N° d'ordre : 252 Année 1998

THÈSE

présentée en vue de
l'obtention du titre de

DOCTEUR
de
**L'ÉCOLE NATIONALE SUPÉRIEURE
DE L'AÉRONAUTIQUE ET DE L'ESPACE**

SPÉCIALITÉ : AUTOMATIQUE, INFORMATIQUE INDUSTRIELLE

par

Humbert FIORINO

ÉLABORATION DE CONJECTURES PAR DES AGENTS COOPÉRANTS

Soutenue le 4 décembre 1998 devant la Commission d'Examen :

MM.	L.	MARCÉ	Président
	R.	ALAMI	
	J.	ERCEAU	Rapporteur
	J.-P.	HATON	Rapporteur
	H.-J.	MÜLLER	
Mme	C.	TESSIER	Directrice de thèse

Thèse préparée au sein du département commande des systèmes et dynamique du vol de l'ONERA centre de Toulouse.

À mes parents,
À mon pétrel des neiges (*Isa Pagodroma Nivea*)...

«Vieux bureaucrate, mon camarade ici présent, nul jamais ne t'a fait évader et tu n'en es point responsable. Tu as construit ta paix à force d'aveugler de ciment, comme le font les termites, toutes les échappées vers la lumière. Tu t'es roulé en boule dans ta sécurité bourgeoise, tes routines, les rites étouffants de ta vie provinciale, tu as élevé cet humble rempart contre les vents et les marées et les étoiles. Tu ne veux point t'inquiéter des grands problèmes, tu as eu bien assez de mal à oublier ta condition d'homme. Tu n'es point l'habitant d'une planète errante, tu ne te poses point de questions sans réponses : tu es un petit bourgeois de Toulouse. Nul ne t'a saisi par les épaules quand il était temps encore. Maintenant, la glaise dont tu es formé a séché, et s'est durcie, et nul en toi ne saurait désormais réveiller le musicien endormi ou le poète, ou l'astronome qui peut-être t'habitait d'abord.»

Antoine de Saint-Exupéry, Terre des hommes.

Remerciements

ou «De l'impertinence d'un impétrant»

Je n'aime pas les remerciements. Cela ne signifie nullement que je ne me sente redevable de personne. Bien au contraire. Mon travail a bénéficié de très nombreuses contributions et d'un entourage facétieux qui a su m'accabler lorsque je doutais et me railler lorsque j'exultais.

Je tiens à saluer toutes les personnes qui ont fait que cette thèse soit possible : M. Jacques Kerbrat, directeur de SUPAÉRO, M. André Fossard, professeur à SUPAÉRO, Mme. Annie Bailhé, responsable de la formation doctorale à SUPAÉRO et M. Jean-Pierre Jung, chef du département de Commande des Systèmes et Dynamique du vol à l'ONERA-CERT.

Je n'oublie pas d'associer à cet hommage tous les membres du jury qui ont consacré une part de leur temps pour assister à ma soutenance et évaluer mon manuscrit : M. Rachid Alami, directeur de recherche au LAAS-CNRS, M. Yves Davaine, ingénieur à DASSAULT AVIATION et membre invité, M. Jean Erceau, ingénieur de recherche à l'ONERA-CHÂTILLON et rapporteur, M. Jean-Paul Haton, professeur à l'Université de Nancy et rapporteur, M. Lionel Marcé, professeur à l'Université de Bretagne Occidentale et président du jury, M. Jürgen Müller, ingénieur à DEUTSCHE TELEKOM et Mme. Catherine Tessier, ingénieur de recherche à l'ONERA-CERT et directeur de thèse émérite[1].

Je n'aime pas ces remerciements parce que la pudeur la plus élémentaire m'interdit de dire tout le bien que je pense de tous ceux qui m'ont fait l'amitié de partager avec moi tant de moments extrêmement gratifiants et si anodins en apparence ; parce que ces personnes n'attendent certainement pas que je leur montre une quelconque déférence ; parce que ces quelques lignes idiotes concluent trois années de bonheurs simples.

Merci à Catherine, Corine, Laurent, Nicolas, Roger, Claude, Patrick et Charles qui, tous les jours, m'ont répété la même leçon,

«ad agusta per augusta».

1. Peut-être préférerait-elle que j'écrive «directrice de thèse émérite», l'adjectif «émérite» ne faisant pas ici référence à un grand âge comme c'est habituellement l'usage – généralement en lieu et place d'un qualificatif plus perfide comme «édenté» –, mais plutôt à ses compétences tout à fait remarquables, à son esprit délié et primesautier, car chacun sait qu'aux âmes bien nées, la valeur n'attend point le nombre des années.

Table des matières

Introduction générale **1**

1 Les approches multiagents **5**
1.1 Thèmes de recherches 6
1.2 Modèles cognitifs et simulation d'organisations sociales 7
1.2.1 Approches cognitives 7
1.2.2 Approches réactives 12
1.3 Résolution distribuée de problèmes 15
1.4 Programmation Orientée Agents 17
1.5 Les protocoles de négociation automatiques 19
1.6 Coordination de systèmes autonomes et robotiques 20
1.7 «Collecticiels» et Interfaces Homme - Machine 21
1.8 Caractéristiques des approches multiagents 23
1.8.1 Notion d'agent 23
1.8.2 Notion d'interaction 26

2 Méthodes de coopération et de négociation **33**
2.1 La négociation 35
2.1.1 Le partage de tâche 35
2.1.2 La résolution de contraintes distribuées 37
2.1.3 Les protocoles de négociation 39
2.2 La coopération 40
2.2.1 Notion de cohérence 41
2.2.2 L'utilisation de connaissances communes 42
2.2.3 Échange d'information de contrôle 47
2.2.4 Élaboration d'une conjecture commune 49

3 Modèle de coopération **55**
3.1 Le cycle de Lakatos 58

3.2 Objectifs du modèle 60
3.2.1 Classe d'applications visées par le modèle 60
3.2.2 Place du modèle par rapport aux travaux existants . . . 62
3.2.3 Caractéristiques du modèle 63
3.3 Modèle de coopération 65
3.3.1 Composition 65
3.3.2 Formalisme de représentation 65
3.3.3 Définition 66
3.4 L'agent coopératif 68
3.4.1 Principe 68
3.4.2 Mise à jour de la conjecture commune 71
3.4.3 Mise à jour de la base de connaissances 75
3.4.4 Décision d'engagement 78
3.5 Le greffier 81
3.5.1 Aspects fonctionnels 81
3.5.2 État global de la fédération 82
3.5.3 Greffier et mise en œuvre 85
3.6 Fonctionnement dynamique du modèle 86

4 Notion de croyance commune 91
4.1 Représentation logique 93
4.1.1 Exemple 93
4.1.2 Représentation du monde par la logique prédicative . . . 94
4.2 Notion de croyances et de préférences 97
4.3 Cadre algébrique pour la fusion des préférences 102
4.3.1 Principe de la fusion endogène 102
4.3.2 Treillis de graphes de préférences 103
4.4 La notion de conflit et de croyance commune 105
4.4.1 Notion de conflit 105
4.4.2 Notion de croyance commune 110
4.5 Un opérateur de fusion sur les préférences 112
4.5.1 Procédure additive 112
4.5.2 Propriétés de la procédure additive 115

5 Préférences fondées sur la distance de Hamming 121
5.1 Un exemple de relation de préférences 121
5.1.1 Définitions 121
5.1.2 Procédure additive et relation de Hamming 124

5.2 Notion de cube 125
5.2.1 Définitions 125
5.2.2 Propriétés 128

6 Synthèse 133
6.1 Dispositif de reconnaissance d'activités 133
6.2 Définitions des éléments du modèle de coopération 135
6.3 L'agent de reconnaissance d'activités 136
6.3.1 Mise à jour de la base de connaissances 136
6.3.2 Mise à jour de la croyance commune 139
6.3.3 Décision d'engagement 146
6.4 Le greffier 147
6.5 Fonctionnement dynamique du modèle 149
6.5.1 Exemple 149
6.5.2 Discussion 157

Conclusion générale 159

Annexes 162

A Démonstrations des propriétés 163
Démonstration de la propriété 5.4 163
Démonstration de la propriété 5.5 164
Démonstration de la propriété 5.6 167

B Modélisation de la coopération 169

C Représentation logique 187
C.1 Logique propositionnelle 187
C.1.1 Syntaxe 187
C.1.2 Sémantique 187
C.1.3 Formules valides, inconsistantes 189
C.1.4 Théorie logique 191
C.2 Logique des prédicats du premier ordre 191
C.2.1 Vocabulaire 191
C.2.2 Interprétation 192
C.2.3 Interprétations de Herbrand 195
C.3 Modèles de Herbrand et mondes possibles 197
C.3.1 Modèles de Herbrand 197

C.3.2 Inconsistance, déduction 198
C.3.3 Décidabilité 200

D Réseaux de Petri colorés **201**

Table des figures

1.1 Intelligence Artificielle Distribuée 7
1.2 Évolution de la notion de programme 19
1.3 Place de la programmation orientée agents 20
1.4 Dualité fonctions/objects 24
1.5 Un système multiagent 26
1.6 Non déterminisme des interactions 27
1.7 Interactions cooperatives 28
1.8 Problème de décomposition 29
1.9 Problème de distribution 30
1.10 Problème d'association 30

2.1 Réseau contractuel : proposition de tâche 36
2.2 Réseau contractuel : offres 36
2.3 Réseau contractuel : évaluation des offres et attribution de la tâche 37
2.4 Processus de recherche 39
2.5 «Partial Global Planning» 48
2.6 Phase de délibération 53

3.1 Gnoptères en soutien à un avion de chasse 56
3.2 Paradigme Preuve - Réfutation 60
3.3 Place du modèle par rapport aux paradigmes de l'IAD 63
3.4 Vue fonctionnelle du modèle 67
3.5 Structure globale du modèle 68
3.6 Vue fonctionnelle des activités locales de l'agent 69
3.7 Admissibilité d'une conjecture 72
3.8 Module de mise à jour 75
3.9 Information issue d'un capteur 77
3.10 Module de mise à jour de la base de connaissances 79
3.11 Module de décision 81
3.12 Vue fonctionnelle du greffier 82

3.13 Fonctionnement dynamique du modèle 86
3.14 Mise en œuvre du modèle 89

4.1 environnement observé par les agents 94
4.2 Ensemble d'interprétations et états possibles de l'environnement 98
4.3 Préférences d'un agent 101
4.4 Principe de fusion 103
4.5 Préférences des agents 104
4.6 Intérêt commun des agents 105
4.7 Exemple de conflit : connaissances contradictoires 106
4.8 Exemple de conflit : connaissances différentes 107
4.9 Échelle des conflits 109
4.10 Calcul des rangs 113
4.11 Procédure additive 114
4.12 Effet d'entraînement 117
4.13 Effet de contradiction 118

5.1 Exemple d'une relation de Hamming 123
5.2 «Distribution» des interprétations 125
5.3 Exemple de cubes orthogonaux 127
5.4 Exemple de cubes parallèles 127
5.5 Plus petit cube englobant 128
5.6 Convictions des agents 129
5.7 Mondes les plus préférés 130

