partir d'une analyse fonctionnelle, cinq différents rôles des composants ou agents et six différentes fonctions nécessaires à une organisation. Les tableaux 3.3 et 3.4 ci-dessous en font une présentation succincte.

Rôle	Description
Médiateur	S'occupe de la gestion des demandes d'exécution et de la répartition des tâches suivant les compétences
Planificateur	Détermine les actions à entreprendre
Décideur	Choisit parmi un ensemble d'actions possibles, celle à entreprendre soit sur la base d'une priorité ou sur celle d'un critère de performance, de but ou d'utilité.
Fournisseur	Rend un service à un autre composant ou agent considéré dans ce schéma comme un client.
Exécutant	Il a pour fonction principale d'exécuter une action lorsqu'il en reçoit l'ordre

Tableau 3.3 : Présentation des rôles des composants ou agent d'une organisation

Fonction		Description
Représentationnelle		Elle se charge de la modélisation de l'environnement et des autres organisations par un ensemble de données et de représentations. Elle est dotée d'une fonction de mémorisation.
Organisationnelle		Elle est responsable de la planification, de l'allocation, du suivi, de la coordination et de la gestion des engagements de chaque composante ou agent de l'organisation : Scheduling des tâches et satisfaction des buts.
Motivationnelle	Conative	Elle se charge de la gestion de l'action vers l'extérieur (génération de buts, tendances, contrats) et se rapporte aux sources, aux limites et à la sélection des activités des organisations. C'est la fonction décideur.
Elaboratrice des contraintes		
Décisionnelle		
Perceptive	Interactionnelle	Elle a en charge l'établissement du lien entre l'organisation et ce qui l'entoure. Elle gère donc l'interaction avec l'environnement ainsi que d'autres agents.
Exécutive		
Productive ou opératoire		Elle a en charge l'ensemble des activités primitives à mettre en œuvre pour la résolution d'un problème ou la réalisation d'une tâche.
Végétative ou Conservatrice		Elle assure la conservation de la structure et des agents, l'acquisition et l'entretien des ressources et tâches pour maintenir et reproduire éventuellement l'organisation. C'est la fonction de survie.

Tableau 3.4 : Différentes fonctions principales d'une organisation

Les relations qui peuvent exister entre ces différentes fonctions sont illustrées dans la figure 3.2 ci-dessous. Il faut noter que sur cette figure, on voit apparaître une fonction perceptive et une fonction exécutive. Ces deux fonctions ne sont que les composantes de la fonction interactionnelle qui assure la fonction d'entrée-sortie ainsi que celle d'interface. Suivant qu'un agent est réactif ou cognitif, certaines de ces fonctions peut ne pas être réellement implémentés.

En analysant les rôles et les fonctions que peuvent être amené à assumer un agent ou un SMA, on réalise que prendre en compte toutes ces contraintes et exigences dans leur conception représente un véritable défi à relever. Les problèmes inhérents à la conception et à l'implémentation d'un agent ou d'un SMA sont multiples et la mise place de l'architecture de l'agent doit intégrer des réponses à un ensemble de questions que se sont posées plusieurs chercheurs [75], [87].

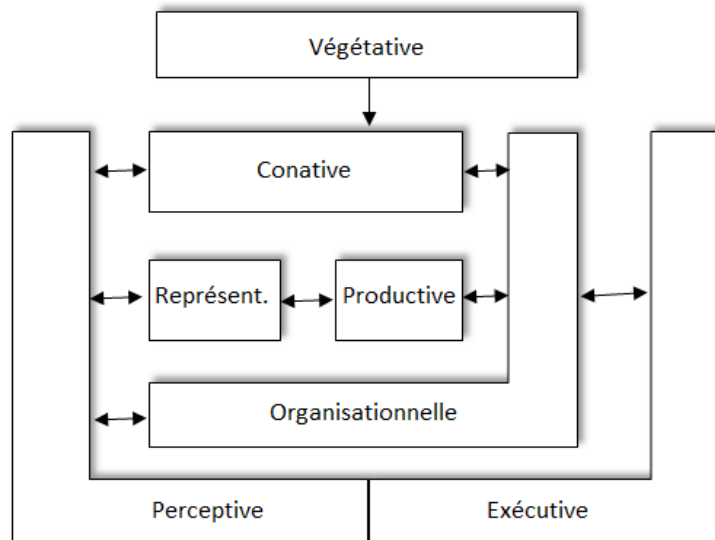


Figure 3.2 : Différentes fonctions d'une organisation.

- Comment réaliser les fonctions d'un agent ou d'un SMA ?
- Composant ou agent va jouer quel rôle ou occuper quelle place ?
- Pourquoi, Quand et Comment les différents composants ou agents vont-ils échanger, communiquer, coordonner leurs actions ?
- Quelles sont les capacités, les possibilités d'agent ou de l'organisation, et comment les utiliser pour agir sur le monde ?
- Comment formuler, décrire, décomposer, et allouer les problèmes et synthétiser les résultats ?
- Comment assurer que les agents agissent de manière cohérente en prenant leurs décisions ou actions, ou en gérant les effets non locaux de leurs décisions locales et en évitant les interactions nuisibles ?
- Comment permettre aux agents individuels de représenter et raisonner sur les actions, plans et connaissances des autres agents afin de se coordonner avec eux ?
- Comment raisonner sur l'état de leurs processus coordonnés (comme l'initialisation ou la terminaison) ?
- Comment reconnaître et réconcilier les points de vue disparates et les intentions conflictuelles dans un ensemble d'agents essayant de coordonner leurs actions ?
- Comment trouver le meilleur compromis entre le traitement local au niveau d'un seul agent et le traitement distribué entre plusieurs agents (traitement distribuée qui induit la communication) ? Plus généralement, comment gérer la répartition des ressources limitées ?
- Comment éviter ou amoindrir un comportement nuisible du système global, comme les comportements chaotiques ou oscillatoires ?

L'analyse fonctionnelle d'un agent ou d'un SMA peut se faire suivant plusieurs dimensions pour affiner ou préciser son architecture. Ferber [75] déclinera cette analyse selon les cinq dimensions que sont : la dimension physique, la dimension sociale, la dimension relationnelle,

celle environnementale, et celle personnelle. Cette décomposition de l'analyse fonctionnelle va permettre de mettre en place un tableau dans lequel on pourra savoir ou non si on doit implémenter explicitement ou non une dimension donnée pour une fonctionnalité donnée. Mieux, en utilisant différents types de symboles (trois principalement) pour marquer les cellules correspondants aux fonctionnalités à prendre en compte ou non, on sera même en mesure de préciser leur niveau de prise en compte (obligatoire, optionnelle, absence). Le tableau 3-5 ci-dessous présente un modèle de ce tableau.

Dimension \ Fonctions	Physique	Sociale	Relationnelle	Environnementale	Personnelles
Représentationnelle					
Organisationnelle					
Conative					
Interactionnelle					
Productive					
Conservative					

Tableau 3.5 : Grille d'analyse fonctionnelle des organisations

Symboles de remplissage des cases : On retiendra les trois symboles qu'on voudra mais en prenant la précaution d'accompagner le remplissage de la grille d'une légende explicative.

Exemple de légende :

☒ : obligatoire ☑ : optionnelle □ : Absente

En dehors de l'analyse fonctionnelle d'une organisation, deux autres facteurs doivent être pris en compte pour relever le défi de la définition de l'architecture d'un agent ou d'un SMA. Il s'agit de l'analyse structurelle et des paramètres de concrétisation de l'agent.

L'analyse structurelle va permettre de définir l'ordonnancement des différentes interactions possibles entre les agents en faisant ressortir les relations abstraites qu'il y a entre eux, et la manière dont ces relations évoluent dans le temps. En d'autres termes, il s'agira de décrire les formes d'interaction (statiques ou dynamiques) auxquelles on peut assister entre les rôles des agents ; de voir quelles structures de subordination il y a entre les agents du point de vue de leur dépendance (maitre/esclave) ; d'envisager les modes de couplage (fixe, variable, évolutif) qu'il peut y avoir entre les composants ou les agents ; enfin de retenir la façon dont l'organisation sera constituée a priori ou a postériori.

Les paramètres de concrétisation d'un agent sont des paramètres qui sont à la fois relatives à sa spécialisation du point de vue compétences personnelles vis-à-vis des autres membres d'une organisation, que par rapport à sa multiplicité ou redondance dans l'organisation. Un compromis devra être trouvé pour ne pas avoir plus qu'il ne faut d'agent multidisciplinaire ou

d'agents hautement spécialisés. Ce qui pose un problème d'optimisation de la valeur d'une fonction de performance organisationnelle $\mu_S(R, H)$, en fonction de la redondance et du degré de spécialisation des agents.

Pour la définition de cette fonction de performance, un certain nombre d'hypothèses devront être posées.

- Tout agent ou composant qui appartient à une organisation a tout au moins une compétence, un rôle.
- Tout travail ou un problème P, au sens d'une organisation, sera considéré comme une tâche non élémentaire ; donc nécessitant plus d'une compétence pour sa réalisation.

Partant de ces deux hypothèses, et en retenant les notations suivantes :

C^p : ensemble des compétences nécessaires à la réalisation du travail P

C_a : ensemble des compétences dont dispose un agent ou un module

$C_a^p = C_a \cap C^p$: ensemble des compétences utiles à un agent ou un module dans sa participation à l'accomplissement d'un travail P.

P_c : nombre d'agents ou de modules disposant de la compétence c.

On définira le degré de spécialisation H_a^p d'un agent par :

$$H_a^p = \frac{\text{card}(C^p) - \text{card}(C_a^p)}{\text{card}(C^p) - 1} \quad (3-1)$$

De même, on définira le degré de multiplicité ou de redondance R^c d'un agent ou d'un composant dans une organisation par :

$$R^c = \frac{P_c - 1}{\text{card}(C) - 1} \quad (3-2)$$

Une généralisation de ces deux formules (3-1) et (3-2) à un système multi-agent donne respectivement :

$$H^p = \frac{1}{\text{Card}(A)} \sum_{a \in A} H_a^p \quad (3-3)$$

$$R = \frac{1}{\text{Card}(C)} \sum_{c \in C} R^c \quad (3-4)$$

Un exemple d'illustration des relations pouvant exister entre la redondance et la spécialisation des agents ainsi qu'une représentation de l'évolution de la fonction de performance en

fonction de ces deux paramètres est observable sur la figure 3.3 ci-dessous. Il sera possible de déduire à partir de cette figure la valeur optimale de la fonction de performance et ainsi trouver le compromis entre agents spécialisés et redondance d'agents non forcément spécialisés.

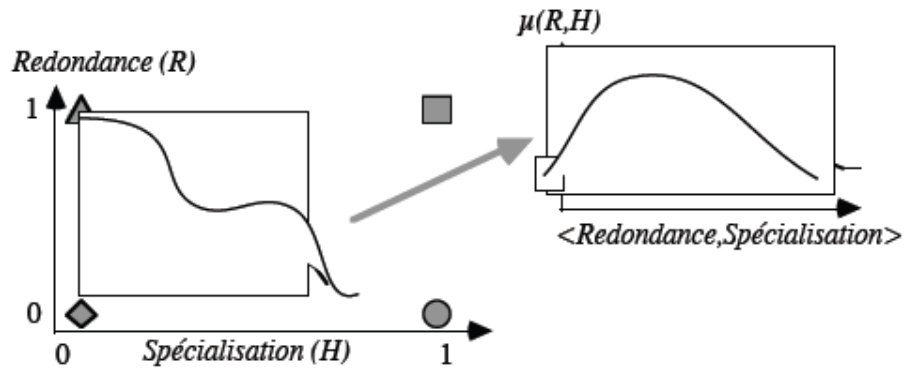


Figure 3.3.: Caractéristiques fonctionnelles, structurelles et de concrétisation

La figure 3.4 ci-dessous présente un résumé de l'ensemble des caractéristiques fonctionnelles, structurelles et de concrétisation à prendre en compte dans la conception de l'architecture d'un agent ou d'un SMA.

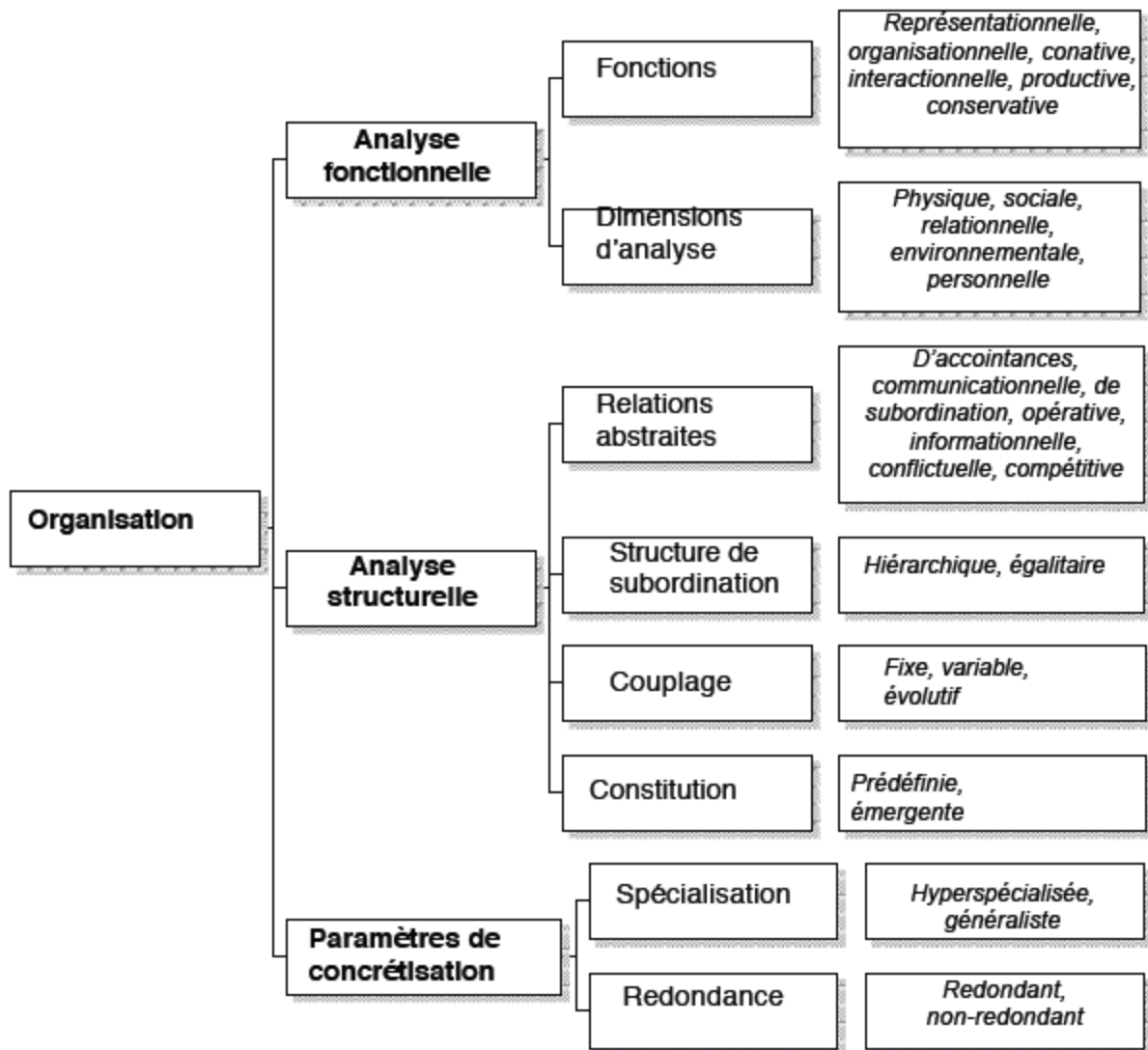


Figure 3.4 : Caractéristiques de conception de l'architecture d'un agent ou d'un SMA

Malgré tout ce travail, Il n'y a pas de standard et il n'y a pas non plus d'approche exclusivement calquée sur une démarche. Ceci explique l'existence de plusieurs architectures :

Architecture modulaire horizontale

- Les architectures à base de tableaux noirs
- L'architecture de subsomption
- Les tâches compétitives
- Les systèmes de production
- Les systèmes à base de classifieurs
- Les architectures connexionnistes
- Les architectures à base de système dynamique
- Architectures multi-agents et acteurs

3.2.6. INTERACTIONS ET COMMUNICATION

3.2.6.1. Interactions

Ferber [75], définit à la page 63 de son ouvrage, l'interaction comme étant la mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques. On retiendra également de ces travaux que les interactions constituent une suite d'actions et de réactions qui peuvent influencer le comportement futur des agents. Une interaction implique d'une certaine manière un contact, soit direct entre agents ou indirect via d'autres agents ou l'environnement ; et est à la fois une conséquence d'action(s) et un élément nécessaire à la constitution d'organisations sociales.

Comme souligné dans [88], Boissier reprendra la définition de Ferber et y apportera une certaine précision pour attirer notre attention sur le fait qu'on ne parlera d'interaction que lorsque la dynamique propre d'un agent est perturbée par les influences des autres. Donc si un agent agit sur l'environnement et que cette action influe finalement sur le comportement d'un autre agent, c'est qu'il y a eu une interaction entre ces deux agents de façon indirecte via l'environnement puisque l'action de l'un a engendré la réaction de l'autre.

Il existe donc plusieurs formes d'interactions et ces dernières nécessiteront ou non des échanges ou communications directes ou non dépendamment du système et des agents en présence (réactifs, cognitifs, hétérogènes ou hybrides).

Ainsi, selon qu'il y a compatibilité entre leurs buts, leurs capacités ou leurs moyens, les interactions qu'il y a entre agents peuvent différer. Dans un système multi-agents les différentes possibilités d'interaction qui peuvent exister sont : la coexistence, cohabitation, la compétition, la négociation, la collaboration et la coopération ; les unes pouvant intervenir dans la réalisation des autres dans certains cas.

La coexistence et la cohabitation sont des formes très rares dans les systèmes multi-agents. On les rencontre souvent dans des systèmes qu'on peut appeler des collections d'agents. Dans ces genres de systèmes les agents travaillent séparément et n'ont pas de but ou d'objectifs communs à atteindre. Ils ont à la rigueur juste besoin de s'éviter ou de tenir compte de l'existence de l'autre dans leur environnement. Par conséquent, s'il doit y avoir une interaction entre ces agents celle-ci ne se manifestera qu'à travers l'environnement.

Lorsque des agents sont en compétition, ils sont en concurrence et donc peuvent partager les mêmes objectifs ou les mêmes ressources pour atteindre des objectifs communs ou divergents. Ces situations peuvent être source de conflits et engendrer une interdépendance négative voire chaotique pour l'ensemble du système. Les interactions dans ce cas peuvent être aussi bien directes qu'indirectes.

La collaboration et la coopération sont deux facettes très proches des notions de partage de connaissance et de tâches par une interdépendance positive qui participe à la performance du groupe. Tout en étant très proches, ces deux notions ne sont cependant pas exactement les mêmes. La coopération qui veut dire opérer ensemble est un processus linéaire de confiance dans lequel les tâches sont partagées et où on procède par juxtaposition ou addition

(on ajoute X_1, X_2, \dots, X_n) pour obtenir la réalisation d'un objectif commun Y , connu d'avance. Dans une pareille interaction il existe très peu de conflit du fait de la répartition des tâches. Les interactions ont lieu souvent lors de l'organisation de la répartition des tâches, de la coordination des tâches et lors de la mise en commun des résultats dans une suite logique. La collaboration qui veut dire quant à elle, travailler ensemble, est processus non linéaire de consensus et de confiance pour converger vers une compréhension partagée d'un résultat qui n'est pas forcément connu à l'avance. On collabore pour échanger, partager et arriver à produire de nouvelles connaissances ou élaborer de nouvelles solutions à un problème commun. Dans une telle forme d'interaction, les situations conflictuelles ne manquent pas pour arriver au consensus. La coordination joue un rôle très important dans la collaboration. Elle est d'ailleurs nécessaire pour toutes sortes d'activités collectives. C'est elle qui permettra de garantir la cohésion du groupe et la cohérence nécessaire pour atteindre les objectifs communs.

Thomas Malone et Krevin Crowston [105] définissent la coordination comme l'acte de gérer les interdépendances des différentes activités exécutées pendant la réalisation d'un but. Les interdépendances regroupent les prérequis, le partage de ressources et la simultanéité.

Suivant cette définition, la coordination recouvre les indices de coopération se rapportant au partage des ressources, à la coordination des actions et à la parallélisation des actions. Deux moyens de mettre en œuvre la coordination peuvent être différenciés la coordination par plans et la coordination par rétroaction [106].

La coordination par plans est basée sur des conventions préétablies, elle détermine quelles sont les tâches à effectuer, quand et par qui, et s'applique à des situations stables et prévisibles. La coordination par rétroaction implique la transmission d'informations non planifiées, elle est adaptée à des situations variables et imprévisibles. La coordination par rétroaction est généralement substituée à la coordination par plans, lorsque surgissent des contingences qui ne s'accordent pas avec les conventions. La coordination nécessite alors des communications pour notifier les divergences par rapport aux conditions planifiées ou prévues, ou bien pour donner des instructions afin que les changements d'activités soient ajustés à ces déviations.

L'un des éléments caractéristiques d'un agent est son autonomie et tous les courants et toutes les écoles s'accordent à reconnaître cet attribut d'un agent. Cette autonomie lui confère le droit de contrôler aussi bien son état que son comportement même si on peut relativiser en prenant en compte la forme de relation abstraite qui lie les agents ; relation qui peut être une relation de subordination statique [75] dans laquelle le fournisseur ne peut refuser la demande du client mais peut cependant en différer sa réalisation sur la base de ses priorités. Cette caractéristique des agents peut être source de conflit et fait appel à la négociation.

La négociation est une discussion dans laquelle des individus intéressés échangent des informations et arrivent à un accord en commun [107]. 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.

Durfee et ses collègues [117] définissent la négociation comme le processus permettant d'améliorer les accords (en réduisant les inconsistances et l'incertitude) sur des points de vue communs ou des plans d'action grâce à l'échange structuré d'informations pertinentes.

La négociation joue donc un rôle fondamental dans les activités d'interaction en permettant aux différents participants de résoudre des conflits qui pourraient mettre en péril des comportements coopératifs ou collaboratifs. Généralement, la négociation est utilisée en intelligence artificielle distribuée et dans les systèmes multi-agents comme un mécanisme pour coordonner un groupe d'agents. Différentes approches ont été développées en s'appuyant sur la riche diversité des négociations humaines dans divers contextes [118], [119], [120]. Suivant le temps qu'elle prend, la négociation peut affecter la performance d'un système. Le temps peut donc jouer un rôle déterminant dans les négociations inter-agents. Krauss [121], [122] a fait des études sur le rôle du temps dans les négociations et s'est bien rendu compte que l'importance qu'un agent accorde au temps peut affecter l'entente finale entre les parties.

Il ressort des définitions ci-dessus, que la communication et la prise de décision sont deux aspects essentiels de la négociation.

3.2.6.2. Communication

Dans un système multi-agent la communication inter-agent est un point essentiel. En effet, bien qu'étant chacun autonome, les agents ont besoin, d'échanger ou de partager des informations (leurs croyances, leurs intentions, leurs désirs, etc.) pour une meilleure coordination de leurs actions dans le cadre de la résolution d'un problème ou juste pour une cohabitation non chaotique.

Comme les interactions, la communication peut-être directe comme indirecte, abstraite ou concrète. Elle peut même être implicite. La communication est un sujet central en informatique en général et particulièrement en programmation systèmes, en systèmes multi-agents et en IAD où des problèmes de synchronisation de processus et des situations d'interférence négative peuvent surgir à tout moment s'il n'y a pas de coordination et une protection des ressources partagées.

Plusieurs chercheurs ont déjà abordé la question et les avis divergent quant à la définition à donner à la notion de communication, tant elle est complexe. Est-ce une fin ou un moyen ?

Une définition générale de la communication est donnée par [89] « La communication est l'ensemble des processus physiques et psychologiques par lesquels s'effectue l'opération de mise en relation de une ou plusieurs personnes – l'émetteur – avec une ou plusieurs personnes – le récepteur -, en vue d'atteindre certains objectifs.»

La communication demande donc une mise en commun d'un certain nombre ressources ou compétences ainsi qu'une prise d'engagement.

Pour certains auteurs, la communication peut être traitée comme une action rentrant dans le cadre de la poursuite d'une intention et par définition on dit que l'action est ce qui modifie l'état global du monde. Mais la question qui se pose est de savoir si une action a toujours pour effet ou conséquence la modification de l'état global du monde ou bien si une action peut quand même ne pas avoir un retour ?

Ferber, dans [75] proposera de voir l'action de façon moins radicale et de la considérer comme une tentative d'influencer le monde pour le modifier selon son désir. L'action sera donc considérée comme un geste, et la conséquence d'un geste peut ne pas toujours suivre l'intention qui l'a causé. Le résultat d'un geste n'est donc pas forcément le but visé par le geste mais le produit de la combinaison de tous les gestes et tentatives en fonction des lois de l'univers dans lequel ces gestes ont été accomplis. Il définira alors la communication comme un système de réponses à la production d'influences produites par les agents.

Il existe donc plusieurs formalismes pour modéliser la notion de communication et ses propriétés. Comme on peut le constater sur la figure 3.5, la communication joue un rôle très important dans la négociation, la coordination, l'allocation de tâches et même la coopération et la collaboration. En effet, toute négociation implique deux éléments essentiels que sont la communication et la prise de décision suite à un accord des parties. Sans communication, il n'y a pas d'action car la communication est un moyen pour accomplir une action. Et sans action il n'y a pas de négociation. La négociation est une forme d'interaction, action et réaction dans une relation d'influence réciproque. De même, sans communication il ne peut pas y avoir d'allocation de tâches qui passe par une négociation et un engagement des parties. La communication permet de faire la coordination de sorte que les interactions n'agissent pas négativement sur le groupe mais qu'au contraire elles participent à une efficacité en limitant ou en gérant les conflits.

En se référant aux différents rôles joués par la communication dans SMA ne serait-ce qu'en se basant sur la figure 3.5 ci-dessus on se rend compte de la pluralité de ses dimensions. Les travaux de [101], [100] et [104] ont permis de faire ressortir quatre dimensions de la communication. Ainsi a-t-on la dimension psychologique, celle sociale, celle structurelle et celle décisionnelle. Les grandes lignes de ces différentes dimensions ont été quelque peu abordées par Thierry Bouron [102] dans sa thèse.

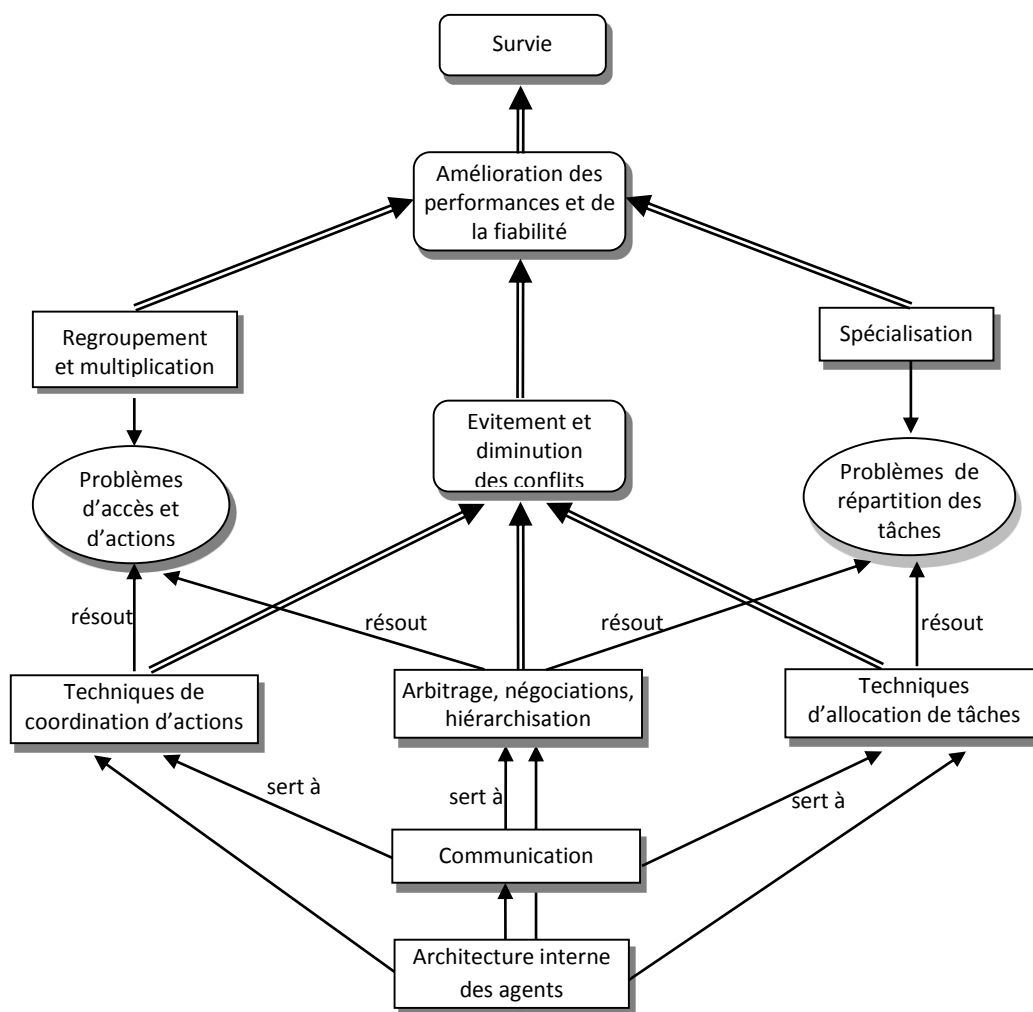


Figure 3.5 : Système des activités de coopération



La nécessité de coopération (le pourquoi ?) entre les agents au sein d'une organisation fait qu'à un moment donné les agents ont besoin de communiquer. Pour s'en rendre compte les agents se basent sur un certain nombre de connaissances ou de faits ou parfois juste un stimulus suivant leur nature. C'est donc la combinaison d'un certain nombre d'états mentaux et ou d'influences qui déclenchent la nécessité de communication chez un agent et répond à la question du quand communiquer ? Et qui montre dans le même temps la dimension psychologique de la communication. Communiquer, c'est mettre en commun et prendre un engagement. D'où la dimension sociale de la communication. Mais avec qui communiquer ?

Sauf en cas de système monolithique, les agents communiquent entre eux et/ou avec leur environnement (ce qui répond à la question : avec qui communiquer ?) ; le tout dans un ensemble d'interactions pour un but bien déterminé et dans le respect d'un ensemble de règles qui régissent l'organisation.

La grande question reste comment définir et représenter les connaissances devant permettre à un agent de se rendre compte qu'il doit communiquer et comment communique-t-il réellement. C'est à ce niveau qu'interviennent les dimensions décisionnelles et structurelles de la communication.

Les agents communiquent soit par messages directs, soit par mémoires partagés ou soit par conventions prédéfinies et partagées, suivant que la communication soit intentionnelle et ou incidente, et suivant le type d'agents en présence.

La communication entre agent nécessite donc un mécanisme ou protocole de transport ou de partage des informations, une stratégie ou un protocole qui régit les interactions, et un langage de communication compréhensible et interprétable par tous les participants. Ce langage doit avoir une syntaxe, une sémantique et d'ontologie.

Dans la partie suivante de ce document, nous allons présenter les langages de communication dans les SMA.

3.2.7. LANGAGES DE COMMUNICATION

Les communications dans les SMA sont à la base de leurs interactions et de leur organisation sociale. Sans communication, l'agent n'est qu'un individu isolé, sourd et muet aux autres agents, refermé sur sa boucle perception-délibération-action [75].

La communication se fait, par messages directs, par mémoires partagées, par conventions prédéfinies, le tout à travers des langages de communication qui utilisent soit une approche basée sur des échanges d'informations entretenus par des directives procédurales, soit une approche déclarative qui modélise la communication par des échanges déclaratifs.

Les langages de communication entre agents ou ACL (Agent Communication Language) en anglais, ont fait l'objet de plusieurs travaux dans la communauté des chercheurs.

La théorie de Claude Shannon [123] a grandement influencée celle de la communication pour qui elle représente un repère. Cette théorie est présentée de façon succincte par la figure 3.6 ci-après :

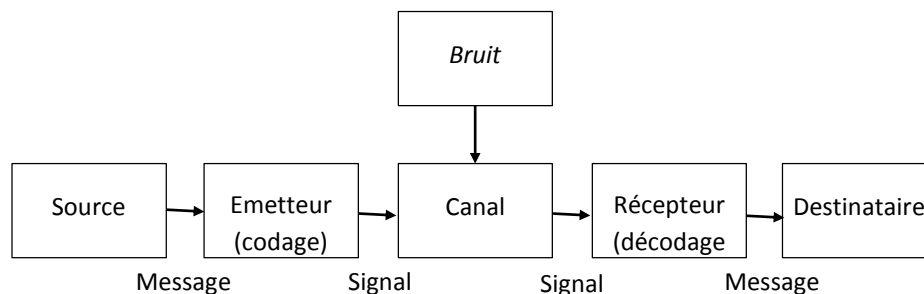


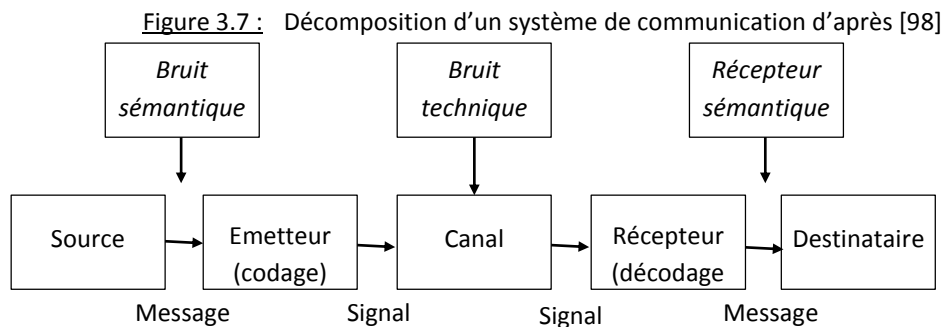
Figure 3.6 : Système de communication d'après Claude Shannon

Émetteur, canal et récepteur = « intermédiaires techniques »

Bruit = toute source d'interférence susceptible de détériorer le signal est donc d'affecter la communication.

- 1) La source d'information énonce un message...
- 2) ... que l'émetteur va décoder et transformer en signal
- 3) Lequel va être acheminé par le canal
- 4) Puis décodé par le récepteur, qui reconstitue un message à partir du signal
- 5) Et le transmet enfin au destinataire.

Warren Weaver dans [98] apportera une amélioration à ce schéma en y introduisant dans un premier temps un bruit sémantique juste après la source pour prendre en compte la possibilité de perturbation ou de déformation du sens du message, puis dans un second temps un récepteur sémantique juste après le récepteur technique pour cette fois-ci donner des caractères sémantique aux mots décodés, conformément aux capacités sémantiques du destinataire.



La communication est ainsi décomposée en un enchaînement de processus dans une structure linéaire.

Les travaux du père de la cybernétique [97] viendront ajouter la notion de rétroaction ou de feedback (positif ou négatif) dans tout processus de communication du fait des interactions, et va rendre la communication non seulement linéaire mais plutôt circulaire.

A ces différentes théories viendra s'ajouter la théorie des actes du langage [127] qui va aborder trois autres aspects (acte locutoire, L'acte illocutoire, acte perlocutoire) de la communication en dehors de celle technique souligné par les travaux de Shannon et Al. Ces nouveaux aspects vont impacter tous les travaux qui vont suivre dans le domaine de la communication et surtout la conception des langages de communication quant à sa syntaxe et de sa sémantique.

En outre, différents travaux sur les langages de communication entre agents ont amené à distinguer trois niveaux dans la structure d'un message :

1. Le niveau du contenu qui est décrit dans un langage de contenu,
2. Le niveau du message qui est défini par le performatif du message, une ontologie, le langage, etc.
3. Le niveau de la communication qui permet de décrire les métadonnées relatives à la communication telles que l'émetteur du message, le destinataire, le protocole utilisé, etc.

Plusieurs langages de communication inter-agents ou ACL (Agent Communication Language) ont alors vu le jour dont les deux les plus célèbres sont le KQML (Knowledge Query and Manipulation Language) du consortium DARPA-KSE (Knowledge Sharing Effort), qui est créé en 1994 mais dont les spécifications n’ont été fournies qu’en 1996 ; et le FIPA-ACL (Fondation for Intelligent Physical Agents – Agent Communication Languages) qui a vu le jour dans les années 2000 des travaux de la fondation FIPA.

3.2.7.1. Le KQML

Le KQML est un langage de communication de haut niveau qui a été développé pour permettre la coopération entre agents cognitifs qu’il considère d’ailleurs disposant chacun d’une base de connaissances virtuelle qui est constituée soit de croyances et/ou de buts. Il opérationnalise la théorie des actes du langage et se veut indépendant de la syntaxe des messages, de la description et de la modélisation des connaissances transmises ou à transmettre (l’ontologie), du mécanisme de transport du message, ainsi que du langage du contenu du message.

La spécification d’un message KQML se fait à l’aide d’un performatif qui représente l’en-tête du message, et d’un certain nombre de couples attribut-valeur.

La syntaxe est la suivante :

```
(<performatif>
  : <attribut 1> <valeur 1>
  : <attribut 2> <valeur 2>
  ...
  : <attribut n> <valeur n>
)
```

La spécification de ce langage a été faite par [90] qui en définit un minimum de 35 performatifs et leur sémantique. Chaque performatif appartient à une catégorie ou type de message et donc l’expression d’un acte du langage. On distingue plusieurs catégories ou classes d’actes du langage. Les *assertifs* permettent de donner une information, les *directifs* servent à donner des ordres, les *promissifs* permettent de s’engager sur une action à venir, les *expressifs* servent à exprimer les sentiments ou les croyances, les *déclaratifs* sont utilisés pour énoncer un fait, les *interrogatifs* servent à demander une information.

Aussi bien les performatifs que les attributs de KQML sont des mots-clés réservés. Les deux tableaux 3.6 et 3.7 ci-dessous présentent quelques performatifs ainsi que quelques attributs.

Performatif	Catégorie	Signification
achieve	directifs	A veut que Y accomplisse une action
Ask-if	interrogatif	L’agent émetteur demande à l’agent récepteur s’il considère que le contenu du message est vrai

Advertise	déclaratif	A indique aux autres agents qu'il peut faire P
ask-one	interrogatifs	A veut une réponse de B à une question
break	déclaratifs	A veut supprimer une communication
sorry	assertifs	A ne peut fournir une réponse plus informative
error	expressifs	A considère que le message de B est mal formé

Tableau 3.6 : Quelques performatifs de KQML

A : agent Expéditeur, B : agent Destinataire, BC : Base de Connaissances d'un agent

Attribut	Signification
:sender	Nom de l'émetteur du message
:receiver	Nom du récepteur du message
:from	l'origine de la performative dans :content quand un forward est utilisé
:to	la destination finale de la performative dans :content quand un forward est utilisé
:in-reply-to	le label attendu dans une réponse à un précédent message (de même que la valeur :reply-with du message précédent)
:reply-with	le label attendu dans une réponse au message courant
:language	Langage dans lequel le contenu est exprimé
:ontology	Ontologie sous-jacente nécessaire pour comprendre le contenu du message
:content	Contenu du message

Tableau 3.7 : Quelques attributs utilisés dans les performatifs de KQML

Parmi les performatifs de KQML, il y en a qui sont destinés aux facilitateurs qu'on peut considérer comme des entités chargés de distribuer des meta-informations aux autres agents et d'offrir des services de communication tels que la retransmission et la distribution des messages (forwarding and broadcasting), la découverte des ressources, le routage basé sur le contenu du message, les appariement.

Les facilitateurs peuvent être des agents intelligents ou simplement des « pages jaunes ».

KQML est théoriquement un langage à trois (3) couches à savoir :

- la couche contenu, au niveau de laquelle les agents doivent s'accorder sur le langage de contenu (KRSL, KIF, LOOM, prolog, Clips, ...) et l'ontologie à utiliser ;
- la couche message, qui est relative au codage du message qu'un agent souhaite transmettre à un autre. Elle prend en compte aussi bien l'expression de l'acte de langage à accomplir, que la description d'une partie des connaissances (données) ;
- la couche communication qui est relative aux informations garantissant l'acheminement des échanges de « packages » entre l'expéditeur et destinataire(s). C'est en quelque sorte l'enveloppe autour du message.

Un exemple d'échange de messages KQML

```

(stream-about
  :sender A1
  :receiver A2
  :reply-with hpj
  :language KIF : ontology imprimantes
  :content HP-Jet
)

(tell
  :sender A2 :receiver A1
  :in-reply-to hpj
  :content (= (resolution HP-Jet) (scalar 300 dpi))
  :language KIF : ontology imprimantes
)

(tell
  :sender A2 :receiver A1
  :in-reply-to hpj
  :content (= (prix HP-Jet) (scalar 190 USD))
  :language KIF : ontology imprimantes
)

(eos
  :sender A2 :receiver A1
  :in-reply-to hpj
)

```

Dans cet échange, l'agent A1 demande à l'agent A2 toutes les informations concernant l'imprimante HPJet, et A2 lui répond par plusieurs messages qui se terminent avec le performatif *eos* qui met fin à la suite de réponses de A2 à la question de A1.

3.2.7.2. Le FIPA-ACL

Malgré les différentes qualités de la KQML et ces nombreuses performatifs, plusieurs chercheurs lui ont fait plusieurs critiques en dénonçant certaines de ses lacunes. Au nombre de celle-ci, on peut citer :

- l'ambiguïté et l'imprécision dans la signification de certains performatifs du fait de leur description en langage naturelle qui peut être source de confusion
- la redondance et/ou l'incohérence de certains performatifs qui ne sont pas nécessaires ou ne sont pas des actes du langage car n'ayant pas le pouvoir de satisfaire les buts des agents émetteurs.
- L'absence de certains performatifs (les promissifs), empêchant ainsi d'exprimer certains actes du langage tels que l'engagement.

Pour corriger le manque de spécification et de formalisation constaté dans KQML, Cohen et Levesque [129] ont proposé la définition d'un minimum de performatifs ayant des propriétés de compositionnalité pouvant permettre d'aboutir à de nouveaux performatifs à partir de combinaisons d'actes de langages plus primitifs. Ils ont ainsi défini à partir de leur théorie de

l'action rationnelle, une sémantique pour quelques-uns des performatifs les plus fondamentaux de KQML.

Dans la même lignée, et pour répondre à l'une des critiques faite au KQML sur le nombre élevé de ses performatifs qui malgré cela ne couvrent pas tous les types de performatifs envisageables, la FIPA a défini un ACL, le FIPA00061 (FIPA ACL Message Structure Specification), composé de 22 performatifs ou actes de communication (communicative acts) dont quatre de base (inform, request, confirm, disconfirm) et ayant une sémantique bien décrite dans le SL (Semantic Language), qui est un langage spécialisé disposant d'une logique multi-modale à différents quantificateurs mettant en relief : la croyance (Belief), le désir (Desire) et l'incertitude (Uncertainty).

Le FIPA-ACL est basé sur la théorie des actes de langage comme le KQML auquel il ressemble beaucoup et dont il reprend la syntaxe sans pour autant en prendre la sémantique. Les noms des performatifs sont différents mais FIPA-ACL reprend la liste des champs de KQML en la complétant par d'autres champs tels que : conversation-id et :protocol ; pour ne citer que ceux-là.

FIPA-ACL s'appuie sur la définition d'un ensemble d'actes de communication primitifs, à laquelle s'ajoutent les autres actes de communication pouvant être obtenus par la composition de ces actes de base, et sur un ensemble de messages prédéfinis que tous les agents peuvent comprendre.

Liste des performatifs de FIPA-ACL

Performatif	Catégorie	Signification
Accept proposal	Négociation	accepte une proposition d'action reçu précédemment
Agree	Exécution d'actions	accepte de réaliser une action demandée
Cancel	Demande d'information/ Exécution d'actions	informe qu'il n'a plus besoin qu'un autre agent réalise une action demandée précédemment
call for proposal	Négociation	fait un appel d'offre pour une action
Confirm	Transmission d'information	informe qu'une proposition est vraie
Disconfirm	Transmission d'information	informe qu'une proposition est fausse
Failure	Traitement d'erreur	informe de son échec dans la tentative de réalisation d'une action
Inform	Transmission d'information	informe de la valeur d'une proposition
Inform-if	Transmission d'information	informe qu'une proposition est vraie
Inform-ref	Transmission d'information	informe de la valeur d'un objet
Not-understood	Transmission d'information/ Traitement d'erreur	informe de son incompréhension d'une action demandée
Propagate	Exécution d'actions	demande à un agent de propager un message à un ensemble d'agents définis
Propose	Négociation	propose de réaliser une certaine action
Proxy	Exécution d'actions	demande à un agent de propager un message à un ensemble d'agents sélectionnés
Query-if	Demande d'information	demande si une proposition est vraie ou non
Query-ref	Demande d'information	demande la valeur d'un objet
Refuse	Exécution d'actions	refuse de réaliser une action
Reject Proposal	Négociation	refuse une proposition d'action

Request	Exécution d'actions	demande à un agent de réaliser une action
Request When	Exécution d'actions	demande à un agent de réaliser une action quand une certaine proposition deviendra vraie
Request Whenever	Exécution d'actions	demande à un agent de réaliser une action à chaque fois qu'une certaine proposition deviendra vraie
Subscribe	Demande d'information	demande à un agent de le prévenir des changements de valeur d'un objet

Tableau 3.8 : Performatifs (Communication Acts) de FIPA-ACL

A quelques éléments près (niveau orthographe) FIPA-ACL reprend pratiquement la liste des attributs de KQML qu'elle complète avec quelques attributs supplémentaires.

Attribut	Signification
:sender	Nom de l'émetteur du message
:receiver	Nom du récepteur du message
:from	l'origine de la performative dans :content quand un forward est utilisé
:to	la destination finale de la performative dans :content quand un forward est utilisé
:in-reply-to	le label attendu dans une réponse à un précédent message (de même que la valeur :reply-with du message précédent)
:reply-with	le label attendu dans une réponse au message courant
:language	Langage dans lequel le contenu est exprimé
:ontology	Ontologie sous-jacente nécessaire pour comprendre le contenu du message
:content	Contenu du message

Tableau 3.9 : Liste des attributs utilisés dans les performatifs de FIPA-ACL

Paramètres	Catégorie du paramètre
performative	Le type d'acte de communication
sender	Participant/Émetteur du message
receiver	Participant/Récepteur du message
reply-to	Participant in communication
content	Content of message
language	Description of Content
encoding	Description of Content
ontology	Description of Content
protocol	Control of conversation
conversation-id	Control of conversation
reply-with	Control of conversation
in-reply-to	Control of conversation
reply-by	Control of conversation

Tableau 3.10 : Paramètres des messages FIPA-ACL

Exemple de message FIPA-ACL

(confirm

```
:sender (agent-identifieur :name i)
:receiver (set (agent-identifieur :name j))
:content "weather (today, snowing)"
:language Prolog
```

)

Explication : L'agent i confirme à l'agent j qu'il est vrai qu'il neige aujourd'hui.

3.2.8. ONTOLOGIES

Pour communiquer il faut bien retenir un langage de communication compréhensible pour tous et un certain nombre de règles régissant la communication pour que les différents acteurs puissent se comprendre. En outre, on peut distinguer 3 couches ou niveaux dans une communication : le niveau syntaxique, le niveau sémantique et le niveau protocoles de communication. Suivant les objectifs visés et la structure des acteurs participant à la communication, les exigences au niveau de chacune de ces couches varient

Pour communiquer, les agents doivent donc être en mesure :

- d'envoyer et de recevoir des messages, faisant ainsi référence au niveau physique de la communication. A ce niveau, les agents doivent communiquer sur des couches physiques et réseaux convenus à cet effet, pour être en mesure de livrer et de recevoir des chaînes ou des objets qui représentent des messages.
- d'analyser les messages (chaînes ou objets reçus) du point de vue de leur syntaxe pour les décoder correctement et identifier voire séparer les différentes parties ou composantes (contenu du message, langue, expéditeur, etc.), et doivent pouvoir analyser le contenu du message.
- de comprendre les messages du point de vue de leur sémantique pour pouvoir satisfaire le but visé par leur envoi. L'agent récepteur doit donc pouvoir comprendre et interpréter de la même manière les symboles reçus et analysés. Ce qui suppose que tous les agents en communication partagent les mêmes descriptions des concepts associés aux symboles utilisés ou que ces symboles sont tellement explicitement exprimés au point de n'engendrer aucune ambiguïté dans leur interprétation.

C'est ce dernier niveau de la communication qui fait intervenir la notion d'ontologie qui est une composante capitale pour garantir le succès des échanges.

En effet, sachant qu'un même thème ou expression peut désigner plusieurs concepts en fonction du contexte. Sachant également que le même concept peut être désigné par

plusieurs termes ou expressions différentes, il est effectivement fondamental, pour éviter toute ambiguïté, de décrire ou de préciser explicitement le cadre d'application des symboles ou concepts véhiculés. Les langages de communication entre agents (ou ACL=Agent Communication Languages) étant pour la plupart une opérationnalisation des actes du langage, hérite de cette notion d'ambiguïté qui est inhérente à tout langage naturel. Ne pas prendre en compte et résoudre une telle situation en amont au niveau de la communication entre agents pourrait engendrer aussi bien la confusion que limiter l'interopérabilité entre agents avec des conséquences imprévisibles. Rappelons-nous qu'une simple erreur d'interprétation d'unité de mesure a été à l'origine de la perte de la sonde Mars Climate Orbiter.

Pour éviter ou tout au moins limiter au maximum aux ACL et aux agents de pareilles situations, certaines dispositions ont été prises à la base pour définir et situer le contexte de la communication afin de garantir une compréhension partagée des informations échangées par les agents.

Gandon [93] dira que « La sémantique des langages n'est pas un problème trivial ! » et définira une ontologie comme « une théorie logique qui rend compte partiellement mais explicitement d'une conceptualisation ». Ou plus simplement comme « un vocabulaire commun et une compréhension du sujet décrit ».

Thomas Gruber [91,92] nous en donnera la définition suivante : «une ontologie est une spécification explicite d'une conceptualisation ».

Le terme ontologie est emprunté à la philosophie où elle désigne *la science de l'être en tant qu'être*. La conceptualisation quant à elle est une représentation mentale générale et abstraite d'une information ou d'un objet. Elle ne varie pratiquement pas et décrit de façon générale et abstraite les connaissances d'un domaine et non un état particulier des choses du domaine.

La conceptualisation décrit de façon abstraite et générale les connaissances d'un domaine et non sur un état particulier des choses du domaine. Sa spécification formelle est requise pour la rendre exploitable et elle se fait en utilisant un langage particulier de modélisation ainsi que des termes particuliers propres au domaine.

Une ontologie décrit un domaine alors qu'une base de connaissances décrit les états particuliers des choses du domaine. Chaque agent a sa base de connaissances et seules les choses exprimables en utilisant une ontologie y sont stockées.

Pour communiquer avec un autre agent, chaque agent doit se baser sur une ontologie pour construire son message. Les ontologies représentent le domaine du discours et sont partagées entre agents pour leur permettre de se comprendre lors d'une communication.

Le but visé par la conception d'ontologie est celui de partage ou de transfert de connaissances et de sa réutilisation. L'ontologie est une spécification qui permet de prendre des engagements ontologiques qui consistent en des accords pour utiliser un vocabulaire pas forcément complet mais compatible à la théorie spécifiée pour un domaine. Une ontologie

peut donc être vue comme un ensemble de définition du vocabulaire formel partagé par les agents participant à une communication et auquel ils s'engagent à se conformer de façon cohérente et uniforme pour garantir la même compréhension des messages échangés dans un domaine de discours et éviter toute ambiguïté. C'est sur la base de ce vocabulaire que sont échangées les requêtes et les assertions entre les agents. Le partage d'une ontologie n'équivaut pas au partage de base de connaissances et ne fait pas obligation aux agents d'être en mesure de répondre à toutes les requêtes formulables à partir du vocabulaire partagé.

Dans le contexte de l'informatique et des sciences de l'information, une ontologie définit un ensemble de primitives de représentation pour modéliser un domaine de connaissances ou de discours. Les primitives de représentation sont généralement des classes (ou des ensembles), les attributs (ou propriétés), et les relations (ou relations entre les membres d'une classe). Les définitions des primitives de représentation incluent des informations sur leur signification et les contraintes relatives à leur application logique et cohérente. Dans le contexte des systèmes de bases de données, une ontologie peut être considérée comme un niveau d'abstraction des modèles de données, analogue aux modèles hiérarchiques et relationnels, mais destiné à la modélisation des connaissances sur les individus, leurs attributs et leurs relations avec d'autres personnes. Les ontologies sont généralement spécifiées dans les langues qui permettent l'abstraction loin des structures de données et des stratégies de mise en œuvre. Dans la pratique, les langues d'ontologies sont plus proches de la puissance expressive de la logique du premier ordre que les langues utilisées pour le modèle de bases de données. Pour cette raison, les ontologies sont dites être au niveau « sémantique », alors que les schémas de base de données sont des modèles de données de niveau « logique » ou « physique ». En raison de leur indépendance vis à vis des modèles de données de niveau inférieur, les ontologies sont utilisées pour intégrer les bases de données hétérogènes, permettant ainsi une interopérabilité entre des systèmes disparates, et spécifiant des interfaces de services de bases de connaissances indépendantes. Dans la pile de technologie des standards du Web sémantique [130], les ontologies sont appelées en tant que couche explicite. Il y a maintenant des langages standards et une variété d'outils commerciaux et open source pour créer et travailler avec les ontologies.

Une ontologie est difficile à concevoir [94]. Quelques grandes étapes de leur conception sont présentées ci-dessous :

- Terminologie consensuelle, avec définition des termes ;
- Concepts, attributs ... ;
- Taxonomie des concepts.

Nous avons également ci-dessous, diverses manières de représenter une Ontologie :

- Formelle : logique, graphes conceptuels, treillis de concepts, ... ;
- Semi-formelle : Frame, langage à objets, RDF, ... ;
- Informelle : langue naturelle, ...

Nous reviendrons plus loin sur la conception et la représentation d'une ontologie lorsque nous allons présenter nos choix et outils de notre système multi-agents de simulation de suivi.

3.3. CONCLUSION

Les SMA se trouvent aujourd'hui à l'intersection de beaucoup de domaines scientifiques et même des domaines relevant d'autres disciplines, soit connexes ou pas. Les SMA représentent une nouvelle approche pour l'analyse, la conception et l'implantation des systèmes informatiques complexes. L'approche orientée agents offre un puissant répertoire d'outils, de techniques, et de métaphores qui ont le potentiel d'améliorer considérablement les systèmes logiciels. Les ACL (Agent Communication Languages) ajoutent une dimension supplémentaire aux protocoles classiques de communication en apportant non seulement la prise en compte de la sémantique de l'information échangée entre l'expéditeur et le destinataire, mais également la modélisation intentionnelle de l'expéditeur par rapport à l'information envoyée.

Il faut toutefois reconnaître que l'implémentation d'un SMA reste assez complexe et demande une très bonne démarche méthodologique dans la phase d'analyse ainsi que de bonnes plateformes adaptées à leur implémentation et respectant les standards et les normes des protocoles d'interaction et de communication. C'est à cet exercice que nous nous proposons de nous livrer dans le chapitre 5 de cette thèse.

CHAPITRE 4 : APPROCHES PROPOSEES POUR LE SUIVI

Sommaire

4.1.	Introduction	67
4.2.	Modélisation des caractéristiques visio-temporelles	68
4.2.1.	Modélisation de la couleur.....	68
4.2.2.	Modélisation de la forme	70
4.2.3.	Modélisation de la vitesse.....	71
4.3.	Application du TBM pour le suivi dans un espace avec caméras distants	72
4.3.1.	Cadres de discernement et référentiels.....	72
4.3.2.	Construction des fonctions de croyance.....	73
4.3.3.	Première combinaison des masses	74
4.3.4.	Deuxième combinaison des masses.....	76
4.3.5.	La Prise de décision	78
4.4.	Conclusion	81

4.1. INTRODUCTION

Après avoir présenté dans les deux premiers chapitres quelques-uns des différents concepts et outils qui sont utilisés en vidéosurveillance et autres domaines de vision par ordinateur pour la modélisation de l'information dans le cadre de la détection, la reconnaissance et le suivi de cible ; puis dans un troisième chapitre, les contours de la modélisation de systèmes multi-agents ainsi que les mécanismes et contraintes de leur mise en œuvre ; nous allons consacrer le chapitre courant à la présentation de nos choix en ce qui concerne la détection et la mise en correspondance dans la mise en place de notre système de suivi multi-capteurs.

Parmi les méthodes d'associations de données, nous avons opté pour le MHT malgré son plus grand défaut qui est l'explosion combinatoire. Les raisons qui justifient ce choix sont relatives à sa capacité de gestion des situations ambiguës à travers le maintien d'hypothèses ; et celle de la gestion de l'initialisation et de la terminaison de pistes qui sont deux notions cruciales dans un système de suivi de cible (véhicule) dans un flux vidéo. Pour la fusion d'information, le modèle de fusion qui nous a paru le plus réaliste et le plus adapté à notre étude est celui de la théorie de l'évident qui permet de prendre en compte l'imperfection (imprécision, incomplétude, incertitude) des mesures dans leur modélisation et dans la prise de décision.

Nous avons particulièrement opté pour le modèle des croyances transférables, proposé par Smets et qui a bénéficié de plusieurs travaux enrichissants. Notre objectif est d'apporter notre contribution à la gestion de la prise de décisions en cas d'ambiguïté en nous appuyant sur notre approche personnelle d'utilisation du MHT.

Nous présentons dans la seconde session de cette partie nos différents modèles pour la représentation de l'information, leur combinaison ainsi que la prise de décision. Dans cette partie, nous montrons également comment nous comptons faire fonctionner le MHT pour améliorer la prise de décision en cas d'ambiguïté.

4.2. MODELISATION DES CARACTERISTIQUES VISIO-TEMPORELLES

4.2.1. MODELISATION DE LA COULEUR

Un des outils efficaces pour la modélisation de la répartition des couleurs d'un objet ou d'une image reste l'histogramme. Un histogramme permet de représenter la distribution des intensités ou des couleurs des pixels d'une image. Plusieurs travaux ont démontré la robustesse des histogrammes et leur invariance face à certaines transformations telles que la rotation, la translation ; et dans une moindre mesure au changement d'angle de vue (point de vue) et au changement d'échelle. Les histogrammes se prêtent bien aux applications temps réel (comme dans notre cas) du fait de leur facilité de calcul.

Mais les histogrammes ont quelques insuffisances dont deux principalement qui nous intéressent. La première c'est que les histogrammes simples ne prennent malheureusement pas en compte la répartition spatiale des couleurs et donc deux images différentes peuvent avoir même histogramme pour peu qu'elles aient la même distribution des couleurs. Une illustration en est faite sur la figure 4.1 ci-dessous.

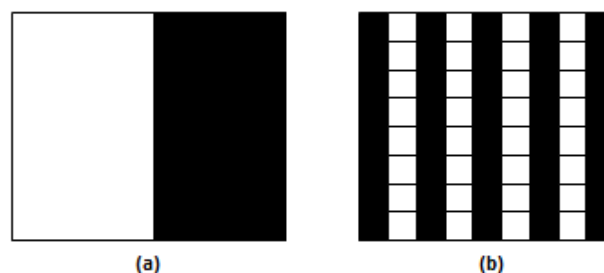


Figure 4.1 : Deux différentes images (a) et (b) ayant même histogramme

Plusieurs travaux ont été effectués pour pallier la première insuffisance. Au nombre de ces travaux, deux ont particulièrement retenu notre attention. Il s'agit dans un premier temps d'histogrammes multi-échelles d'histogrammes de couleurs, proposé dans [15] et [16] par le même auteur. Ces histogrammes présentent l'avantage de prendre en compte la distribution spatiale des couleurs ainsi que la robustesse au bruit pour une meilleure mise en

correspondance. Ils ont prouvé leur efficacité par rapport aux histogrammes simples. Il s'agit d'une famille d'histogrammes \mathbf{p} élaborés à différentes échelles τ d'une image.

Si I désigne une image donnée, son histogramme multi-échelles s'obtient en effectuant la convolution de cette image avec des filtres gaussiens $G(\tau)$ présenté dans [17] et [18].

$$G(\tau) = \frac{1}{2\pi\tau^2} \exp\left\{-\frac{x^2+y^2}{2\tau^2}\right\}, \text{ avec } \tau \text{ représentant l'écart type du filtre.} \quad (4-1)$$

L'histogramme de l'image convoluée est noté : $\mathbf{p}^\tau (I) = \rho(I * G(\tau))$, avec $*$ opérateur de convolution.

L'histogramme multi-échelles sera donc le vecteur défini par : $\mathbf{p} = [\mathbf{p}^0, \dots, \mathbf{p}^\tau, \dots, \mathbf{p}^{N-1}]$

Dans le cadre de la mise en correspondance, une distance a également été proposée par [16]. Cette distance est définie par :

$$d^{HM}(p, q) = \frac{1}{N} \sum_{\tau=0}^{N-1} d(p^\tau, q^\tau) \quad (4-2)$$

Avec $d(p^\tau, q^\tau)$ la distance entre les histogrammes p^τ et q^τ . Cette distance peut être la distance de Mahalanobis ou la distance Bhattachayya. Notons au passage que cette dernière distance est très pratique pour la comparaison d'histogrammes.

La seconde approche qui nous a semblée intéressante compte tenu de la qualité de ses résultats est celle proposée par [9] dans sa thèse. Il s'agit d'utiliser la caractéristique verticale d'un objet pour l'identifier. Pour le calcul de la caractéristique verticale, [9] procède à la projection des pixels de l'objet sur son axe principal. La moyenne des couleurs est alors calculée pour chaque canal. La hauteur de l'objet est fixée à l'avance ; au besoin un re-échantillonnage est fait. Cette méthode offre l'avantage d'obtenir un traitement très rapide en temps de calcul, robuste au changement d'angle de vue et qui prend en compte dans une moindre mesure la répartition spatiale des couleurs.

Le fait de prendre en compte seulement la caractéristique verticale d'une image n'est normalement pas suffisant pour son identification. En effet, il sera très facile de présenter deux voire plusieurs images totalement différentes du point de vue de leurs motifs mais donnant les mêmes résultats avec cette approche, laissant croire que ces deux images sont identiques ou tout au moins proches alors que ce n'est pas le cas. Les images de la figure 4.2 ci-dessus en sont une illustration. Cependant, comme de telles situations ne sont pas très courantes au niveau des objets de type véhicule, nous pensons très intéressante une telle forme de signature d'une image.

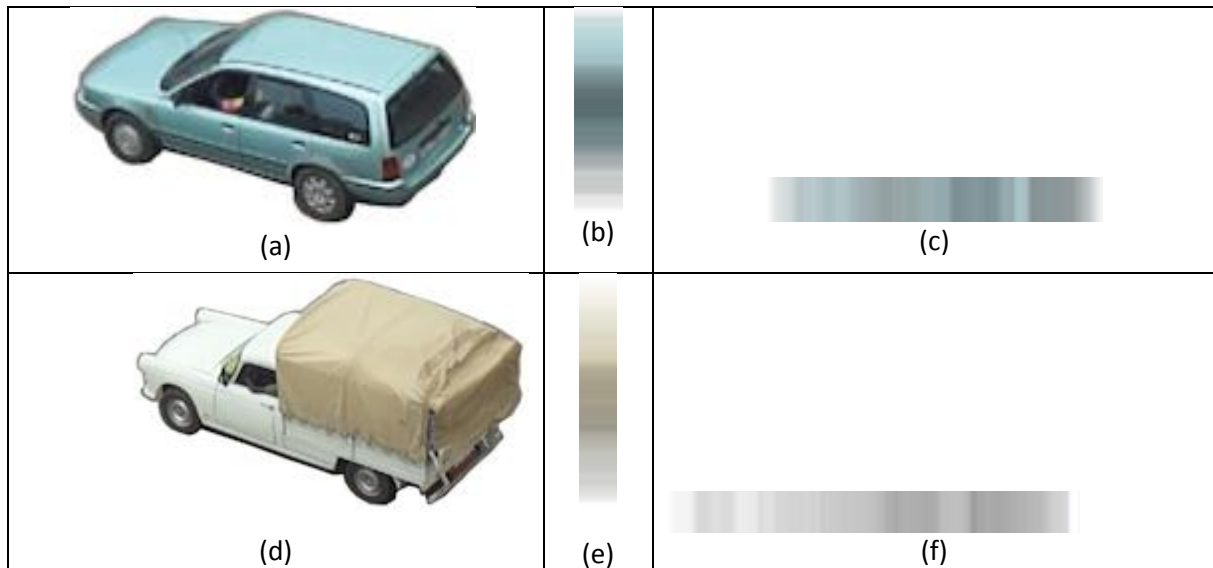


Figure 4.2 : Deux images avec leurs projections verticales et horizontales

(a) première image source, (b) projection verticale de (a), (c) projection horizontale de (a)
 (d) première image source, (e) projection verticale de (d), (f) projection horizontale de (d)

Après avoir fait des tests avec la projection verticale et celle horizontale que nous avons décidé nous-même d'expérimenter, nous avons décidé de retenir les deux types de projections qui obéissent à la même logique mais qui ne véhiculent pas les mêmes qualités d'information suivant l'objet en présence. Ainsi, avons-nous retenu d'utiliser une projection horizontale pour les véhicules de type voiture ou camion ; et une projection verticale pour les véhicules de type motocyclette. Notons que la notion de motocyclette inclue également dans notre contexte, la ou les personnes dessus (conducteur).

A partir de ces deux approches, nous comptons proposer une fusion d'informations pour donner une modélisation plus robuste de la répartition spatiale des couleurs de nos objets suivis.

La deuxième insuffisance est la sensibilité des histogrammes aux changements d'illumination et aux conditions d'éclairage. Plusieurs pistes de solutions ont été explorées dans [20], [21] et [22]. Après étude de ces différentes approches qui présentent aussi bien des points forts que des points faibles, [16] propose pour diminuer sensiblement l'effet du changement d'illumination, d'utiliser une distance qu'il a mise en place à partir d'une modification de la distance de Matusita. Le principe de cette distance est basé sur un changement de repère aux vecteurs représentant les histogrammes avant de calculer la distance de Matusita.

4.2.2. MODELISATION DE LA FORME

Pour modéliser la forme des objets, nous allons nous baser sur deux éléments qui sont : les dimensions de l'objet, la distance entre l'objet et le capteur.

Pour les dimensions des objets, nous avons deux possibilités : opter pour un modèle tridimensionnel polyédrique et plus précisément un modèle parallélépipédique inspiré de plusieurs travaux effectués dans le domaine des transports ; ou opter pour un modèle rectangulaire plus simple. L'objectif dans

les deux cas est d'obtenir une boîte englobante. Pour ce qui est des coordonnées et/ou de l'angle de vision, un repère en dimension 3 et une projection dans le plan seront utilisés suivant le cas.

Une boîte englobante est la plus petite boîte rectangulaire ou parallélépipédique avec des côtés parallèles aux axes x et y (puis z dans le cas d'un parallélépipède), qui contient entièrement un objet.

Les boîtes englobantes ont le plus souvent une bonne distance de Hausdorff, mais l'intersection entre boîtes englobantes est très coûteuse en général et le seul cas où les tests de leurs intersections est simple à mettre en œuvre reste celui où les boîtes sont définies dans un même repère. Dans [26], les auteurs ont fait des comparaisons dans divers repères, en calculant des boîtes englobantes de boîtes englobantes, et ils proposent dans [23] une meilleure solution avec un algorithme efficace pour la comparaison de deux boîtes englobantes quelconques.

4.2.3. MODELISATION DE LA VITESSE

Connaitre ou évaluer la vitesse instantanée d'un objet à une date t donnée n'est pas difficile à partir d'un certain nombre de mesures. Mais prédire l'évolution de cette vitesse dans le temps et l'espace n'est pas du tout chose facile car plusieurs facteurs non maîtrisables peuvent influencer sur cette évolution. Dans cette partie, nous allons juste construire un modèle approximative permettant de faire une estimation de la vitesse. Cependant, pour le rendre réaliste, nous allons y intégrer la notion d'intervalle flou (ou utiliser une dispersion gaussienne) pour en garantir la flexibilité.

Partant de l'équation cinématique suivante :

$$d = v_0.t + \frac{1}{2} a.t^2 \quad (4-3)$$

on peut déduire par dérivation que :

$$v(t) = v_0 + a.t \quad (4-4)$$

v_0 : vitesse local au départ d'un capteur (caméra)

a : module d'accélération de l'objet

d : distance entre deux capteur (caméra)

On sait également que la vitesse d'un objet (voiture ou personne) à un instant peut être influencée par la densité du trafic environnant. Généralement cette vitesse est inversement proportionnelle à la densité du trafic à l'instant et peut-être représentée comme suit :

$$V(t) = v_{\max}.(1 - \rho(t)/\rho_{\max}) \quad (4-5)$$

v_{\max} : vitesse maximale de l'objet suivant sa famille ou classe d'appartenance

ρ_{\max} : densité maximale à partir de laquelle des objets peuvent être considérés à l'arrêt

$\rho(t)$: densité d'objets sur le parcours à l'instant t . Il est proportionnel au nombre d'objet par unité de voie ou d'espace.

A partir des équations (4-4) et (4-5) on peut déduire un modèle d'accélération qui se présente comme ci-dessous :

$$a(t) = [v_{\max} \cdot (1 - \rho(t)/\rho_{\max}) - v_0] \cdot 1/t \quad (4-6)$$

Naturellement, ce modèle d'accélération est loin d'être réaliste car ne prend pas en compte tous les paramètres spatio-temporels pouvant influencer sur le comportement dynamique des objets.

Aussi, pour une meilleure exploitation des prédictions sur le déplacement des objets en zones aveugles, nous allons utiliser une distribution Gaussienne pour modéliser les dates d'arrivée. L'idée est de les considérer comme étant des moyennes de distributions Gaussiennes.

4.3. APPLICATION DU TBM POUR LE SUIVI DANS UN ESPACE AVEC CAMERAS DISTANTES

Le suivi global multi-caméras utilise les données haut-niveaux provenant des suiveurs locaux. Le véritable problème qui se pose se rapporte à la difficulté de mise en correspondance à cause de l'existence de zones aveugles non couvertes par aucune caméra. Cette difficulté sera d'autant plus importante que l'étendu des zones aveugles sera plus grand. En effet, la trajectoire des objets suivis, doit être construite à partir des observations et d'éventuelles connaissances d'un expert du domaine sur la dynamique du comportement des objets en tenant compte de l'environnement qui n'est pas forcément le même d'une caméra à l'autre ; chose qui peut ajouter de l'imprécision aux mesures et occasionner de non détections ou des erreurs de détection.

La ré-identification des objets déjà connus par d'autres caméras se fera à partir des résultats issus des suivis locaux qui doivent mettre à disposition les deux niveaux d'informations suivantes dont nous avons parlé plus haut :

- les caractéristiques visuelles de l'objet (histogramme des couleurs, boîte englobante en relation avec la classe de l'objet, position de l'objet) qui représente sa signature.
- les contraintes spatiaux-temporelles de l'objet (vitesse et modèle d'accélération, densité du trafic dans la zone aveugle)

A partir des différentes caractéristiques visuelles de chaque objet détecté par un capteur 'k', on va mettre en place dans la phase d'association de données, des distances qui vont servir à la construction des différentes fonctions de masses ; chacune correspondant à une caractéristique visuelle.

4.3.1. CADRES DE DISCERNEMENT ET REFERENTIELS

La définition du cadre de discernement est la première étape dans le processus de modélisation des données. Elle est préalable à la phase de mise en place des fonctions de masse de croyance. Les différentes situations observables au niveau de chaque caméra à un instant t sur une scène multi-caméras sont :

- la perception d'objets déjà identifiés (objets connus, donc attendus)
- la perception de nouveaux objets (soit attendus, soit totalement nouveaux dans le système)
- la disparition d'objets (soit par sortie de scène, soit par occultation, soit par erreur d'identification)

Pour modéliser toutes ces catégories de situations, nous allons introduire les notations suivantes issues des travaux de David Mercier [72], inspirées des travaux de Rombaut [70] et de Gruyer [73].