6.1 Dispositif de reconnaissance d'activités 134
6.2 Parking surveillé par les agents 135
6.3 Contraintes du domaine 140
6.4 Intersection vide 141
6.5 Convictions et contraintes 142
6.6 Image perçue par l'agent ψ à la date t_1 149
6.7 Image perçue par l'agent φ à la date $t1$ 150
6.8 Exemple d'élaboration d'une croyance commune 152
6.9 Image perçue par l'agent ψ à la date $t2$ 154
6.10 Image perçue par l'agent ψ à la date t_3 155
6.11 Images perçues par les agents ψ et μ à la date t_4 156
6.12 Conjecture non admissible 157

B.1 Déclaration des variables et des couleurs du réseau 169

B.2 Greffier : conjecture modifiée 169
B.3 Greffier : admissibilité globale 170
B.4 Greffier : complétude 171
B.5 Greffier : conformité 172
B.6 Greffier : conjecture disponible 173
B.7 Greffier : accès aux états locaux 174
B.8 Agent α : module de mise à jour de la conjecture 175
B.9 Agent α : module de mise à jour de la base de connaissances . . 176
B.10 Agent α : réception des informations 176
B.11 Agent α : admissibilité 177
B.12 Agent α : conjecture modifiée 178
B.13 Agent α : décision d'engagement 179
B.14 Agent β : module de mise à jour de la conjecture 180
B.15 Agent β : module de mise à jour de la base de connaissances . . 181
B.16 Agent β : réception des informations 181
B.17 Agent β : admissibilité 182
B.18 Agent β : conjecture modifiée 183
B.19 Agent β : décision d'engagement 184
B.20 Mise en œuvre du modèle 185

C.1 Un état possible de l'environnement de Fred 193
C.2 Un état possible de l'autre environnement de Fred 193
C.3 Modèles où Fred est sur le vélo 197
C.4 Fred cherche son vélo 198

D.1 Réseau de Petri interprété 202

Liste des tableaux

1.1 Notions présentées 7

2.1 Notions présentées 35
2.2 Exemple de matrice de coût pour les agents J et K 43

3.1 Variables d'état locales 71
3.2 Variables d'état globales 83
3.3 Récapitulatif des fonctions du greffier 87
3.4 Récapitulatif des variables d'état du modèle 87
3.5 Récapitulatif des postulats du modèle 88
3.6 Récapitulatif des variables de décision d'un agent 88
3.7 Récapitulatif des fonctions d'un agent 88

4.1 Graphe de préférences de $\leq_\psi$ 102

C.1 Table de vérité pour la négation 188
C.2 Table de vérité pour les connecteurs dyadiques 188
C.3 189

Introduction générale

La machine est un outil et non un but pour l'homme, disait Antoine de Saint-Exupéry[2] ; comparable à la charrue du paysan, c'est l'instrument par lequel il découvre le vrai visage de la terre. Échoué au cœur du Sahara, sur un plateau en forme de tronc dont toute évasion était impossible, il éprouvait une joie puérile en arpentant ce territoire que nul n'avait encore souillé : «*J'étais le premier à faire ruisseler, d'une main à l'autre, comme un or précieux, cette poussière de coquillages. Le premier à troubler ce silence.*» Son sentiment d'exaltation fut soudain exacerbé par un caillou noir, objet fort insolite sur ce terrain sédimentaire constitué de trois cents mètres d'épaisseur de coquillages :«*une nappe tendue sous un pommier ne peut recevoir que des pommes, une nappe tendue sous les étoiles ne peut recevoir que des poussières d'astres ; jamais aucun aérolithe n'avait montré avec une telle évidence son origine.*»

De l'euphorie, les opérateurs du Jet Propulsion Laboratory de la Nasa en ont sans aucun doute aussi éprouvée lorsque la sonde Mars Pathfinder s'est posée sur le sol martien, le 4 juillet 1997, jour de la fête de l'Indépendance. Sa mission était d'y acheminer un véhicule d'exploration de la taille d'un jouet d'enfant, le Sojourner microrover (Volpe et al., 1997) ; téléopéré depuis la Terre, les commandes mettent, ce jour-là, près de 11 minutes[3] pour parcourir les 192 millions de kilomètres qui séparent le véhicule du centre de pilotage. Ceci explique pourquoi Sojourner avait été conçu pour fonctionner de manière semi-autonome («supervised autonomous control») : les opérateurs assignent au robot des objectifs, c'est-à-dire des endroits à atteindre, celui-ci devant naviguer de façon autonome entre les roches et les irrégularités du sol.

Les systèmes entièrement autonomes sont encore du domaine de la recherche, même si certaines applications impliquant des robots dans un environnement peu évolutif dont les caractéristiques sont connues *a priori* (e. g. le transport de containers dans le port de Rotterdam (Robert, 1996)) sont effectives. Loin des rêves chimériques de vie artificielle, l'objectif de ces travaux

2. Antoine de Saint-Exupéry, *Terre des Hommes*.
3. la durée de transmission a parfois atteint 20 minutes.

est de doter ces systèmes des capacités nécessaires à l'exploration d'environnements hostiles (e. g. réacteurs nucléaires, milieu sous-marin) et à l'accomplissement de missions trop contraignantes ou trop dangereuses pour l'homme.

Afin d'être totalement autonomes vis-à-vis d'un opérateur humain, les machines doivent mettre en œuvre des systèmes de perception, de décision et d'action performants par rapport un environnement «réaliste» : à partir d'observations incomplètes ou entachées d'incertitudes, elles doivent identifier les objets qui les entourent; reconnaître les événements provoquant des changements d'état de l'environnement ; raisonner sur les conséquences de ces changements d'état et choisir les actions permettant de remplir la mission qui leur a été assignée. Ces étapes (perception, décision, action) constituent la boucle de contrôle du système autonome.

Le projet PERCEPTION (Barrouil et al., 1998; Tessier, 1997; Castel et al., 1996; Perception, 1996) qui a été mené au sein de l'ONERA, a contribué à l'étude et au développement de méthodes permettant d'élaborer et de mettre à jour une représentation du monde réel dans le cadre de cette boucle fermée perception - décision - action. Le projet répondait ainsi aux besoins opérationnels de veille, surveillance, renseignement, reconnaissance, des systèmes autonomes.

L'extension de cette boucle de contrôle à des ensembles intégrant des entités indépendantes dont il faut coordonner les activités est un thème de réflexion qui se situe dans la continuité de ce projet.

Cette thèse aborde la problématique de l'élaboration d'une représentation du monde réel dans un contexte distribué : plusieurs entités ayant sur la situation réelle des points de vue différents doivent *coopérer* pour construire une représentation *commune* de la situation globale. Ces entités, que nous désignerons par le terme générique d'«agents», sont hétérogènes car leurs capteurs sont de natures différentes (e. g. capteur infrarouge, radar) et elles ne perçoivent donc pas nécessairement les mêmes attributs de l'environnement. De plus, à cause de leur perception locale, leurs connaissances sont incertaines, contradictoires, voire erronées ; par conséquent, ces connaissances ne correspondent pas, en général, à l'état réel du monde. C'est pourquoi elles sont qualifiées de *croyances* de l'agent. L'élaboration d'un point de vue commun ne consiste donc pas simplement à juxtaposer des points de vue locaux : les agents doivent résoudre des conflits portant sur leurs croyances afin d'aboutir à un résultat commun. De façon plus générale, nous développons l'idée selon laquelle ces conflits sont à la base de l'élaboration d'une *conjecture commune*.

Dans ce contexte, nous développons notre contribution selon les points suivants :

- nous établissons un état des lieux critique du domaine de la coopération entre agents (chapitre 1 et 2) ;

- nous proposons un modèle générique de coopération pour des systèmes autonomes devant accomplir une tâche collective comme l'élaboration d'une situation observée ou d'un plan partagé (chapitre 3) : il s'agit d'une liste de principes pour la conception d'agents coopératifs et d'un protocole d'interaction spécifiant les échanges d'informations entre agents qui ont pour but de modéliser la problématique de l'élaboration coopérative d'une conjecture commune. Ce protocole a été formalisé par des réseaux de Petri. En amont de la mise en œuvre, cette étude constitue un travail de synthèse et de formalisation *a posteriori* de méthodes de coopération effectives. Contrairement à de nombreux travaux du domaine, nous ne cherchons pas à modéliser la notion de coopération dans une organisation sociale (e. g. colonie de fourmis, états mentaux dans un groupe) : notre objectif est plutôt de structurer[4] la notion de coopération entre systèmes autonomes afin d'ouvrir la voie à de nouvelles méthodes et à d'autres modèles intégrant un plus grand nombre d'applications. Ce travail vise donc à apporter une contribution à l'édification de bases formelles qui semblent faire défaut à la notion de coopération ;

- plus spécifiquement, nous nous intéressons *au contenu* de la conjecture et nous définissons la notion de *conflit* et de *croyance commune* (chapitre 4 et 5) : étant donné un ensemble de croyances éventuellement contradictoires, nous spécifions les propriétés que doit vérifier cette croyance commune. L'objectif est de proposer un cadre formel pour deux types d'applications, la coordination de systèmes autonomes et les systèmes d'aide à la décision : dans le premier cas, nous postulons qu'un groupe d'agents doit respecter une boucle de contrôle semblable à celle que suivrait un agent isolé : tout comme un agent doit mettre à jour ses connaissances sur l'environnement pour construire un plan d'actions adéquat, un groupe d'agents doit collectivement élaborer une croyance commune pour se coordonner efficacement. Dans le second cas, la croyance commune, visualisée sur un support approprié, est un outil d'analyse permettant à un opérateur supervisant un groupe d'agents autonomes (e. g. un essaim de gnoptères[5] assistant un avion de chasse dans une mission de

4. Il ne s'agit donc pas d'un modèle prédictif au sens où on l'entend dans les sciences expérimentales.

5. synonyme d'avion sans pilote ou «drone». Mot créé à l'ONERA à partir de racines grecques, γνῶσις («gnôsis»), la connaissance, γνώμη («gnomè»), la pensée, et πτερόν («pteron»), l'aile.

reconnaissance ou d'attaque) de prendre une décision dans un contexte dynamique et distribué.

De notre point de vue, l'élaboration d'une croyance commune est un cas particulier de la coopération de systèmes autonomes devant accomplir une tâche collective ;

- nous illustrons cette recherche sur un problème concret d'élaboration d'une croyance commune dans le cadre de la reconnaissance d'activités (chapitre 6).

Chapitre 1

Les approches multiagents : un état de l'art

Introduction

La communauté des chercheurs en IAD - Intelligence Artificielle Distribuée, et de façon plus particulière celle qui s'intéresse aux *systèmes multiagents*, connaît une évolution importante comme en atteste le nombre de publications et de conférences internationales consacrées à ce sujet (e. g. ICMAS - International Conference on MultiAgent Systems, International Conference on Autonomous Agents) : il est maintenant possible de dégager de l'expérience acquise au travers de nombreux projets de recherche, un certain nombre de concepts et paradigmes fédérateurs.

L'IAD met en exergue la notion d'interactions, c'est-à-dire à la mise en relation d'«agents» par le biais d'actions réciproques ; le terme «agent» désigne des entités qui peuvent être physiques, comme des gnoptères ou des robots, ou purement informationnels, comme des systèmes experts ou des automates interconnectés à travers un réseau afin d'effectuer des transactions commerciales ou des recherches documentaires.

L'autonomie est une autre caractéristique cardinale du domaine : les agents sont autonomes par rapport à un système de décision centralisé dont l'absence se justifie par diverses contraintes liées à l'application mise en œuvre : limitation de la bande passante des canaux de communications qui impose de traiter une grande quantité des données au niveau de chaque agent ; discrétion des communications ; recherche de performances par la parallélisation des traitements ; modélisation «naturelle» de systèmes géographiquement répartis.