X_i : objet i perçu à l'instant t . $i \in I_p = \{1, 2, \dots, N\}$, où N désigne le nombre total d'objets perçu à l'instant t .

Y_j : objet j connu à l'instant d'avant, donc $t-1$. $j \in I_c = \{1, 2, \dots, M\}$, où M désigne le nombre total d'objets connu à l'instant $t-1$.

* : apparition ou disparition d'objet suivant le cas.

Ces notations vont nous permettre d'explicitier toutes les associations possibles et de construire des cadres de discernement cohérents [71].

- $\Omega_{i,j} = \{o_{i,j}, \overline{o_{i,j}}\}$; où $o_{i,j}$ représente l'hypothèse selon laquelle l'objet perçu X_i correspondrait à l'objet connu Y_j ; et $\overline{o_{i,j}}$ représente la négation de cette hypothèse.
- $\Omega_{X_i} = \{Y_1, Y_2, \dots, Y_M, *\}$; avec chacun des Y_j connus qui est vu comme candidat à la mise en correspondance avec l'objet X_i perçu à l'instant. * représentant ici l'apparition de l'objet X_i .
- $\Omega_{Y_j} = \{X_1, X_2, \dots, X_N, *\}$ avec chacun des X_i perçus qui est vu comme candidat à la mise en correspondance avec l'objet Y_j connu. * représentant ici la disparition de l'objet Y_j ou son occultation par un autre ou d'autres objets.

4.3.2. CONSTRUCTION DES FONCTIONS DE CROYANCE

Plusieurs travaux et recherches ont permis de mettre en place un certain nombre d'approches pour le calcul des masses d'évidence ; chaque méthode étant adaptée à des domaines d'application bien définis. On peut citer les travaux de Shafer [43], de Denoeux [45, 54], Vannoorenberghe et Denoeux [49], de Appriou [50, 51], de Yager [53], de Lefevre [52], de Yann Permal et al. [47], pour ne citer que ceux-là.

Comme nous l'avions annoncé plus haut dans le premier chapitre, notre approche va se baser sur les travaux de Denoeux [45, 54] et ceux de Y. Permal et al. [47] pour la définition de nos masses de croyance suivant le modèle :

$$\begin{cases} m^{\Omega_{i,j}}(\{o_{i,j}\}) = \beta \phi_{i,j}(e_{i,j}) \\ m^{\Omega_{i,j}}(\{\overline{o_{i,j}}\}) = \beta (1 - \phi_{i,j}(e_{i,j})) \\ m^{\Omega_{i,j}}(\Omega_{i,j}) = 1 - \beta \end{cases} \quad (4-7)$$

avec :

- β : constante caractérisant le degré de fiabilité de la source ; $0 < \beta < 1$
- $\phi_{i,j}$: fonction décroissante monotone vérifiant $\begin{cases} \phi_{i,j}(0) = 1 \\ \lim_{x \rightarrow \infty} \phi_{i,j}(x) = 0 \end{cases}$ (4-8)

la fonction qui est généralement utilisée est de la forme :

$$\phi_{i,j}(d_{i,j}) = e^{-\gamma_{i,j}(d_{i,j})^2} \quad (4-9)$$

- $e_{i,j}$: mesure de dissimilarité entre les caractéristiques de l'objet perçu X_i et celles de l'objet connu Y_j .
- $\gamma_{i,j}$ est un paramètre d'ajustement de l'ampleur de la décroissance de la masse en fonction de la distance de mise en correspondance (similarité) utilisée. Sa valeur est généralement fixée à 1.

4.3.3. PREMIERE COMBINAISON DES MASSES

Une première fusion d'information sera faite à partir de 3 fonctions de masses, qui sont définies dans un même référentiel, pour obtenir une nouvelle fonction de masse $m^{\Omega_{i,j}}$ par la règle combinaison de Dempster non normalisée.

$$m^{\Omega_{i,j}} = m_{couleur}^{\Omega_{i,j}} \oplus m_{forme}^{\Omega_{i,j}} \oplus m_{date}^{\Omega_{i,j}} \quad (4-10)$$

Ceci permet de construire $N \times M$ fonctions de croyances, chacune pour une mise en correspondances entre un objet perçu X_i et un objet connu Y_j , avec comme référentiel $\Omega_{i,j} = \{o_{i,j}, \overline{o_{i,j}}\}$.

Nous allons présenter ci-dessous les éléments nous permettant de mettre en place chacune de ces fonctions de masse.

4.3.3.1. Niveau date

Il s'agit ici de caractériser la date probable de ré-identification par une caméra distante, d'un objet préalablement identifié à une date antérieure par une autre caméra.

La première étape de cette partie est la détermination en local (au niveau de la caméra courante) de la vitesse moyenne de chacun des objets suivis.

Toutes les caméras étant fixes, on peut connaître à tout moment la distance entre deux caméras voisines. Nous reviendrons plus loin sur notre modèle et la structure de données de représentation du système multi- caméras de sorte à connaître facilement et à tout instant, la distance entre deux caméras voisines.

Nous allons déterminer le temps moyen nécessaire à chacun des objets pour parcourir la distance entre deux caméras. A partir de cette distance et d'un modèle d'accélération nous allons avoir une équation à une inconnue permettant d'avoir la date moyenne recherchée à partir des formules (4-3) et (4-6).

$$d = \frac{1}{2} \left[V_{max} \left(1 - \frac{\rho(t)}{\rho_{max}} \right) + V_0 \right] \cdot t \quad (4-11)$$

Soit t_{moy} une solution de cette équation, la précision de t_{moy} dépendra de la qualité de la fonction de densité $\rho(t)$ choisie. Mais n'ayant en réalité aucune solution efficace d'approximation de $\rho(t)$, nous avons plutôt choisi de considérer que la vitesse des objets est constante sur tout leur trajet. Ce qui nous donne à partir l'équation (1-3) :

$$t_{moy} = \frac{d}{V_0} \quad (4-12)$$

Cette approximation du temps de parcours n'est pas du tout réaliste. Alors, nous appuyant sur les travaux de [62, 63, 74], nous construisons plutôt un intervalle temporel flou ou DOP (Domain Occurrence Possibility) autour de t_{moy} pour exprimer un intervalle de dates probables d'arrivée d'un objet au niveau d'un autre capteur (caméra).

On peut ainsi construire une fonction de vraisemblance puis la convertir en fonction de masse. Nous avons opté dans cette partie pour l'utilisation de la règle d'Appriou [74] pour la définition de la fonction de masse $m_{date}^{\Omega_{i,j}}$. Cette fonction de masse est définie de la manière suivante :

$$\begin{cases} m_{date}^{\Omega_{i,j}}(\{o_{i,j}\}) = 0 \\ m_{date}^{\Omega_{i,j}}(\{\overline{o_{i,j}}\}) = \beta (1 - R.L(O|o_{i,j})) \\ m_{date}^{\Omega_{i,j}}(\Omega_{i,j}) = 1 - \beta (1 - R.L(O|o_{i,j})) \end{cases} \quad (4-13)$$

4.3.3.2. Niveau Couleurs

Pour caractériser la couleur des objets nous avons proposé dans la sous session III-1-1) de notre premier chapitre d'utiliser des descripteurs de haut niveau. Il s'agit d'un histogramme d'histogrammes multi-échelles et des caractéristiques verticales puis horizontales des objets suivis.

Avec chacune de ces caractéristiques, on va mettre en place la distance d'histogramme multi-échelles associée telle que présenté à l'équation (4-2). En utilisant la formule présentée en (4-9), nous obtenons la quantité $\phi_{i,j}(d_{i,j})$ nous permettant de définir nos deux premières fonctions de masse permettant de caractériser la couleur sur les cadres de discernement $\Omega_{i,j}$.

Ces cadres étant totalement exhaustifs, nous utilisons la règle de Dempster (somme orthogonale) pour les combiner pour avoir une seule masse $m_{couleur}^{\Omega_{ij}}$ pour caractériser la couleur avec la fonction de masse de l'équation (3-1).

$$\begin{cases} m_{couleur}^{\Omega_{ij}}(\{o_{ij}\}) = \beta \phi_{ij}(e_{ij}) \\ m_{couleur}^{\Omega_{ij}}(\{\overline{o_{ij}}\}) = \beta (1 - \phi_{ij}(e_{ij})) \\ m_{couleur}^{\Omega_{ij}}(\Omega_{ij}) = 1 - \beta \end{cases} \quad (4-14)$$

4.3.3.3. Niveau forme

Pour la forme, nous allons utiliser une boite englobante et plus précisément un rectangle englobant. Une classification des objets sera faite à partir de la valeur de la surface englobante de ces véhicules. Cette surface est calculée à partir des deux mesures de gabarit (longueur, hauteur) et est représenté par un nombre (vecteur) flou.

A partir de ces mesures floues, nous construisons, comme nous avons eu à le faire plus haut, la fonction de masse $m_{forme}^{\Omega_{ij}}$ à partir des fonctions de vraisemblance.

Au total, le nombre de fonctions de croyance obtenu (NxM) devient très vite assez important et croissant en fonction des objets perçus et connus ; avec autant de référentiels que de fonctions de croyances.

4.3.4. DEUXIEME COMBINAISON DES MASSES

Les différentes fonctions de masse de croyance ci-dessus obtenues sont combinées pour obtenir une unique fonction de croyance permettant de faire la correspondance entre tous les objets perçus et ceux connus ; d'identifier les nouveaux objets ; de déduire les éventuelles disparitions d'objets.

Mais au préalable, un référentiel commun (nouveau cadre de discernement) est créé pour rester conforme aux exigences de la loi de combinaison de Dempster. Les deux référentiels possibles qui prennent en compte toutes les hypothèses possibles permettant d'évaluer la correspondance entre objets perçus et objets connus sont les référentiels Ω_{X_i} et Ω_{Y_j} qui ont été présentés plus haut.

La définition des masses sur ces nouveaux référentiels à partir des masses sur les NxM référentiels $\Omega_{i,j}$ se fera par opération de raffinement des $\Omega_{i,j}$ vers Ω_{X_i} , respectivement Ω_{Y_j} . Les deux nouvelles masses obtenues par combinaison conjointe se définissent comme suit :

$$m^{\Omega_{X_i}} = \bigoplus_{j \in I_p} m_j^{\Omega_{X_i}} \quad (4-15)$$

$$m^{\Omega_{Y_j}} = \bigotimes_{i \in I_c} m_i^{\Omega_{Y_j}} \quad (4-16)$$

Où

$m_j^{\Omega_{X_i}} = m^{\Omega_{i,j} \uparrow \Omega_{X_i}}$ est la fonction de masse obtenue par l'opération d'extension vide de $m^{\Omega_{i,j}}$ du référentiel $\Omega_{i,j}$ au référentiel Ω_{X_i}

$m_i^{\Omega_{Y_j}} = m^{\Omega_{i,j} \uparrow \Omega_{Y_j}}$ est la fonction de masse l'opération d'extension vide de $m^{\Omega_{i,j}}$ du référentiel $\Omega_{i,j}$ au référentiel Ω_{Y_j} .

On note que :

$$m^{\Omega_{i,j} \uparrow \Omega_{X_i}}(\rho_{i,j}^X(A)) = m^{\Omega_{i,j}}(A), \quad \forall A \subseteq \Omega_{i,j} \quad (4-17)$$

$$m^{\Omega_{i,j} \uparrow \Omega_{X_i}}(\rho_{i,j}^Y(A)) = m^{\Omega_{i,j}}(A), \quad \forall A \subseteq \Omega_{i,j} \quad (4-18)$$

où $\rho_{i,j}^X$ et $\rho_{i,j}^Y$ sont les opérateurs de raffinement définis respectivement par :

$$\rho_{i,j}^X : \Omega_{i,j} \rightarrow \Omega_{X_i}, \text{ avec } \begin{cases} \rho_{i,j}^X(\{o_{i,j}\}) = \{Y_j\} \\ \rho_{i,j}^X(\{\overline{o_{i,j}}\}) = \{\overline{Y_j}\} \end{cases} \quad (4-19)$$

et

$$\rho_{i,j}^Y : \Omega_{i,j} \rightarrow \Omega_{Y_j}, \text{ avec } \begin{cases} \rho_{i,j}^Y(\{o_{i,j}\}) = \{X_i\} \\ \rho_{i,j}^Y(\{\overline{o_{i,j}}\}) = \{\overline{X_i}\} \end{cases} \quad (4-20)$$

On a :

$$m^{\Omega_{X_i}}(Y_k) = \sum_{\Omega_{A_j} = Y_k} \prod_{j \in J} m_j^{\Omega_{X_i}}(A_j) \quad (4-21)$$

$$m^{\Omega_{X_i}}(\overline{K}) = \sum_{\Omega_{A_j} = \overline{K}} \prod_{j \in J} m_j^{\Omega_{X_i}}(A_j) \quad (4-22)$$

En particulier

$$m^{\Omega_{X_i}}(\{*\}) = \prod_{j \in J} m_j^{\Omega_{X_i}}(\overline{A_j}) \quad (4-23)$$

$$m^{\Omega_{X_i}}(\Omega_{X_i}) = \prod_{j \in J} m_j^{\Omega_{X_i}}(\Omega_{X_i}) \quad (4-24)$$

$$m^{\Omega_{X_i}}(\emptyset) = \sum_{\Omega_{A_j} = \emptyset} \prod_{j \in J} m_j^{\Omega_{X_i}}(A_j) \quad (4-25)$$

Les détails de ces calculs sont disponibles dans [65].

4.3.5. LA PRISE DE DECISION

Il s'agit du niveau pignistique du Modèle des Croyances Transférables (MCT). Dans cette phase de prise de décision, Smets propose l'utilisation d'une distribution de probabilités en lieu et place des structures du niveau crédal (croyance, plausibilité, etc.). Il existe plusieurs approches mais la plus utilisée est celle reposant sur le calcul de probabilité pignistique qui est un compromis. Il définit pour cela la probabilité pignistique **BetP** qui se présente comme suit :

$$BetP(\omega_i) = \frac{1}{1-m(\emptyset)} \sum_{A \ni \omega_i} \frac{m(A)}{|A|} \quad \forall \omega_i \in \Omega \quad (4-26)$$

Une fois le **BetP** calculé, la décision est prise en choisissant l'élément ω^* possédant la plus grande probabilité pignistique. Mais nous allons adapter cette façon de prendre la décision à notre cas et nous prendrons les décisions à partir de la probabilité pignistique qui maximise la loi jointe $BetP^{\Omega_{X_1} \times \dots \times \Omega_{X_N}}$ en nous inspirant des travaux de D. Mercier et al. [65]. L'avantage de cette approche est de prendre directement des décisions en faisant des associations entre n-uplets plutôt que sur chaque objet de façon individuelle. Elle permet d'obtenir une association solution (c'est un n-uplet) plutôt que n associations solutions pour aboutir à un n-uplets solution.

4.3.5.1. Mise en œuvre du MHT dans la prise de décision

Pour améliorer la mise en correspondance dans le cas des ambiguïtés, nous proposons un suivi temporel orienté observation. La solution serait une adaptation du MHT qui est orientée observation.

Les situations d'ambiguïtés d'association peuvent résulter de problèmes d'initialisation de pistes, de situations d'occultations, de situations de fausses alarmes ou même de problèmes d'imprécision dans la définition des fonctions de masse ou de fiabilité des sources d'information.

A chaque fois qu'il y aura ambiguïté, nous tenterons de la lever en déployant le MHT en partant des probabilités pignistiques de la loi jointe pour construire un arbre d'hypothèses tout en émettant des contraintes pour élaguer des hypothèses ne répondant pas à un seuil d'acceptabilité afin d'éviter d'avoir une explosion exponentielle. Les optimisations de l'algorithme du MHT pour réduire les hypothèses existent à plusieurs niveaux. Les trois grands axes d'optimisation que nous avons présenté plus haut sont :

- 1- partitionnement spatial en sous-groupes indépendant pour diminuer la taille des problèmes d'association
- 2- élagage en largeur de l'arbre des hypothèses en le gardant que les **N** meilleures hypothèses (N à définir) pour rendre la complexité du MHT plutôt polynomiale qu'exponentielle.

- 3- élagage en profondeur de l'arbre en jouant sur la dimension temporelle pour limiter la durée en termes de nombre maximal d'images à explorer (3 à 4)

En considérant par exemple qu'on connaît quatre objets Y_1, Y_2, Y_3, Y_4 à un instant $t-1$ (donc attendus à l'instant t) et qu'à l'instant t , on observe trois objets X_1, X_2, X_3 .

Une hypothèse d'association à l'instant t pourrait être par exemple : « X_1 est associé à Y_1, X_2 a disparu, X_3 est associé à Y_2 ». On peut ainsi avoir jusqu'à $3 \times (4 + 1)$ hypothèses dont les moins probables seront élaguées.

Ainsi, soit à un instant $t-1$ M objets connus, supposons qu'à l'instant t on perçoit N objets. On a théoriquement $M \times (N+1)$ hypothèses $h_1, h_2, \dots, h_{M \cdot (N+1)}$, de la forme :

$$h_i = \{(X_1, Y_1), \dots, (X_k, Y_M)\} \quad \text{avec} \quad X_k \in \Omega_{Y_j} = \{X_1, X_2, \dots, X_N, *\} \quad \text{et} \quad i \in \{1, 2, \dots, M \cdot (N + 1)\}$$

A chaque hypothèse sera associée sa probabilité pignistique jointe.

Le fonctionnement du MHT se présente comme suit :

- La construction de l'arbre du MHT commence avec l'univers Ω_{t_0} des objets connus ainsi que les hypothèses retenues parmi les N meilleures hypothèses d'associations.
- Chaque branche de l'arbre d'hypothèse sera une hypothèse d'association et à chaque itération, chaque hypothèse va donner naissance à un sous arbre d'hypothèses par le principe de propagation des hypothèses en supprimant naturellement à chaque niveau celles qui deviennent improbables et en regroupant celles très proches.
- Une mise à jour des probabilités de chaque nouveau nœud est faite à partir des probabilités conditionnelles ainsi que du calcul des probabilités pignistiques de la loi jointe des nouvelles associations issus de nouvelles observations. Chaque hypothèse h_i met à jour ses propres pistes indépendamment des autres hypothèses, et sans tenir compte de fait qu'il y a des pistes communes qui peuvent être identiques.
- Une mise à jour des intervalles flous d'apparition est aussi faite en prenant en compte le temps écoulé depuis la dernière observation.

Donc à chaque acquisition, on génère toutes les hypothèses à partir des hypothèses précédentes et des observations courantes. On calcule les probabilités pignistiques et on élague les hypothèses non probables. On regroupe éventuellement celles très proches, puis on passe à l'itération suivante tant que la condition d'arrêt n'est pas satisfaite.

Pour la prise de décision, on procède à l'évaluation des hypothèses à l'instant t . Cette évaluation se fait en calculant leurs probabilités par une mise à jour des probabilités des hypothèses parents (instant $t-1$) en tenant compte des associations entre les objets perçus (à l'instant t) et les objets connus (à l'instant $t-1$). L'hypothèse courante h_k au temps t_k sachant les mesures Z_k du temps t_0 jusqu'au temps t_k est évaluée à partir des associations pistes/objets au temps t_k représentée par $h(k)$ et l'hypothèse parent h_{k-1} au temps t_{k-1} ainsi

que les mesures Z_k . La probabilité d'une hypothèse h_k peut être calculée par la règle de Bayes suivante :

$$P(h_k | Z_k) = P(h(k), h_{k-1} | Z(k), Z_{k-1})$$

$$= \frac{1}{c} * P(Z_k | h(k), h_{k-1}, Z_{k-1}) * P(h(k) | h_{k-1}, Z_{k-1}) * P(h_{k-1} | Z_{k-1})$$

Où :

c : est une constante de normalisation ; et $h(k)$: hypothèse d'association pistes/objets au temps t_k

Un algorithme d'explication du processus est présenté ci-dessous.

Algorithme :

En entrée :

- la liste des objets attendus (donc connus à l'instant d'avant)
- la listes probabilités pignistiques des nœuds si profondeur > 0 (pas racine de l'arbre)
- La liste des objets perçus au cours de l'observation courante.

Début

Mettre en place les fonctions de masse de croyance

Tant que non condition d'arrêt Faire

- A chaque observation on regarde la liste des objets
- Lister les associations possibles
- Retenir les N meilleurs associations probables (élagage en largeur)
- Calculer les probabilités pignistiques jointe de chaque association
- Sélectionner la probabilité pignistique maximale et calculer son écart par rapport aux probabilités pignistiques des autres hypothèses.
- Si condition d'arrêt en profondeur non remplit alors
 - o Retenir toutes les hypothèses dont l'écart avec le maximum est inférieur à un seuil et les ajouter à la liste des hypothèses éligibles
- FinSi
- Si une seule hypothèse alors
 - o mettre à jour la liste des connus et aller à la fin
- Sinon
 - o calculer les probabilités conditionnelles de chaque nœud de la liste et élaguer les autres noeuds
- FinSi
- Si condition d'arrêt en profondeur non remplie alors
 - o reprendre le même processus avec de nouvelles observations
- FinSi

FinTant que

Fin

La terminaison de l'arbre MHT doit forcément être suivie d'une prise de décision par rapport au suivi en prenant la meilleure branche solution de l'arbre. Dans notre cas, il s'agit tout juste de la meilleure association issue de cet algorithme.

4.4. CONCLUSION

Les caractéristiques visuelles des objets permettent de les caractériser, de les classer et de les identifier éventuellement. Il est souvent rare qu'une seule caractéristique visuelle suffise pour discriminer deux objets de même catégories. On a alors recours à un ensemble des caractéristiques donc la mise en commun permet de renforcer ou d'affiner la description d'un objet au point de le différencier d'autres objets de sa famille. Dans partie de notre thèse, nous avons présenté la modélisation des différentes caractéristiques visuelles que nous avons estimé pertinentes pour la descriptions de nos objets d'intérêt (véhicules) pour aboutir aux différentes mesures de similarité ou de dissimilarité sur lesquels nous nous basons pour construire des fonctions de masses de croyance que nous utilisons dans le TBM.

Le TBM est un bon outil de représentation, de combinaison de données et de prise de décision en présence d'incertitude. Mais il peut arriver qu'on ait besoin, en cas d'ambiguïté dans la prise de décision, de retarder temporellement cette prise de décision pour renforcer les hypothèses permettant de lever éventuellement l'ambiguïté. Pour ce faire, nous avons eu recours au MHT qui agit comme un mécanisme de fusion temporelle et construit un arbre d'hypothèses dont le parcours en largeur et en profondeur permet de lever une partie des ambiguïtés d'association de données. Pour éviter que le MHT ne dégénère en une combinatoire explosive, une optimisation de la construction de l'arbre d'hypothèse en est faite en imposant une restriction dans le temps (donc en profondeur) et en largeur en ne retenant à chaque fois que les hypothèses les plus probables des N meilleures d'hypothèses.

CHAPITRE 5 : ARCHITECTURE MULTI-AGENTS PROPOSEE

Sommaire

5.1.	Introduction	82
5.2.	Architecture logique du système de suivi	83
5.2.1.	Structures de données utilisées	85
5.3.	Architecture du Système Multi-Agents	86
5.3.1.	Modélisation de système Multi-Agents	86
5.3.2.	Outils et environnements d'implémentation.....	100
5.3.3.	Conception détaillée du système multi-agents.....	114
5.4.	Conclusion.....	123

5.1. INTRODUCTION

L'intérêt des SMA et de leur modélisation vient de leur flexibilité, de leur portabilité et de leur adaptabilité à beaucoup de types de problèmes complexes ou situations. En effet dans un système multi-caméras, leur extensibilité peut permettre de gérer de façon dynamique la configuration du système à travers des agents (SMA) : nombre de caméras, rôle de chaque caméra, les hypothèses de suivi, nombre de ressources logicielles destinées à faire les différents calculs pour atteindre les buts du système, avec la possibilité d'avoir certaines ressources redondantes et d'autres plutôt très spécialisées et très limitées en nombre ; le tout pour assurer un traitement parallèle et garantir la fluidité nécessaire au temps réel.

Dans cette partie de notre travail, nous allons présenter l'architecture logique et la structure fonctionnelle de notre système de suivi multi-caméra, et partant de cette description, arriver à identifier les différentes composantes du système. Nous allons utiliser une méthodologie d'analyse et de modélisation inspirée d'un certain nombre d'approches utilisées dans la littérature pour la modélisation de SMA. Nous présentons également nos choix d'environnement de développement multi-agents en les justifiant et en montrant comment associer ces différents outils pour atteindre nos objectifs.

5.2. ARCHITECTURE LOGIQUE DU SYSTEME DE SUIVI

Notre système est composé d'un ensemble de caméras distribuées dans un objectif de vidéosurveillance. Le système doit pouvoir réagir en temps réel en se basant sur les ressources/données mises à disposition à travers ce réseau de capteurs.

Le suivi global sera basé sur un superviseur qui s'appuie sur les données provenant de suivis locaux au niveau des différents capteurs. Les traitements bas niveau se feront au niveau de chaque capteur ou du système abstrait le représentant et seules les données haut niveau, issues de ces traitements, seront transmises au niveau superviseur ou à d'autres capteurs via un réseau IP local ou distant comme l'illustre la figure 5.1 ci-dessous.

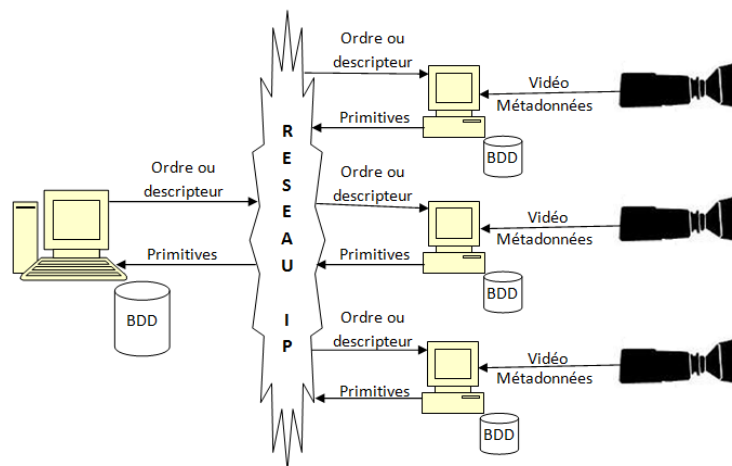


Figure 5.1: Organisation logique de notre architecture de suivi

Cette organisation décentralisée présente l'avantage de réduire les contraintes de bande passante au niveau du réseau puisque seuls les descripteurs des objets pertinents perçus et quelques autres informations haut niveau sont transmis via le réseau à destination d'un superviseur ou d'un autre capteur.

Pour la mise en œuvre fonctionnelle, la gestion des échanges d'informations ainsi que le traitement de ces informations dans un tel système, un ensemble de structures de données devra être mis en place. Ainsi le réseau de capteurs pourra être représenté par un graphe non orienté dans lequel chaque sommet représentera une caméra et chaque arête correspondra à la distance en termes de tronçon de route d'une caméra à une autre qui lui est voisine. En terme de structure de données, ce graphe sera représentée par une matrice carré M définie par :

$$M_{i,j} = \begin{cases} \text{longueur de l'arête joignant les sommets } i \text{ et } j, \text{ si l'arête existe} \\ \text{une valeur négative prédéfinie (par exemple } -1) \text{ sinon} \end{cases}$$

Ainsi pour une carte routière d'une région représentée par la figure (Fig. 5.2) ci-dessous, nous pouvons avoir la représentation sous forme de graphe représenté par la figure (Fig. 5.3).

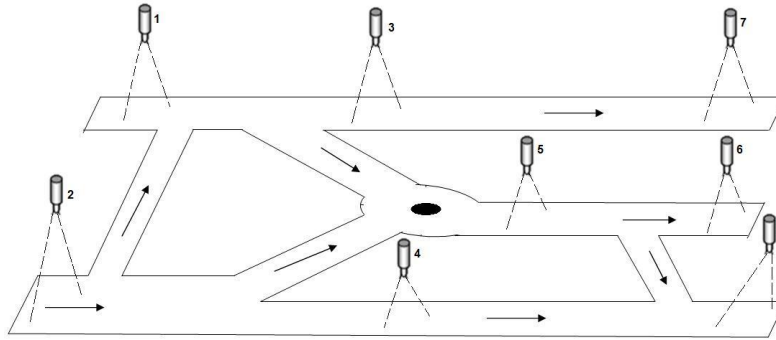


Figure 5.2: Axes routiers représentant des zones de suivi multi-caméras

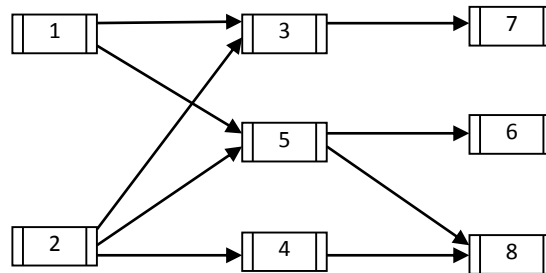


Figure 5.3: Réseau de capteurs associé au schéma ci-dessus, sous forme de graphe orienté

Les échanges et les traitements étant de plusieurs ordres, les données échangées entre les capteurs ou entre les capteurs et le superviseur seront de plusieurs catégories. Le résultat instantané du suivi local, contenu dans un agent, sera envoyé à chaque fois au superviseur ou à un nœud voisin pour un suivi global. Les informations qu'on peut retrouver dans un tel résultat seront par exemple, le numéro de la caméra qui envoie l'information, les informations d'apparence (dimension, histogramme couleur avec prise en compte d'information spatiale), les informations relatives à ses contraintes temporelles, et celles liées au statut de l'objet perçu.

Les informations d'apparence vont permettre de mettre en place l'indicateur d'identité de chaque objet perçu. Cet indicateur représente pour chaque objet perçu, le degré de croyance en ce qu'il correspond à un objet précis connu.

Les informations liées aux contraintes temporelles serviront à mettre en place des indicateurs relatifs au modèle de vitesse et d'accélération de l'objet. Ce modèle va se faire soit par apprentissage, soit par estimation approximative. Ceci permettra de mettre en place des contraintes temporelles et de gérer la durée de vie ainsi que des délais d'observation ou d'apparition d'objets à travers la définition de fenêtres temporelles [3]. Enfin, les indicateurs liés au statut de l'objet, vont nous permettre de mettre en place une stratégie de levée d'ambiguïté en cas d'occlusion.

Tous ces indicateurs vont nous permettre de définir des fonctions de masse de croyance qui sont utilisées pour la fusion d'information et la gestion de la prise de décision dans le cadre de la théorie des croyances transférables présentée plus haut.

5.2.1. STRUCTURES DE DONNEES UTILISEES

Pour la mise en œuvre fonctionnelle, la gestion des échanges d'informations ainsi que le traitement de ces informations dans un tel système, un ensemble de structures de données devra être mis en place. Nous présentons donc dans cette partie les différentes structures de données que nous nous proposons d'utiliser dans la représentation des différents types de données que nous aurons à manipuler au sein de notre système.

Les échanges et les traitements étant de plusieurs ordres, ces données seront de plusieurs catégories à savoir :

- les données haut niveau que les systèmes de caméras locales vont transmettre à chaque fois via le réseau à destination d'un superviseur ou des caméras voisines suite à leurs traitements bas niveau lors du suivi local. Ces données sont :
 - le numéro ou l'identifiant du capteur transmettant les données ;
 - la famille du capteur (distant ou co-localisé) sur la base du contexte ;
 - le nombre total d'objets d'intérêt perçus par le capteur ;
 - pour chaque objet perçu,
 - ses coordonnées (donc sa position),
 - sa distance qui le sépare du capteur (pour des besoins de mise à l'échelle),
 - sa durée de vie dans le champ de vision du capteur. Cette durée est estimée à partir d'un intervalle temporel flou qui prend en compte le modèle de vitesse et celui d'accélération de l'objet,
 - ses informations d'apparence telles que : ces dimensions, son histogramme des couleurs avec prise en compte d'informations spatiales,
 - son statut : seul ou dans un groupe du fait d'une occlusion,
 - son degré de visibilité.
- Les informations stockées au niveau de chaque nœud (objet capteur) :
 - le type de capteur ;
 - les coordonnées du capteur ;
 - son champ de vision (défini par une fonction mathématique) ;
 - la précision de ses mesures ;
 - la liste de ses proches voisins ;
 - les distances du capteur à chacun de ses proches voisins.
- La structure de données contenant réellement toutes les données (messages) que nos différents capteurs vont s'échanger. Cette structure de données aura comme minimum de champ,
 - Émetteur (adresse IP)

- Récepteur (adresse IP)
- Identifiant
- Nombre de sauts restant
- Langage utilisé
- L'ontologie (Utilisée pour identifier la source du vocabulaire; le moyen pour le receveur d'interpréter le contenu du message)
- Le type de message (liste des types de message)
- Contenu du message

5.3. ARCHITECTURE DU SYSTEME MULTI-AGENTS

L'architecture logicielle que nous voulons mettre en place doit nous permettre de piloter le matériel de suivi multi-caméras, de faire les associations de données (mise en correspondance) pour aboutir à la ré-identification de véhicule à partir de caractéristiques visio-temporelles.

L'objectif du système est donc de permettre de faire du suivi multi-caméras avec des zones non couvertes (zones aveugles) et d'arriver à ré-identifier au niveau d'une caméra des véhicules précédemment identifiés par une autre caméra.

Chaque caméra du système de caméras distribuées étant chargée de capturer le flux vidéo de la scène observée, sera considérée dans notre architecture comme une source de données vidéo. Ces données pouvant donc être des données temps-réel ou des données pré-filmées au niveau des différentes caméras d'observation et jouées de façon synchrone dans notre système de sorte à virtualiser la réalité terrain.

Dans notre cas, nous avons opté pour la seconde solution de fait de n'avoir pas à disposition toute la logistique (matérielle, logicielle, réseau) nécessaire pour des tests complets à partir des données du terrain. Toutefois, ces données qui sont essentiellement des captures vidéo de plusieurs caméras, seront les sources exploitées

Dans nos tests, nous nous sommes contenté dans un premier temps de 2 cameras positionnées sur un segment d'une voie sans intersection, puis dans un second temps, nous sommes passé à trois ou quatre caméras réparties de part et d'autre d'un circuit routier disposant d'un carrefour.

5.3.1. MODELISATION DE SYSTEME MULTI-AGENTS

Le cycle de vie d'un projet démarre généralement par la phase d'émergence de ses besoins et objectifs. Cette phase est suivie d'une phase d'analyse sur la base du cahier des charges et de l'environnement. Viennent ensuite les phases de conception, de réalisation ou d'implémentation, puis celle de clôture ou de livraison.

Aussi bien dans la phase d'analyse que dans celle de conception, il est souvent conseillé d'adopter une ou plusieurs méthodologies afin de mieux identifier et formaliser de façon structurée les différentes étapes devant permettre de passer des spécifications du projet à sa réalisation concrète, tout en garantissant la cohérence dans l'enchaînement des étapes. Une telle démarche offre également l'avantage de mettre à disposition une documentation descriptive de tout le processus ayant conduit la réalisation depuis les spécifications jusqu'à la livraison. Elle présente également, suivant le cas, l'avantage d'une abstraction dans la conceptualisation opérationnelle en se détachant de toutes contraintes liées à un outil ou environnement spécifique d'implémentation.