De manière générale, les agents réalisent des tâches spécifiques avec une

connaissance limitée de leur environnement qui est évolutif, incertain, voire imprédictible, et où la complexité des situations rencontrées exige de prendre des décisions locales : c'est le cas de robots évoluant dans un milieu hostile ou d'ordinateurs programmés pour réagir aux fluctuations d'un marché. En accomplissant leur tâche à partir d'un point de vue local, les agents interagissent pour satisfaire leurs propres buts (e. g. vendre ou acheter au meilleur prix), ou pour réaliser un but spécifique (e. g. établir un diagnostic de panne).

Selon les applications, les agents ont donc un *but commun* ou des *buts concurrents* : dans le premier cas, nous parlerons de *coopération* et dans le second, de *négociation*. Cette distinction nous sert de grille de lecture du domaine :

- nous présentons les différents thèmes de recherches multiagents ;
- nous développons leurs caractéristiques génériques c'est-à-dire la notion d'agent et la notion d'interaction ;
- nous faisons la synthèse des différentes méthodes d'interaction (i. e. coopération *versus* négociation, cf. chapitre 2).

1.1 Thèmes de recherches

Chronologiquement, les systèmes multiagents sont une évolution de systèmes experts comme HEARSAY-II (Erman et al., 1980; Lesser and Corkill, 1981; Corkill et al., 1982) vers les systèmes distribués, dans le but d'acquérir des caractéristiques comparables.

La rapidité d'exécution par la parallélisation des tâches et l'optimisation de l'accès aux ressources communes sont des propriétés visées. D'un point de vue qualitatif, l'adaptabilité aux applications naturellement réparties, la modularité des fonctions du système et la robustesse aux défaillances sont les avantages les plus recherchés.

Cette «filiation» entre Intelligence Artificielle (IA) et systèmes distribués explique la diversité des thèmes de recherches qui tendent soit vers des problématiques inspirées par les sciences cognitives, soit vers des problématiques de conception et de programmation des systèmes. Nous avons ainsi répertorié six axes principaux :

- Modèles cognitifs et simulation d'organisations sociales ;
- Résolution distribuée de problèmes ;
- Programmation Orientée Agents ;

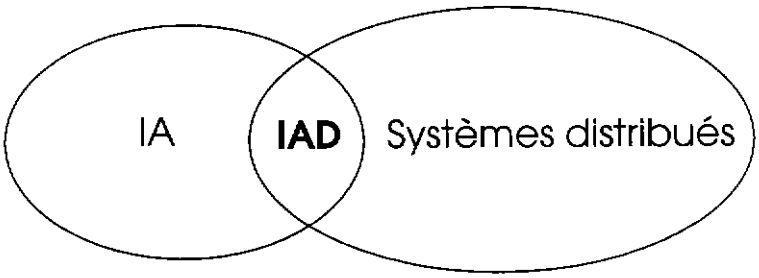


FIG. 1.1 – *Intelligence Artificielle Distribuée*

- Protocoles de négociation automatique ;
- Coordination de systèmes autonomes et robotiques ;
- «Collecticiels» et Interfaces Homme - Machine ;

	Agent	Interaction
Modèles et simulations	cognitif/réactif	états mentaux de groupe/auto-organisation
Programmation orientée agents	processus communicants	organisation dynamique
Négociation automatique	«surrogates»	offre/demande
Coordination	systèmes autonomes	planification distribuée
IHM, collecticiels	hommes/machines	interfaces de communication
Résolution distribuée	tâches de résolution	échange de résultats

TAB. 1.1 – *Notions présentées*

Le tableau 1.1 présente les notions qui sont abordées dans ce chapitre.

1.2 Modèles cognitifs et simulation d'organisations sociales

1.2.1 Approches cognitives

Présentation des travaux

Certains travaux (Cohen and Levesque, 1990; Wooldridge, 1994; Singh, 1994; Wooldridge and Jennings, 1994; Wooldridge and Jennings, 1995; Jen-

nings, 1992; Seguin, 1992; Grosz and Kraus, 1993; Werner, 1989), marchant sur les brisées des sciences cognitives, ont pour objet de formaliser le raisonnement humain et, de façon plus générale, leurs attitudes sociales. L'un de leurs objectifs est d'élaborer des modèles de comportements pouvant servir de spécifications à la conception de systèmes coopératifs. M. Wooldridge et N. R. Jennings (Wooldridge and Jennings, 1994) écrivent ainsi que : «la résolution coopérative de problèmes est une caractéristique commune et importante des sociétés humaines, et il semble de plus en plus évident qu'il en sera de même dans les futurs systèmes informatiques.»

Le point de départ de ces modèles est la planification, la représentation de l'action par des graphes de changements d'état et les formalismes comme STRIPS (Seguin, 1992; Chapman, 1987; Garcia, 1993) : les agents ont des buts à accomplir, c'est-à-dire un état final à atteindre, en appliquant des opérateurs de transformation d'état (i. e. des actions) à l'état initial. La succession des opérateurs menant de l'état initial à l'état final est un plan.

Les modèles cognitifs - on emploie aussi le terme «BDI» pour *Beliefs, Desires and Intentions* - postulent que les actions d'un agent, tout comme celles d'un être humain, doivent être dictées par des représentations abstraites du monde, incluant la représentation de ses propres capacités, ses buts ainsi que ceux des autres agents : ces représentations sont les *états mentaux* («intentional states») :

- une *croyance* est une base de faits incertains sur le monde : c'est une représentation du monde perçue comme vraie par l'agent, et qu'il met à jour ;
- un *désir* représente une action potentielle ;
- l'*intention* est la capacité d'un agent à contrôler ses actions ; elle décrit tout ce qui permet de passer à l'acte et elle engendre le processus de planification en raisonnant sur les croyances. L'intention est une attitude mentale complexe, fondée sur la notion de but (parmi les états futurs possibles, l'agent effectue un choix) et sur celle d'engagement, c'est-à-dire la persistance des buts (l'engagement d'un agent à réaliser un but choisi).

Voici quelques éléments tirés des travaux de H. J. Levesque et P. R. Cohen (Cohen and Levesque, 1990; Levesque et al., 1990).

L'engagement d'un agent x à réaliser un but p signifie qu'il croit que p est faux, mais qu'il veut que p soit vrai, et qu'il continue de vouloir que p doive être vrai tant qu'il croit que ses justifications q sont vraies ou que p ne sera

jamais vrai : l'engagement est donc défini comme un but persistant.

Pour un agent x, avoir l'intention de faire une action se définit par : x a l'intention de faire une action s'il a comme but persistant de faire cette action et s'il l'a faite en croyant qu'il était en train de la faire ; l'intention est par conséquent une action délibérée.

H. J. Levesque et P. R. Cohen postulent qu'il existe aussi des états mentaux au niveau social bâtis sur le même schéma que les états mentaux d'un agent. Ainsi, tout comme ils définissent un but persistant et une intention pour un agent, ils définissent un but persistant collectif pour un groupe d'agents. À cette fin, ils modifient la définition d'un but persistant pour un agent car le groupe n'est pas la simple union de tous les agents : cette définition de la notion de but «mou» (« *Weak Goal* ») recouvre trois cas distincts :

- l'agent x a comme but que p soit «éventuellement» vrai,
- ou x pense que p est déjà vrai et il veut que ce fait soit mutuellement admis,
- ou il pense que p ne sera jamais vrai et il a pour but de faire en sorte que cela soit mutuellement admis.

La notion de but a été «affaiblie» parce qu'un agent ne peut pas faire l'hypothèse que tous les autres agents continuent à poursuivre le même but que lui ; il se peut qu'un agent ait découvert que p était inatteignable et soit en train d'en informer chaque agent.

Leur définition de l'état d'un groupe d'agents poursuivant un but collectivement est la suivante : tous les membres du groupe ont pour but p si le groupe croit que p n'est pas vrai, et il a pour but p tant que le groupe ne croit pas :

- que p est vrai ;
- ou que p ne sera «jamais» vrai ;
- ou que les justifications du but sont fausses.

Les travaux de H. J. Levesque et P. R. Cohen sont à la base de nombreux développements. Ainsi, B. Grosz et S. Kraus proposent un modèle de collaboration pour la planification (Grosz and Kraus, 1993) : de leur point de vue, il ne suffit pas de spécifier les états mentaux des agents, il faut aussi définir des opérateurs permettant de raisonner sur les états mentaux.

À cet effet, ils définissent deux opérateurs dans leur langage, *Int.to* c'est-à-dire l'intention de faire quelque chose et *Int.th* c'est-à-dire l'intention que quelque chose soit réalisé.

Pour illustrer la différence entre leurs deux opérateurs, ils donnent l'exemple suivant : deux agents, Jan et Sandy se sont mis d'accord pour préparer un repas ensemble. Leur plan partagé pour préparer ce repas inclut l'intention de (*Int.to*) Jan de préparer l'apéritif, l'intention de (*Int.to*) Sandy de faire les courses ainsi que leur plan partagé de préparer un dessert. Le plan partagé pour préparer le repas comprend aussi l'intention de Sandy que (*Int.th*) Jan puisse s'occuper de l'apéritif et l'intention de Jan que (*Int.th*) Sandy puisse faire les courses.

Critiques des approches cognitives

De façon générale, les modèles cognitifs et les définitions des états mentaux ne font pas l'objet d'un consensus parmi les chercheurs du domaine. L'intérêt même de ces travaux est critiqué :

- les modèles sont formalisés avec un langage logique utilisant des modalités «auto-épistémiques» (e. g. l'agent *croit qu'il croit p*). D'après E. Grégoire (Grégoire, 1988), ils peuvent difficilement être mis en œuvre : «on doit plutôt les considérer comme des modèles de compétences vers lesquels devraient converger des modèles effectifs.»

  L'un des inconvénients les plus importants de ces formalismes réside dans leur sémantique semblable à celle des mondes possibles : chaque monde est un état possible de la réalité et on passe d'un monde à l'autre par des actions élémentaires. Il n'est donc pas possible de rendre compte d'événements simultanés.

  Par ailleurs, à partir de chaque monde, chaque agent détermine ses croyances grâce à sa propre relation d'accessibilité vers les autres mondes. Ainsi, l'assertion «l'agent x croit p» est vrai pour l'agent x, en *un point donné* d'une séquence d'actions, si p est vraie dans tous les mondes avec lesquels il est en relation à partir de ce point. La valeur de vérité d'une formule qui peut aussi contenir des modalités temporelles s'évalue quant à elle par rapport à une *séquence d'actions données*. On en déduit que pour trouver un modèle d'une formule pour un agent, il faut parmi toutes les séquences possibles trouver la séquence et le point de cette séquence qui la rendent vraie. Comme cette procédure doit être répétée pour chaque agent du groupe, on conçoit que ces modèles n'aient pas de finalité opératoire du fait de l'explosion combinatoire des séquences d'actions pos-

sibles.

Remarque : de plus, les problèmes classiques de représentation des actions et de leurs conséquences devraient être traités pour tous les agents - quelles sont les pré-conditions à une action («qualification problem») ? quelles sont les propriétés inchangées dans un monde («frame problem») ? quels sont les nouveaux faits («ramification problem») ?