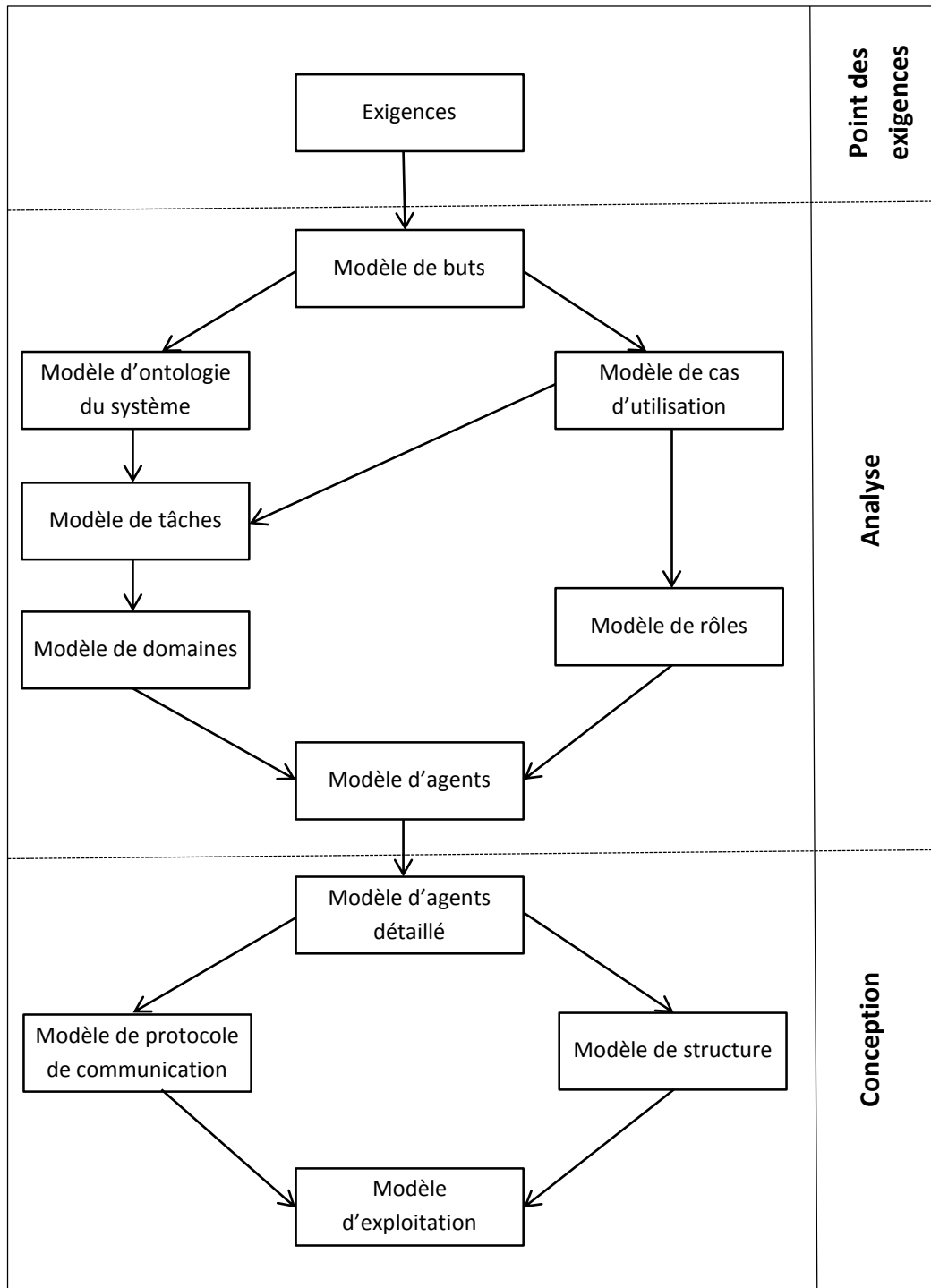
Pour ce qui nous concerne, c'est-à-dire le cas de la modélisation de SMA, il existe plusieurs méthodologies ainsi que plusieurs plateformes de modélisation des SMA mais aucune des méthodologies n'est à ce jour vraiment normalisée de façon à être un standard adopté de tous pour analyser et concevoir des SMA. De même, la plupart des plateformes implémentent une catégorie d'agents ou de SMA de sorte à ne répondre ou couvrir que les besoins de programmation de certains domaines et ne permettent ainsi que la mise en œuvre de certains types d'applications.

Parmi les méthodologies et les langages de modélisation multi-agents qui existent, on peut citer AALAADIN, ADELFE, GAIA, MASCommonKADS, MaSE, OMASEINGENIAS, PASSI, ROADMAP, TROPOS, VOYELLES, Agent UML, MAS-ML, AML pour les plus connus. De même on distingue parmi les plateformes de développement multi-agents, les outils suivants : CORMAS, DIMA, GEAMAS, MAGIQUE, MADKIT, MASK, MERCURE, MOCAH, OSACA, SYNERGIC, ADELFE, JADE, AGENT, ZEUS.

Notre objectif dans cette partie n'est pas de faire l'état de l'art des différentes approches mais de tirer profit des points forts de certaines de ces approches pour construire notre démarche d'analyse, de modélisation et de conception de notre SMA.

Ainsi, après quelques recherches et études, nous avons fait le constat que quelle que soit l'approche retenue, on remarque généralement trois grandes étapes dans le cycle de développement d'un SMA. Une première étape correspond à la présentation ou la définition des exigences et des besoins du système à mettre en œuvre (Phase d'identification). Il s'en suit une seconde étape correspondant à la phase d'analyse de ces exigences en suivant une méthodologie pour dégager les différents organisations ou groupes ainsi que les différents rôles joués par les acteurs du système qui ne sont rien d'autres que les agents (Phase de définitions). La troisième qui est celle de conceptualisation opérationnelle, est une étape qui permet d'apporter des détails sur la structure, les caractéristiques et les modèles de communication ou d'interaction des différents types d'agents entre eux ou avec l'environnement, en vue de leur implémentation conformément aux différentes spécifications (Phase de conception). A ces trois étapes, est ajoutée celle de réalisation ou de mise en œuvre. Les éléments essentiels de ces trois étapes sont présentés sur la figure 5.4 ci-dessous.

Figure 5.4: Etapes méthodologiques de conception et de réalisation



Suivant la complexité du système à mettre en place, certaines composantes de ces étapes peuvent être modélisées ou non en fonction de leur pertinence et de leur apport en terme de précision pour la phase de mise en œuvre.

Nous adoptons cette démarche méthodologique dans la suite de cette session pour la modélisation du SMA que nous envisageons mettre en œuvre.

5.3.1.1. Point des exigences du système en termes de fonctionnalités

Comme présenté plus haut, les exigences fonctionnelles du système se résument en :

- Permettre l'acquisition de flux vidéo provenant de plusieurs sources (caméras) distribuées sur des axes routiers dans le cadre de suivi multi-caméras,
- Permettre la détection locale de véhicules au niveau de chaque source,
- Permettre la création locale de caractéristiques visio-temporelles (signatures) de chaque véhicule détecté en vue d'en faire un suivi global,
- Permettre de faire de l'association de données et la prise de décision à l'aide du TBM avec possibilité de levée d'ambiguïté par déploiement du MHT.

La prise en compte de ces exigences va nous permettre de conduire la phase d'analyse pour déboucher sur la modélisation complète du système puis sa réalisation.

5.3.1.2. Les étapes de la phase d'analyse

Identification des buts du système : Diagramme des buts

On peut définir un but comme étant l'énoncé prescriptif d'une intention qu'un système doit satisfaire grâce à la coopération de ses agents [95]. L'identification des buts d'un système se fait à partir d'un ensemble de besoins fonctionnels. Dans notre cas, le but principal du système est de permettre de faire du suivi multi-caméras de véhicules avec ré-identification. Ce but est celui du système global. Au niveau des suivis locaux nous avons des buts locaux qui participent à la concrétisation du but global. Ces buts peuvent être considérés comme des sous-buts ou des buts partiels ou élémentaires. On peut en citer principalement trois que sont : détecter des véhicules, identifier de façon unique chaque véhicule détecté, et enfin arriver à associer les véhicules perçus avec ceux connus.

La réalisation de chacun de ces trois buts est subordonnée à celle de leurs sous buts respectifs, jusqu'à arriver aux buts feuilles.

La figure 5.5 présente le diagramme des buts ainsi que les relations entre ces buts. Ainsi, on note sur ce diagramme, en dehors des buts, les relations de composition (décomposition « and ») ainsi que celles de précédence (« precedes ») et de déclenchement (triggers()) qu'il y a entre les buts.

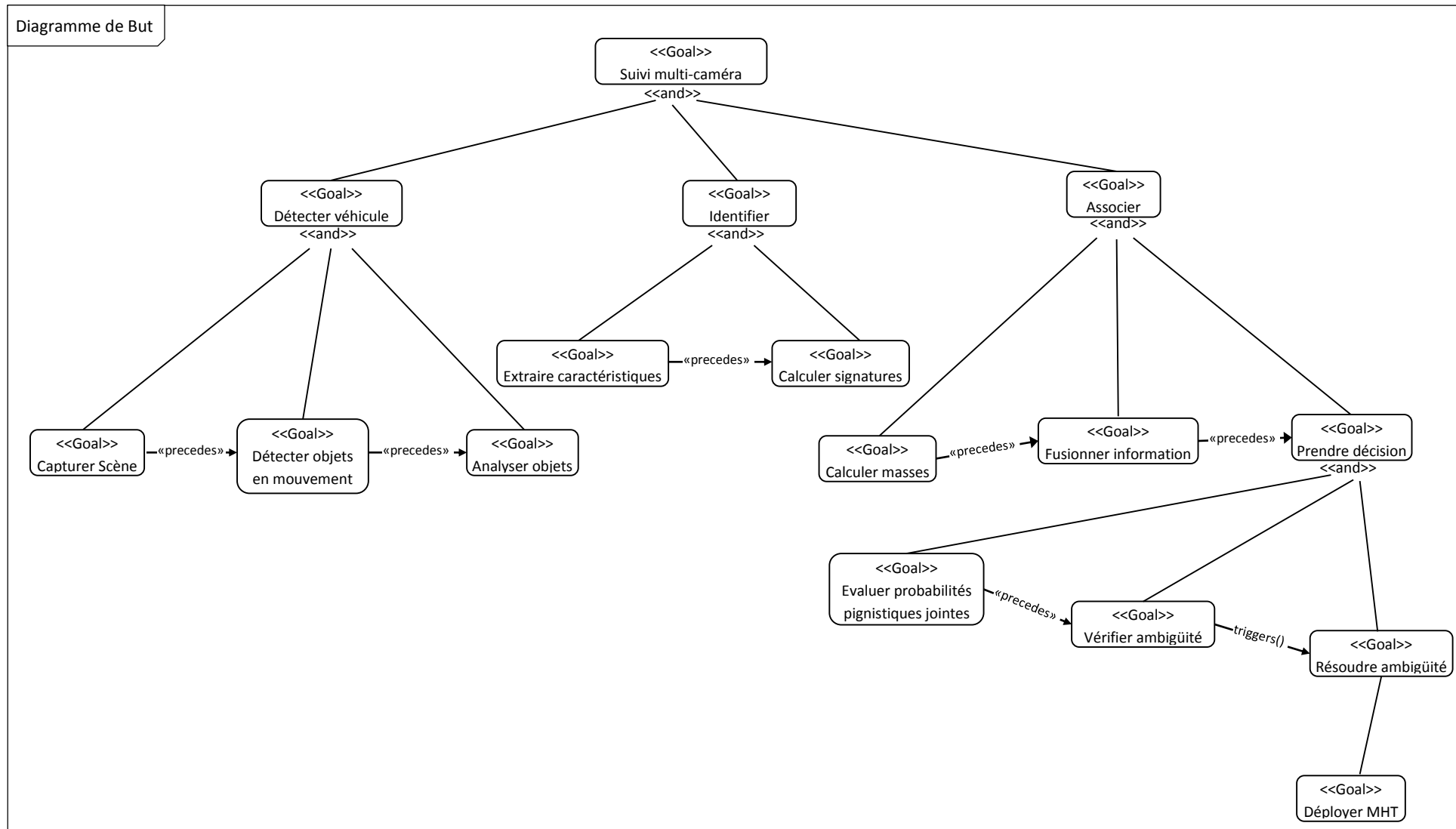


Figure 5.5: Diagramme des buts du système

Identification des rôles des agents du système : Diagramme des rôles

Le diagramme de rôle définit les rôles d'agents, les relations éventuelles entre les rôles d'agents. Le rôle d'un agent décrit le comportement de ce dernier dans une organisation vis-à-vis des autres agents et de son environnement pour atteindre ou réaliser un ou plusieurs buts. Le stéréotype «achieves» permet de relier le rôle au(x) but(s) qui permet de réaliser.

Pour la réalisation d'un but, un rôle doit être doté («requires») d'une ou de plusieurs capacités («capability»). La capacité d'un agent est la spécification de ce qu'il est en mesure de faire et dans quelles conditions. Chaque capacité se traduit par un plan d'exécution décrivant un comportement. Un agent peut avoir plusieurs capacités, et la mise en œuvre d'une capacité à un instant lui confère à cet instant un état qui est induit par son comportement (rôle) à l'instant. Comportement qui peut changer à un autre instant du fait d'un nouveau rôle si l'agent en a les facultés.

A partir du diagramme des buts (figure 5.5) nous avons pu identifier un certain nombre de rôles permettant de réaliser les différents buts. Il s'agit de :

- Extraction vidéo
- Reconnaissance
- Prototypage
- Modélisation croyances
- Analyse probabilités
- Déploiement MHT

La figure 5.6, présente la liste des rôles ainsi que les différentes relations de contraintes et celles de réalisations auxquelles ils sont assujettis.

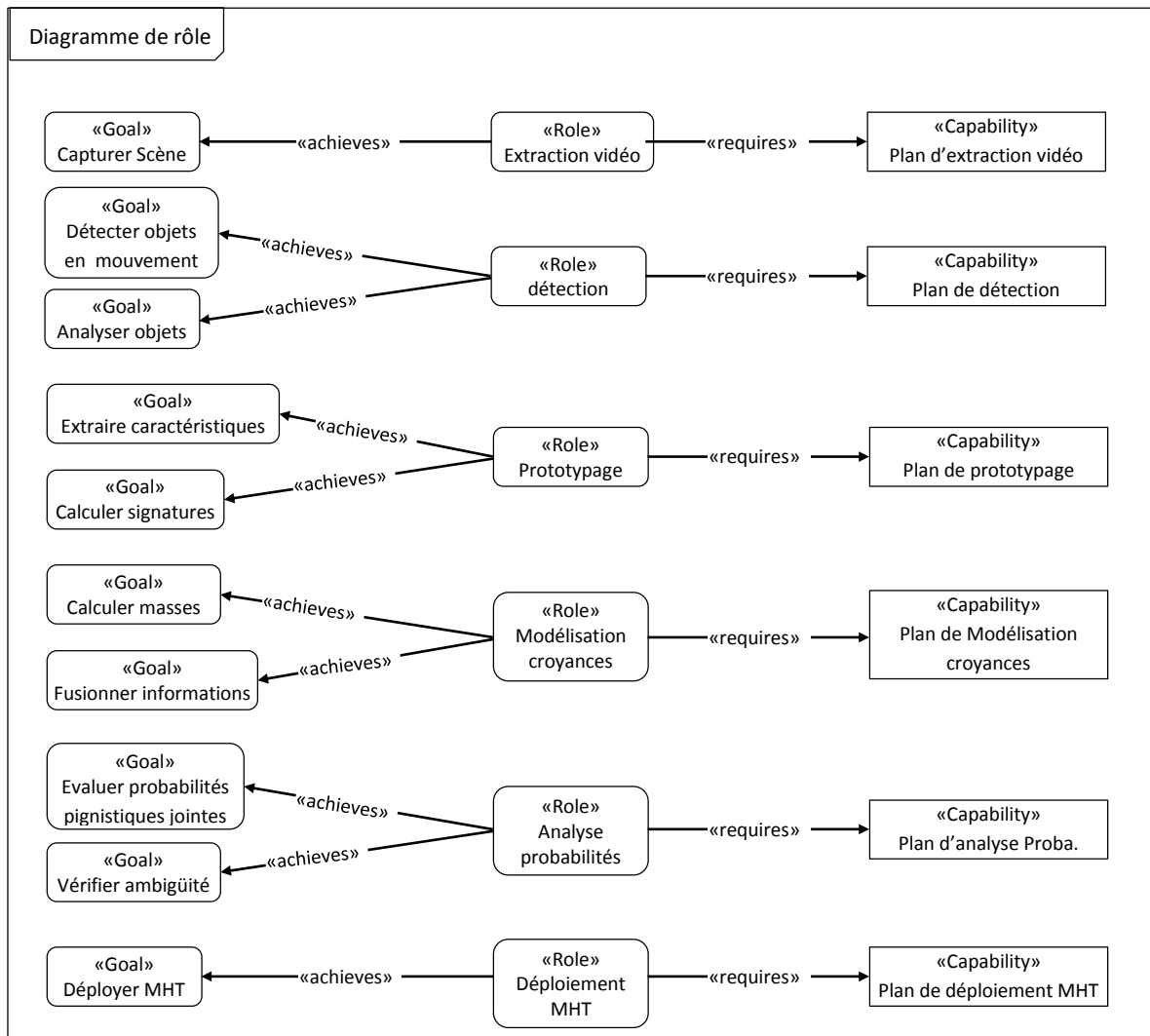


Figure 5.6: Diagramme des rôles des agents du système

Identification des cas d'utilisation : Diagramme des cas d'utilisation

Le diagramme de cas d'utilisation est un diagramme d'UML 2.0 qui montre les interactions fonctionnelles entre les acteurs et le système étudié. En général les acteurs sont des acteurs externes (humain ou non, ou autres systèmes externes). Un cas d'utilisation est un ensemble de scénarii liés par un but utilisateur commun. Un scénario est une séquence d'étapes décrivant une interaction entre un utilisateur et le système.

Malgré que les agents d'un SMA ne soient pas des acteurs externes au système auquel ils appartiennent, nous avons décidé de présenter ce diagramme pour donner un aperçu d'un certain nombre de buts des agents du système et les relations fonctionnelles entre ces buts et les agents. Le diagramme de cas d'utilisation peut permettre de valider les buts du système et déboucher éventuellement sur un raffinement des rôles qui en découlent.

La figure 5.7 donne une présentation synthétisée du diagramme de cas d'utilisation de notre système multi agent de suivi multi-caméras.

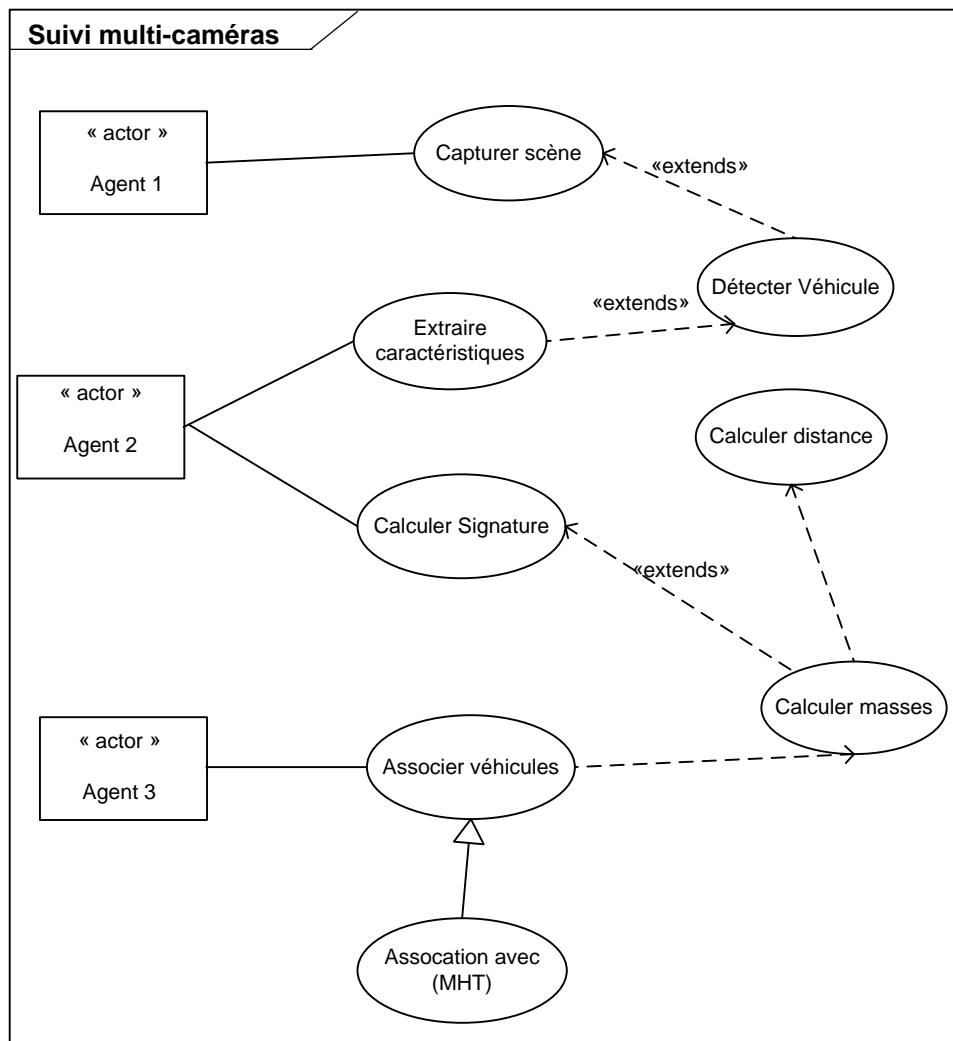


Figure 5.7: Diagramme synthétisé de cas d'utilisation du système

Identification des agents : Diagramme d'agents

Le système multi-agents à mettre en place est constitué d'un ensemble d'agents représentant des entités actives qui agissent et interagissent dans un environnement logiciel commun. Les véhicules sont considérés comme de simples objets du système qu'il faut identifier et suivre. L'action des agents, suivant leurs catégories, consistera à percevoir leur environnement, à analyser les données qu'ils reçoivent pour y détecter les véhicules, à transformer les données pour produire des métriques et des fonctions de masse de croyance, à synthétiser ces informations en les fusionnant, à prendre des décisions d'association sur la base de leurs croyances. Les données produites par certains agents sont transmises et utilisées par d'autres agents dans un processus de collaboration pour la résolution d'un problème global de suivi multi-capteurs.

Le cycle de vie d'un agent peut se résumer en trois aptitudes que sont :

- perception ;
- décision ;
- action.

Un agent, suivant qu'il est cognitif ou réactif possède ou non des aptitudes à raisonner à la fois sur ses connaissances et sur ses croyances. Dans tous les cas, il s'agit de connaissances opératoires qui peuvent consister, à interpréter un signal ou à réagir à un message en provenance d'un autre agent ou de l'environnement.

Pour identifier les agents de notre système et mettre en place de diagramme d'agent, nous partons des buts et des rôles des composantes du système ; mais également d'un certain nombre de caractéristiques non exhaustives d'un agent que sont :

- l'identifiant
- le(s) rôle(s)
- l'(les) organisation(s)
- la (les) capacité(s)
- le(s) service(s)

L'identifiant permet désigner de façon unique chaque agent dans un système ou une organisation. L'identifiant fait partie des paramètres pouvant permettre de retrouver ou même de reconnaître un agent dans un processus de coopération.

Le rôle définit le comportement de l'agent dans la société à un moment donné ou de façon permanente dans l'accomplissement d'un ou de plusieurs buts. Un agent peut jouer un ou plusieurs rôles, comme plusieurs agents peuvent jouer le même rôle et donc seront en compétition sur ces rôles.

L'organisation définit les relations entre les rôles. Un agent peut faire partir de plusieurs organisations dans lesquelles il peut jouer le même rôle ou des rôles différents. Une organisation peut être définie comme, un agencement de relations entre composants ou individus ; agencement qui produit une unité, ou système, doté de qualités inconnues au niveau des composants ou individus. L'organisation lie de façon interrelationnelle des éléments ou événements ou individus divers qui dès lors deviennent les composants d'un tout. Elle assure solidarité et solidité relative, donc assure au système une certaine possibilité de durée en dépit de perturbations aléatoires.

La capacité spécifie ce qu'un agent est en mesure de faire et dans quelle(s) condition(s).

Le service décrit l'activité ou prestation qu'un agent peut effectuer ou offrir, et qu'il propose aux autres agents.

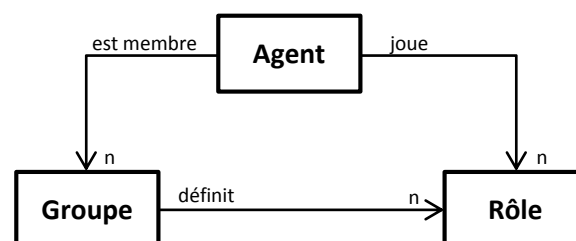


Figure 5.8: Modèle organisationnel des agents

Partant de toutes ces considérations, nous sommes arrivé à la proposition de quelques agents et du diagramme d'agents de la figure 5.9 qui décrit les interactions pouvant exister entre eux. On y voit, les différents agents, leurs capacités ainsi que les relations et interactions qu'il y a entre les agents du SMA.

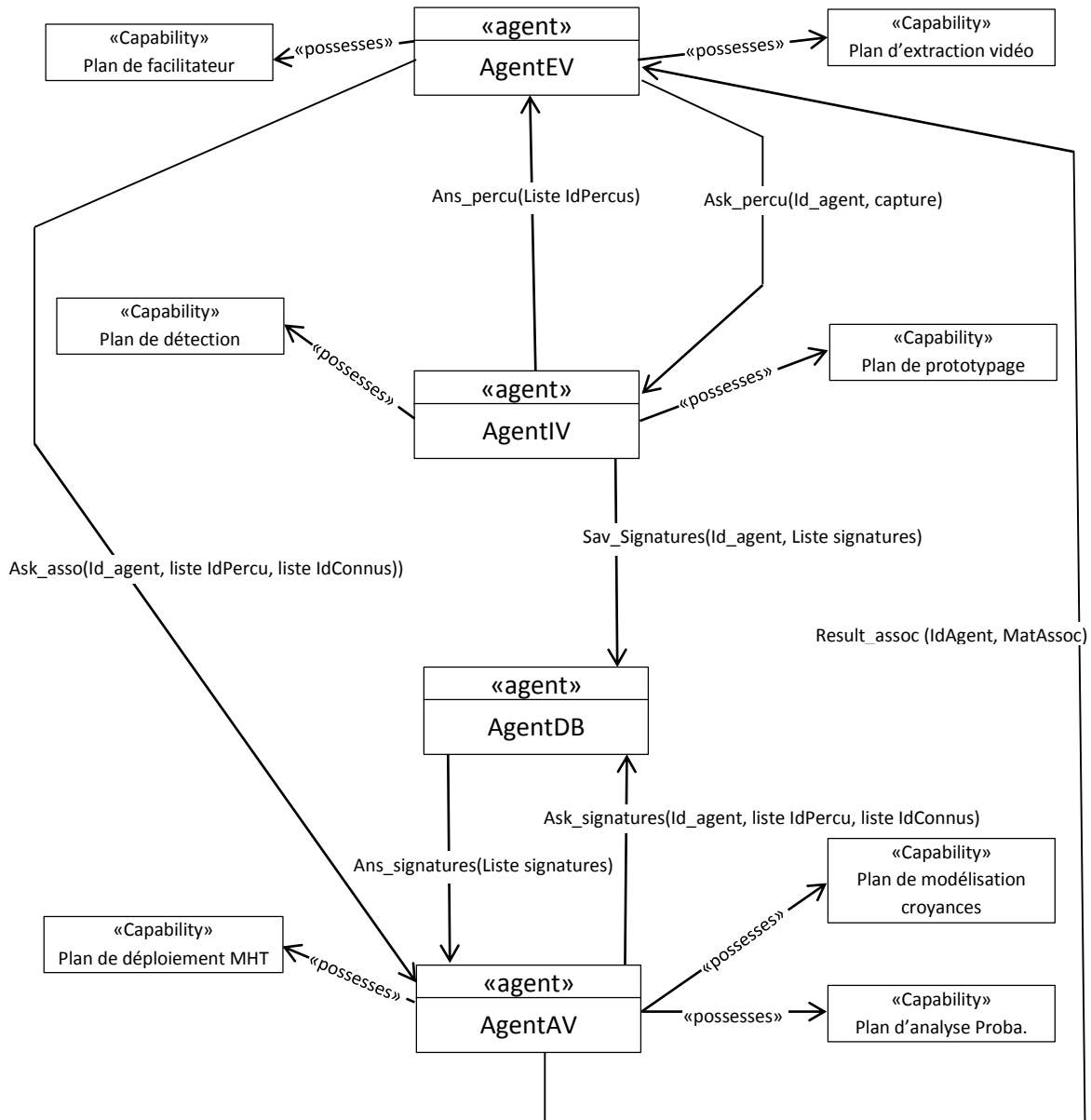


Figure 5.9: Diagramme d'agents

Agents Extraction Vidéo, noté agentEV : Ils sont directement associés aux différentes caméras du système et travaillent directement au niveau de chaque caméra sur les données brutes produites par celle-ci. Il y en aura autant que de caméras dans le système de suivi multi-caméras.

Agents d'analyse vidéo, noté AgentAV : ils sont chargés d'analyser en temps-réel un flux vidéo issu d'une caméra et mis à disposition par les agentEV à travers un processus de coopération.

Ils se basent sur un certain nombre de primitives pour détecter dans ces flux, les objets d'intérêt (des véhicules) pour les extraire et en calculer les signatures.

Agents Ré-Identificateur de Véhicules noté **AgentIV** : ils ont en charge la finition du processus de suivi global. A cet effet, ils jouent plusieurs rôles.

Agent de Production de Masse : Il aura pour rôle le calcul des masses de croyance. Cet agent peut ne pas exister si nous décidons d'ajouter ses rôles à d'autres agents de notre système.

Agent facilitateur ou page jaune : dialogue régulièrement avec les agents chargés d'exécuter les tâches pour connaitre leurs capacité et disponibilité puis les stocke dans sa base de données afin de proposer leurs services à ceux qui en feront la demande pour la réalisation d'une tâche.

5.3.1.3. Description fonctionnelle du système

La description fonctionnelle du système va se présenter à travers la présentation des différents plans que nous avons mis en place pour accomplir les différentes tâches du système. Un plan est défini par une séquence d'instructions permettant d'atteindre un but.

Les figures 5.10 à 5.15, présentent sous forme d'automates ou de diagrammes d'état-transition, les différents plans des agents du système.