- les modèles ne sont pas «réalistes» : les agents sont logiquement omniscients c'est-à-dire que l'ensemble des croyances d'un agent contient toutes les formules valides pouvant être déduites. Ils n'ont donc pas comme les êtres humains des capacités d'inférence limitées. Bien que des travaux aient été menés pour corriger cet inconvénient (Fagin and Halpern, 1985), les modèles d'action des agents restent déterministes, c'est-à-dire qu'étant donné un état du monde parmi l'ensemble des mondes possibles, une action mène à un et un seul nouveau monde. Par ailleurs ce monde est statique, c'est-à-dire que seules les actions des agents peuvent modifier l'état de ce monde (Seguin, 1992).

- les modèles se fondent entièrement sur le postulat philosophique de l'intention (Bratman, 1992) et ne reposent sur aucune donnée expérimentale. À ce sujet, une étude de B. Pavard (Bressolle et al., 1996) dans le domaine du contrôle aérien montre que les interactions dans un groupe sont un phénomène complexe et difficile à appréhender. Il a pu ainsi observer que les interactions entre des contrôleurs chargés de la régulation du trafic aérien ne respectent pratiquement jamais le cadre réglementaire extrêmement rigide ; les communications entre les pilotes et les contrôleurs ou entre les contrôleurs (contrôleur organique et contrôleur radar) ne suivent pas le protocole prédéfini et les actes de langages sont pluri-adressés et pluri-évalués. Il y a donc un fonctionnement implicite du groupe qui s'oppose au fonctionne explicite défini par le règlement, mais qui rend la coopération possible. On peut ainsi observer une superposition de modes de contrôle : le «feedback» est le mode standard, le pilote confirme qu'il a bien reçu le message du contrôleur ; mais, il existe d'autres modes informels comme le contrôle parallèle (le contrôleur organique vérifie les consignes du contrôleur radar), la délégation de contrôle (le contrôleur demande au pilote de le rappeler plus tard), ou le contrôle distribué (le contrôleur demande au pilote de le rappeler lorsqu'il sera arrivé au niveau d'une balise). Cette étude a été réalisée pour comprendre comment concevoir les collecticiels («groupware»), dont le fonctionnement ne doit pas empêcher les interactions informelles et rigidifier le protocole de communication au risque d'entraver la bonne marche du groupe.

- la justification des modèles cognitifs comme spécification de systèmes co-

opératifs est fondée sur un contresens. En effet, ces modèles essaient de répondre à la question suivante : *pourquoi* les agents cognitifs interagissent-ils ? Or, si les comportements sociaux chez l'homme se justifient par le jeu des connaissances «épistémiques» (i. e. ses croyances, ses intentions), les interactions entre machines sont justifiées par la nécessité de réaliser une tâche collective assignée par un concepteur. On ne se pose donc pas la question de savoir *pourquoi* les agents coopèrent : les réponses à cette question ne sont d'aucune utilité tant qu'on ne met pas au point des machines avec des besoins primaires comme trouver des sources d'énergie dans leur environnement, survivre et se multiplier.

1.2.2 Approches réactives

Parallèlement au courant cognitif, sont apparus d'autres modèles d'agents qualifiés de réactifs (Rosenschein and Kaelbling, 1995; Brooks, 1991; Agre and Chapman, 1987; Brooks, 1986; Dario et al., 1991; Zeghal and Ferber, 1993; Stirling, 1993; Steels, 1990). Ces travaux sont dus aux critiques portées essentiellement aux thèses cognitives et contestent la nécessité de représentations abstraites (i. e. la manipulation de symboles) pour simuler un comportement intelligent (Brooks, 1991). Ainsi, nos comportements sont souvent des réactions immédiates et bien déterminées à des situations inattendues. Le problème du choix d'une séquence d'actions adéquates n'est donc pas une question centrale ; D'après P. E. Agre (Agre and Chapman, 1987) : «lorsque le caillou sur lequel vous marchez cède, il n'est plus temps de chercher un plan ou de démontrer un théorème.»

Ces critiques aboutissent à l'idée que l'intelligence est «située» dans le monde et non pas concentrée dans un système découplé du monde comme un système expert ou un démonstrateur de théorèmes : un comportement intelligent est le résultat de l'interaction d'un agent avec son environnement et non pas une propriété intrinsèque à l'agent. Il s'ensuit que les agents réactifs ont des structures internes peu abstraites et des comportements de type stimuli-réponses.

Pour démontrer la justesse de ces thèses, plusieurs expérimentations ont été menées :

- R. Brooks a mis au point des robots fondés sur une architecture de «subsumption» (Brooks, 1986). Contrairement à l'architecture classique (Perception, 1996; Tessier, 1997) qui enchaîne séquentiellement les fonctions de perception, planification et exécution, l'architecture de «subsumption» est un empilement de fonctions élémentaires comme «éviter un obstacle» ou «explorer». Celles-ci sont câblées par des composants

physiques, entre les capteurs et les effecteurs. Ainsi, lorsqu'un capteur détecte un obstacle, la commande des moteurs qui effectuent la fonction «explorer» est inhibée par les signaux de la fonction «éviter un obstacle». L'intérêt d'une cette architecture est de pouvoir, de façon incrémentale, donner de nouvelles aptitudes élémentaires au robot qui finalement, du point de vue de l'observateur, exhibe un comportement intelligent.

- De même, P. E. Agre et D. Chapman ont développé un programme appelé PENGI (Agre and Chapman, 1987), qui joue de façon «satisfaisante» à un jeu vidéo commercial : un personnage se déplace dans un labyrinthe composé de blocs qu'il peut pousser pour fuir ses ennemis. Le programme enchaîne des «routines», c'est-à-dire des séquences d'actions pré-définies, en fonction de situations prototypiques. Ainsi, les deux règles suivantes,

  (R1) si poursuivi alors fuir ;

  (R2) si obstacle alors pousser ;

  produisent le comportement itératif suivant :

  (I) fuir jusqu'à heurter un obstacle, pousser l'obstacle et répéter fuir ;

  Les règles ne représentent pas la procédure itérative : la boucle *émergeant* de **R1** et **R2** résulte de l'interaction de celles-ci avec la situation rencontrée par le héros du jeu.

Les résultats des simulations concernant l'émergence de comportements non pré-définis et adaptés de la part d'un agent, ont conduit de nombreux chercheurs à l'étude de l'*auto-organisation* d'un groupe d'agents. Ce terme est issu des théories relatives aux systèmes complexes (Prigogine and Nicolis, 1977) et se réfère à l'apparition «spontanée» d'une structure par l'interaction de ses éléments constituants : l'auto-organisation semble être un concept clé pour la compréhension de nombreux phénomènes en physique, en chimie et dans les sciences de la vie. Par exemple, une termitière est un effet de bord de l'activité des termites : aucun d'entre eux n'a pour «but» de construire la termitière ; elle est le résultat d'une «coopération» non intentionnelle qui s'est organisée par le biais de gradients de températures et dont la principale conséquence est l'amélioration de la survie des termites (Theraulaz and Bonabeau, 1996).

Plusieurs chercheurs en Intelligence Artificielle tentent de simuler des phénomènes d'auto-organisation :

- W. C. Stirling (Stirling, 1993) s'est intéressé au problème des «producteurs - consommateurs - transporteurs» (Durfee and Montgomery, 1991) où deux équipes de deux robots partagent un espace commun. L'un des coéquipiers est chargé de transporter un jeton du site de production à

un site de stockage, et l'autre, du site de stockage au site de consommation. Le problème de coordination consiste, pour les deux coéquipiers, à converger vers le même site de stockage. L'incertitude liée au choix du site décroît au cours du temps car les agents observent tous la même situation ; leurs hypothèses sont donc corroborées par une boucle de contrôle sur l'environnement. Cet environnement partagé joue, de façon subtile, le rôle d'une structure de communication.

- L. Steels (Steels, 1990) a simulé un groupe de robots explorant une zone inconnue : leur but est de trouver des objets (e. g. du minerai) et de les ramener à un véhicule collecteur. Les comportements élémentaires sont couplés suivant l'architecture de «subsumption» et les robots communiquent en émettant autour d'eux un champ de potentiel (e. g. un son). L. Steels évalue les avantages et les inconvénients des différents comportements par rapport au temps théorique que mettraient les robots pour rassembler tous les objets s'ils connaissaient leurs emplacements (i. e. temps pour aller chercher directement l'objet puis revenir au véhicule collecteur). Signalons que l'idée des champs de potentiels a été également utilisée pour l'évitement de collision et la coordination du trafic aérien (Zeghal and Ferber, 1993).

- Le projet MANTA (Drogoul and Ferber, 1992) a permis la simulation de l'organisation sociale d'une colonie de fourmis. L'approche réactive est séduisante car elle permet à des éthologues de coder simplement leurs connaissances (i. e. les comportements des insectes), d'étudier l'influence de paramètres environnementaux (e. g. la luminosité, l'humidité) et de tester certaines de leurs hypothèses : en particulier, on s'est intéressé au lien entre le comportement local et spécifique d'un agent et le comportement global de la société d'agents réalisant les fonctions attendues ; c'est ce qu'A. Drogoul appelle «le passage micro-macro» : comment des agents qui n'ont pas de représentation des autres et qui ne sont sensibles qu'à leurs stimuli laissés dans l'environnement peuvent-ils produire une organisation sociale cohérente, c'est-à-dire assurant sa survie et sa reproduction ?

Remarques :

- les outils de simulation du projet MANTA ont également été utilisés pour la validation de l'«*éco-résolution*» qui est un paradigme de résolution de problèmes (cf. paragraphe 1.3) ;
- les approches réactives et cognitives ne sont pas antinomiques ; dans les premières, les agents interagissent implicitement par l'intermédiaire d'un environnement partagé alors que dans les secondes, la coordination s'opère grâce à l'envoi de messages.

1.3 Résolution distribuée de problèmes

Les problèmes abordés par les systèmes multiagents ne sont pas intrinsèquement distribués : ils peuvent être résolus par une approche centralisée. Les expérimentations menées cherchent précisément à évaluer l'intérêt d'une approche distribuée en termes de temps de calcul ou de «facilité» de programmation.

On peut distinguer trois classes de problèmes traités par une approche multiagent :

- L'interprétation distribuée : le domaine est distribué et les agents peuvent avoir tous la même expertise, mais aucun n'a une vision globale du problème à résoudre. Au contraire, elle est partielle et éventuellement erronée ; le but est d'intégrer et d'analyser des données distribuées, généralement de bas niveau, comme par exemple les signaux provenant de capteurs, pour générer un modèle sémantique d'un plus haut niveau d'abstraction. Pour ce type de problèmes, une architecture centralisée a l'inconvénient d'être peu robuste aux pannes et risque de constituer un goulot d'étranglement pour les communications, car la bande passante pour récupérer des données brutes doit être très importante. En revanche, une architecture répartie exploite le parallélisme de traitement des agents qui n'échangent entre eux qu'un plus petit nombre de données abstraites. DVMT (Durfee et al., 1987) pour le pistage de véhicules est le système précurseur de ce type d'applications

  Remarque : ces applications doivent être rapprochées des problèmes de fusion multicapteur qui tendent aussi à adopter une architecture distribuée (Luo and Kay, 1989; Waltz and Buede, 1986; Berge-Cherfaoui and Vachon, 1991).