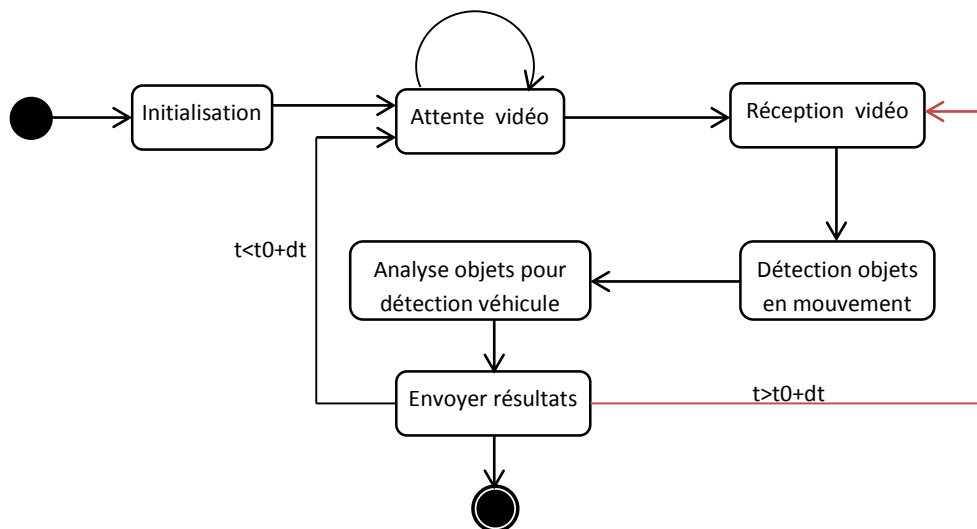


Figure 5.10: Plan de détection

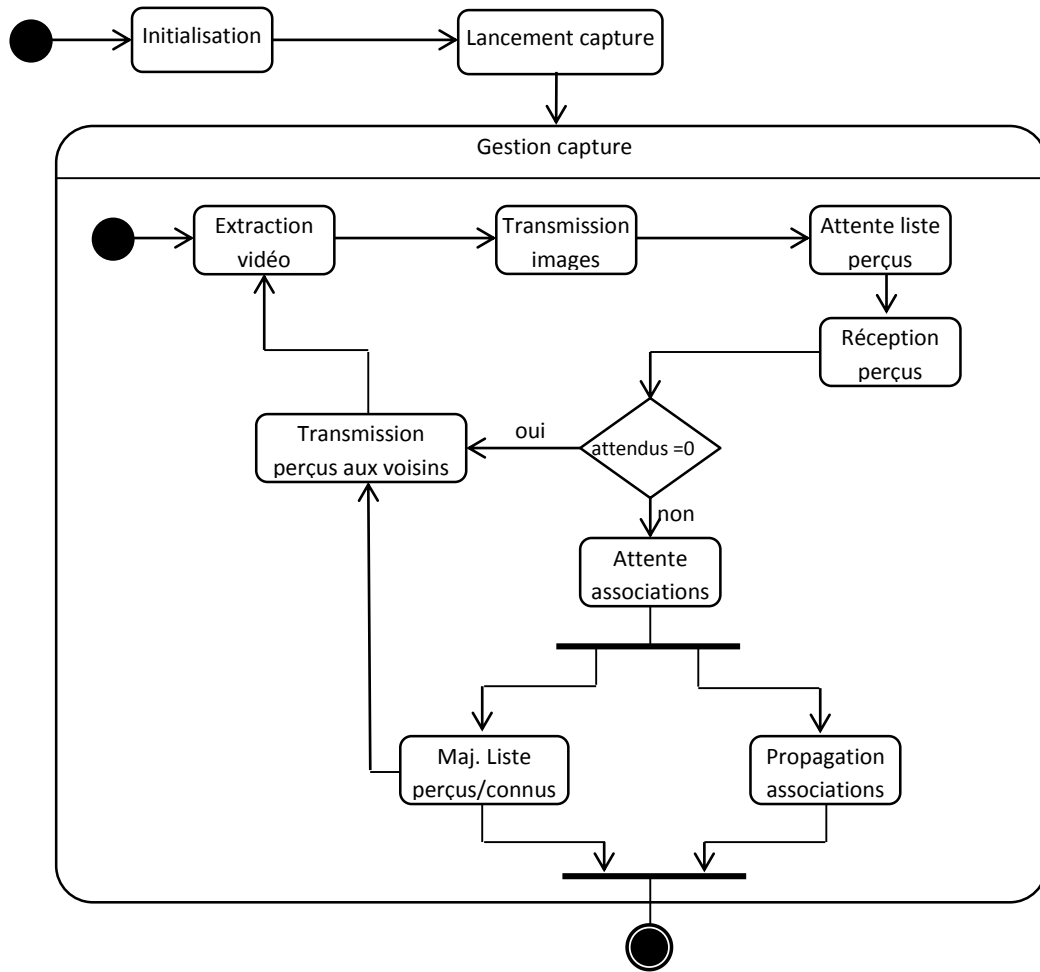


Figure 5.11: Plan d'extraction et de gestion suivi local

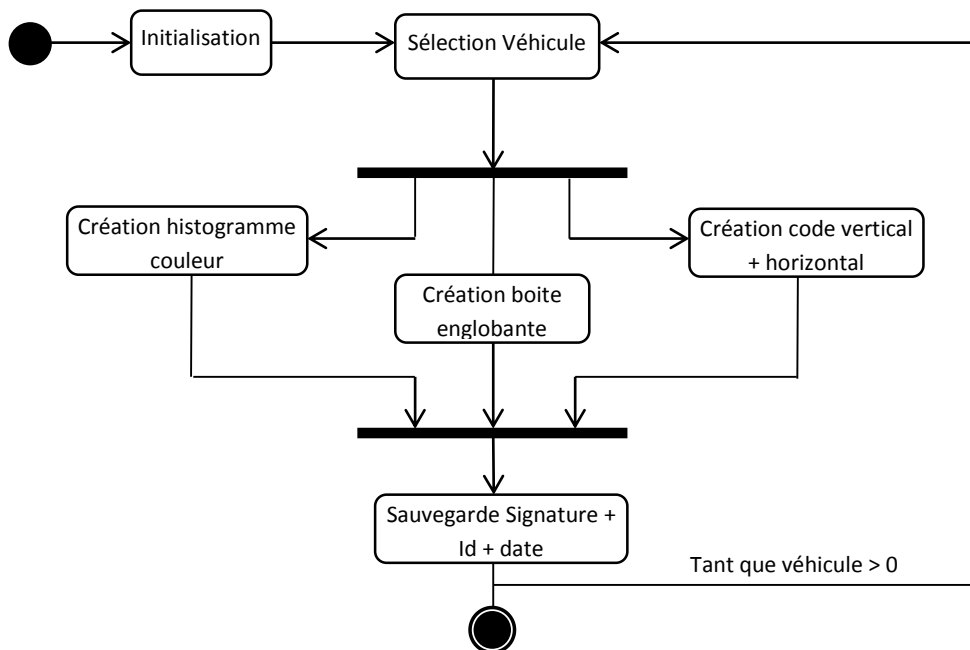


Figure 5.12: Plan de prototypage

Figure 5.13: Plan de modélisation croyance

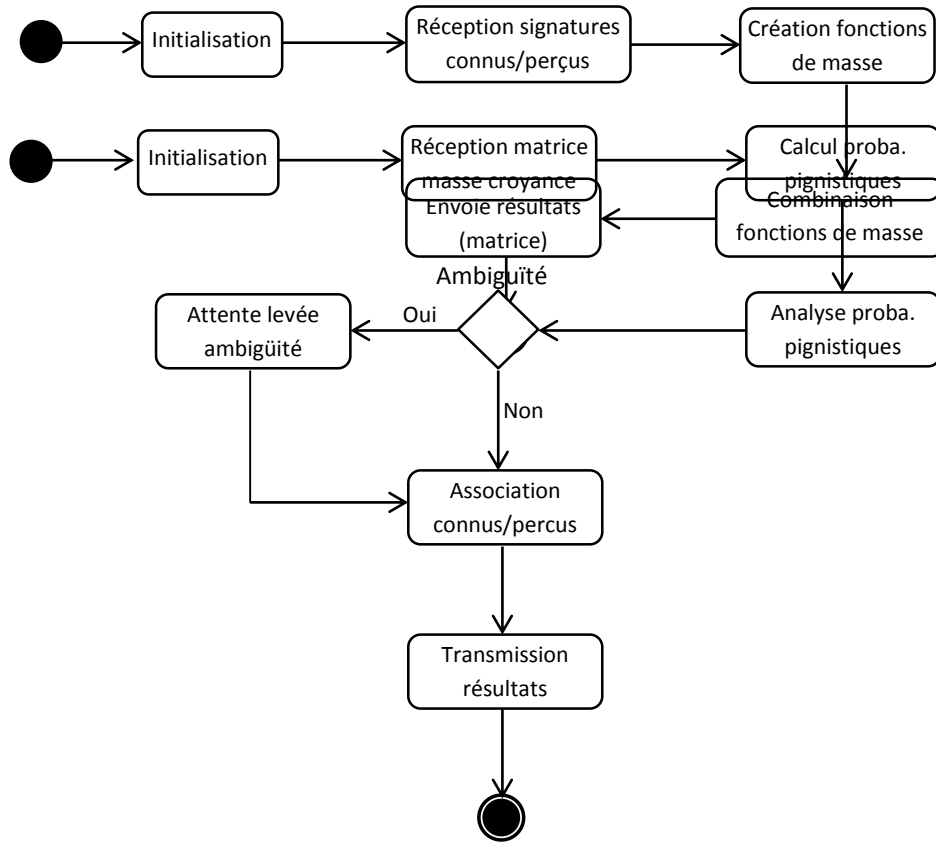


Figure 5.14: Plan d'analyse probabilité/Association

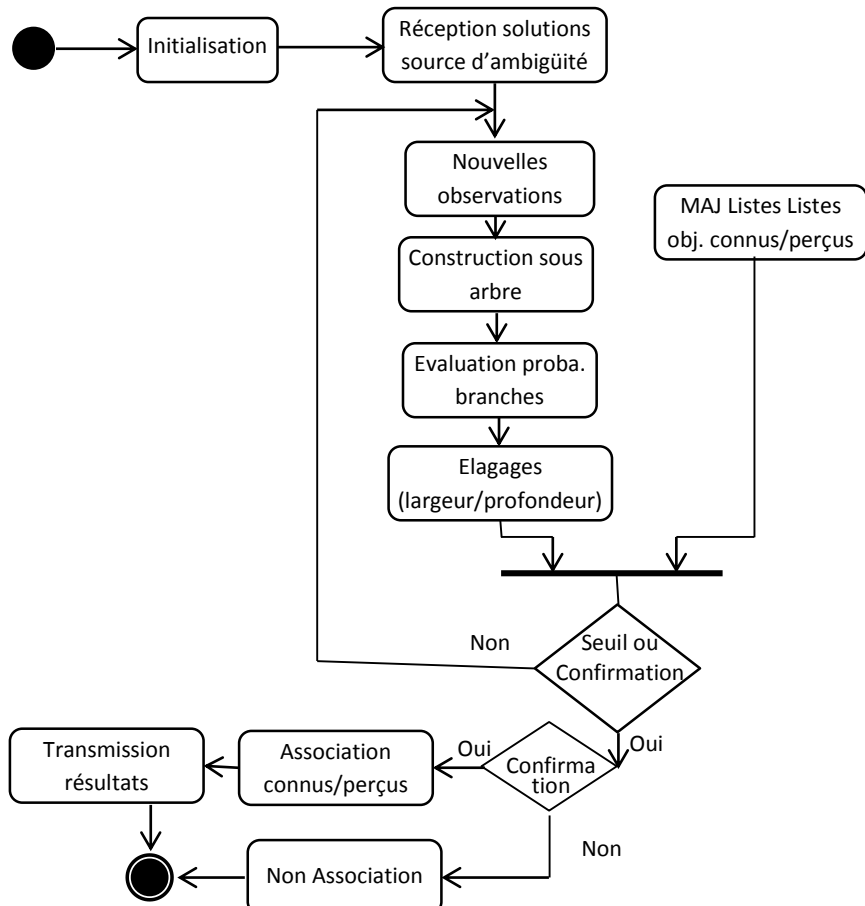


Figure 5.15: Diagramme de plan de déploiement MHT

5.3.2. OUTILS ET ENVIRONNEMENTS D'IMPLEMENTATION

Au-delà des concepts et des différentes approches méthodologiques qui restent des outils théoriques servant de support dans la phase de modélisation des SMA, la phase d'implémentation est une phase cruciale puisque c'est elle qui permet la concrétisation SMA. Cette phase de mise en œuvre est généralement assez difficile et complexe et nécessite donc de faire des choix judicieux d'outils et d'environnements de développements en privilégiant certains critères de facilitation de développement ainsi que d'autres liés à la facilité de création et de gestion d'environnement multi-agents. Le meilleur procédé pour mettre en place de façon rapide, efficace et cohérente un SMA reste l'utilisation d'une plate-forme conçue à cet effet. Parmi ces critères on peut citer : la méthodologie utilisée, la facilité d'apprentissage, la souplesse de l'outil, la disponibilité d'outil de débogage, le support au développement, la génération de code, l'extensibilité du code, la facilité de déploiement, la disponibilité de la documentation, le support implémentation, la facilité de gestion du SMA, la communication inter-agent.

Dans cette partie de notre travail, nous présentons nos choix en y apportant les justificatifs ayant conduit à les adopter.

5.3.2.1. Etude comparatives de quelques outils et environnements

Parmi les outils et environnements les plus connus on peut citer par ordre alphabétique : AgentBuilder, AgentTool, DECAF, Jack, JADE, JAFMAS/JIVE, Madkit, Zeus.

Une actualisation d'une étude comparative qui avait déjà été faite auparavant nous a permis de mettre en place le tableau suivant sur la base de l'évolution à ce jour de chacun de ces outils.

Critères \ Outils	AgentBuilder	AgentTool	DECAF	Jack	JADE	JAFMAS/JIVE	Madkit	Zeus
la méthodologie utilisée	4	3	0	0	0	3	3	4
la facilité d'apprentissage	1	3	3	0	0	1	2	1
la souplesse de l'outil	1	0	0	3	3	2	3	1
la disponibilité d'outil de débogage	4	2	2	0	3	1	4	4
le support au développement	4	4	2	1	0	2	2	4
la génération de code	1	1	0	0	0	1	0	3
l'extensibilité du code	1	0	1	4	4	1	3	2
la facilité de déploiement	2	1	1	2	4	1	3	2
la disponibilité de la documentation	4	1	1	3	3	1	3	4
le support implémentation	4	2	0	1	0	2	1	4
la facilité de gestion du SMA	3	1	0	0	4	0	4	3
la communication inter-agent	4	2	2	3	4	2	3	4
Note finale (sur 48)	33	20	12	17	25	17	31	36

Tableau 5.1 : Tableau comparatif d'outils de développement de SMA

Si on doit s'en tenir à cette étude, il ressort de ce tableau que les deux meilleurs outils sont dans l'ordre Zeus et AgentBuilder. Cependant, dans le cadre de ce travail, nous avons décidé d'écarter ces deux outils malgré leurs différents atouts. En effet, ces outils qui sont les plus complets du domaine sont également les plus complexes, les plus contraignants et les moins souples. Ils demandent beaucoup de temps pour leur maîtrise qui exige entre autres l'appropriation préalable et obligatoire de langage ou de technique de modélisation sur lequel chacun d'eux se base dans la phase de mise en œuvre. De plus, ils présentent des limites par rapport à leur déploiement, leur extensibilité ainsi que leur réutilisabilité. En outre, l'utilisation de KQML comme langage de communication a été également un critère qui a joué à notre niveau en défaveur de ces outils en raison des critiques formulées (plus haut) à l'endroit de ce langage.

Enfin, lorsqu'on analyse de plus près le tableau comparatif, on se rend compte que certains critères vont de pair avec le critère relatif à la méthodologie utilisée. De ce fait, leur présence dans ce tableau pénalise à plusieurs titres tous les outils n'intégrant pas du tout la prise en charge de ce critère. En ne prenant pas en compte les deux critères marqués en rouge dans le tableau ci-dessus, on se rend compte qu'il y a deux autres outils qui peuvent retenir notre attention en dehors des deux premiers soulignés plus haut. Il s'agit de Madkit et de JADE.

5.3.2.2. Choix de JADE

Avant de présenter notre choix et d'apporter les justificatifs qui le motivent, nous tenons à citer Michael Wooldridge dans l'avant-propos de l'ouvrage " *Multi-Agent Programming : Languages, Platforms and Applications*", Springer, 2005. Ainsi que John-Jules Ch. Meyer dans l'avant-propos de l'ouvrage " *Multi-Agent Programming : Languages, Tools and Applications*" Springer, 2009.

Je cite,

Michael Wooldridge : *Every programmer knows that what makes a "good" programming language is at least in part a kind of magic: there is an indefinable "rightness" to the best languages, that make them somehow easier, more fun, more natural, just better to use.*

John-Jules Ch. Meyer : *Researchers are struggling to find the best way to program these highly complex but potentially very useful systems! In fact, by a very bold kind of induction (1, 2, many!), I dare forecast that the present volume will not be the final word on this matter either. It is my strong conviction that, as with so many things in life, including programming in a broad sense, there is no unique best way to program MAS, no unique language or platform that is evidently the optimal choice for every situation, and that, in the end, the best choice of language to use will heavily depend on the application at hand.*

Fin de citations.

Le plus grand reproche formulé à l'endroit de JADE reste qu'il ne se base sur aucune spécification méthodologique et ne met pas vraiment à disposition d'outils graphiques d'assistance à l'implémentation et/ou au développement. Son interface graphique se limite à la gestion et supervision des agents. Dans le même temps, et nonobstant ces reproches qu'on lui oppose, JADE présente certains avantages très importants et très déterminants qu'on ne saurait occulter. Il s'agit de la facilité de déploiement, l'extensibilité du code, la facilité de communication des agents et la facilité de gestion des SMA dans l'environnement. Au-delà de ces avantages, JADE présente plusieurs autres avantages qui le démarquent de plusieurs plateformes.

A la lumière de ces constats notre choix s'est donc porté sur JADE dont les insuffisances au niveau conceptuel sont cependant contournables en adoptant ou en élaborant soi-même une démarche méthodologique cohérente ; et c'est ce que nous nous avons eu à faire plus haut en privilégiant une méthodologie qui fait abstraction de toutes les exigences ou contraintes liées à un outil ou environnement spécifique de mise en œuvre.

Nous allons donc présenter ci-dessus quelques caractéristiques et fonctionnalités de JADE qui en font pour nous l'outil idéal et l'un des plus ouverts et libres conformément à nos choix d'implémentation.

JADE est une plateforme logicielle ou un framework implémenté en JAVA et permettant le développement et la gestion de SMA en respect des normes FIPA [131]. Sa création a été motivée par le besoin de disposer d'un outil répondant aux normes FIPA et permettant de valider les spécifications de cette norme relative aux SMA. Il est open source depuis 2000 et est distribué sous la licence LGPL (Library Gnu Public Licence). Il est donc disponible en libre téléchargement ainsi que sa documentation et un ensemble d'exemples sous forme de codes sources. Ces contributeurs sont très nombreux, assurant ainsi son évolution et un support en cas de problèmes. Contrairement à beaucoup d'autres plateformes, il continue d'être amélioré et sa version actuelle (jade 4.3.1) est de mars 2013.

Sa conformité aux normes de la FIPA exige de lui de disposer tout au moins :

- d'un canal de communication entre Agents (**ACC**, Agent Communication Channel) ;
- d'un système de gestion des agents (**AMS**, Agent Management System) ;
- d'un annuaire facilitateur ou page jaune (**DF**, Directory Facilitator);
- du support du langage de communication FIPA-ACL.

JADE fournit une couche homogène qui permet de simplifier l'implémentation des fonctionnalités basiques voire complexes d'applications distribuées utilisant l'abstraction d'agents logiciels en restant conforme aux exigences des normes FIPA. Il est composé d'un environnement d'exécution des Agents (runtime Environment) qui héberge les agents ; d'une librairie de classes pour faciliter la mise en œuvre de SMA ; et d'une suite d'outils graphiques pour une meilleure gestion et supervision de la plateforme des agents.

JADE garantit une interopérabilité transparente et une compatibilité totale avec toutes plates-formes entièrement conformes aux spécifications FIPA2000 qui constituent le cadre normatif dans lequel les agents FIPA peuvent exister, fonctionner et communiquer, tout en adoptant une architecture interne indépendante. A ce jour, JADE est pratiquement la seule plate-forme multi-agents qui propose une bibliothèque de protocoles d'interaction facilitant au programmeur l'implémentation d'agents interactifs.

5.3.2.3. Présentation de l'architecture de JADE

La figure 5.16 ci-dessous présente l'architecture d'une plateforme JADE. Elle est composée d'un ensemble de conteneurs qui sont des instances JADE. Ces conteneurs peuvent être sur la même machine, comme elles peuvent être réparties sur plusieurs machines d'un réseau. Chaque instance ou conteneur peut héberger/contenir plusieurs agents. Chaque plateforme JADE contient obligatoirement un conteneur principal que nous désignerons dans la suite ce document par MC (Main Container) auprès duquel s'enregistrent tous les autres conteneurs dès leur lancement. Chaque conteneur est identifié par un nom logique qui lui est associé.

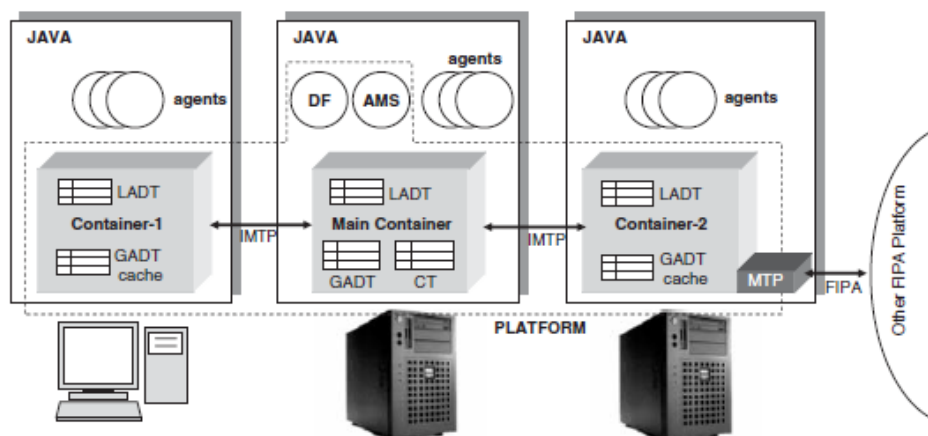


Figure 5.16: Architecture d'une plateforme JADE

CT = Container Table

GADT = Global Agent Descriptor Table

LADT = Local Agent Descriptor Table

AMS = Agent Management System

DF = Directory Facilitator

IMTP = Internal Message Transport Protocol

MTP = Message Transport Protocol

Cette architecture répond aux exigences de la norme FIPA [132] relative à la conception de SMA. En effet, les spécifications FIPA97 de la norme FIPA, établissent les règles normatives permettant aux agents d'inter-opérer dans un SMA. Ces spécifications décrivent le modèle de référence d'une plate-forme multi-agents où ils identifient les rôles de certains agents clés nécessaires à la gestion de la plate-forme, et spécifient le contenu du langage de gestion des agents et l'ontologie du langage.

Le MC a en charge la gestion de la table des conteneurs enregistrés (CT) ; la gestion de la table de description globale des agents (GADT) ; hébergement du AMS et du DF. En effet, le MC, au-

delà des autres agents qu'il peut héberger, contient toujours deux agents spéciaux AMS et DF qui sont lancés automatiquement à son démarrage.

Chaque conteneur dispose localement de sa propre table locale de description de ces agents (LDAT) de sorte que lors de la recherche d'un agent, le conteneur consulte d'abord sa table LDAT et c'est seulement en cas d'échec qu'il se réfère au MC pour obtenir de la GADT les références distantes de l'agent puis les enregistre dans son cache (GADT Cache) pour un éventuel usage futur.

Le **AMS** fournit un service de nommage unique et représente l'autorité qui supervise la plateforme en assurant la gestion du cycle de vie des agents (création, attente, destruction, etc.). Chaque agent, à sa création, doit s'enregistrer à son niveau pour obtenir un identifiant unique et valide qui lui servira d'authentifiant lors de ces accès et utilisations du système. Quant au **DF**, il fournit un service de pages jaunes permettant aux agents d'enregistrer leurs services ou de retrouver les agents fournisseurs de services dont ils ont besoin.

Chaque conteneur dispose localement de sa propre table locale de description de ces agents (LDAT) de sorte que lors de la recherche d'un agent, le conteneur consulte d'abord sa table LDAT et c'est seulement en cas d'échec qu'il se réfère au MC pour obtenir de la GADT les références distantes de l'agent puis les enregistre dans son cache (GADT Cache) pour un éventuel usage futur.

Le système étant dynamique, la gestion des éventuelles migrations, terminaisons et/ou apparitions de nouveaux agents est prise en compte et les différentes mises à jours des tables sont effectuées pour garantir la continuité de service voire la tolérance aux pannes en mettant en place au besoin un mécanisme de réplication.

Une représentation de la plateforme, mettant en exergue les mécanismes de communication et d'interaction entre agents, est présentée à la figure 5.17 ci-dessous.

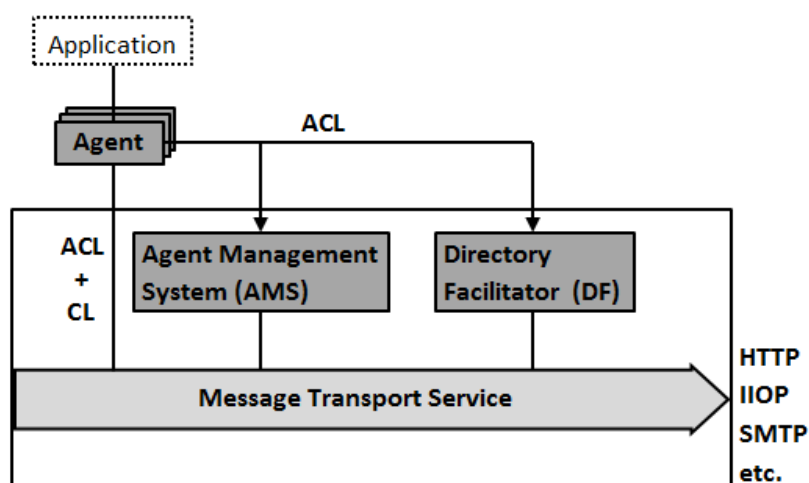


Figure 5.17: Plateforme multi-agents aux normes FIPA

JADE contient les bibliothèques de Java (classes) qui sont requises pour le développement des applications à base d'agents. Chaque agent a un AID (Agent Identifier) qui est un identificateur

qui est une structure de données qui lui est associée à sa création, qui le caractérise de façon unique et qui intègre son nom et son adresse. Chaque agent est autonome et proactif et dispose de son propre thread dans le système à sa création.

5.3.2.4. Les agents et leurs comportements dans JADE

Tout système SMA est mis en place pour résoudre au moins un problème. Il a donc des objets à satisfaire ou des buts à atteindre. Pour atteindre ces buts, chaque agent se voit attribuer des rôles. Pour jouer un rôle donné, un agent doit avoir un comportement et sa capacité se mesure à l'ensemble des tâches (rôles) qu'il peut accomplir ou comportements qu'il peut avoir. JADE permet d'implémenter des agents cognitifs ou réactifs et chaque agent peut se voir ajouté un ou plusieurs comportements (behaviours). En effet, à la création les agents sont comme des coquilles vides auxquelles il faut ajouter des comportements implémentant des services/fonctionnalités. Les comportements dans JADE sont classifiés en plusieurs catégories. Ainsi distingue-t-on aussi bien des comportements simples que des comportements composés. La figure 5.18 ci-dessous présente la hiérarchie entre les comportements en termes de généralisation et de spécialisation.

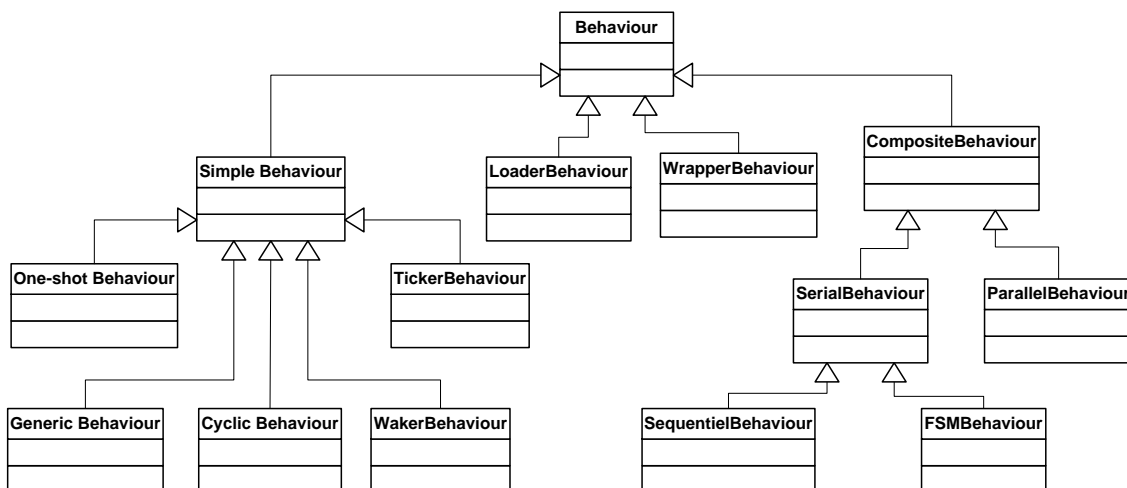


Figure 5.18: Modélisation UML de la classification hiérarchique des comportements

Les comportements simples (classiques) peuvent être de cinq sortes dont deux sont des comportements planifiés :

- **les one-shot behaviour** (éphémères) qui ne s'exécutent qu'une seule fois ;
- **les cyclic behaviour** (cycliques) qui s'exécutent de façon répétitive ;
- **les generic behaviour** dont la planification du comportement et de la terminaison est laissée à la discrétion du programmeur.
- **le WakerBehaviour** permet d'exécuter une tâche après un intervalle de temps défini par le programmeur ;
- **le TickerBehaviour** permet d'exécuter périodiquement une tâche en définissant sa périodicité.

Les comportements composés sont destinés à l'exécution de tâches complexes nécessitant l'association de plusieurs sous-comportements. Cette association permet d'obtenir un comportement composite et peut se faire suivant trois approches :

- **Les SequentielBehaviour** permettent d'organiser les sous-comportements de façon séquentielle ou linéaire. Ainsi, l'exécution du comportement composite commence par celui du premier sous-comportement, s'enchaîne avec le suivant de la liste dès que le sous-comportement courant prend fin, et ainsi de suite pour se terminer juste après l'exécution du tout dernier sous-comportement.
- **Les FSMBehaviour** permettent d'organiser les sous-comportements comme des états d'un automate à états finis. Le comportement composite est donc vu comme cet automate à états finis. Il est composé d'un seul état initial, mais peut avoir plusieurs états intermédiaires ou finaux. Le passage d'un état à un autre est une transition dont l'avènement est subordonné à la survenue d'un événement.
- **Les parallelBehaviour** permettent de créer des comportements composites qui exécutent en parallèle plusieurs sous comportements. Les sous-comportements sont donc lancés les uns à la suite des autres sans attendre le précédent ne termine son exécution.

Chaque agent est doté d'une capacité et les différents rôles qu'il joue ou peut jouer au sein d'une organisation témoignent de cette capacité. Un agent peut donc être amené à jouer un ou plusieurs rôles pour atteindre ou participer à atteindre un but. Autrement, il peut avoir des comportements (behaviour) différents en fonction du contexte ou du but à atteindre. De même, un rôle peut être joué parallèlement par plusieurs agents de façon concurrente, complémentaire ou alternative. Tout dépend de la complexité, de la criticité ou de la fréquence de ce rôle et des ressources disponibles.

Dans JADE, lorsqu'un agent doit accomplir plusieurs tâches (jouer plusieurs rôles) il revient au programmeur d'ordonner ces tâches dans sa méthode *setup()* qui permet d'initialiser l'agent, d'enregistrer dans la page jaune ses capacités et de démarrer le tout premier comportement. Les différentes opérations effectuées par un comportement sont définies dans sa méthode *action()* et la fin de l'exécution ou l'état d'exécution d'un comportement est retournée par sa méthode *done()*. Lorsque la valeur de cette dernière est à *true*, l'exécution du comportement est supposée terminée et est retiré par l'agent de son ordonnancement. Les méthodes *setup()* et *takeDown()* jouent respectivement le rôle de constructeur et de destructeur dans un agent et il est donc recommandé de ne plus implémenter ces méthodes dans la classe de l'agent. La figure 5.19 ci-dessous présente le cycle de vie d'un agent et de l'ordonnancement de l'exécution des comportements d'un agent.

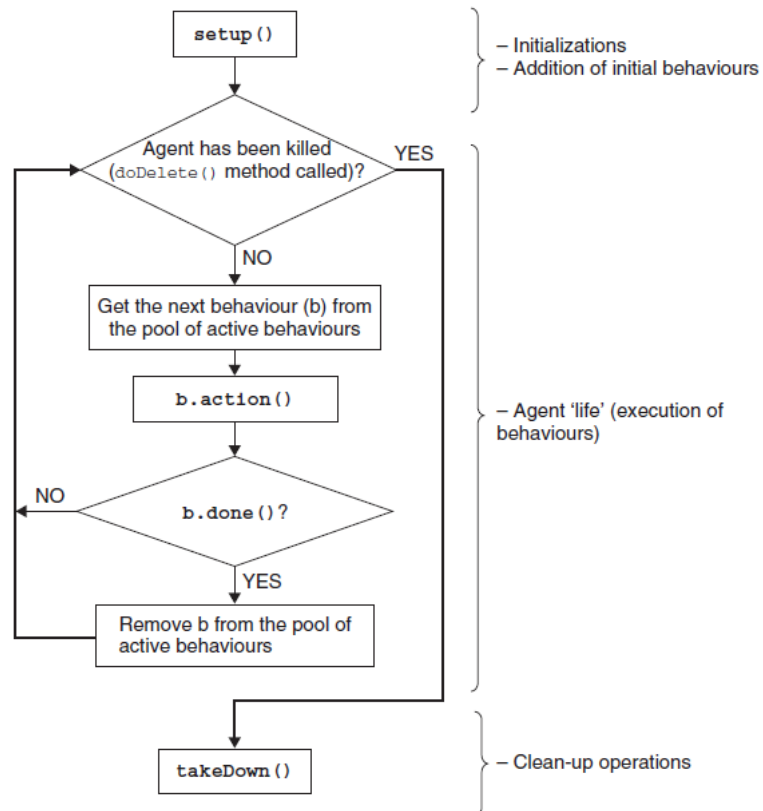


Figure 5.19: Cycle de vie d'un agent et de ses comportements dans JADE

Lorsqu'un agent est créé, il est initialisé, mais n'est pas encore actif. Il ne peut dans cet état faire aucune action car non encore enregistré. Il devient actif par son invocation (appel). La figure 5.20 ci-dessous on a la présentation du cycle de vie d'un agent et son diagramme d'états-transitions dans JADE.

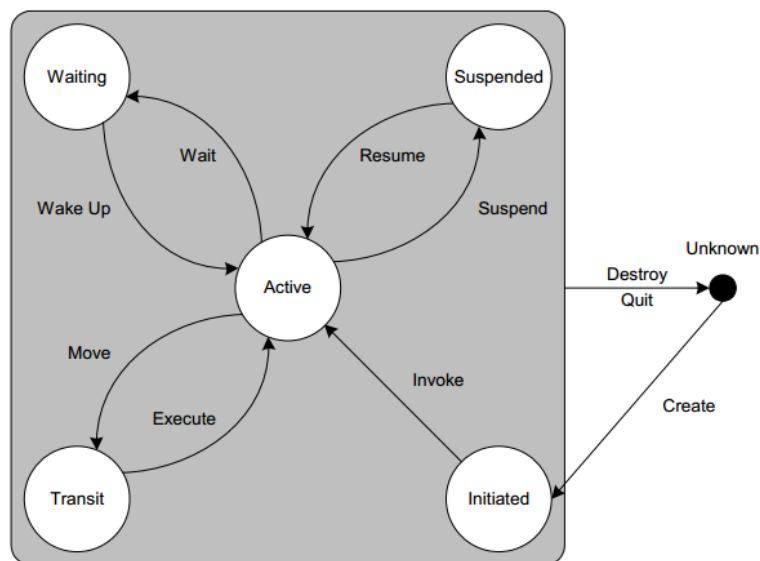


Figure 5.20: Cycle de vie d'un agent tel que défini par FIPA

L'activation d'un agent lui permet de s'enregistrer auprès de l'agent AMS de la plateforme JADE. Dès lors, un nom (identifiant unique) et une adresse lui est associé et il peut exécuter ses comportements sur la base de sa planification et communiquer avec les autres agents de son environnement suivant ses sollicitation et ses désirs. L'état actif est un état central et le seul peut exécuter des comportements (Behaviours) ou transiter vers un autre état en dehors de sa fin de vie.

5.3.2.5. La Communication dans JADE

La communication inter-agents est probablement la caractéristique la plus fondamentale de JADE et elle est mise en œuvre conformément aux spécifications FIPA. Le paradigme ou modèle de communication de JADE est basé sur la transmission de message asynchrone. Ainsi, chaque agent dispose d'une «boîte aux lettres» qui est une file d'attente de messages d'agents vers laquelle l'environnement d'exécution de JADE achemine les messages envoyés à l'agent en question par d'autres agents. Chaque fois qu'un message arrive dans la file d'attente, une notification est envoyée à l'agent concerné. Cependant, le principe d'autonomie de l'agent fait qu'il peut décider de récupérer et de traiter ou non un message de sa file d'attente. Toutefois, la programmation de ce mécanisme est laissée aux choix conceptuels du programmeur. Le processus de communication est illustré à la figure 5.21 ci-dessus.

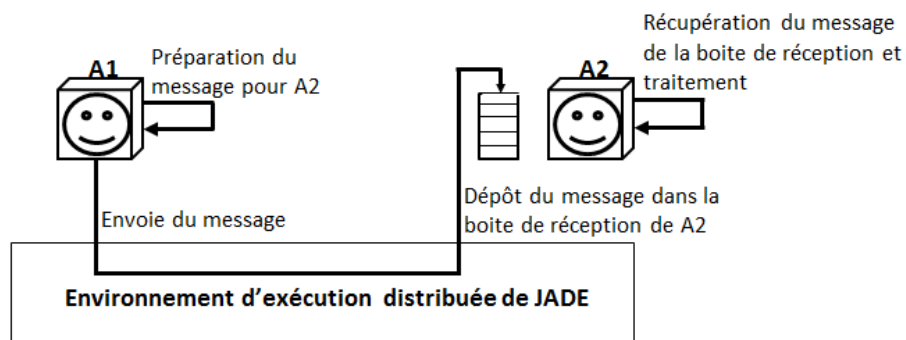


Figure 5.21: Mécanisme de communication par envoi de messages

Le format particulier des messages dans JADE est conforme à celui du FIPA-ACL. Chaque message contient les champs suivants :

- l'émetteur (sender) ;
- le(s) destinataire(s) (receivers) ;
- l'acte de communication ou Performatif : renseigne sur l'intention de l'émetteur ;
- le contenu (content) : correspond à l'information à échanger ou véhiculer ;
- le langage du contenu (content language) : notifie la syntaxe utilisée pour exprimer le contenu. Pour une communication efficace, il est important que l'émetteur et le destinataire puisse encoder et analyser les expressions du message conformément à cette syntaxe ;

- l'ontologie (ontology) : indique le vocabulaire des symboles utilisés dans le contenu. Pour une communication efficace, aussi bien l'émetteur que le destinataire, doivent interpréter de la même façon les symboles utilisés ;
- certains champs supplémentaires sont utilisées pour contrôler plusieurs conversations simultanées (Gestion concurrence) et pour préciser l'expiration du délai de réception de réponse telle que : conversation-id, reply-with, in-reply-to, reply-by...

Un message de JADE est un objet de la classe *jade.lang.acl.ACLMessage* qui met à disposition toutes les méthodes (accesseur et mutateurs) pour accéder à l'ensemble des champs ou attributs spécifiés par le format FIPA-ACL et listés ci-dessus. Tous les performatifs définis dans la spécification FIPA sont mappés comme des constantes dans la classe *ACLMessage*. La classe *jade.core.Agent* met à la disposition de chaque agent (instance de cette classe) la méthode *send()*. Cette méthode permet à l'agent, lorsqu'il l'invoque, d'envoyer de message à d'autres agents.

Ainsi, pour envoyer un message, l'agent crée une instance de la classe *ACLMessage*, invoque dessus les différentes méthodes pour préparer le message à savoir : *addReceiver()*, *setLanguage()*, *setOntology*, *setContent()*, etc. ; puis invoque sa propre méthode *send()* pour envoyer le message ainsi préparé.

Tous les messages destinés à un agent sont stockés pour lui dans une liste de type FIFO par l'environnement d'exécution de JADE. Pour la lecture des messages, l'agent invoque sa méthode *receive()* qui lui retourne un objet de la classe *ACLMessage* contenant toutes les informations relatives au message en tête de liste, puis le pointeur de la liste se positionne sur le message suivant sauf en cas de fin de liste.

Un agent peut définir des modèles de messages qu'il voudrait lire. Ainsi, lors de l'appel de la méthode *receive()*, il lui passe en paramètre le modèle souhaité pour le lire que le premier message de sa boîte aux lettres, conforme à ce modèle, si un tel message existe.

5.3.2.6. Protocoles d'interaction dans JADE

FIPA propose des normes et des spécifications pour les systèmes orientés agent et pour l'interaction entre agents. Ces spécifications servent de canevas pour la conception et le développement d'applications orientées agents et participent à accroître l'interopérabilité entre les agents ainsi que les applications orientées agents. Les spécifications relatives à la communication inter-agents définissent les protocoles d'interaction entre les agents.

La spécification d'un protocole d'interaction se traduit concrètement par la mise en place d'un scénario ou d'une séquence bien définie de messages échangés entre les participants (agents). Le respect de cette séquence permettra de caractériser ou d'identifier le protocole mis en œuvre. Le choix du protocole ou des protocoles à implémenter est fonction de chaque situation concrète et des besoins. Chaque protocole spécifie l'enchaînement des messages

entre les agents. Ils définissent les messages en donnant des précisions sur le performatif à utiliser, les destinataires du message, l'ontologie utilisée ainsi que le contenu du message.

Un des avantages de la plateforme JADE est qu'elle implémente les principales spécifications de protocoles d'interaction proposées par FIPA dans FIPA-ACL.