- La coopération de spécialistes : l'expertise globale est distribuée, aucun agent n'a la capacité de résoudre le problème tout seul ; en revanche, le domaine est non distribué et les systèmes de cette catégorie sont généralement composés d'agents purement informationnels (e. g. bases de données, systèmes experts). Dans SMIEG (Bouzouane, 1995), chaque agent est un système expert, l'expert en sinistres, l'expert financier et l'expert juridique. Ils examinent le problème, une déclaration de sinistre, depuis leur point de vue, et réunissent leurs résultats pour aboutir à une expertise commune.

La réalisation de systèmes informatiques dans lesquels on peut «facilement» intégrer des nouvelles fonctionnalités et faire évoluer les anciennes, est un argument en faveur des approches multiagents. L'objectif de la

coopération d'experts est de faire travailler ensemble des programmes hétérogènes provenant parfois de différents concepteurs. En encapsulant ces programmes dans des entités génériques, c'est-à-dire des agents, indépendantes du concepteur initial et de l'environnement matériel, on peut réaliser d'importantes économies dans la conception (e. g. réutilisation de code déjà existant et provenant de différentes sources) et donner plus de modularité (i. e. échange et maintenance des fonctionnalités accrus). Cette méthode de conception a déjà été appliquée à un système de gestion de la distribution de l'énergie électrique avec le projet ARCHON (Wittig et al., 1994).

- l'éco-résolution est l'application des principes des approches réactives à la résolution de problèmes. Elle se démarque des méthodes de résolution distribuée où les agents ont un but commun - résoudre un problème donné - en considérant que les agents ont des buts concurrents qu'ils cherchent chacun à satisfaire.

Ainsi, MASC (Ghedira, 1993), développé à l'ONERA-CERT traite le problème de l'allocation de tâches à des ressources partageables d'un point de vue multiagent : MASC associe des principes d'éco-résolution (Ferber, 1995; Ferber and Jacopin, 1990) et de recuit simulé, les tâches et les ressources étant considérées comme des agents. Conformément à ces principes, ces agents tentent de satisfaire des objectifs propres : une tâche «désire» être placée au mieux et une ressource «être utilisée» tout en minimisant son taux d'occupation. Les résultats théoriques prouvent que cette méthode peut converger asymptotiquement vers une solution optimale, et les divers tests effectués montrent une grande rapidité d'exécution par rapport à d'autres méthodes (recherche opérationnelle ou programmation logique avec contraintes).

Les «éco-agents» reposent sur le modèle d'«acteurs» (Agha, 1986) (i. e. des processus concurrents communiquant par l'envoi de messages asynchrones) ayant des comportements extrêmement simples. Les interactions entre agents permettent de faire «émerger» la solution du problème, la principale difficulté de l'éco-résolution résidant dans le codage des «éco-agents» - comment décomposer un problème en agents dont le comportement mène à une solution ?

Remarque : l'éco-résolution peut aussi être comparée aux méthodes classiques (Barr and Feigenbaum, 1982) (i. e. «centralisée») de recherche d'une solution dans un graphe d'état où trouver une solution c'est aller d'un état initial à un état final souhaité : la solution existe donc *a priori*

et on cherche à trouver un chemin optimal (i. e. une séquence d'opérateurs de changements d'état) par l'utilisation des heuristiques adéquates. Dans le cas de l'éco-résolution, trouver une solution c'est atteindre un état stable par consensus : l'une des ambitions de cette approche est de maîtriser l'explosion combinatoire inhérente à certains problèmes d'Intelligence Artificielle (cf. (Chapman, 1987; Korf, 1987; Ephrati and Rosenschein, 1994) pour les problèmes de planification).

1.4 Programmation Orientée Agents

Du point de vue de la résolution de problèmes à forte combinatoire, le paradigme de programmation multiagent (Shoham, 1993) est en compétition avec les méthodes de résolution distribuée (Durfee et al., 1987; Lesser and Corkill, 1981) («DPS - Distributed Problem Solving») : on ne peut cependant pas les opposer car les systèmes multiagents découlent de paradigmes de résolution distribuée innovateurs ayant progressivement glissé sur le terrain de l'Intelligence Artificielle.

Le «tableau noir» est l'un de ces paradigmes : il s'agit d'une structure de données hiérarchisée et partagée par plusieurs processus (Erman et al., 1980; Lesser and Corkill, 1981). Il a été utilisé pour la première fois par HEARSAY-II qui est un système de reconnaissance de la parole : le premier niveau du tableau est celui des signaux acoustiques échantillonnés que les processus idoines doivent transformer en portion de signaux, syllabes, mots, séquences de mots, phrases, jusqu'au dernier niveau qui interprète les phrases énoncées par l'opérateur pour exécuter une action (e. g. une recherche documentaire).

Le système procède donc par abstraction des données ; les processus qui sont appelés des «sources de connaissances» font passer les données d'un niveau d'abstraction n à un niveau $n+1$ où elles deviennent des «hypothèses», c'est-à-dire des tentatives de solutions partielles. Les sources de connaissances se déclenchent de façon opportuniste (i. e. dès que toutes les entrées d'une source sont instanciées par les résultats d'une autre source) et concurrentielle (i. e. en parallèle et éventuellement sur les mêmes données). Certaines sources ont pour fonction d'associer les hypothèses entre elles et donc de produire des hypothèses plus «globales» (e. g. un syntagme à partir de mots).

On peut comparer ce type de mécanisme à la résolution d'un puzzle : les sources de connaissances essaient d'assembler les différentes pièces en îlots de pièces formant à leur tour des ensembles plus importants de pièces... Ainsi le nombre d'îlots isolés décroît rapidement et le système converge de plus en plus vite vers le puzzle complet : cette démarche a été appelée «island driven» (Les-

ser and Corkill, 1981). On constate par ailleurs que la boucle de résolution est itérative : elle consiste à générer un ensemble d'hypothèses cohérentes, par un cycle intégration d'hypothèses/tests de cohérence, satisfaisant la spécification du problème.

DVMT (Durfee et al., 1987; Durfee and Lesser, 1987) s'inspire des principes de HEARSAY-II et met en œuvre un réseau de tableaux noirs appelés «nœuds». V. Lesser et D. Corkill ont expérimenté cette architecture à partir des données recueillies par des capteurs acoustiques couvrant des zones géographiques adjacentes. L'objectif du système est de reconstituer les trajectoires de véhicules ayant traversé ces zones : chaque nœud est responsable d'une zone géographique bien délimitée à l'intérieur de laquelle il élabore des fragments de trajectoires à partir de ses signaux acoustiques. La structure du système est plate et correspond au maillage géographique. Les nœuds échangent des informations concernant les portions de trajectoires qu'ils élaborent. Différentes stratégies de communication ont été testées dans le but d'améliorer les performances du système (en termes de temps de calcul) : en particulier, les nœuds peuvent décider de traiter de façon prioritaire les signaux provenant de zones limitrophes ; un nœud ne recevant pas de signaux peut choisir de fusionner les trajectoires partielles élaborées par d'autres nœuds. La coopération entre nœuds est induite par des structures d'interaction prédéfinies par un concepteur pour résoudre un problème particulier.

Les résultats de DVMT ont permis d'envisager des architectures dans lesquelles les agents, c'est-à-dire les modules de traitements, ont la capacité de coordonner eux-mêmes leurs connaissances, leurs compétences et leurs tâches pour conjointement résoudre un problème donné. Cela suppose que les agents aient des connaissances relatives aux capacités et aux rôles des autres agents (i. e. des modèles d'accointances) et éventuellement sur les caractéristiques de leur environnement : les agents ne seraient pas capables d'organiser de façon dynamique la résolution sans raisonner sur ces connaissances. Par conséquent, la notion d'agent est plus abstraite que celle de «processus» ou d'«objet» (au sens des langages à objets) puisqu'elle rend compte de l'hétérogénéité des modules de traitement (notamment en ce qui concerne leurs compétences respectives) ou de leur situation dans un environnement complexe.

La figure 1.2 montre l'évolution de la notion de programme, qui est une séquence d'actions atomiques (e. g. lire ou écrire un registre) à celle d'un agent, qui est un processus communicant capable de raisonner sur les autres agents de son environnement.

Par ailleurs, la programmation orientée agents est un paradigme de l'In-

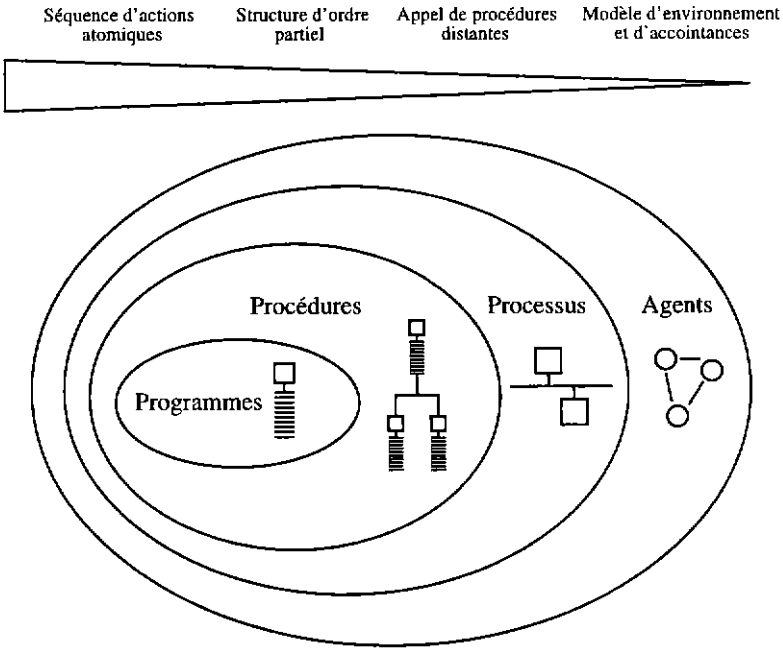


FIG. 1.2 – *Évolution de la notion de programme*

telligence Artificielle Distribué (cf. figure 1.3) : il participe à l'évolution des systèmes distribués et répond à un besoin d'outils de programmation d'interactions d'un haut niveau d'abstraction : ainsi, le modèle maître-esclave caractérise les échanges d'informations et de signaux de synchronisation entre processus parallèles ; le modèle lecteurs-écrivains permet de concevoir des processus travaillant avec une zone de données partagée ; le modèle d'acteurs (Agha, 1986) spécifie le fonctionnement de processus concurrents par échanges de messages ; les réseaux contractuels (Smith and Davis, 1981) sont une forme élémentaire de négociation pour la délégation de tâches entre agents.
Remarque : il existe de nombreuses plates-formes ou langages dédiés au développement de systèmes multiagents comme par exemple MADKIT (Gutknecht and Ferber, 1998) ou LALO (Gauvin, 1995).

1.5 Les protocoles de négociation automatiques

Les protocoles de négociation (Rosenschein and Zlotkin, 1994; Zlotkin and Rosenschein, 1993) mettent en œuvre des agents antagonistes, c'est-à-dire ayant des buts concurrents et défendant des intérêts divergents : les agents sont des automates se substituant à des personnes ou des organisations sociales concurrentes («surrogates») et soumises aux lois d'un marché (Rosenschein, 1993). Ainsi, les agents sont des machines effectuant automatiquement

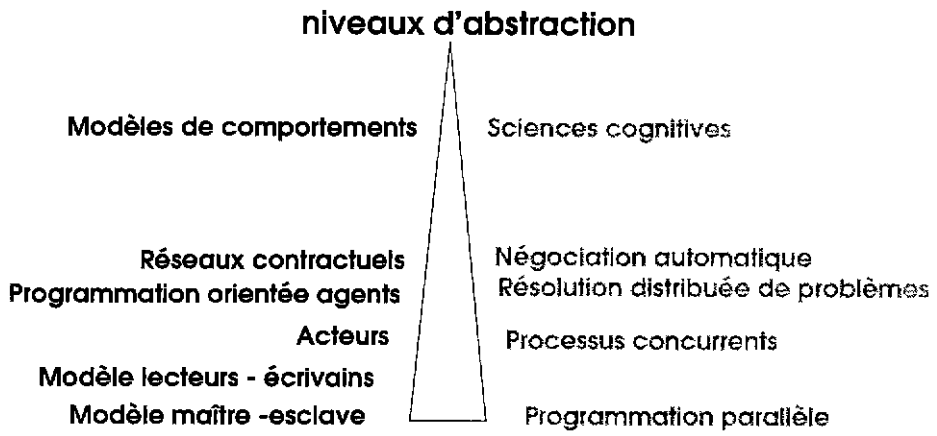


FIG. 1.3 – *Place de la programmation orientée agents*

des transactions financières, ou des relais permettant d'interconnecter différents réseaux de télécommunications en fonction de leurs besoins : on envisage (Rosenschein and Zlotkin, 1994), dans un proche avenir, des téléphones se connectant automatiquement aux opérateurs offrant les meilleurs prix à un instant donné ou des programmes recherchant les meilleures offres de services (e. g. réservation de voyages, location de voitures etc.) Ces applications sont actuellement à l'étude ; elles cherchent à mettre en œuvre des mécanismes de négociation automatique visant à maximiser la satisfaction locale des agents.

Remarque : nous présentons les méthodes de négociation au chapitre 2.

1.6 Coordination de systèmes autonomes et robotiques

Dans les applications de ce domaine, les agents évoluent dans un environnement physique et l'objectif est de coordonner un ensemble d'actions interagissant entre elles.

La coordination par planification, pour un seul agent, est en soit un problème difficile à résoudre, comme l'ont démontré D. Chapman et K. E. Korf (Korf, 1987; Chapman, 1987) : si b est le nombre moyen de nouveaux états qui peuvent être générés depuis un état donné en appliquant un seul opérateur, et d la profondeur du problème (i. e. la longueur du chemin optimal allant de l'état initial à l'état final), la complexité du problème de planification en termes de temps est de l'ordre de b^d. Mais cette complexité se trouve fortement augmentée si on ajoute d'autres agents. En effet, dans une structure distribuée,

un planificateur centralisé élabore tous les plans d'actions locaux et doit considérer l'assignation de chaque opérateur à chacun des n agents. Dans de telles conditions, coordonner de façon cohérente et efficace des agents, c'est-à-dire choisir l'action adéquate à accomplir à un moment donné, revient à évaluer la situation de plusieurs points de vue, et la complexité est de l'ordre de $(n \times b)^d$; c'est pourquoi certains travaux portent sur la réduction de la complexité des problèmes de coordination en répartissant la tâche de planification sur l'ensemble des agents (Ephrati and Rosenschein, 1994; Georgeff, 1983).

Ainsi, K. Azarm and G. Schmidt (Azarm and Schmidt, 1997) ont développé une méthode de planification distribuée pour la résolution de conflits entre robots mobiles : il s'agit d'un problème de partage d'une ressource (i. e. la zone d'évolution) et une méthode classique consiste à affecter à chaque robot un niveau de priorité et à calculer les différentes trajectoires les unes après les autres. Le robot de plus haut niveau de priorité planifie sa trajectoire en ne tenant compte que des obstacles fixes. Les robots de plus basses priorités doivent calculer leurs trajectoires en tenant compte des mouvements des robots plus prioritaires qu'eux. Cette méthode peut conduire à des interblocages lorsque des robots se rencontrent dans des impasses : K. Azarm and G. Schmidt améliorent cette méthode en proposant un algorithme d'assignation dynamique des niveaux de priorité.

D'autres méthodes dites «réactives» ont été proposées : ainsi, on a envisagé d'appliquer une méthode fondée sur la notion de champs de forces pour l'évitement de collision entre avions (Zeghal and Ferber, 1993).

Parmi les applications de coordination, on peut par exemple citer l'organisation de trafic aérien (Cammarata et al., 1983), les procédures d'évitement (Zeghal and Ferber, 1993), la gestion de chaînes de production (Ouzrout et al., 1996), et l'organisation du trafic de robots de manutention (Robert, 1996).

Remarque : les problèmes de planification et de coordination en général se distinguent de la résolution de problèmes : les agents évoluent dans un environnement dynamique et doivent réagir à ces changements. Ainsi, dans MARTHA (Robert, 1996), des robots partagent la même aire d'évolution : la boucle de coordination de leurs déplacements est entrelacée avec la boucle d'exécution.

1.7 «Collecticiels» et Interfaces Homme - Machine

Les interfaces Homme-Machine peuvent être conçues comme un système distribué composé de deux agents coopérants. Des recherches sont menées dans

ce sens pour établir un véritable dialogue (Chaudron et al., 1995; Chaudron et al., 1997b; Chaudron et al., 1998), sans ambiguïtés. En effet, l'échange d'information peut conduire dans certaines situations à de graves accidents : ce peut être le cas, par exemple, d'une mauvaise interprétation des instruments de vol par un pilote.

Plus prosaïquement, face à des machines de plus en plus complexes, l'être humain dont la capacité de réaction à des situations imprévues et de stress est limitée, a besoin d'outils d'analyse contextuelle, c'est-à-dire capables de donner un sens à l'état d'un système - Que fait la machine? Dans quel but? Son comportement est-il cohérent avec mes objectifs? Ceci, aucun environnement technique aussi complet soit-il, ne peut l'établir simplement (Wooldridge and Jennings, 1995) : prenons, par exemple, le cas de la mise au point d'un programme et de la correction des bogues ; sans l'assistance (bien que souvent rudimentaire) du compilateur qui émet des messages d'erreurs, il est difficile de trouver les erreurs de syntaxe dans le code.

Un autre exemple est celui de la surveillance d'un réseau de transport d'électricité : la tâche de surveillance et de diagnostic incombe normalement à un opérateur qui doit manuellement transmettre les données d'un système expert à un autre. Dans le projet ARCHON (Wittig et al., 1994), cette fonction a été automatisée par des systèmes communiquant de façon autonome, ce qui a permis d'alléger la «charge cognitive» de l'opérateur, le laissant à des activités plus complexes, pour lesquelles il est mieux adapté.

D'autres applications sont conçues pour mettre en relation des groupes d'agents «humains» afin de travailler coopérativement et prendre des décisions en commun. Regroupées sous le sigle CSCW («Computer-Supported Cooperative Work») ou CSHC («Computer-Supported Human Cooperation»), elles sont développées pour la coordination de groupes de travail, comme par exemple la création d'agendas et l'acheminement automatique d'information, par l'intégration de fonctionnalités collaboratives dans les outils de bureautiques (Favier, 1995; Maes, 1994).

Les collecticiels sont donc des systèmes informatiques qui assistent un groupe de personnes engagées dans une tâche commune et qui fournissent une interface à un environnement partagé. On peut distinguer trois fonctions principales : la production et le partage d'objets informatiques (e. g. un rapport, une feuille de calcul) ; la communication entre les utilisateurs en faisant référence à un objet informatique ; la coordination des utilisateurs. Les salles de réunions informatisées, les éditeurs partagés de textes et le courrier électronique sont des exemples de collecticiels. Ainsi, dans le cas du courrier électronique,

il existe des logiciels qui lisent les entêtes des messages, ce qui déclenche des actions d'archivage, de destruction ou de réponse en fonction de leur contenu.

1.8 Caractéristiques des approches multiagents

Un système multiagent n'est pas totalement comparable à un système réparti ; en effet, les algorithmes répartis sont fondés sur des modules de traitement distribués, mais sans autonomie. Les systèmes multiagents ne disposent pas d'un mécanisme centralisé de contrôle et les interactions ne sont pratiquement pas définies par le concepteur. Par conséquent, l'approche multiagent convient à des applications où la capacité d'adaptation des agents aux changements d'état de l'environnement ou à ceux des autres agents, est une caractéristique importante. Plus précisément, les principales caractéristiques des domaines d'applications concernés sont les suivantes (Bouron, 1992):

- certaines tâches requièrent plusieurs agents pour être réalisées ;
- plusieurs d'entre elles peuvent être effectuées simultanément ;
- leurs caractéristiques et leur nombre changent continuellement et de façon imprévisible ;
- la connaissance et le contrôle sont distribués, ce qui permet un fonctionnement concurrent des agents ;
- les agents ne connaissent pas la totalité de l'ensemble des tâches à réaliser.

Par ailleurs, nous avons vu que les thèmes de recherches portent d'une part, sur la modélisation des agents (i. e. modèle cognitif ou réactif) et d'autre part, sur la modélisation de leurs interactions (i. e. résolution de problèmes, négociation, coordination).

Par conséquent, nous allons développer une synthèse «transversale» aux thèmes de recherches, comportant deux parties : la notion d'agent et la notion d'interaction.

1.8.1 Notion d'agent

Quelle que soit l'application, on constate que les agents réalisent un certain nombre de tâches génériques :

Tâches de production : l'agent met en œuvre des capacités spécifiques pour atteindre son but ; déplacer un objet, résoudre un problème ; traiter les données issues d'un capteur.

Tâches de contrôle : elles sont de deux types différents :

- d'une part, l'agent choisit les tâches ou les actions à réaliser ;
- d'autre part, l'agent coordonne ses activités internes avec les autres agents (e. g. mise à jour d'un agenda de tâches).

Tâches de mise à jour de connaissances : l'agent met à jour sa représentation de l'environnement, enregistre les événements et les actions entreprises, mémorise la solution partielle à un problème, fait évoluer ses états mentaux.

Tâches de perception - action : il s'agit de l'interface entre l'agent et son environnement (e. g. les capteurs et les effecteurs) ainsi que les communications avec les autres agents ; on distingue généralement deux modes de communications, par partage d'information dans une structure commune (i. e. un tableau noir) (Engelmore and Morgan, 1988) et par envoi de messages (Agha, 1986; Hewitt, 1977; Shoham, 1993).