Pour toutes conversations entre agents, JADE distingue le rôle d'initiateur (l'agent qui commence la conversation) de celui du participant ou répondeur (l'agent qui entre dans une conversation après avoir été contacté par un autre). JADE fournit pour les deux rôles des classes de comportement (Behaviour) prêt à l'emploi pour suivre la plupart des conversations des protocoles d'interaction FIPA. Ces classes sont disponibles dans le package `jade.proto` et libèrent le programmeur de la charge de la mise en œuvre de tous les contrôles relatifs à la circulation des messages lorsque deux ou plusieurs agents interagissent. Ces classes fournissent un certain nombre de méthodes que les programmeurs peuvent redéfinir au besoin pour prendre en compte les spécificités associées au domaine qui ne puisse être généralisée. Tous les comportements de l'initiateur prennent fin et sont supprimés de la file d'attente des tâches de l'agent dès qu'ils atteignent un état final du protocole d'interaction. Cependant, dans un souci de réutilisation des objets Java représentant ces comportements, tous les initiateurs implémentent un certain nombre de méthodes de réinitialisation (`reset()`) avec les paramètres appropriés. Notons qu'à chaque comportement de communication de l'initiateur on peut associer la réaction simultanée de plusieurs intervenants. Mais tous les comportements de répondeurs, au contraire, sont cycliques et ils sont réordonnés automatiquement via la méthode `reset()` dès qu'ils atteignent un état final du protocole d'interaction ; offrant ainsi au développeur la possibilité de limiter le nombre maximum de comportements d'intervenants que l'agent doit exécuter en parallèle.

Pour chaque acte de communication, la norme FIPA précise les préconditions de l'envoi d'un message ainsi que le l'effet normal attendu ou le but poursuivi par l'initiateur de l'acte de communication. Dans un scénario normal d'un acte de communication, une fois les préconditions remplies et le désir ou but du message précisé, l'initiateur envoie le message au destinataire(s) ; déclenchant ainsi l'acte de communication. Lorsque c'est un message qui nécessite une réponse : un REQUEST ou un Call For Proposals (CFP) par exemple, le ou les destinataires peuvent y répondre de trois façons. Soit le destinataire n'a pas compris la demande reçue alors il en informe l'expéditeur à travers sa réponse ; soit il ne veut pas rendre le service demandé et il en donne la raison ; soit il accepte d'offrir le service demandé sans exclure dans ce cas la possibilité d'échec.

Néanmoins, il peut arriver qu'un message ne produise pas l'effet attendu du fait de l'autonomie et de la liberté de l'agent destinataire de répondre ou non au message conformément à son modèle de lecture de messages (priorité ou choix). Pour pallier cette situation non recommandable, il est souhaitable de faire en sorte le protocole d'interaction permet à l'initiateur du message de vérifier si son message a atteint ou non l'objectif visé. Ceci lui permet de savoir quel comportement adopter pour atteindre son objectif.

Il existe plusieurs protocoles de communication ou d'interaction parmi lesquels on a FIPA-Request, FIPA-Request-When, FIPA-Subscribe, FIPA-Query, FIPA-Contract-Net, FIPA-Propose, FIPA-Interated-Contract-Net, FIPA-English-Auction, FIPA-Dutch-Auction, FIPA-Brokering, FIPA-Recruiting.

Ci-dessous, nous présentons très brièvement quelques-unes de ces protocoles qui nous intéressent ainsi que leurs diagramme UML qui en présentent les descriptions symboliques. Tous les détails à propos des différentes spécifications des protocoles d'interaction de la norme FIPA sont disponibles sur le site de la FIPA.

Le protocole FIPA-Request s'utilise généralement pour demander à un ou plusieurs agents d'accomplir une tâche, de collecter les résultats et de les intégrer. Le protocole FIPA-Request-When en est une variante. Le protocole FIPA-Subscribe est quant à lui utilisé pour établir un accord de notification avec un autre agent pour que ce dernier envoie une notification (alerte) à chaque fois qu'une condition est remplie. FIPA-Query permet à un agent de demander une information à un autre agent soit sous forme de demande de valeur de vérité d'une information, soit sous forme de demande d'informations à propos d'un objet bien identifié. FIPA-Contract-Net est un protocole d'interaction à travers lequel un agent (généralement un coordonnateur) peut lancer un appel d'offre à l'endroit de plusieurs autres agents (les contractants qui s'étaient enregistrés comme fournisseur de ce service).

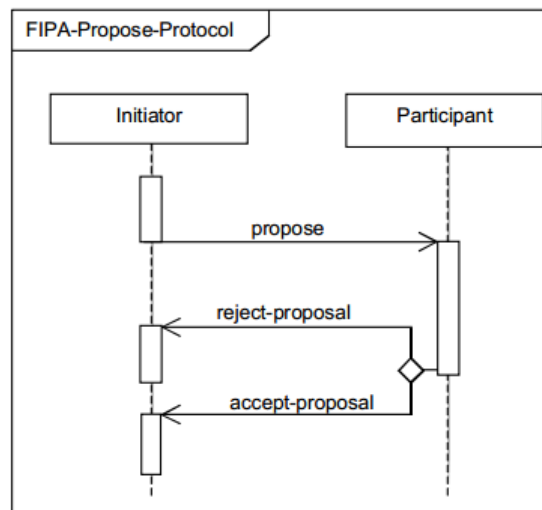


Figure 5.22: Diagramme UML du protocole FIPA-Propose

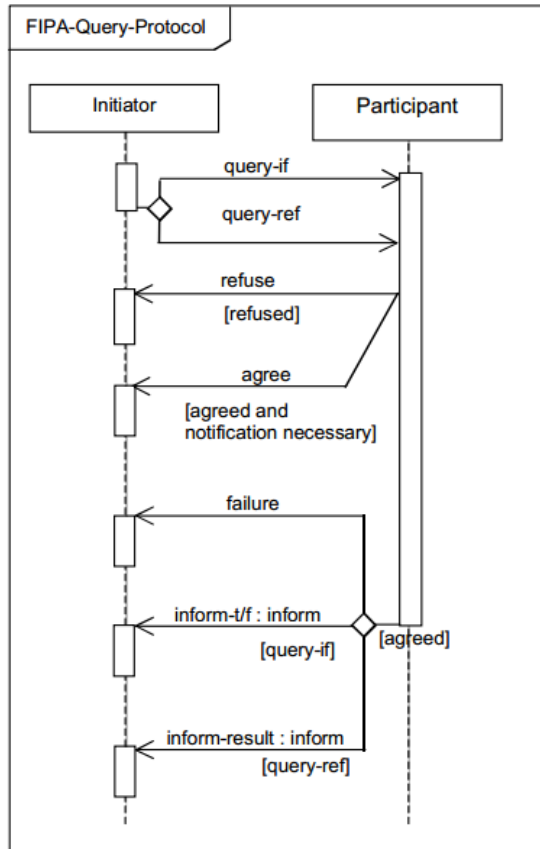


Figure 5.23: Diagramme UML du protocole FIPA-Query

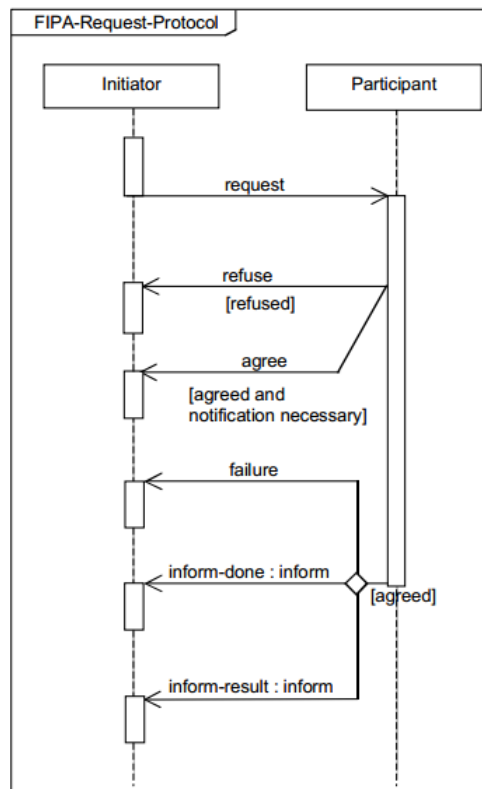


Figure 5.24: Diagramme UML du protocole FIPA-Request

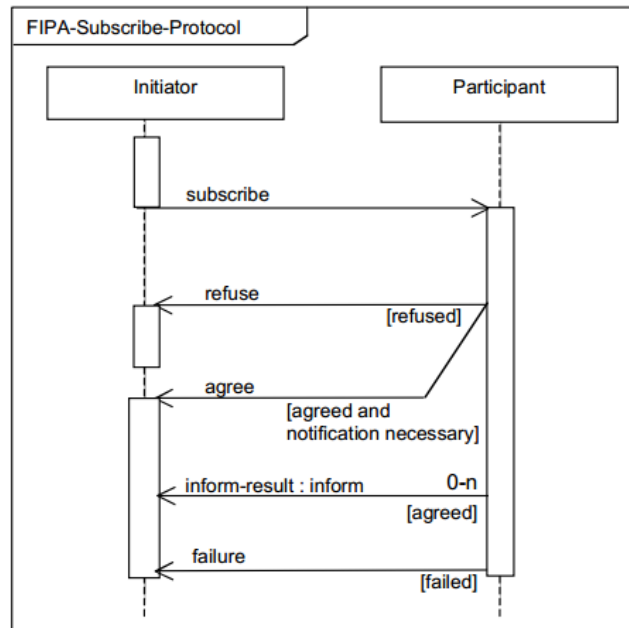


Figure 5.25: Diagramme UML du protocole FIPA-Subscribe

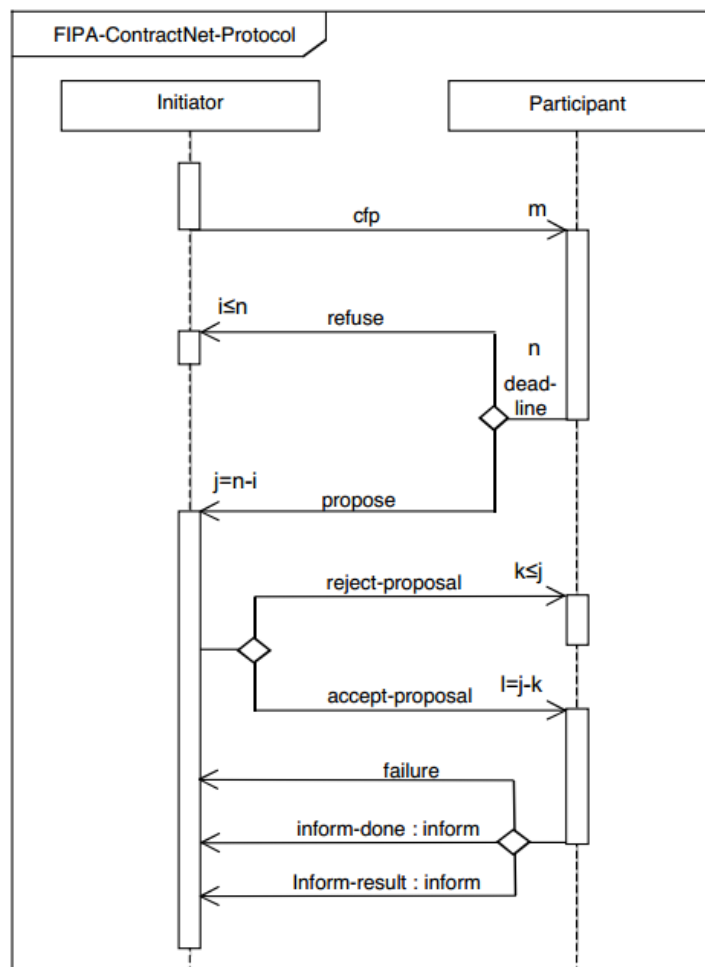


Figure 5.26: Diagramme UML du protocole FIPA-ContractNet

5.3.3. CONCEPTION DETAILLEE DU SYSTEME MULTI-AGENTS

Dans cette partie de notre travail, nous avons essayé de suivre une démarche proposée dans la méthodologie de JADE [113] dans la phase de conception. Ceci nous permettra de mieux organiser nos agents et de rendre leur fonctionnement conforme aux exigences de cette plateforme et à celles de la norme FIPA.

5.3.3.1. Raffinement et réorganisation des agents.

Dans notre diagramme d'agents, présenté à la figure 5.9, nous avons eu à recenser quatre catégories d'agents indépendamment de ceux naturellement mis à disposition par le système. De ces quatre catégories, certains étaient démultipliés parce qu'il fallait en avoir autant que de caméras positionnées. Un récapitulatif de ces agents ainsi que leurs rôles se présente dans le tableau 5.2 ci-dessous.

Agent	Rôle ou Responsabilité
<u>agentEV</u> Agent d'extraction Vidéo	<ol style="list-style-type: none"> 1. Allocation et initialisation de la structure de capture de capture pour lire le flux vidéo provenant d'une caméra ou d'un fichier vidéo. 2. Extraction d'image à partir de la vidéo. Lecture de la vidéo image par image. 3. Servir d'agent facilitateur en entretenant un service de page jaune. 4. Gestion des listes des objets connus et ceux perçus 5. Planificateur des tâches des autres agents
<u>agentAV</u> Agent d'analyse vidéo	<ol style="list-style-type: none"> 1. Détection de véhicules en mouvement (objets d'intérêt) 2. Etablissement de la signature et de l'identifiant de chaque objet d'intérêt détecté 3. Sérialisation de cette signature puis envoi à un autre agent pour sauvegarde dans une base de données 4. Envoi de l'identifiant de la signature à l'agentAV pour mise à jour de la liste des objets perçus.
<u>agentSbD</u> Agent gestionnaire de la base de données des signatures	<ol style="list-style-type: none"> 1. Gestion de la base de données des objets attendus et des objets perçus. 2. Mise à disposition de ces informations aux agents les désirant
<u>agentIT</u> Agent de ré-identification et de suivi	<ol style="list-style-type: none"> 1. Déploie le TBM et fait la fusion d'informations puis la prise de décision 2. Déploie éventuellement le MHT en cas d'ambiguïté pour retarder la prise de décision afin de lever l'ambiguïté.

Tableau 5.2 : Récapitulatif des agents et de leurs rôles d'après notre analyse initiale

Les recommandations pour raffiner ces agents sont les suivantes : éviter de dupliquer des données ; éviter de dupliquer les codes d'accès aux ressources ; éviter de dédoubler les agents tant que c'est possible, chaque agent doit être sur une seule machine ; éviter les agents très surchargés et très complexes pour ne pas avoir de problème à les concevoir et à les maintenir.

Au vu de ces recommandations, nous avons décidé de revoir notre approche agent et redéfinir les rôles de certains agents. En effet, à la lecture de ce tableau, il ressort que l'agentAV joue plusieurs rôles qui ne sont pas forcément du même ordre et qui lui vont l'amener à beaucoup communiquer avec les autres agents et en même temps assurer des tâches critiques telles que la capture et l'extraction de des images de la captures. En effet, sans ces tâches, les autres n'auraient pas leur raison d'être. De même, au niveau de l'agentAV, nous constatons que ses tâches sont dans la continuité des tâches de l'agentEV et qu'en retour toutes ses tâches lui font rendre compte à l'agentEV.

Tout ceci crée une interdépendance entre les deux agents. Nous avons donc décidé de les fusionner pour en faire un seul agent, tout en lui enlevant les rôles qui ne participent pas au traitement des données vidéo. Ce nouvel agent, nous l'avons nommé agentTV pour dire agent chargé des Traitements Vidéo.

A partir ce cet instant, nous avons constaté qu'il ne reste que les traitements participants à la fusion d'informations et le prise de décision de ré-identification. En regardant les rôles des agents agentSbD et agentIT, on se rend compte qu'ils se partagent ces traitements. Nous avons donc décidé de les fusionner également en leur ajoutant en plus de leurs rôles initiaux, les rôles dont l'agentTV a été dépossédé. CE nouvel agent, nous l'avons nommé agentTBM.

Notons que l'agentTV existera autant de fois qu'il y aura de source de capture vidéo (caméra ou fichier). Par contre, l'agentTBM sera unique et se chargera de la gestion de toutes les mises en correspondances du système. Au-delà de ces deux catégories d'agent, les autres agents de notre système seront ceux existant déjà dans l'environnement de JADE à savoir : le AMS et le DF qui pourra au besoin être surchargé en lui ajoutant des rôles de planificateur et de facilitateur dans notre système.

Le tableau 5.3 ci-dessous nous présente la mise à jour de la liste de nos agents ainsi que leurs nouveaux rôles.

Agent	Rôle ou Responsabilité
<p style="text-align: center;"><u>agentTV</u> Agent chargé des Traitements Vidéo</p>	<ol style="list-style-type: none"> 1. Allocation et initialisation de la structure de capture de capture pour lire le flux vidéo provenant d'une caméra ou d'un fichier vidéo. 2. Extraction d'image à partir de la vidéo. Lecture de la vidéo image par image. 3. Détection de véhicules en mouvement (objets d'intérêt) 4. Etablissement de la signature et de l'identifiant de chaque objet d'intérêt détecté 5. Sérialisation de cette signature puis envoie de la signature sérialisée à l'agentTBM pour mis à jour de sa base de données 6. Mise à jour de la densité de trafic local

<u>agentTBM</u> Agent chargé de la fusion des informations et de ré-identification	<ol style="list-style-type: none"> 1. Gestion du graphe orienté des caméras 2. Gestion de la base de données des objets attendus et des objets perçus. 3. Déploie le TBM et fait la fusion d'informations puis la prise de décision 4. Déploie éventuellement le MHT en cas d'ambiguïté pour retarder la prise de décision afin de lever l'ambiguïté. 5. Mise à disposition des informations d'association aux agents ayant souscrit.
---	--

Tableau 5.3 : Nouvelle liste des agents suite à la phase de conception

5.3.3.2. Recensement des interactions

Dans l'accomplissement de certains de leurs rôles ou pour participer à la réalisation d'un bût commun ou complémentaires, les agents sont souvent appelé à s'échange des informations.

Nous avons recensé dans cette partie toutes les interactions possibles entres nos agents, les préconditions de ces interactions, les protocoles d'interaction approprié à mettre en œuvre, les différents acteurs (initiateur et participants) concernés par ces interactions, ainsi que toutes autres informations devant participer à faciliter la mise en œuvre réelle de nos agent. Le point de ce travail se trouve dans le tableau 5.4 ci-dessous.

Agent	Interaction	N° rôle	Protocol	Type d'acteur	Avec qui	Précondition
<u>agentTV</u>	Enregistrement des services proposés	-	-	I	Agent DF	Création
<u>agentTV</u>	Réponse : Transfert de signature	5	fipa-subscribe	R	agentTBM	A la fin de chaque détection d'objet d'intérêt
agentTV	Contact des chargés de ré-identification	6	fipa-request	I	Agent DF	Première signature transmise
agentTV	Diminution de la densité du trafic	6	fipa-subscribe	I	agentTBM	Toujours
agentTBM	contacts des fournisseurs de signatures	2	fipa-request	I	Agent DF	Dès que actif
agentTBM	Ajout de signature aux listes	2	fipa-subscribe	I	agentTV	Toujours
agentTBM	Réponse : Informer d'une association	5	fipa-subscribe	R	agentTV	A chaque ré-identification d'objet préalablement annoncé
agentTBM	Enregistrement des services proposés	-	-	I	Agent DF	Création
Agent DF	Réponse aux demandes d'enregistrement	-	-	R	agentTV agentTBM	Les agents ont fait de demande d'enregistrement

Tableau 5.4 : Les agents de notre système et leurs interactions

5.3.3.3. Les modèles de messages

Chaque agent dispose d'une boîte de réception dans laquelle tous ses messages lui sont envoyés et sont normalement accessibles par ordre d'arrivée. Cependant, de par son autonomie, un agent peut décider d'appliquer un filtre et ne sélectionner que les messages répondant à un critère. Dans de pareilles situations, il a recours dans son comportement aux modèles de message appropriés pour gérer sa boîte aux lettres, privilégiant ainsi les messages qui l'intéressent.

Nous présentons ci-dessous dans le tableau et pour chaque interaction le modèle de messages utilisé par nos agents pour lire les messages de leurs boîtes aux lettres.

Agent	Interaction	N° rôle	Protocol	Type d'acteur	Avec qui	Précondition	Template
agentTV	Enregistrement des services proposés	-	-	I	Agent DF	Création	Conv-id
agentTV	Réponse : Transfert de signature	5	fipa-subscribe	R	agentTBM	A la fin de chaque détection d'objet d'intérêt	perf = subscribe prot= subscribe, InReplyTo
agentTV	Contact des chargés de ré-identification	6	fipa-request	I	Agent DF	Première signature transmise	Conv-id
agentTV	Diminution de la densité du trafic	6	fipa-subscribe	I	agentTBM	Toujours	Conv-id
agentTBM	contacts des fournisseurs de signatures	2	fipa-request	I	Agent DF	Dès que actif	Conv-id
agentTBM	Ajout de signature aux listes	2	fipa-subscribe	I	agentTV	Toujours	Conv-id
agentTBM	Réponse : Informer d'une association	5	fipa-subscribe	R	agentTV	A chaque ré-identification d'objet préalablement annoncé par lui	prot= subscribe, InReplyTo
agentTBM	Enregistrement des services proposés	-	-	I	Agent DF	Création	Conv-id
Agent DF	Réponse aux demandes d'enregistrement	-	-	R	agentTV agentTBM	Les agents ont fait de demande d'enregistrement	-

Tableau 5.5 : Les agents de notre système et leurs interactions

5.3.3.4. Enregistrement et recherche de services ou de fournisseur de services

JADE met à disposition un agent facilitateur : Directory Facilitator (DF), conformément aux spécifications de la FIPA. La DF est comme un annuaire ou page jaune. Tout agent qui désire mettre à disposition ses services, les enregistre auprès du DF. De même, tout autre agent visiteur pourrait interroger DF pour retrouver les agents fournissant les services dont ils ont besoin.

Ci-dessous, nous présentons sur les figures 5.27 et 5.28 une forme de diagrammes de classes pour décrire les enregistrements et les recherches de services.

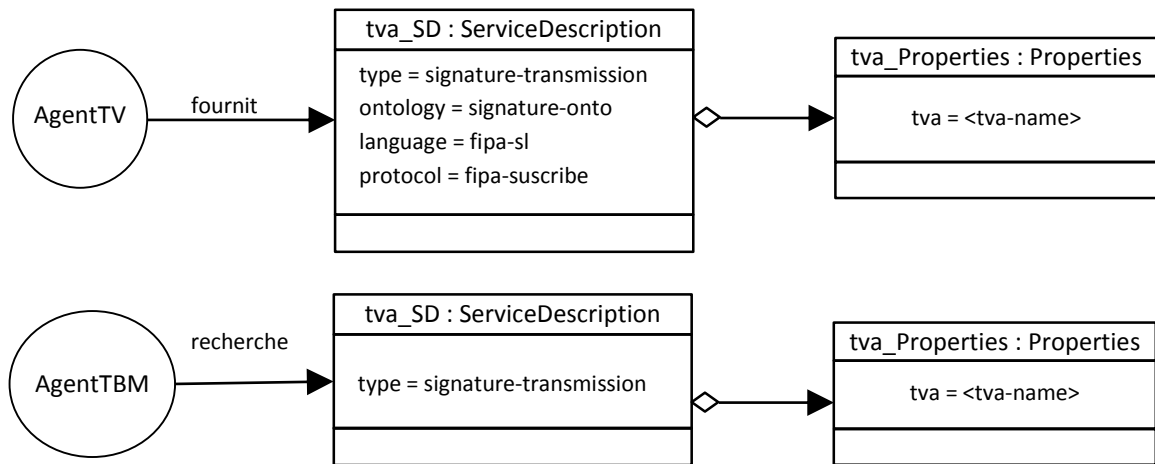


Figure 5.27: Enregistrement et recherche de service de traitement vidéo

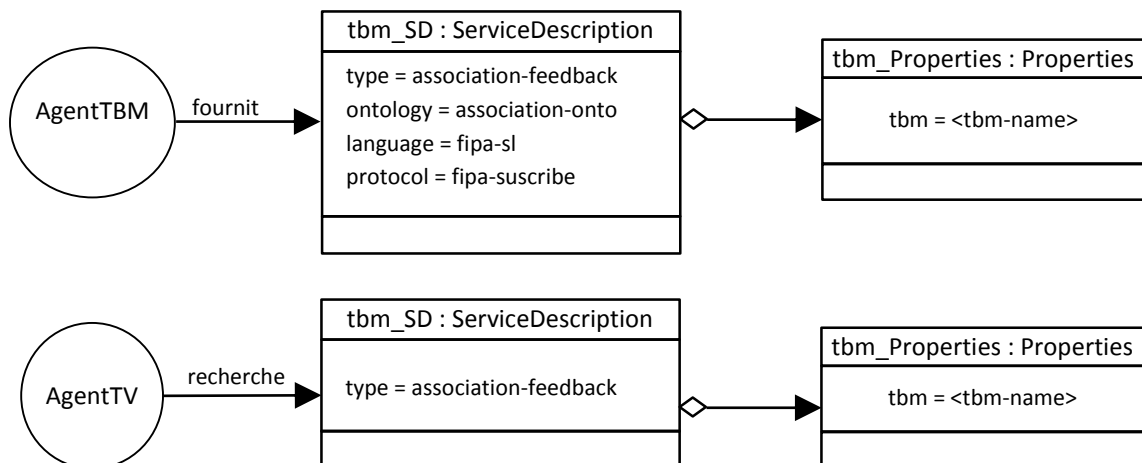


Figure 5.28: Enregistrement et recherche de service de fusion d'information et de re-identification

5.3.3.5. Interactions avec des ressources

Les seules ressources gérer dans le système multi-agents sont : Dans un premier temps, le graphe orienté représentant le nombre de caméras de notre système ainsi que les relations (voisin, etc.) qui existes entres ces caméras. Dans un second temps la base de données des listes d'objets connus et celles perçus.

Pour le graphe orienté, comme nous avons eu à le présenter plus haut figure 5.3, il sera représenté par une matrice qui sera initialisé dès le départ par l'utilisateur avant le démarrage de la simulation. La seule interaction avec cette matrice sera juste d'y aller rechercher soit directement la liste des voisins (saut suivant), soit de partir d'un nœud et de remonter à un prédécesseur puis rechercher tous ces voisins sauf celui utilisé pour remonter à lui. Fondamentalement cette ressource (matrice ou graphe) sera une ressource passive et elle

sera géré par l'agent DF pour sa construction ; par l'agentTBM pour les accès afin d'y faire des recherches.

Pour ce qui est de la base de données des différentes listes d'objets connus et celles perçus, nous avons opté pour un tableau (plus précisément une matrice $N \times 2$) de listes chaînées de signatures. Chacune ligne de cette matrice représente les objets connus et ceux connus par un agentTV. La première colonne représente les listes des différentes listes d'objets connus, tandis que la seconde colonne représente la liste des objets perçus. Donc autant de lignes que d'agentTV. Il est initialisé au lancement de la simulation et est directement géré par l'agent TBM.

Il ressort de tout ceci qu'il n'y aura pas véritablement de base de données à gérer au sens SQL du terme et qu'il n'y aura pas d'interaction avec de ressources externes aux agents.

Cependant cette logique peut être remise en cause en cas de dédoublement de l'AgentDF pour raison de surcharge d'activité par exemple. En effet, suivant la densité des trafics au niveau des différents agentTV, en fonction de leur nombre et en fonction des distances entre les agentTV (en termes de portion de route dans le graphe orienté), il peut y arriver que l'agentTBM soit surchargé surtout du fait qu'il s'agit d'un traitement en temps réel. Dans ces conditions, il est possible de lui adjoindre un ou d'autres agentTBM pour en augmenter leur performance globale. Beaucoup de modifications devront alors être faites car ces agents vont partager aussi bien le graphe orienté que les listes des objets perçus et ceux connus ; et il faudra gérer les accès concurrents. Il faudra aussi revoir les communications entre agents et les protocoles de communication à certains niveaux.

Nous n'avons pas exploré cette possibilité dans ce travail. Nous le gardons comme perspective d'amélioration du travail.

5.3.3.6. Interfaces utilisateurs

Dans notre application de simulation, l'utilisateur n'interagit pas avec les agents. Il n'intervient au début que pour définir les paramètres nécessaires au déroulement de la simulation. Ainsi, il construit le graphe orienté en définissant le nombre de caméras utilisé, la distance entre les différentes caméras voisines (en termes de segment de route). Mais tout ceci en remplissant juste une matrice. Suite à cela, il précise les chemins vers les différentes sources vidéo utilisées pour faire la simulation.

L'intervention de l'utilisateur se limite à ces actions ci-dessus citée. Le reste de la simulation se déroule sous forme de visualisation des différentes fenêtres de flux vidéo et d'affichage des résultats une fois la simulation terminée.

5.3.3.7. Présentation des comportements internes des agents

Les comportements des agents permettent d'implémenter leurs différents rôles et capacités.

Dans cette partie de notre travail, nous présentons les différentes classes de JADE permettant d'implémenter les comportements de nos agents ainsi que les protocoles d'interaction utilisés par les agents à l'intérieur de certains de ces comportements pour collaborer avec d'autres agents. Pour chaque rôle, les types de comportements (behaviour) utilisés seront aussi présentés. Le tableau ci-dessous nous présente la synthèse de ces descriptions.

Agent	Responsabilité / sous responsabilité	Type de comportement à implémenter	Type de comportement à implémenter	Protocole éventuellement utilisé
agentTV	Allocation et initialisation de la structure de capture de capture pour lire le flux vidéo provenant d'une caméra ou d'un fichier vidéo.	OneShotBehaviour	OneShotBehaviour	Pas d'échange, donc pas de protocole
	Extraction d'image à partir de la vidéo	CyclicBehaviour	SequentialBehaviour	
	Détection de véhicules en mouvement (objets d'intérêt)	OneShotBehaviour		
	Etablissement de la signature et de l'identifiant de chaque objet d'intérêt détecté	OneShotBehaviour		
	Sérialisation de cette signature puis envoi de la signature sérialisée à l'agentTBM pour mis à jour de sa base de données	OneShotBehaviour		subscribe, InReplyTo
	Mise à jour de la densité de trafic local	OneShotBehaviour		
agentTBM	Gestion du graphe orienté des caméras	OneShotBehaviour	SequentialBehaviour	Pas d'échange, donc pas de protocole
	Gestion de la base de données des objets attendus et des objets perçus.	CyclicBehaviour		
	Déploie le TBM et fait la fusion d'informations puis la prise de décision	FSMBehaviour		
	Déploie éventuellement le MHT en cas d'ambiguïté pour retarder la prise de décision afin de lever l'ambiguïté.	CyclicBehaviour		
	Mise à disposition des informations d'association aux agents ayant souscrit	OneShotBehaviour	subscribe, InReplyTo	

Tableau 5.6 : Tableau descriptif des rôles de nos agents et des comportements JADE associés

5.3.3.8. Approche de communication

Pour que les agents puissent communiquer, ils doivent s'envoyer des messages mais surtout comprendre les messages des uns et des autres. Pour cela ils doivent avoir un protocole de communication et partager le même langage et vocabulaire. Une sémantique doit être définie pour le contenu des messages et les terminologies utilisables doivent également préciser pour assurer la bonne compréhension et la bonne interprétation des messages.

Suivant le contexte, la mise en œuvre de cet ensemble de choses peut conduire à la définition d'une ontologie dans l'espace de communication des agents. Cependant, cette approche, bien qu'étant disponible dans JADE, n'est vraiment utile et avantageuse que lorsqu'on se trouve dans le besoin de faire communiquer des agents hébergés sur des plateformes et systèmes d'agents différents. Cela permet de garantir une interopérabilité entre différents systèmes d'agents.

Dans notre cas, tous nos agents sont hébergés dans le même système d'agent qui est JADE. Ce système est dans un environnement homogène qui tourne sur la même machine et dans le pire des cas dans un même réseau local. Pour cette raison, en lieu et place d'une solution de communication basée sur une ontologie, nous avons préféré un système de communication basé sur l'utilisation d'objets java comme contenu des messages en y ajoutant l'implémentation de l'interface `java.io.Serializable` pour assurer la sérialisation des objets (donc du contenu de nos messages) avant leur envoi (méthode `send()`).

Ainsi, dans les fichiers des classes respectives de nos agents `agentTV` et `agentTBM`, nous avons défini comme classes externes les classes `infoSignature` et `reIdentifie`.

La classe `infoSignature` permet à un `agentTV` de créer et de renseigner un objet avec toutes les informations (histogramme, dimension du rectangle englobant, vitesse à la détection, densité du trafic lors de la détection, agent auteur de la détection) permettant d'établir la signature d'un véhicule détecté, ainsi que l'agent auteur de cette information. Puis d'envoyer, après sérialisation, ces informations à l'`agentTBM` pour traitement.

La classe `reIdentifie` permet à l'`agentTBM`, après chaque bonne association, d'envoyer en retour à chaque `agentTV` approprié le nombre de ses objets ré-identifiés parmi ses objets préalablement perçus. Ceci permet à l'`agentTV` de mettre à jour les informations relatives à la densité de trafic sur son axe routier.

5.4. CONCLUSION

Ce chapitre nous a permis de présenter les différentes étapes de la mise en place de notre système multi-agents de suivi multi-caméras, depuis l'analyse jusqu'à la conception détaillée. L'utilisation d'un système multi-agents est justifiée dans un premier temps par la nécessité de gestion en parallèle de plusieurs caméras et de la fusion d'informations pour une prise de décision. Elle se justifie également par le caractère auto-adaptatif de notre système suite à une augmentation ou à une diminution du nombre de caméras à gérer.

Une amélioration aurait été de rendre automatique le nombre d'agentTMB du système pour répondre aux contraintes temporelles du temps réel suite à toutes variations (augmentation ou diminution) de trafic ou celle du nombre de caméras (donc d'agentTV) dans le système.

Nous avons opté, dans notre application actuelle, pour la non utilisation d'ontologie pour ne pas alourdir inutilement les communications entre agents mais dans le cas où on aurait plus d'un agentTBM dans le système, notre approche actuelle souffrirait de quelques insuffisance en matière de communication entre certains agents. En effet, certains protocoles de communications vont changer pour permettre aux agentTV de procéder par élection de l'agentTMB à qui soumettre leurs données de signature. De même, les agentTMP devront coopérer et partager certaines ressources pour garantir une mise en correspondance fiable.

Pour la conception et la mise en œuvre de notre système, nous avons opté pour l'utilisation de JADE que nous avons trouvée comme l'une des meilleures plateformes ou framework de développement et la gestion d'applications SMA en respect des normes FIPA.

Le chapitre suivant présente quelques résultats de nos implémentations et tests.

CHAPITRE 6 : IMPLEMENTATIONS ET SIMULATIONS

Sommaire

6.1.	Introduction	124
6.2.	Aspects techniques du système	124
6.2.1.	Matériel utilisé	124
6.2.2.	Calibration des caméras	125
6.2.3.	Framework et environnements de développement	128
6.2.4.	Données utilisées	128
6.3.	Expérimentations et évaluations	129
6.3.1.	Evaluation de nos descripteurs	129
6.3.2.	Résultats de suivi et ré-identification de notre système	138
6.4.	Conclusion	141

6.1. INTRODUCTION

Dans cette partie nous allons présenter nos différentes expérimentations en commençant par décrire les conditions ainsi que les données et outils ayant servi à leur réalisation. La comparaison des résultats de nos expérimentations aux résultats attendus ou aux résultats obtenus dans d'autres conditions ou des suites d'autres travaux de recherche utilisant d'autres approches, nous permettra de juger de la qualité et de la pertinence de nos solutions.

Nous allons dans un premier temps tester nos fonctions de distances et une fois rassuré de leur robustesse nous passerons à la mise en place d'un jeu de données devant nous permettre de mettre en place nos fonctions de masses de croyance de notre approche évidentielle.

Nous allons analyser nos résultats et les comparer à celles obtenus dans un de nos travaux antérieurs et ceux d'autres chercheurs.