Remarque : pour un domaine donné, il n'est pas toujours simple pour le concepteur d'un système multiagent d'identifier ce que doit être un agent. On retrouve la dualité classique entre une analyse centrée sur les objets ou sur les fonctions (Ferber, 1995). Dans le premier cas, on considère qu'un agent est

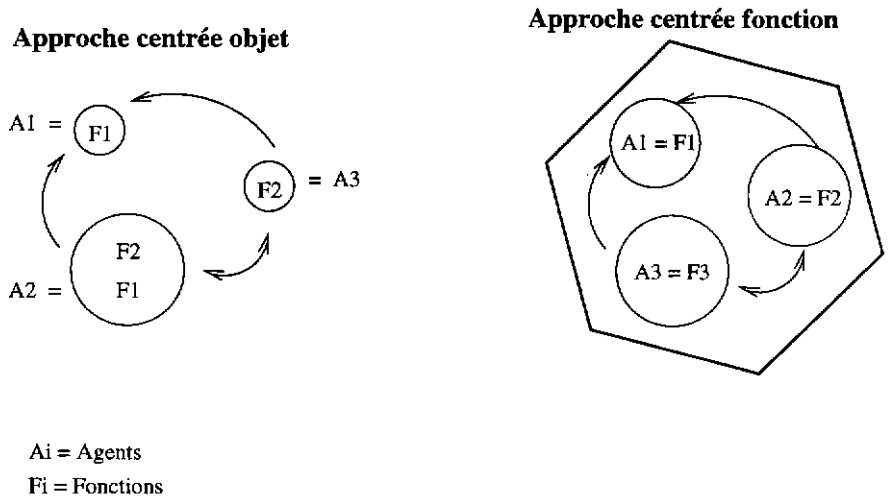


FIG. 1.4 – *Dualité fonctions/objects*

objet (un système autonome ou un capteur couvrant une zone géographique) réalisant des fonctions (e. g. transport de container, surveillance d'un site) tandis que, dans l'autre cas, les agents sont des fonctions réalisées par un objet ; c'est le cas, par exemple d'un système multi-expert où les agents sont les

modules d'expertise. La figure 1.4 montre une approche «centrée objet» dans laquelle les agents en interaction $A1$, $A2$ et $A3$ se partagent trois fonctions $F1$, $F2$ et $F3$; l'approche «centrée fonction» montre un système à l'intérieur duquel trois agents exécutant des fonctions spécifiques sont en interaction.

On constate qu'il est difficile de définir dans le cas général la notion d'agent et de système multiagent. Nous retiendrons ici des définitions adaptées de J. Ferber (Ferber, 1995) :

On appelle agent une entité physique ou virtuelle qui :

- est capable d'agir dans un environnement ;
- peut communiquer directement avec d'autres agents ;
- est mue par un ensemble des buts ;
- possède des ressources propres ;
- est capable de percevoir de manière partielle son environnement ;
- ne dispose que d'une représentation partielle de cet environnement ;
- possède des compétences et offre des services ;
- peut éventuellement se reproduire ;
- tend à satisfaire ses objectifs, en tenant compte des ressources, des compétences, et des informations dont il dispose.

On appelle système multiagent, un système composé des éléments suivants :

- un environnement, c'est-à-dire un espace disposant généralement d'une métrique ;
- un ensemble d'objets : ces objets sont situés, c'est-à-dire que, à tout objet, il est possible, à un moment donné, d'associer une position dans l'environnement. Ces objets sont passifs, c'est-à-dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents ;
- un ensemble d'agents représentant les entités actives du système ;
- un ensemble de relations qui unissent des objets et des agents entre eux ;
- un ensemble d'opérations permettant aux agents de percevoir, produire, consommer, transformer et manipuler des objets ;
- des opérateurs chargés de représenter l'application de ces opérations et la réponse des objets à ces opérations ;

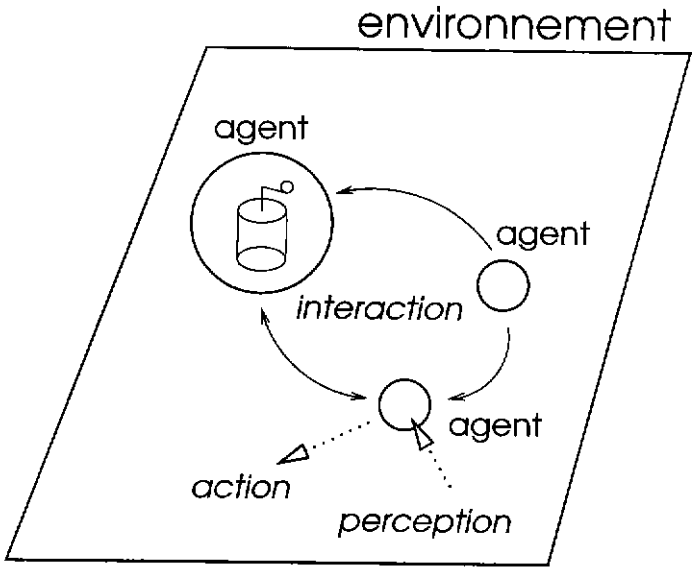


FIG. 1.5 – *Un système multiagent*

1.8.2 Notion d'interaction

La capacité de contrôle local des agents (i. e. leur autonomie) fait que les tâches de production, les informations communiquées par les autres agents ou perçues dans l'environnement sont traitées de façon asynchrone. Ce mode de fonctionnement pose le problème du fonctionnement cohérent du groupe d'agents. Selon J. P. Verjus (Verjus, 1991) : «Dans un système réparti, on ne peut faire l'hypothèse de l'existence d'une heure unique, car il n'existe aucune garantie que les horloges des différents sites soient synchrones. Il faut donc pouvoir construire un ordre global strict des événements, valable pour tous les sites du système et obtenu par coopération entre tous les processus concernés». Les systèmes multiagents, au même titre que tous les systèmes répartis, connaissent donc des problèmes de synchronisation des tâches (l'existence de délais de propagation d'une information produit des décalages entre les différents centres de décision), d'ordonnancement d'événements, de terminaison (comment détecter une propriété stable, comme l'arrêt de tous les modules, à partir d'informations locales et partielles ?) et de fiabilité (cf. problème-type dit des «généraux byzantins»[1]). Cela entraîne un comportement indéterministe du système (Florin, 1996) :

- non déterminisme des communications : «la requête expédiée est-elle bien

1. Soit n machines interconnectées dont p sont réputées non fiables : comment s'assurer que les informations reçues par les machines fiables sont les mêmes ? Le problème n'a pas de solution pour $p > \frac{n}{3}$ et le principe des solutions repose sur l'existence d'un consensus majoritaire (Verjus, 1991).

arrivée ?»

- non déterminisme des exécutions : «la requête a-t-elle été prise en compte ?»
- non déterminisme dû aux pannes.

Mais, ce qui caractérise plus particulièrement les systèmes multiagents, c'est le *non déterminisme des interactions* : dans ce cas, même la connaissance complète et exacte de l'état et de la structure du système, ainsi que de la valeur de ses entrées, ne permet pas de connaître ses sorties. Selon C. Hewitt (Hewitt, 1986), ces systèmes ne se comportent pas comme des machines de Turing dans la mesure où les opérations peuvent être affectées par des communications asynchrones provenant de leur environnement externe ; leur nature n'est même pas de type probabiliste et ils sont le siège de conflits et d'informations contradictoires. Dans les systèmes multiagents, le contrôle est distribué, et les

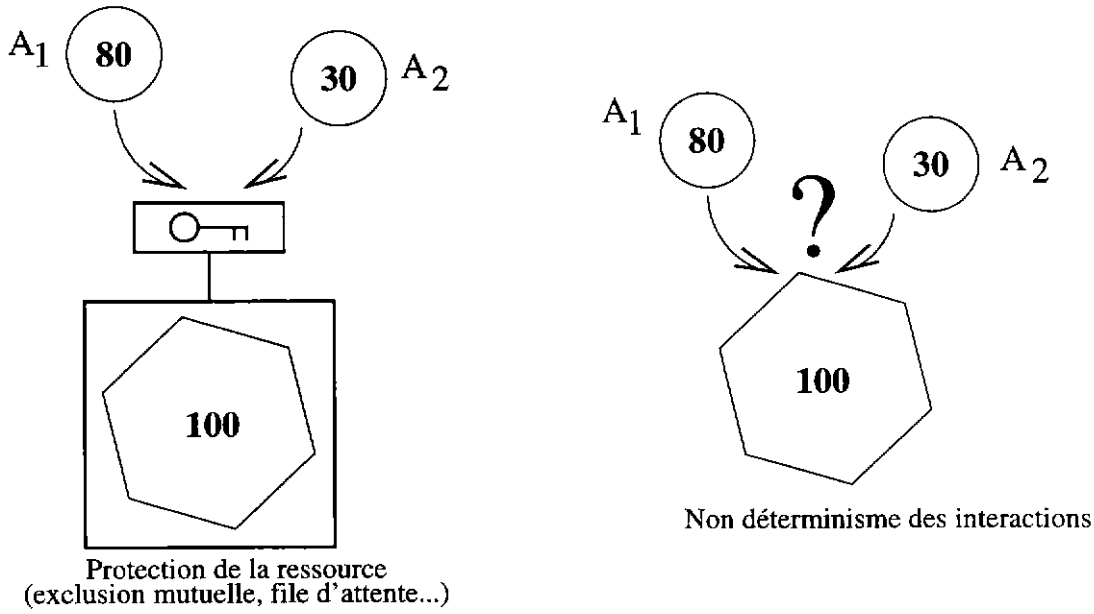


FIG. 1.6 – *Non déterminisme des interactions*

agents ne maîtrisent pas *a priori* les «ressources» : les ressources ne sont pas gérées par des mécanismes qui déterminent quel est l'agent opérant sur une ressource utilisable sur le mode de l'exclusion mutuelle et de la file d'attente. La figure 1.6 montre deux agents $A1$ et $A2$ voulant accéder simultanément à une même ressource : l'agent $A1$ veut en occuper 80 % et l'agent $A2$, 30 %. Du point de vue des agents, leurs requêtes sont cohérentes par rapport à la ressource, mais elles ne peuvent pas être satisfaites conjointement. Dans une approche «classique» des systèmes distribués, ce type de conflits est résolu par des mécanismes prédéfinis par le concepteur (e. g. file d'attente). Dans une

approche multiagent (cf. figure 1.6, dessin de droite), les interactions sont indéterminées et les agents doivent coordonner leurs actions. On constate, par ailleurs, que la connaissance des caractéristiques d'entrée du système (dans le cas de la figure 1.6, les deux agents et leurs requêtes), ne permet pas de déterminer *a priori* ses caractéristiques de sortie (i. e. quel agent accédera à la ressource le premier).

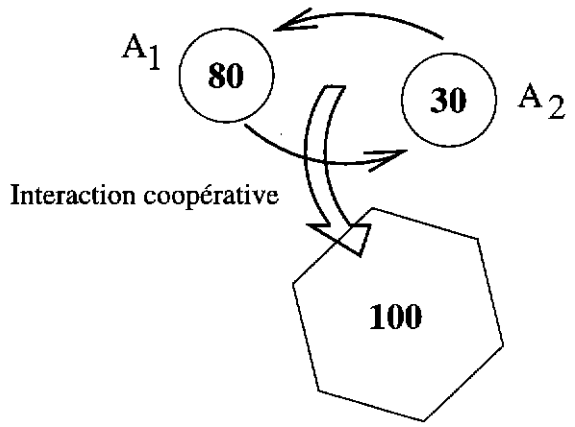


FIG. 1.7 – *Interactions cooperatives*

La coopération compense l'absence de moyens de régulation et rend *cohérent* le fonctionnement d'un système par nature non déterministe (cf. figure 1.7).

Coopérer, c'est donc «décider qui fait quoi», c'est-à-dire prendre les décisions adéquates relatives aux actions qui doivent être entreprises, par quel agent et à quel moment. On peut caractériser l'adéquation des décisions par rapport aux tâches assignées aux agents par trois propriétés (adaptées de (Jennings, 1994)) qui doivent être au minimum vérifiées :

Propriété de couverture : la totalité des tâches doivent être effectuées par au moins un agent ;

Propriété de connexité : les agents coordonnent les interactions de leurs tâches locales (e. g. conflits d'accès à une ressource, redondance d'une tâche, ordonnancement d'un sous-ensemble de tâches) afin que celles-ci aboutissent à un comportement cohérent par rapport à des critères dépendant de l'application (e. g. réalisation effective de la tâche collective, critères de performance, de validité, de complétude lorsqu'il s'agit de résoudre un problème) ;

Propriété de capacité : les propriétés de couverture et de connexité doivent être vérifiées relativement à un réseau de communication et des ressources prédéfinies (e. g. bande passante du réseau, puissance de calcul, zones d'évolution des agents).