6.2. ASPECTS TECHNIQUES DU SYSTEME

6.2.1. MATERIEL UTILISE

- Un ordinateur portable ASUS X53S, Intel core I5-2410M
4 Go de RAM, 500 Go de disque dur
Carte graphique Geforce GT520 1Go dédié
Ecran 15", Webcam avec micro intégré, Wifi, Windows 7
- Deux caméras réseau AXIS 206M Mégapixel, avec 6 différentes résolutions vidéo disponibles (maximum 1280 x 1024 pixels) et support de HDTV (High Definition TV)
- Deux appareils photo numériques
 - Kodak EasyShare Z1012 IS, 10.1 Megapixels, HD 720p, 30ips
 - SONY cyber-shot DSC-W380, 14,1 Megapixels, HD ready: 1280 x 720p (30fps), VGA: 640 x 480 (30fps)

6.2.2. CALIBRATION DES CAMERAS

Lorsque nous utilisons une caméra pour faire la capture vidéo, nous faisons passer une information 3D du monde réel en une information numérique vidéo (succession d'images) mais cette fois-ci en 2D. Il s'agit donc d'une projection d'un espace de dimension 3 dans un autre espace de dimension 2 par la caméra. Plusieurs modèles existent pour décrire cette transformation géométrique. Le modèle le plus utilisé en vision par ordinateur est le modèle sténopé basé sur une projection en perspective.

La réalisation efficace de cette projection nécessite la connaissance de paramètres aussi bien intrinsèques qui modélisent la géométrie interne et les caractéristiques optiques du capteur de la caméra, que celles extrinsèques qui lient le repère du capteur au repère associé à la scène où se trouvent les objets de référence utiles pour la calibration. La figure 6.1 ci-dessous présente les différents repères pris en compte dans la modélisation géométrique d'une caméra pour faire le passage du monde réel 3D au plan image de la caméra.

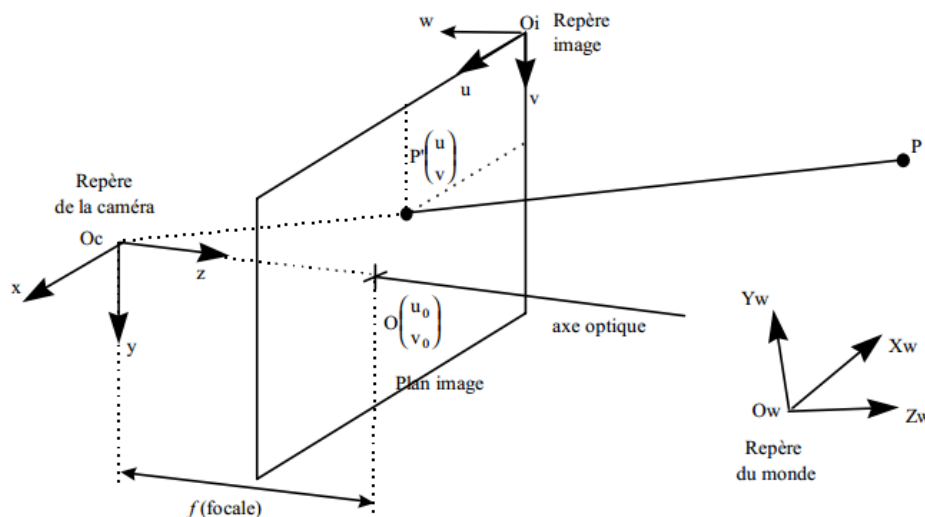


Figure 6.1: Passage de l'univers 3D au plan image 2D

La transformation du repère caméra au repère image s'obtient de la façon suivante :

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} k_u & 0 & 0 \\ 0 & k_v & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \\ w_0 \end{bmatrix} \quad (6-1)$$

Nota :

x', y', z' sont les coordonnées du point P' , projection du point P du repère de monde réel, dans le repère de la caméra.

u_0, v_0, w_0 sont les coordonnées de l'origine du repère caméra.

u, v, w représentent les coordonnées de la projection du point P' dans le repère image.

k_u, k_v représentent respectivement le facteur d'échelle horizontal et le facteur d'échelle vertical (en pixels/mm) permettant de passer du plan rétinien (repère caméra) au plan image du repère image.

La composante w étant toujours nulle, cette équation s'écrit plus simplement :

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} k_u & 0 \\ 0 & k_v \end{bmatrix} \cdot \begin{bmatrix} x' \\ y' \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (6-2)$$

En utilisant les relations qui existent entre le point P' et le point P , on peut déduire :

$$\begin{cases} u = \alpha_u \cdot \frac{r_{11} \cdot P_{Xw} + r_{12} \cdot P_{Yw} + r_{13} \cdot P_{Zw} + t_X}{r_{31} \cdot P_{Xw} + r_{32} \cdot P_{Yw} + r_{33} \cdot P_{Zw} + t_Z} + u_0 \\ v = \alpha_v \cdot \frac{r_{21} \cdot P_{Xw} + r_{22} \cdot P_{Yw} + r_{23} \cdot P_{Zw} + t_Y}{r_{31} \cdot P_{Xw} + r_{32} \cdot P_{Yw} + r_{33} \cdot P_{Zw} + t_Z} + v_0 \end{cases} \quad (6-3)$$

Avec : $\alpha_u = fk_u$, $\alpha_v = fk_v$, f la distance focale, t_x, t_y, t_z les coordonnées du vecteur translation entre O_c et O_w .

Ce qui nous intéresse le plus dans le cadre de notre étude, c'est de pouvoir établir une correspondance entre les dimensions réelles des véhicules et celles des objets (véhicules) observés dans le plan image de nos vidéos tout en tenant compte de la profondeur ou de la perspective linéaire qui agit sur les dimensions des objets de même catégorie mais situés à des profondeurs différentes dans la même scène comme l'illustre la figure ci-dessous.



Figure 6.2: Effet de la profondeur sur la taille d'un objet

Pour résoudre ce problème, nous pouvons faire les projections des points d'extrémité de nos objets du monde réel dans le plan image, puis en calculer les distances euclidiennes. Mais dans

notre cas, nous avons pris des mesures réelles sur le terrain à plusieurs niveaux de la route de la scène filmée, puis nous avons comparé ces mesures à celles obtenus dans les vidéos pour les mêmes objets pour établir un rapport de proportionnalité.

Dans un premier temps, nous avons utilisé les dimensions (longueurs) réelles et celles issus de prises de vue (vidéo) des signalisations horizontales (lignes blanches) sur la voie. Dans un second temps, nous avons fait la même chose mais cette fois-ci avec des prises de vue (vidéo) de véhicules situés à différentes profondeurs dans les vidéos (figure 6.2).

Ces différentes mesures, au niveau de chacune de nos caméras, nous ont permis d'établir les correspondances entre nos objets en grandeur nature dans un référentiel réel 3D et nos données dans le plan image 2D de nos vidéos. Cela nous a permis de faire des correspondances entre les mesures en unité pixels du repère image et celles en centimètre du repère du monde réel.



Figure 6.3: Lignes routières de même taille mais à différentes profondeurs.

Par exemple, sur le terrain, la mesure réelle de la longueur des deux lignes horizontales ci-dessus de la figure 6.3 donne la même valeur qui est de 3m.

Dimensions dans le repère réel de monde	
Longueur des traits	3m
Distance entre les deux traits	5,5m

Tableau 6.1. Mesures de lignes dans le repère du monde

Mais on se rend compte que ces mesures ne donnent pas les mêmes valeurs à cause de l'effet de profondeur qui entraîne des facteurs de grandissement différents.

Dimensions dans le plan image		
La longueur du premier trait (le plus proche)	164 pixels	20,83mm
La longueur du second trait (le plus éloigné)	111 pixels	14,10mm
Distance entre les deux traits	21 pixels	2,67mm

Tableau 6.2. Mesures de lignes dans le repère image

Soit g le facteur de grandissement, on a : $g = \text{longueur dans le plan image} / \text{longueur réelle}$.

Pour les deux traits visibles sur la figure 6.3, on obtient les facteurs de grandissement suivants :

- $g = 20,83/3000 = 0,00694$
- $g = 14,10/3000 = 0,00470$

La profondeur pouvant jouer sur la taille des boites englobantes et fausser les calculs liés à leur comparaison, nous utilisons un facteur multiplicatif fonction de la position dans l'image de la ligne de base du rectangle englobant par rapport au bas de l'image pour réajuster ses dimensions avant de passer à la comparaison.

6.2.3. FRAMEWORK ET ENVIRONNEMENTS DE DEVELOPPEMENT

Les différents frameworks et environnements de développement utilisés sont

- OpenCV 2.3, bibliothèque graphique libre C/C++ pour le traitement d'images et de vidéos.
- JavaCV, librairie java implémentant bibliothèque OpenCV. C'est en réalité une bibliothèque d'interfaçage d'OpenCV basée sur la JNA (Java Native Access).
- Eclipse 3.7.0, notre environnement de programmation en java
- JADE 4.3.0, framework open source de développement et de gestion de systèmes multi-agents.

6.2.4. DONNEES UTILISEES

Les axes routiers sur lesquels nous avons eu à travailler sont des autoroutes sur lesquelles passent toutes sortes de véhicules y compris les motos. On y rencontre souvent aussi des piétons au niveau des chaussées latérales.

Pour nos expérimentations, nos caméras ont été placées suivant deux schémas. D'une part, sur un axe routier avec une voie de sortie de l'autoroute ; puis dans un second temps sur un axe sans autres entrée ni sorties. Nous y avons recueilli des données vidéo que nous avons traité par la suite au niveau de notre système pour voir les résultats.

Pour pouvoir utiliser les détecteurs de Haar de Viola-Jones, nous avons dû faire de l'apprentissage de données pour mettre en place nos fichiers XML. Pour faire l'apprentissage, nous avons utilisé les captures vidéo provenant de nos différentes caméras dont nous avons extrait des milliers d'images à l'aide d'un petit code que nous avons écrit dans OpenCV. De ces images, nous avons sélectionné 900 contenant de un à plusieurs véhicules. Nous avons dédoublé ces images pour avoir au total 1800 images par symétrie et par d'autres transformations pour agir sur l'éclairage de certaines images.

Sur les 1800 images, nous avons mis de côté 400 pour faire les tests à la fin de l'apprentissage. Les 1400 restants ont été utilisés comme base de données des images positives de l'apprentissage. Il faut dire aussi que sur les 1400 images positives, nous avons aussi volontairement mis près de 200 images de véhicules dont l'arrière ou l'avant est partiellement voilé. Ceci dans le souci d'augmenter les chances de détection de véhicules même en cas d'occultation.

Pour mettre en place la base des images négatives, nous avons utilisé aussi bien des bibliothèques d'images n'ayant rien à voir avec les véhicules que les images prises sur les routes sans véhicules. De tout cet ensemble, nous avons extrait 960 images pour construire notre base d'images négatives.

Après plusieurs jours d'essai et de traitements, nous sommes arrivés à avoir des fichiers XML nous garantissant les taux de détection suivants :

- Bonne détection : 98,99 %
- Fausses détections : 0,97%
- Non détection : 0,033%

Tous ces éléments réunis, nous sommes passé à la phase de mise en place et de test de notre système multi agents de suivi multi caméras.

6.3. EXPERIMENTATIONS ET EVALUATIONS

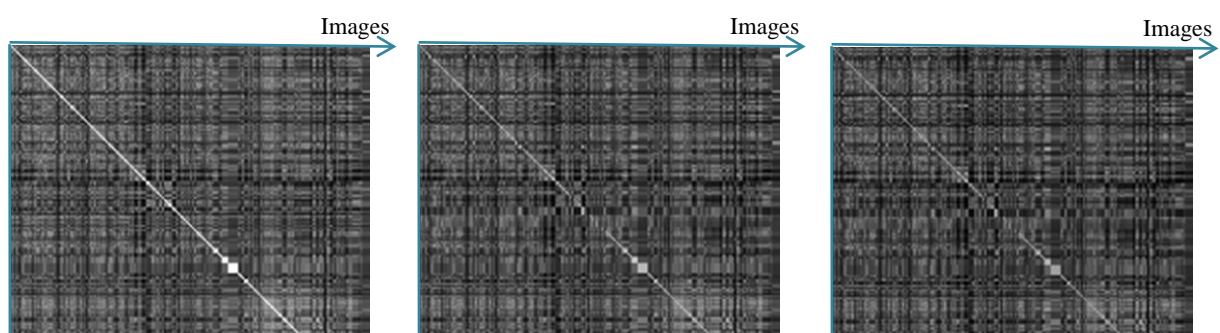
Dans cette partie, nous présentons notre démarche d'évaluation de notre système de suivi et de ré-identification. Pour nous assurer de la qualité de nos modèles, notamment pour la construction de notre modèle basé sur la théorie de l'évidence et plus précisément sur le modèle de croyances transférables, nous présentons dans un premier temps la validation de la qualité de nos descripteurs ; puis nous terminerons par la présentation de quelques résultats de nos expérimentations.

6.3.1. EVALUATION DE NOS DESCRIPTEURS

6.3.1.1. Descripteurs visuels

Le tout premier descripteur que nous avons évalué est celui relatif à la couleur de nos objets. Il s'agit d'histogrammes et plus particulièrement d'histogrammes multi-échelles et des caractéristiques verticales puis horizontales des objets à suivre.

La figure 6.4 ci-dessous présente quelques résultats issus de nos expérimentations.



(a)

(b)

(c)

Figure 6.4: Images créées à partir de distances entre histogrammes d'images d'une base de données. On a utilisé 180 images de véhicules prises toutes de profil. (a) est créée à partir d'histogrammes d'images originales. (b) et (c) sont créées à partir d'histogrammes multi-échelles.

Sur ces différentes figures, chacun des axes est numéroté de 1 à 180, et chaque numéro correspond à chacune des 180 images de notre base de données. Les images sont générées à partir de distances entre histogrammes. La distance utilisée ici est la distance de Bhattacharyya. La valeur a_{ij} de chaque pixel est obtenue de la façon suivante :

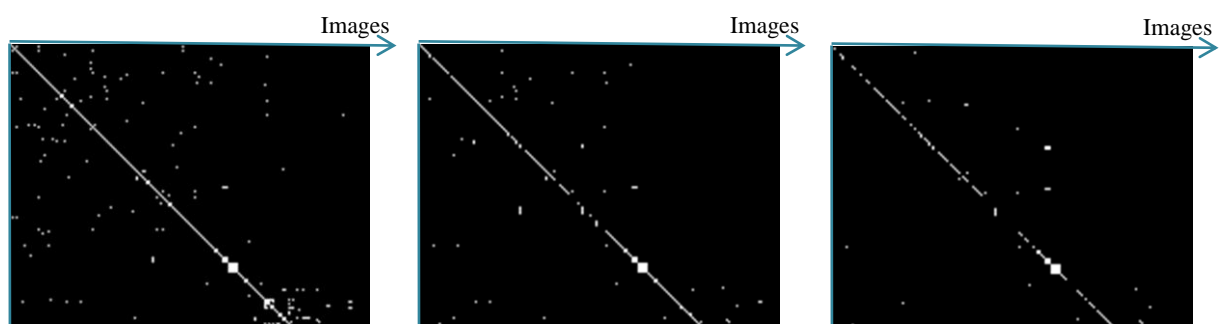
$$a_{ij} = 255 * (1 - d(H_i, H_j)) \quad (6-1)$$

d : désigne la distance de Bhattacharyya

H_i : désigne l'histogramme de l'image numéro i de la base de données.

A l'analyse de ces images, il ressort plusieurs constats :

- Les diagonales en blancs confirment que toutes les images identiques sont bien associées comme on devait s'y attendre.
- La présence de plusieurs endroits de niveau de gris relativement proche ou très proche du blanc montre qu'il y a bien plusieurs autres images de voitures très proches les uns des autres.
- les zones sombres ou noires représentent les voitures qui ne présentent aucune similarité du point de vue de leurs couleurs.
- Enfin, en regardant avec un peu plus d'attention les 3 images (a)(b)(c), on constate que les histogrammes multi-échelles sont plus discriminants que les histogrammes de base ou originaux. En effet, de la gauche vers la droite, c'est-à-dire de l'image (a) à l'image (c) en passant par l'image (b), on remarque que la densité de la couleur noire évolue de façon croissante ; éliminant ainsi certaines fausses associations. Ceci est beaucoup plus remarquable sur la figure 6.5 ci-dessous. Les images de cette figure sont obtenues en procédant à un seuillage de ses différentes images de la figure précédente (figure 6.4). On arrive ainsi à ne faire ressortir (en blanc) que les images qui présentent beaucoup plus de similarité du point de vue de leurs couleurs.



(a)

(b)

(c)

Figure 6.5: Images obtenues après seuillage des images de la figure précédente.

Il faut aussi remarquer que les histogrammes multi-échelles peuvent présenter des inconvénients lorsqu'on augmente trop la valeur d'échelle. En effet, sur la même figure ci-dessus, on peut remarquer au niveau de (b) et (c) que la première diagonale n'est pas uniformément blanche au fur et à mesure que l'on augmente d'échelle. Ceci peut donc entraîner de faux négatifs. Il est alors important de faire un bon calibrage pour tirer profit des avantages très importants des histogrammes multi-échelle.

Le second descripteur que nous présentons ici est celui relatif à la forme des objets à suivre. Dans cette partie nous avons basé nos travaux sur les boites englobantes pour mettre en place notre descripteur, puis l'évaluer. Il a été question à ce nouveau de déterminer la boite englobante minimale. Nous avons opté dans un premier temps pour la détermination du rectangle minimum englobant de chaque objet d'intérêt. Notre objectif étant de mettre en place une distance entre ces descripteurs de forme pour discriminer des véhicules objets de suivi et de ré-identification.

La figure 6.6 ci-dessous est une illustration de la représentation de rectangle englobant.



Figure 6.6: Exemples de rectangles minimums englobant

Les rectangles minimums englobants nous seront utiles à double titre. Dans un premier temps, ils vont nous permettre de catégoriser les véhicules (les poids lourds, les utilitaires, les véhicules légers, etc.). Dans un second temps, ils vont nous permettre de mettre en place un descripteur de véhicule même si de par leurs caractéristiques, les rectangles ne sont pas vraiment des descripteurs assez discriminants. En effet, deux véhicules peuvent avoir pratiquement les mêmes dimensions (longueur, largeur) sans pour autant avoir de réelles similarités. Par contre, lorsque ces dimensions ne concordent pas il serait très peu probable voire improbable d'établir une quelconque similarité entre ces véhicules et c'est cet aspect qui nous intéresse.

Pour comparer deux rectangles on peut se baser sur deux de leurs caractéristiques clés, à savoir les dimensions de la diagonale et celle de l'angle de la pente de cette diagonale. En effet, deux rectangles qui ont des diagonales de même longueur ainsi que même angle de pente, sont identiques.

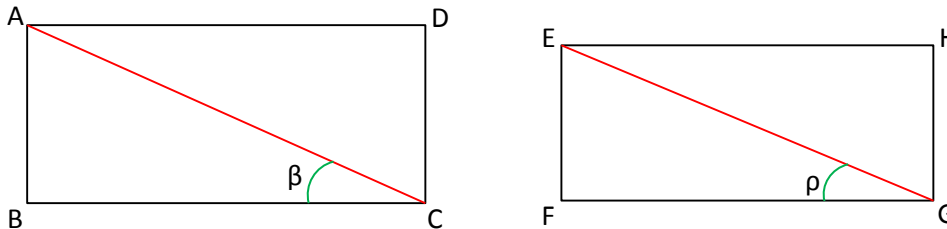


Figure 6.7: Comparaison de deux rectangles

Si $\beta = \rho$ et $AC = EG$ alors $AB = EF$ et $BC = FG$ ce qui revient à dire que les deux rectangles sont identiques. Naturellement β et ρ sont toutes strictement inférieures à $\frac{\pi}{2}$

On peut également utiliser directement les dimensions des cotés orthogonaux des deux rectangles. Nous avons opté pour cette solution et nous nous sommes basés sur la distance de Hausdorff pour proposer une distance afin d'exprimer la similarité entre deux rectangles R1 et R2 de la façon suivante :

$$d(R1, R2) = 1 - \frac{(long_1 + long_2) * (larg_1 + larg_2)}{4 * \max(long_1, long_2) * \max(larg_1, larg_2)} \quad (6-4)$$

Pour évaluer cette distance, nous avons utilisé la même approche que celle utilisée pour l'évaluation des couleurs en nous basant sur la même base de 180 images de véhicules.

6.3.1.2. Descripteurs temporels

La couleur et la forme (rectangle englobant) ne suffisent pas souvent pour discriminer deux véhicules. En effet, trop de véhicules ont à la fois la même couleur ou des couleurs très proches, et la même forme ou des formes très proches. La probabilité de faire de mauvaises associations en ne se basant que sur ces deux attributs est très élevée car il sera très facile de prendre un véhicule pour un autre. Fort heureusement, les véhicules qui font l'objet de notre étude sont des véhicules en mouvement sur un axe routier. Notre ici est donc de faire recours à un descripteur temporel et plus précisément un descripteur spatio-temporel pour compléter la description d'un véhicule et arriver ainsi par fusion d'information à discriminer deux véhicules en mouvement sur un axe routier.

En connaissant le modèle d'accélération d'une voiture et la distance entre deux capteurs (des caméras disposés sur les axes routiers) délimitant un tronçon, il devrait être possible de déterminer ou d'estimer le temps de parcours de ce tronçon, et donc la date d'arrivée du véhicule au niveau du second capteur. Malheureusement la réalité n'est pas aussi facile car plusieurs facteurs peuvent influencer le modèle d'accélération d'un véhicule et donc sa vitesse instantanée. Au nombre de ces facteurs on peut citer : l'état de la route, la luminosité ou la

visibilité, la densité du trafic, le débit du trafic, la présence d'obstacles, la distance à parcourir, le conducteur, le type, la longueur et le poids du véhicule.

Ces différents éléments qui ne sont pas pour la plupart prédictibles, font qu'il serait prétentieux de vouloir faire une modélisation réaliste et précise de l'accélération d'un véhicule quand on sait que les véhicules n'ont presque jamais un mouvement uniformément accéléré. Leur mouvement connaît plusieurs phases ou cycles non uniformes de variation de l'accélération à causes des facteurs cités plus haut.

Notre objectif dans cette partie de notre travail est d'arriver à modéliser le temps de parcours d'un tronçon AB. Autrement, arriver à modéliser les dates d'arrivée d'un véhicule au niveau d'un capteur B, sachant que ce véhicule a été précédemment détecté au niveau d'un capteur A avec comme informations disponibles : la vitesse instantanée V_a de ce véhicule lors de son passage au niveau du capteur A, la distance du trajet entre les deux capteurs (segment AB), la densité de véhicules sur le tronçon AB.

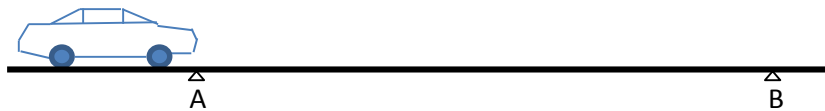


Figure 6.8: Parcours d'un tronçon (segment AB) par un véhicule : estimation du temps de parcours à partir de la vitesse, de la densité de véhicule sur le tronçon et de la longueur du tronçon.

Autrement, à quel moment une détection au niveau du capteur B peut-elle être prise pour celle d'un véhicule préalablement détecté par le capteur A ?

Connaissant la vitesse instantanée d'un véhicule au départ d'un capteur situé à l'entrée d'un tronçon, connaissant la densité de véhicules sur ce tronçon à cet instant, il est possible de construire ou d'estimer la vitesse moyenne du véhicule sur ce tronçon par apprentissage en se basant sur l'analyse de données empiriques provenant de mesures de capteurs et stockées dans une base de données, après éventuelle élimination des valeurs trop particulières ou atypiques.

Plusieurs travaux et recherches ont été menés sur l'estimation du temps moyen de parcours [108], [109], [110] et plusieurs approches et techniques de recueil et de traitement de données existent pour faire une meilleure estimation du temps de parcours. Certaines de ces techniques, dont celle de l'indicateur de vitesse moyenne ou indicateur BRP, mis au point par Bonvalet F. et Robin-Prévallée Y. [114] semble robuste aux perturbations locales du fait de l'utilisation de données issues d'un grand nombre de points de mesure.

En général, une catégorie de ces techniques exploite des outils liés à l'infrastructure routière (capteurs au sol, boucle magnétique, vidéo, relevée manuelle de plaque minéralogique, etc.), tandis qu'une autre repose sur des stratégies de mesures embarquées à bord de véhicules tests. Pour plus de détails sur les différentes approches d'estimation du temps de parcours, se référer à [109] qui en présente une bonne synthèse. La mise en commun de ces divers

moyens complémentaires par fusion d'informations permet d'aboutir à des résultats suffisamment robustes comme cela a été le cas dans [111], [112]. Ces résultats sont exploitables pour construire un système semi-supervisé d'apprentissage pour faire l'estimation du temps de parcours d'un tronçon bien déterminé.

Notre idée et contribution dans cette partie est de nous appuyer sur ces travaux pour mettre en place une démarche d'estimation du temps de parcours, en utilisant une base de données d'apprentissage, interrogeable pour avoir le temps moyen de parcours d'un tronçon par un véhicule, connaissant les trois données fondamentales que sont : sa vitesse instantanée à l'entrée du tronçon, la densité de véhicules sur le tronçon, et la longueur du tronçon ; tout ceci par catégorie de véhicule suite à une classification des véhicules. Nous avons décidé de n'utiliser que trois catégories de véhicules (les poids lourds, les utilitaires, les véhicules légers) et nos travaux se limitent aux réseaux urbains de voies rapides du fait de la complexité de mise en œuvre de méthodes d'estimation de temps de parcours sur des voies de centre-ville à fort maillage et interconnexion avec plusieurs carrefours disposants de feux tricolores.

Notre démarche a été la suivante : Soit AB un tronçon pour lequel on veut estimer le temps de parcours d'un véhicule. A l'entrée de ce tronçon, on enregistre la date, la catégorie et la vitesse instantanée de chaque véhicule. Le tronçon étant découpé en plusieurs segments représentatifs, on détermine les temps de parcours effectifs des véhicules sur chacun des segments. Un cumul des temps de parcours par segment, donne le temps total de parcours du tronçon AB.

Comme nous l'avons souligné plus haut, malgré la qualité des travaux qui ont porté sur l'estimation du temps de parcours d'un tronçon et les différentes approches proposées, il demeure qu'elle reste imparfaite puisqu'imprécise et incomplète car issue d'extrapolation spatio-temporelle de données recueillies par endroit. Elle reste donc très peu réaliste étant donné que rien ne peut permettre de prédire le comportement réel dans le temps d'un véhicule sur un axe routier et donc son temps de parcours de cet axe. Le temps de parcours est une valeur qui se construit au cours du temps puisque tant qu'un véhicule n'est pas sorti du tronçon de route sur laquelle il se trouve, on ne peut connaître le temps réel qu'il va mettre pour réaliser le parcours de ce tronçon.

Le temps de parcours devient donc une notion ambiguë et parfois très peu représentative surtout en cas d'évolution régulière des situations spatio-temporelles pouvant engendrer de variabilité de comportement des conducteurs et influencer sur le trafic.

L'idée de la base d'apprentissage, c'est d'arriver à concevoir un système expert d'aide à la décision de sorte qu'à chaque fois qu'un véhicule sera détecté au niveau du capteur A, il suffira d'identifier sa catégorie, de déterminer sa vitesse instantanée, de consulter dans une base dynamique la valeur de la densité du trafic, pour lui associer un temps moyen de parcours ainsi qu'un écart type pour le tronçon qu'il veut entamer.

N'ayant pas pu réunir toutes les conditions logistiques et financières pour mettre en œuvre les différentes techniques de relevés puis faire une fusion d'informations pour construire un système expert robuste et exploitable, nous avons plutôt opté pour l'utilisation d'équations cinématiques pour faire une estimation du temps de parcours.

Ainsi, en partant de la formule cinématique de l'expression de la distance en fonction du temps ci-dessous :

$$x(t) = x_0 + V_0 t + \frac{1}{2} a t^2 \quad (6-5)$$

En posant $d(t) = x(t) - x_0$ on a :

$$d(t) = V_0 t + \frac{1}{2} a t^2 \quad (6-6)$$

On déduit par dérivation que :

$$V(t) = V_0 + a t \quad (6-7)$$

De même, la vitesse d'un véhicule à un instant est influencée par la densité du trafic environnant et est inversement proportionnelle à la densité du trafic à cet instant [115], suivant la formule ci-après :

$$V(t) = V_{max} \cdot \left(1 - \frac{\rho(t)}{\rho_{max}} \right) \quad (6-8)$$

où

v_{max} : vitesse maximale de l'objet sur l'axe routier suivant sa classe d'appartenance

ρ_{max} : densité maximale à partir de laquelle des objets peuvent être considérés comme étant à l'arrêt

$\rho(t)$: densité d'objets sur le parcours à l'instant t. Il est proportionnel au nombre d'objets par unité de voie ou d'espace.

Pour évoluer, nous avons considéré qu'à tout mouvement non forcément uniformément accéléré sur un axe donné, on peut toujours associer une valeur moyenne de l'accélération pour construire par approximation un mouvement uniformément accéléré. A partir de cette approximation, une égalité entre les deux expressions ci-dessus de la vitesse en (6-7) et (6-8) nous permet de déduire une expression de l'accélération et son remplacement dans la première équation de la distance en (6-4) nous donne l'équation ci-dessous :

$$d(t) = \frac{1}{2} \left[V_{max} \left(1 - \frac{\rho(t)}{\rho_{max}} \right) + V_0 \right] \cdot t \quad (6-9)$$

N'étant pas en mesure de donner une expression algébrique de la fonction $\rho(t)$, nous avons cependant la possibilité d'en connaître la valeur à tout instant t par décompte.

Ainsi, si D est le segment de route dont on veut estimer le temps de parcours connaissant la vitesse V_0 du véhicule à l'entrée du segment, la densité du trafic ρ_0 à l'instant où le véhicule entre sur segment, puis la densité maximale ρ_{max} du trafic sur ce segment, on a :

$$D = \frac{1}{2} \left[V_{max} \left(1 - \frac{\rho_0}{\rho_{max}} \right) + V_0 \right] \cdot t_{moy} \quad (6-10)$$

On en déduit facilement l'expression du temps moyen t_{moy} mis par le véhicule pour effectuer ce trajet.

$$t_{moy} = \frac{2 \cdot D}{V_{max} \left(1 - \frac{\rho_0}{\rho_{max}} \right) + V_0} \quad (6-11)$$

C'est cette expression que nous avons utilisé en lieu et place du système d'apprentissage pour faire nos tests. Mais en dépit de la qualité des travaux qui ont trait à l'estimation du temps de parcours, son estimation reste encore imparfaite car imprécise et incomplète. En effet, rien ni personne ne peut prédire exactement le comportement d'un véhicule sur une route. Le temps de parcours se construit au fil du temps et tant qu'un véhicule n'est pas à la fin d'un trajet, on ne peut pas savoir avec précision le temps qu'il mettra pour effectuer le parcours.

Pour prendre en compte l'incertitude et l'imprécision de notre estimation du temps de parcours, nous avons décidé de le modéliser par une distribution Gaussienne. En effet, chaque temps de parcours étant en réalité un temps moyen estimé, il peut donc être considéré comme la moyenne d'une distribution gaussienne dont on connaît l'écart type (estimation par apprentissage). La fonction de cette distribution est présentée ci-dessous.

$$f(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{(t - \mu)^2}{2\sigma^2} \right] \quad (6-12)$$

En partant de la formule de la distance de Mahalanobis [116] défini par l'expression (6-13) et en considérant le cas unidimensionnel, la mesure de la distance entre un temps t de parcours réellement observé et le temps moyen μ estimé est donnée par la formule (6-14).

$$D(X_i, X_k) = \sqrt{(X_i - X_k)^T \Sigma^{-1} (X_i - X_k)} \quad (6-13)$$

$$D(t, \mu) = \sqrt{\frac{(t - \mu)^2}{\sigma^2}} = \frac{|t - \mu|}{\sigma} = \frac{|t - t_{moy}|}{\sigma} \quad (6-14)$$

6.3.1.3. Récapitulatif des descripteurs

Le tableau 6.3 ci-dessous présente un récapitulatif de l'évaluation de nos descripteurs visuels. Nous présentons juste après son commentaire, notre démarche pour l'évaluation de notre descripteur temporel.

Descripteur	Bonnes associations	Mauvaises association
histogramme multi-échelles	89,22%	10,88%
Boite englobante minimale	56,76%	43,24%

Tableau 6.3. Résultats d'associations de nos descripteurs visuels

L'analyse de ce tableau montre que l'histogramme multi-échelles est un outil assez robuste pour la comparaison d'images. Il est invariant à la rotation, à la translation et très peu sensible au changement d'échelle et d'angle. Son insuffisance se situe juste au niveau de sa non prise en compte d'information temporelle pour discriminer deux images.

Le rectangle englobant est par contre un descripteur très grossier. Il n'est vraiment efficace que pour discriminer deux véhicules de catégories différentes. Il est donc utilisé dans notre travail pour compléter le premier descripteur visuel.

Le dernier descripteur, relatif au temps de parcours a été un peu difficile à évaluer du fait que nous n'avons pas pu réunir suffisamment de mesures pour mettre en place un système d'apprentissage robuste. Cependant, nous avons pu faire quelques tests à partir de la fonction présentée dans la formule (6-11) qui nous ont permis de constater la cohérence de son évolution aussi bien par rapport à l'évolution de la densité du trafic, que par rapport à l'évolution de la vitesse initiale à l'entrée du segment dont on veut évaluer le temps de parcours. Les figures 6-9 et 6-10 ci-dessous en donnent des illustrations.

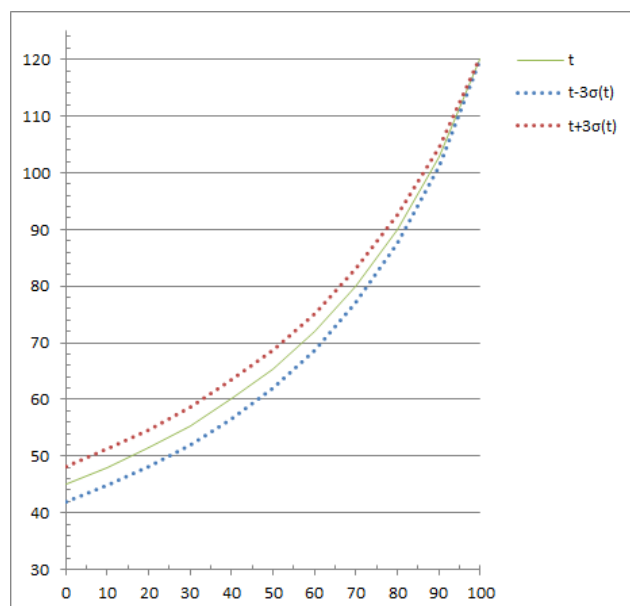


Figure 6.9: Evolution du temps de parcours en fonction de la densité du trafic

Sur cette figure, l'axe horizontal représente les valeurs de la densité du trafic (nombre de véhicules par km). L'axe vertical représente les temps de parcours en considérant $V_0=60\text{km/h}$, $V_{\text{max}} = 130\text{km/h}$, $D=1\text{km}$ et $\rho_{\text{max}}=100$.

La courbe du milieu, en vert, représente l'évolution du temps de parcours estimé. La surface délimitée par les deux courbes de part et d'autre de la courbe de t , représente le domaine d'appartenance du temps de parcours en prenant en compte les imprécisions dans les mesures et les calculs.

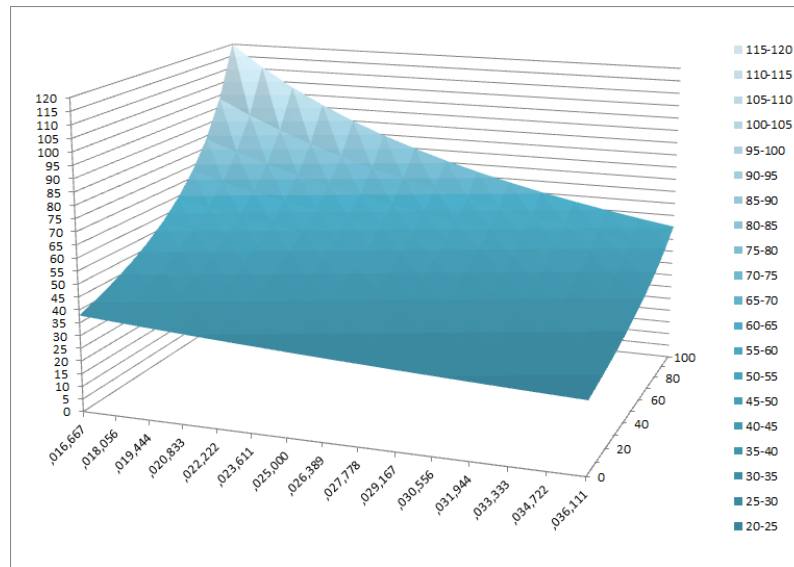


Figure 6.10: Evolution du temps de parcours en fonction de la densité du trafic et de la vitesse V_0 d'entrée

Sur cette figure, le temps de parcours est représenté par l'axe vertical, la vitesse initiale et la densité du trafic représentent le plan horizontal.

6.3.2. RESULTATS DE SUIVI ET RE-IDENTIFICATION DE NOTRE SYSTEME

Nous présentons ci-dessous, et de façon sommaire le fonctionnement global de notre système. Nous en donnons quelques détails ainsi que quelques résultats.

Fonctionnement global du SMA de suivi

- 1) Capture de séquences vidéo
- 2) Détection de mouvement : codebook ou soustraction simple de background. (Notre objectif à travers cette partie est d'arriver à mettre en place un masque des objets en mouvement. Ce masque sera utilisé pour mieux isoler les objets en mouvement et faire des histogrammes robustes et moins bruités).
- 3) Application de l'algorithme de Viola-Jones pour identifier les objets d'intérêt (Véhicules) parmi ceux en mouvement
- 4) Création de rectangle englobant autour de chacun des objets d'intérêt
 - a. Si objet précédemment détecté dans la scène, alors pas de traitement (pas de rectangle)
 - b. Sinon, pour chaque objet d'intérêt, utiliser coordonnées du rectangle englobant pour capturer aussi bien l'image ainsi que son masque (depuis une autre fenêtre) ; puis appliquer masque à l'image. (On obtient comme résultat la suppression de l'arrière-plan dans l'image).

- De là, calculer le rectangle minimum englobant puis l'histogramme de l'image contenue dans ce rectangle minimum englobant. Mettre en place la signature de l'objet et lui associe un identifiant unique.
- 5) Ajouter signature aux listes appropriées (pour prochaine détection)
 - 6) Pour tout suivi local, vérifie si liste d'objets attendus est vide ou non.
 - a. Si oui, ajouter l'identifiant de l'objet perçu à la liste des objets perçus et aller à l'étape 7.
 - b. Sinon, chercher dans le graphe la liste des capteurs (caméras) suivants et leur envoyer l'identifiant de l'objet détecté aussi que l'estimation pour chacun d'eux, de sa date d'arrivée à son niveau en se basant sur les distances entre capteurs voisins, la densité du trafic et la vitesse de l'objet au moment de la détection.
 - c. Dans tous les cas, stockage de la signature dans le dossier approprié (après sérialisation, et enregistrement des informations y relatives.
 - 7) Déclencher le processus de mise en correspondance (TBM) si objets attendus
 - 8) En cas d'association, faire la mise à jour des listes des objets attendus en y supprimant tous les objets associés. Mettre à jour dans ces conditions les densités du trafic sur les branches concernées. Envoyer un message à tous les voisins attendant le même objet, de ne plus l'attendre. Ceci va leur permettre de mettre aussi à jour leur liste d'objets attendus.
 - 9) Retourner à 1.

Nota : Pour savoir si un objet a été déjà détecté ou pas dans une scène, on fait du suivi local.

1. Une fois que Viola-Jones a permis d'identifier tous les objets d'intérêt dans un frame, on dispose d'un tableau contenant les centres de gravité ou barycentres des objets détectés dans ce frame. On fait la sauvegarde de ce tableau dans une variable **prev**.
2. Un nouveau frame est capturé et on procède de même mais cette fois-ci dans une variable **new**. Chaque élément du tableau **new** est comparé à tous les éléments de la liste sauvegardée dans **prev**. Le point qui va maximiser (au-delà d'un seuil) la distance minimum aux points de la liste **prev** sera le bon candidat (nouvel objet). En effet, nos objets étant d'une certaine taille, il doit exister une distance minimale (seuil) qui sépare chaque objet des autres.
3. On ne calcule que la signature des nouveaux objets.



Figure 6.11: Images de véhicules filmés à partir de caméras et de scènes différentes

La figure 6.11 ci-dessus, présente des captures de véhicules observées par deux différentes caméras positionnées à 1km l'une de l'autre sur un tronçon d'autoroute. On peut y voir que les images situées dans la même colonne présentent le même véhicule, vu par l'une et l'autre caméra. On peut y remarquer également que le contraste des images n'est pas le même et que les tailles des véhicules.

Pour corriger l'effet du contraste dans nos traitements, nous avons égalisé nos histogrammes pour ajuster leur contraste en répartissant mieux les intensités sur l'ensemble de la plage de valeurs possible. Pour corriger l'influence de la profondeur sur les dimensions des boites englobantes, nous avons eu recours au facteur de grossissement suggéré plus haut.

Après plusieurs captures et tests, nous présentons ci-dessous dans le tableau 6.1 les résultats de nos suivis comparativement à ceux obtenus avec d'autres approches.

approche	bonnes associations	non association	mauvaises associations
NN	64,5%	0%	35,5
TBM	74,55%	20%	5,45%
TBM+MHT	81,83%	12,72%	5,45%

Tableau 6.4. Résultats expérimentaux de ré-identification d'objets

Ces résultats sont assez encourageants compte tenu de nos conditions de travail et des estimations parfois grossières utilisées. Ils nous permettent de confirmer la qualité de nos métriques ainsi que celle de nos fonctions de masse dans le cadre du déploiement du TBM. On remarque également que l'utilisation du MHT à travers le maintien temporelle de piste pour retarder la prise de décision en cas d'ambiguïté, permet d'apporter une amélioration de l'ordre de 7,28% à la qualité de la ré-identification. Le taux, tout de même un peu élevé des non-associations ou des mauvaises associations s'explique partiellement par l'inexactitude dans l'estimation du temps de parcours et de la difficulté à discriminer deux véhicules ayant presque les mêmes caractéristiques visuelles.

6.4. CONCLUSION

Les résultats de nos travaux nous montrent que le TBM est un bon outil de représentation, de combinaison de données (fusion d'information) et de prise de décision dans le domaine de l'incertitude et de l'imprécision des mesures. Le MHT, agit comme un mécanisme de fusion temporelle d'information et permet de résoudre une partie des problèmes d'association de données en cas d'ambiguïtés.

Dans ce travail, la qualité des résultats dépend beaucoup de l'exactitude de l'estimation du temps de parcours. Lorsque cette estimation est trop éloignée de la réalité, les résultats sont totalement biaisés. Une prise en compte dans nos modèle de plusieurs autres facteurs affectant le comportement d'un conducteur et la vitesse d'un véhicule, pourrait nous aider à améliorer le model d'estimation du temps de parcours et ainsi impacter positivement sur nos résultats. En outre, le modèle utilisé dans notre travail pour l'estimation du temps de parcours est un modèle macroscopique qui ne permet pas de prendre en compte les comportements autonomes et individuels des véhicules à travers leurs conducteurs. Une piste d'amélioration serait d'utiliser un modèle d'agents dans lequel chaque véhicule sera représenté par un agent qui évolue de façon autonome et interagit dans l'environnement autoroutier avec d'autres agents (véhicules et autres équipements).

Conclusion générale et perspectives

Bilan de la thèse

Les principaux objectifs visés par cette thèse sont d'une part le suivi multi-caméras de véhicules sur un vaste étendu avec des zones aveugles, puis dans un second temps la proposition d'un environnement ou d'une architecture logicielle pour la gestion de ce suivi. L'approche utilisée pour le suivi a été le Modèle de Croyances transférable ou Transferable Believe Modele (TBM) en anglais. Ce choix a été opéré à cause de la capacité du TBM à modéliser et à combiner des connaissances imparfaites pour arriver à prendre des décisions en intégrant éventuellement le risque ou le gain lié à cette prise de décision. Dans ce travail, notre apport a été de proposer dans un premier temps des caractéristiques visio-temporelles pour élaborer les signatures des objets à suivre puis d'utiliser ses informations de signature pour construire des fonctions de masses de croyance utilisées dans le TBM. Dans un second temps, nous avons, en nous basant sur les travaux de [69], proposé une approche d'utilisation de MHT pour la levée d'ambiguïté dans la prise de décision dans le TBM tout en réduisant considérablement tout risque d'explosion combinatoire.

L'utilisation des histogrammes multi-échelles comme un des descripteurs des caractéristiques visuelles des objets permet de prendre en compte la répartition spatiale des couleurs des objets suivis et de renforcer la robustesse de la signature de chacun de ses objets. Cependant, le fait que plusieurs véhicules aient les mêmes caractéristiques visuelles ne facilite pas la tâche dans leur discrimination lors de la ré-identification. Les caractéristiques temporelles de ses véhicules deviennent les seules informations permettant de les discriminer. Mais l'imprévisibilité du modèle d'accélération d'un véhicule sur un tronçon de route vient complexifier la robustesse et l'efficacité de cette information capitale dont nous n'avons pu faire qu'une approximation un peu grossière mais très cohérente de par la qualité de nos résultats. Des perspectives d'améliorations existent et nous en présenterons certaines dans nos perspectives.

Dans un troisième temps, nous avons proposé une architecture multi-agents pour la gestion logicielle de tout le système de suivi et de ré-identification des objets (véhicules) d'une caméra à l'autre. L'intérêt d'une telle architecture s'explique par la nécessité de gestion distribuée des différents mécanismes et étapes complexes qui interviennent dans un suivi multi-caméras. Nous avons distingué deux niveaux de suivi. Le suivi local au niveau de chaque caméra et un suivi global qui s'appuie sur les données haut niveau issues des suiveurs locaux. Les caméras ou les systèmes les représentant communiquent et échangent des informations en temps réel pour disposer d'informations actualisées indispensables pour la fusion d'information et la ré-identification. Un SMA nous a permis de garantir facilement l'autonomie de chaque caméra dans notre implémentation tout en mettant à disposition les mécanismes de coopération pour une vidéosurveillance globale. L'autre avantage de notre SMA est que

l'ajout et/ou la suppression d'une ou de plusieurs caméras du système peut se faire de façon dynamique en mettant juste à jour le graphe orienté modélisant le réseau de caméras réparties le long des axes routiers (figure 5.3). Cependant, chaque ajout de caméra entraîne l'ajout au minimum d'un agent de type agentTV pendant que le nombre d'agent de type agentTBM ne changera pas. Cet état de choses peut influencer les performances de l'agentTBM et nuire à la qualité des traitements temps réel. Il serait donc opportun de trouver une solution dynamique pour éviter de surcharger l'agentTV qui reste une pièce maîtresse de notre système de ré-identification.

Perspectives

Nous entrevoyons plusieurs perspectives d'évolution de nos travaux pour répondre à certaines de nos questionnements ou pour améliorer ou optimiser le fonctionnement de notre système.

La qualité de nos résultats repose entièrement sur la robustesse de nos descripteurs et des signatures qui en découlent. Nous avons vu que les boîtes englobantes restent des caractéristiques visuelles très peu discriminantes. En lieu et place, on pouvait envisager d'utiliser une autre caractéristique visuelle plus discriminante telle que les contours par exemple. De même, les caractéristiques visuelles ne permettent pas à elles seules de certifier la ré-identification d'un véhicule dans un suivi multi-caméras sans aucune zone de chevauchement. Plusieurs faux positifs sont envisageables du fait qu'en général plusieurs véhicules de même modèle ou non ont les mêmes couleurs et pratiquement la même forme. La caractéristique temporelle devient une composante primordiale pour discriminer deux véhicules ayant pratiquement les mêmes caractéristiques visuelles. Or, la détermination du temps de parcours ou plus précisément son estimation reste une tâche très complexe et une question d'actualité. Dans notre système, nous en avons fait une estimation avec une approche macroscopique ne tenant pas compte dans un premier temps de l'autonomie de chaque véhicule et/ou de son conducteur, puis dans un second temps des interactions entre les véhicules, et entre les véhicules et l'environnement. Lorsque cette estimation ne reflète pas la réalité, les résultats de la fusion d'information sont totalement biaisés et pénalisent la ré-identification. Il aurait été certainement plus judicieux et plus réaliste d'envisager une modélisation microscopique des véhicules et de leurs conducteurs en utilisant un système multi-agents pour modéliser les véhicules et leurs interactions, et en prenant en compte dans le modèle la plupart des facteurs (environnementaux, émotionnels, etc.) pouvant influencer le comportement d'un conducteur et affecter le modèle d'accélération du véhicule.

Une amélioration serait possible également au niveau de TBM et du MHT dans la phase de prise de décision.

Au niveau de notre modèle multi-agents pour le suivi et la ré-identification, il serait intéressant de le faire évoluer pour la prise en compte de façon dynamique et optimale du nombre

d'agentTBM à utiliser suivant le nombre de caméras en présence et suivant la densité du trafic au niveau des différents caméras, plutôt que de n'en utiliser qu'un seul tels que c'est le cas pour le moment. Une telle architecture nécessitera cependant l'intégration d'autres protocoles de communications telles que le FIPA-ContractNet-Protocol pour faire des appels à proposition à l'endroit des agentTBM s'il y en a plus d'un, et sélectionner la meilleur offre en terme de disponibilité. L'intégration des ontologies dans les actes de communication serait aussi une marque de facilitation d'interopérabilité au cas où nos agents arriveraient à être réellement répartis sur plusieurs plateformes interconnectées via un réseau informatique local ou étendu.

Bibliographie

- [1] K. Toyama, J. Krumm, B. Brummit and B. Meyers, *Wallflower: Principles and Practice of Background Maintenance*, In Proc. IEEE Int. Conf. on Computer Vision (ICCV'99), Kerkyra, Corfu, Greece, Sep. 1999, pp. 255–261.
- [2] H. Wang and D. Suter. A consensus-based method for tracking: *Modelling background scenario and foreground appearance*. Pattern Recognition, 40(3) :1091-1105, 2007.
- [3] N. Verbeke, thèse : *Suivi d'objets en mouvement dans une séquence vidéo*, Paris Descartes, 2007, pp. 15-42
- [4] M. Julien Richefeu, thèse : *Détection et analyse du mouvement sur système de vision à base de rétine numérique*, Université Paris 6, pp. 95-157
- [5] Y.-F. Ma and H.-J. Zhang, *Detecting Motion Object by Spatio-Temporal Entropy*, In Proc. IEEE Int. Conf. on Multimedia and Expo (ICME 2001), Tokyo, Japan, Aug. 2001, pp. 265–268.
- [6] J. Guo, E. S. Chng and D. Rajan, *Foreground Motion Detection by Difference-Based Spatial Temporal Entropy Image*, In Proc. IEEE Region 10 Conf. (TenCon 2004), Chiang Mai, Thailand, Nov. 2004, pp. 379–382.
- [7] N. Verbeke, N. Vincent, *Détection de Mouvements Cohérents dans une Séquence Vidéo*, 4ème conférence internationale SETIT, 2007
- [8] A. Ilyas, M. Scuturici and S. Miguet, *Real time foreground-background segmentation using a modified codebook model*, In AVSS09, pages 454–459, 2009.
- [9] A. ILYAS, *Object Tracking and Re-identification in Multi-Camera Environments*, LIRIS, Université Lumière, Lyon 2, 2011.
- [10] B. D. Lucas and T. Kanade, *An iterative image registration technique with an application to stereo vision*. In Proc. 7th Int. Joint Conf. On Artificial Intelligence, pp 674-679, Vancouver, Canada, 1981.
- [11] R. Cutler and L.S. Davis, *View-based detection and analysis of periodic motion*. In Proc. 14th Int. Conf. on Pattern Recognition (ICPR'98), volume 1, pages 495-500, Brisbane, Australie, 1998.
- [12] J. Rittscher, J. Kato, S. Joga and Blake, A., *A probabilistic background model for tracking*, Proc. 6th European Conf., Computer Vision (ECCV 2000), volume II, pp. 336-350, Dublin, Irlande, 2000.

- [13] C. Stauffer, W. Eric, and L. Grimson. *Learning patterns of activity using real time tracking*. Transaction on Pattern Analysis and Machine Intelligence, 22(8):747–757, 2000.
- [14] K. Kim, T. Thanarat, H. Chalidabbhogse, D. Harwood, and L. Davis. *Real time foreground-background segmentation using codebook model*. Real-Time Imaging, 11 (3):172–185, 2005.
- [15] M. Mikram, R. Megret, Y. Berthoumieu and M. Donias, *Multi-Scale Histograms for Kernel-Based Object Tracking*, ISCCSP, 2006.
- [16] M. MIKRAM, Thèse - *Suivi d'objets dans une séquence d'images par modèle d'apparence : conception et évaluation*, UNIVERSITÉ DE BORDEAUX I, 2008.
- [17] J. Koenderink, *The Structure of Images*, *Biological Cybernetics* 50, 363-370, 1984.
- [18] A. Witkin, *Scale-Space Filtering*, International Joint Conference on Artificial Intelligence (IJCAI), 1019-1022, 1983.
- [19] R. Cucchiara, C. Grana, M. Piccardi and A. Prati, *Detecting moving objects, ghosts, and shadows in video streams*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 10, (Sept. 2003), pp. 1337-1342, ISSN 0162-8828, 2003
- [20] D. Comaniciu, V. Ramesh and P. Meer, *Kernel-based object tracking*, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (5), 564–577, 2003.
- [21] A. Lehuger, P. Lechat, N. Laurent and P. Pérez, *Maximisation du pouvoir discriminant des espaces colorimétriques pour le suivi d'objets* RFIA, 2006.
- [22] D. Jobson, Z. Rahman and G. Woodell, *Properties and performance of a center/surround retinex*, IEEE Transactions on Image Processing 6, 451–462, 1997.
- [23] S. Gottschalk, M.C. Lin, D. Manocha, *OBBTree : a Hierarchical Structure for Rapid Interference Detection*, SIGGRAPH'96 Conference Proceedings, Computer Graphics annual conference series, New Orleans, pp 171-180, 1996.
- [24] Y. Bar-Shalom and E. Tse, *Tracking in a cluttered environment with probabilistic data association*, Automatica, vol. 11, pp. 451-460, Sept. 1975
- [25] V. Cherfaoui, *Contribution à la gestion des incertitudes en fusion multicapteurs. Application à la perception du contexte de conduite*, HDR, Université de technologie de compiègne, 2009.
- [26] A. Garcia-Alonso, N. Serrano and J. Flaquer, *Solving the Detection Problem*, IEEE Computer Graphics and Applications, pp 36-43, Mai 1994.

- [27] B. Jida, Thèse : *Intégration du contexte par réseaux bayésiens pour la détection et le suivi multi-cibles*, ULCO, 2008.
- [28] Y. Bar-Shalom and T. Foreman, *Tracking and Data Association*, New York, Academic Press Inc, 1988.
- [29] S. Blackman, R. Popoli , *Design and Analysis of Modern Tracking Systems*. Artech House, 1999.
- [30] Y. Bar-Shalom and X. R. Li, *Multitarget-Multisensor Tracking : Principles and Techniques*, Storrs, CT: YBS Publishing, 1995
- [31] T. Foreman, Y. Bar-Shalom and M. Scefte, *Sonar tracking of multiple targets using joint probabilistic data association*, IEEE J. Oceanic Eng., vol. OE-8, pp. 173-183, 1983
- [32] D. B. Reid, *An algorithm for tracking multiple target*, IEEE Trans. Automatic Control, vol. AC-24 : 843-854, 1979
- [33] L. A. Zadeh, *Fuzzy Sets and Information Granularity*, In M. Gupta, R. Ragade, and R. Yager, editors, *Advances in Fuzzy Set Theory and Applications*, pp. 3-18. North-Holland, Amsterdam, 1979.
- [34] L. A. Zadeh, *Fuzzy Sets. Information and Control*, 8, pp. 338-353, 1965.
- [35] D. Dubois et H. Prade, *Théorie des possibilités*, Masson, Novembre 1987
- [36] P. Smets, *Medical Diagnosis : Fuzzy Sets and Degree of Belief*. In Colloque International sur la Théorie et les Applications des Sous-Ensembles Flous, Marseille, 1978.
- [37] L. A. Zadeh. *Fuzzy Sets as a Basis for a Theory of Possibility*. *Fuzzy Sets and Systems*, 1, pp. 3-28, 1978.
- [38] D. Dubois and H. Prade, *Fuzzy Sets and Systems : Theory and Applications*, Academic Press, New-York, 1980.
- [39] D. Dubois and H. Prade, *Possibility Theory*, Plenum Press, New-York, 1988.
- [40] I. Bloch, *Information Combination Operators for Data Fusion : A Comparative Review with classification*, IEEE Transactions on Systems, Man, and Cybernetics - Part A : Systems and Humans, 26(1):52_67, 1996.
- [41] M. Grabisch et P. Perny, *Logique floue, principes, aide à la décision*, chapitre Agrégation multicritère, pages 219-258. Hermes Science Publication, 2003.
- [42] A. Dempster, *Upper and lower probabilities induced by multi-valued mapping*, *Annals of Mathematical Statistics AMS-38*, pp. 325-339, 1967.

- [43] G. Shafer, *A Mathematical Theory of evidence*, Princeton University Press, Princeton, NJ, 1976.
- [44] M. Menard, E. H. Zahzah and A. Shahin, *Mass Function Assessment : Case of Multiple Hypotheses for the Evidential Approach*, In Europto Conf. on Image and Signal Processing for Remote Sensing, Taormina, Italy, September 1996.
- [45] T. Denoeux, *A k-Nearest Neighbor Classification Rule Based on Dempster-Shafer Theory*. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 25(5):804-813 (1995)
- [46] P. Cucka and A. Rosenfeld, *Evidence-based Pattern Matching Relaxation*, Technical Report CAR-TR-623, Center of Automation Research, University of Maryland, 1992.
- [47] Y. Permal et al., *Initialisation des masses d'évidence par les Okm pour la théorie des fonctions de croyance*. Application aux bioprocédés, Atelier "Fouille de Données Complexes", EGC, 2009.
- [48] P. Vannoorenberghe, *Un état de l'art sur les fonctions de croyance appliquées au traitement de l'information*. Information-Interaction-Intelligence (I3) 3(2), 1–37,2003.
- [49] P. Vannoorenberghe et T. Denoeux, *Diagnostic de la pollution atmosphérique par une approche RDF utilisant les fonctions de croyance*, Colloque Automatique et Environnement A&E 2001, Saint-Etienne, 4-6 juillet 2001.
- [50] A. Appriou, *Uncertain data aggregation in classification and tracking processes*, In « Aggregation and Fusion of Imperfect Information », B. Bouchon-Meunier, ed., Physica Verlag, 1998.
- [51] S. Fabre, A. Appriou, X. Briottet, *Presentation and Description of two Classification Methods Using Data Fusion Based on Sensor Management*, Information Fusion Journal, Vol. 2, n° 1, Elsevier, March 2001.
- [52] E. Lefevre, P. Vannoorenberghe et O. Colot, *Using Information Criteria in Dempster-Shafer's Basic Belief Assignment*, In International Conference on Fuzzy Systems, pages 173_178, Séoul, Corée, Août 1999.
- [53] R. Yager, *On the Dempster-Shafer Framework and New Combination Rules*, Informations Sciences, 41:pp. 93-137, 1987.
- [54] T. Denoeux, *Application du modèle des croyances transférables en reconnaissance de formes*, Traitement du Signal, 14(5):pp. 443-451, 1997.
- [55] P. Smets, *Belief functions on real number*, Int. J. Approx, Reasoning : Forthcoming, 2004.

- [56] M. Rombaut, *Decision in multi-obstacle matching process using Dempster-Shafer's theory*, AVCS'98, Amiens France, 1998.
- [57] B. Voir et al., *Les méthodes de raisonnement dans les images*, ENST, département TSI, 2004.
- [58] A. Martin, *La fusion d'informations*, cours ENSIETA - Réf. : 1484, 2005.
- [59] P. Smets, *The Combination of Evidence in the Transferable Belief Model*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(5): pp. 447-458, 1990.
- [60] P. Smets, *Constructing the pignistic probability function in a context of uncertainty*, Uncertainty in Artificial Intelligence, 5: pp. 29-39, 1990.
- [61] Ph. Smets and R. Kennes, *The Transferable Belief Model*, Artificial Intelligence, 66(2) : pp. 191-234, 1994.
- [62] C. Motamed, O. Wallart, *Object tracking over wide area using distributed vision*, Application to freeway surveillance. Traitement du Signal 2003, Vol. 20, GRETSI, France 87-100, 2003.
- [63] C. Motamed, *Motion detection and tracking using belief indicators for an automatic visual-surveillance system*, Image and Vision Computing, Volume 24, Issue 11, 1 November, 1192-1201, 2006.
- [64] E. Ramasso, M. Rombaut, D. Pellerin : *Modèle des Croyances Transférables : Représentation des connaissances, Fusion d'informations, Décision*, hal-00283144, version 1, 2008.
- [65] D. Mercier et al., *Association pour le suivi d'objets dans le cadre des fonctions de croyance, appliquée aux véhicules intelligents*, LGI2A, RNTI, 2008.
- [66] Y. F. Ma et H. J. Zhang, *Detecting motion object by spatio-temporal entropy*. IEEE Int. Conf. on Multimedia and Expo (ICME 2001), pp. 265-268, Tokyo, Japon.
- [67] P. Smets, *An axiomatic justification for the use of belief function to quantify beliefs*. In IJCAI-93, éditeur, International Joint Conference on Artificial Intelligence, pages 598-603. Morgan Kaufman, San Mateo, Ca, 1993.
- [68] P. Smets, *The normative representation of quantified beliefs by belief functions*, Artificial Intelligence, 92(1-2): pp. 229-242, 1997.
- [69] E. Etievent, thèse : *Assistance à l'indexation vidéo par analyse du mouvement*, INSA de Lyon, Laboratoire Reconnaissance de Formes et Vision, 2002.

- [70] M. Rombaut, *Decision in multi-obstacle matching process using the theory of belief*, AVCS'98, pp. 63–68, 1998.
- [71] M. Rombaut and V. Cherfaoui, *Decision making in data fusion using dempster-shafer's theory*, 3th IFAC Symposium on Intelligent Components and Instrumentation for Control Applications, Annecy, France, 1997.
- [72] D. Mercier et al., *Association pour le suivi d'objets dans le cadre des fonctions de croyance*, appliquée aux véhicules intelligents, *Revue des Nouvelles Technologies de l'Information*, Vol. E-18, pp. 25-50, Février 2010.
- [73] D. Gruyer and al., *Credibilistic multi sensor fusion for real time application, application to obstacle detection and tracking*. IEEE International Conference on Advanced Robotics, ICAR'2003, University of Coimbra, Portugal, 2003.
- [74] A. Appriou, *Probabilités et incertitude en fusion de données multi-senseurs*, *Revue Scientifique et Technique de la Défense*, vol. 11, p. 27-40, 1991.
- [75] J. Ferber. *Les Systèmes Multi Agents : vers une intelligence collective*, InterEditions, 1995
- [76] N.R. Jennings and M.J. (eds) Wooldridge. *Agent Technology: Foundations, applications, and Markets*. Springer-Verlag: Heidelberg, Germany, 1998
- [77] I. Jarras et B. Chaib-draa. multiagents, *Série Aperçu sur les systèmes Scientifique*, 2002s-67
- [78] S. Moujahed. *Approche Multi-Agents auto-organisée pour la résolution de contraintes spatiales dans les problèmes de positionnement Mono et Multi-Niveaux*, Université de Technologie de Belfort-Montbéliard, 7 décembre 2007
- [79] B. Mughadam and A. Pentland, (1997), '*Probabilistic visual learning for object representation*', *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19(7), 696–710.
- [80] O. Boissier, *Multi-Agent Systems, cours Master Web Intelligence*, 2012
- [81] M. Baldoni and al. *Declarative Agent languages and Technologies III*, Third International Workshop, DALT 2005
- [82] M. Wooldridge, *Agent-based software engineering, IEEE Proceedings on Software Engineering*, p.26–37, 1997
- [83] J. Ferber, *Multi-Agent System: An Introduction to Distributed Artificial Intelligence*. Addison Wesley, 1999

- [84] B. Chaib-draa, *Distributed Artificial Intelligence: An overview*. In A. Ken, J.G. Williams, C. M. Hall, and R.Kent, editors, *Encyclopedia Of Computer Science And Technology*, volume 31, pages 215-243. Marcel Dekker, Inc, 1994.
- [85] B. Chaib-draa, *Interaction between agents in routine, familiar and unfamiliar situations*. *International Journal of Intelligent and Cooperative Information Systems*, 1(5):7-20, 1996.
- [86] B. Moulin and B. Chaib-draa. *An overview of distributed artificial intelligence*. In G. M. P. O'Hare and N. R. Jennings, editors, *Foundations of Distributed AI*, pages 3-54. John Wiley & Sons : Chichester, England, 1996.
- [87] A. Drogoul. *Cours : l'Intelligence Artificielle Distribuée*, Equipe MIRIAD, Thème OASIS, LIP6 – Université Paris 6, 2001
- [88] D. Zeguida, *Une architecture multi-agents d'un système d'apprentissage avec compagnons*, France de magister, Université de Annaba, 2003
- [89] D. Anzieu et J-Y. Martin, *La dynamique des groupes restreints, 12ème édition*, Presses Universitaires de France – PUF, 2000
- [90] Y. Labrou, *Semantics for an Agent Communication Language*. PhD thesis, UMBC, 1996.
- [91] T. R. Gruber, A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2):199-220, 1993
- [92] T. R. Gruber, *Toward principles for the design of ontologies used for knowledge sharing*. Presented at the Padua workshop on Formal Ontology, March 1993, later published in *International Journal of Human-Computer Studies*, Vol. 43, Issues 4-5, November 1995, pp. 907-928
- [93] F. Gandon, *Ontology engineering a survey and a return of experience*. Technical report, INRIA, 2002
- [94] T. Gruber, *in the Encyclopedia of Database Systems*, Ling Liu and M. Tamer Özsu (Eds.), Springer-Verlag, 2009
- [95] A. v. Lamseerde, *Requirement Engineering, From System Goals to UML Models to Software Specifications*. John Wiley & Sons Ltd, West Sussex, 2009
- [96] E. Morin, *La Méthode (t.1): la Nature de la Nature*, Le Seuil, 1977
- [97] N. Wiener, *Cybernetics or control and communication in the animal and the machine*, 1948
- [98] Shannon & Weaver, *Théories mathématiques de la communication*, 1949

- [99] R. Querrec, *Les Systèmes Multi-Agents pour les Environnements Virtuels de Formation, Application à la sécurité civile*, thèse de doctorat Informatique, Laboratoire d'Informatique Industrielle (LI2/ENIB), Université de Bretagne Occidentale. Octobre 2002
- [100] D. Sperber, D. Wilson, 1986, *Relevance Communication and Cognition*, Oxford, Basil Blackwell, 1986
- [101] T. W. Malone, *Modeling Coordination in Organizations and Markets*, Management, Sciences vol. 33, n°10, pp. 1317-1332, 1987
- [102] T. Bouron, *Structures de communication et d'organisation pour la coopération dans un univers multi-agents : Une contribution à la définition d'un modèle de programmation agent basé sur les concepts d'engagement et d'acte de langage*, Université Paris , 1992
- [104] J. R. Galliers, thesis : *A Theoretical Framework for Computer Models of Cooperative Dialogue, Acknowledging Multi-agent Conflict*, Open University (UK), 1988
- [105] T. W. Malone and K. Crowston, *What is coordination theory and how it can help design cooperative work systems*, proceedings of the Conference on Computer-Supported Cooperative Work, Tora Bikson and Frank Halasz editors, pp 357-370, 1990
- [106] J. G. March and H. A. Simon, *Organizations*, Wiley, New York, 1958
- [107] R. G. Smith and R. Davis, *Frameworks for Cooperation in Distributed Problem Solving*, IEEE Transactions on Systems, Man and Cybernetics, SMC-11(1):61-70, 1981
- [108] C. Buisson, J-B. Lesort, *Comprendre le trafic routier, Méthodes et calculs*. CERTU, Éditions du Certu, France, 2010
- [109] F. Pesteil et Al., *Les temps de parcours : Estimation, diffusion et approche multimodale*. CERTU, Éditions du Certu, France, 2008
- [110] J. Bertrand et Al., *Mesure du temps de parcours par analyse de la signature électromagnétique des véhicules*. CETE de l'ouest, Laboratoire Régional des Ponts et Chaussées d'Angers, France, 2007
- [111] A. Ben Aissa, E. Lefevre, N-E. El Faouzi, *Classification multi-source via la théorie des fonctions de croyance : application à l'estimation des temps de parcours*, Revue de Statistique Appliquée, 2009
- [112] E. Lefevre, N-E. El Faouzi, *Estimation et fusion des Temps de parcours routiers par la théorie de l'évidence*, Workshop Fouilles de données complexes dans un processus d'extraction de connaissances, 6ème journée d'Extraction et Gestion des Connaissances, EGC'2006, 2006

- [113] M. Nikraz, G. Caireb and P.A. Bahria, *A Methodology for the Analysis and Design of Multi-Agent Systems using JADE*
- [114] F. Bonvalet, Y. Robin-Prevallee, *Mise au point d'un indicateur permanent des conditions de circulation en Ile-de-France*, Revue T.E.C. n° 84-85, 1987
- [115] B. Greenshields and Al., *A study of traffic capacity*. In *Proceedings of the Highway Research Board*, vol. 14, pp. 967–976 (1935)
- [116] P. Mahalanobis, *On the generalized distance in statistics*. *Proceedings of National Institute of Sciences of India*. Vol. 2, No. 1, pp. 49-55 (1936)
- [117] E.H. Durfee and al., *Negotiating Task Decomposition and Allocation Using Partial Global Planning*, *Distributed Artificial Intelligence*, 2, pp. 229-243, 1989.
- [118] S. E. Conry, and al., *Multistage negotiation in distributed constraint satisfaction*. *IEEE Transactions on Systems, Man, and Cybernetics*, December 1991.
- [119] R. Davis and R. Smith, *Negotiation as a metaphor for distributed problem solving*. *Artificial Intelligence*, 20(1): 63-109, January 1983.
- [120] S. J. Rosenschein and G. Zlotkin, *Rules of Encounter: Designing Conventions for Automated Negotiation Among Computer*. MIT Press, Boston, MA, 1994.
- [121] S. Kraus and J. Wilkenfeld, *Negotiations over time in a multi-agent environment*. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 56-61, Sydney, Australia, 1991.
- [122] S. Kraus, J. Wilkenfeld, and G. Zlotkin, *Multi-agent negotiation under time constraints*, *Artificial Intelligence*, 75(2):297-345, 1995.
- [123] C.E. Shannon and W. Weaver, *Mathematical Theory of Communication*, University of Illinois Press, urbana, 1964
- [124] H. S. Nwana et Martlesham Heath. *Software Agents: An Overview*, 1996.
- [125] P. Maes, *Designing autonomous agents*, MIT press, 1990
- [126] P. Maes, *Artificial life meets entertainment: lifelike autonomous agents*, *Comm. Of the ACM*, vol. 38, pp.108-114, 1995.
- [127] J. R. Searle, *Les Actes de langages – Essai de philosophie du langage*, trad. H. Pauchard, Paris, Hermann Editeur, Collection Savoir, 1972 (1969).
- [128] A. Martin, *Modélisation et gestion du conflit dans la théorie des fonctions de croyance*, Université de Bretagne occidentale, France, 2009

- [129] P.R. Cohen, H.J. Levesque, *Communicative actions for artificial agents*, Proceedings of the 1st International Conference on Multi-Agent Systems (ICMAS'95), pages 65–72, San Francisco, CA, USA, 1995.
- [130] , T. Berners-Lee, J. Hendler and O. Lassila, *The Semantic Web*, Scientific American, May 2001
- [131] P.G. Jade, Bellifemine, G. Caire, T. Trucco, G. Rimassa, *JADE : Programmer's Guide*, Février 2002.
- [132] FIPA, *FIPA '97 Specification*, Foundation for Intelligent Physical Agents, Geneva, Switzerland, 1997