Or, vérifier ces propriétés, c'est résoudre trois problèmes génériques définis par E. H. Durfee et V. R. Lesser (Durfee and Lesser, 1989) :

Problème de décomposition : les agents décident de la meilleure façon de décomposer la tâche collective en un ensemble de sous-tâches pour en réduire la complexité et permettre une résolution distribuée (cf. figure 1.8).

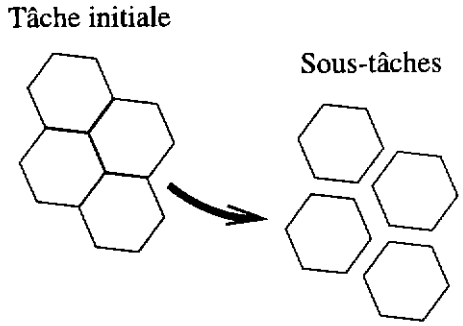


FIG. 1.8 – *Problème de décomposition*

Problème de distribution : c'est un problème d'allocation des sous-tâches et des ressources communes (cf. figure 1.9). Les agents cherchent la meilleure répartition possible en fonction de paramètres qui dépendent de l'application (e. g. capacités spécifiques et charge de résolution d'un agent à un instant donné). Ce problème a fait l'objet de nombreux travaux issus des idées de R. Smith et R. Davis (Smith and Davis, 1981) et de leur protocole de négociation «*contract net*».

Problème d'association : les agents échangent des informations sur leurs activités pour détecter des relations de dépendance (cf. figure 1.10) qui sont, de notre point de vue, au nombre de trois :

(i) Les agents peuvent être en conflit pour l'accès à une ressource ou ne partagent pas le même point de vue sur la situation globale ou produisent des actions incompatibles. Il existe une littérature abondante qui traite ce sujet (cf. (Adler et al., 1989) pour une synthèse). Le cadre formel le plus souvent cité est celui des protocoles de négociation où les agents échangent itérativement des propositions et

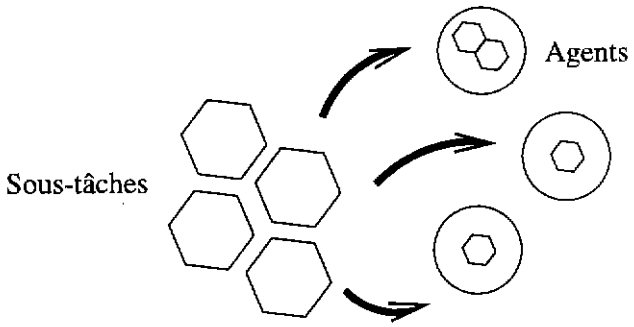


FIG. 1.9 – *Problème de distribution*

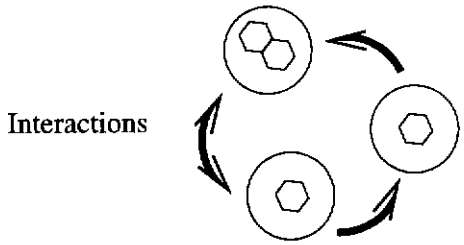


FIG. 1.10 – *Problème d'association*

des contre-propositions jusqu'à trouver un accord «équitable» pour l'ensemble des parties.

(ii) Les agents peuvent agir de façon redondante. Le problème ici consiste pour le concepteur à trouver le meilleur compromis entre la robustesse et les performances de son système. Cet équilibre peut être atteint dynamiquement par les agents qui s'échangent des informations sur leurs activités courantes. Si on dispose de stratégies qui permettent de déterminer quelles sont les informations pertinentes à transmettre, on peut contraindre l'espace de recherche d'un agent et lui éviter de refaire le travail d'un autre agent. Dans le cas contraire, on risque de «distraire» l'agent par le traitement d'informations non pertinentes ou même d'induire des décisions erronées de sa part.

(iii) Généralement, les tâches locales doivent être partiellement ordonnées dans le temps. Cela revient donc à élaborer un agenda commun qui tient compte à la fois des contraintes locales (e. g. estimation du temps nécessaire à la réalisation d'une tâche locale, gestion des ressources) et de la décomposition initiale de la tâche collective.

Remarque : T. Bouron (Bouron, 1992) propose six indices pour évaluer les interactions :

(i) la coordination d'actions ;

(ii) la parallélisation des tâches ;

(iii) le partage des ressources ;

(iv) la robustesse ;

(v) la non-redondance ;

(vi) la non-persistance des conflits.

Conclusion

Nous avons abordé les différents thèmes de recherches multiagents qui sont la modélisation cognitive et la simulation d'organisations sociales, la programmation orientée agents pour la résolution distribuée de problèmes, l'éco-résolution, les protocoles de négociations automatiques, la coordination de systèmes autonomes et les collecticiels.

Puis, nous avons dégagé deux notions communes à tous ces thèmes de recherches : nous avons proposé quelques caractéristiques génériques de la notion d'agent ; nous avons montré que la notion d'interaction est synonyme de non déterminisme pour les systèmes multiagents. Nous en avons conclu que, contrairement aux systèmes distribués «classiques» où les interactions sont structurées par le concepteur, dans les systèmes multiagents, les agents prennent en charge la tâche de coordination.

Dans le chapitre suivant, nous étudions les méthodes expérimentées pour réduire l'incertitude des agents, c'est-à-dire leur permettant de se coordonner de façon cohérente : nous distinguons deux classes de méthodes, la coopération et la négociation, selon que les agents ont un but commun ou des buts concurrents.

Chapitre 2

Méthodes de coopération et de négociation

Introduction

La notion de coopération semble insaisissable. Certains la définissent comme «une dynamique collective aboutissant à un état émergeant bénéfique»[1].

La coopération serait donc un phénomène contingent ou une attitude non intentionnelle des agents. A. Drogoul (Drogoul and Ferber, 1992) considère même que :«la coopération doit émerger de la division du travail et de la compétition entre les agents pour accomplir une tâche.» De façon paradoxale, il présente ainsi la compétition entre agents comme la source de la coopération.

J. R. Galliers, au contraire, postule que la coopération est une attitude qui est explicitement inscrite dans les comportements des agents (Galliers, 1989) : la coopération exige que les agents identifient un but commun, s'engagent dans des actions visant à réaliser ce but commun, soient aptes à reconnaître que les autres agents sont engagés vis-à-vis du même but, et qu'ils soient aptes à éviter les conflits ou à les résoudre. Or, toutes ces exigences peuvent être atteintes par la manipulation formelle des états mentaux. Dans la coopération, il y aurait donc une composante de soutien, c'est-à-dire une attitude intentionnelle qui consiste à aider les autres agents même en l'absence de conflits.

E. Werner constate, quant à lui, qu'il y a deux types de coopération (Werner, 1989). La coopération «négative» se produit lorsque deux agents évitent d'accomplir une action simultanément : il existe des «régions critiques» qu'un

1. Journées sur le Contrôle Réparti dans les Applications Coopératives, Université Pierre et Marie Curie, Paris, 30-31 mai 1996.

seul agent est autorisé à occuper à un instant donné et qui sont donc traitées comme des «zones d'exclusion». Ces situations se rencontrent le plus souvent lorsque les agents sont faiblement couplés et ont des «buts indépendants». Dans ce type de situations, on peut faire l'hypothèse de «connaissance limitée» c'est-à-dire que la connaissance de l'état ou de la région d'une autre action n'est requise que si cette action est dans une région critique et la communication nécessaire à la coopération se limite souvent à l'échange de «signaux de synchronisation». À l'opposé, lorsque les agents sont fortement couplés et partagent des buts communs, il faut mettre en oeuvre une coopération «positive». Dans ce cas, les agents ont besoin de l'aide des autres pour accomplir leurs buts. Il se crée alors des «zones de coopération» (notion de besoin) à côté de «zones d'exclusion» (notion d'évitement).

Tous ces travaux, bien qu'apparemment contradictoires, ont en commun de se poser la même question : *qu'est-ce que la coopération ?* Notre démarche est différente. Nous ne considérons que les deux seules situations suivantes :

- les agents ont un but commun : les agents ont été conçus pour résoudre un problème, élaborer un diagnostic de panne, construire un plan d'actions. Nous avons constaté que leur objectif est alors de se coordonner (cf. paragraphe 4.2) c'est-à-dire de produire un comportement globalement *cohérent* (Bond and Gasser, 1988). En fonction des applications, la cohérence du système est évaluée en termes de résolution de conflits, rapidité de calcul, capacité à trouver une solution.

- les agents ont des buts concurrents : ce peut être le cas d'agents virtuels se substituant à des organisations ou des personnes physiques pour effectuer des transactions automatiques. Dans ces situations, les agents ont des intérêts divergents qu'ils tentent de satisfaire. Parfois, comme c'est le cas de l'éco-résolution, on peut définir des agents antagonistes pour tirer parti des gains de performance liés à une résolution distribuée.

Par ailleurs, nous nous intéressons aux différentes méthodes multiagents :

- nous postulons que la coopération a pour objet de produire des comportements cohérents dans le cas d'agents partageant le même but et nous qualifions de «coopératives» les méthodes mises en œuvre par ce type d'agents ;

- de la même façon, nous considérons que la négociation a pour objet la «satisfaction maximale» des agents à buts concurrents. Cela suppose que les agents cherchent le «meilleur accord possible» : suivant les applications, cet accord est évalué par des critères de gains, de coûts ou de validité.

Nous présentons ces deux approches en insistant plus particulièrement sur les méthodes de coopération.

Agents	Coopération	Négociation
Ils ont...	un but commun	des buts concurrents
Ils cherchent...	la cohérence globale	leur satisfaction locale

TAB. 2.1 – *Notions présentées*

Le tableau 2.1 présente les notions qui sont abordées dans ce chapitre.

2.1 La négociation

Les travaux relatifs à la négociation automatique prennent une place prépondérante dans les approches multiagents. On assiste, en effet, à d'une part une automatisation de plus en plus poussée des activités humaines et d'autre part, à l'interconnection toujours croissante des réseaux informatiques. Ainsi, l'automatisation de certaines transactions boursières et des opérations de courtage est le symbole paradigmatique de cette évolution. Le transport routier de fret, le transport aérien et les aéroports sont aussi des secteurs susceptibles de connaître ces bouleversements à court terme (Rosenschein and Zlotkin, 1994). Le développement des réseaux de télécommunications, qui touche tout le monde, permet d'envisager également la mise en concurrence des différents opérateurs (e. g. téléphonie mobile, diffusion par câble, par satellite) et des prestataires de services (e. g. chaînes de télévision, éditeurs de logiciels) : le «commerce électronique» est déjà une réalité.

2.1.1 Le partage de tâche

R. Smith et R. Davis (Smith and Davis, 1981) sont les précurseurs des méthodes de négociation automatiques. Leur protocole appelé «contract-net» permet l'allocation de tâches sur un réseau d'agents qui coordonnent leurs activités en établissant des «contrats» entre eux.

Le protocole comporte trois phases :

(1) Dans la première phase, un agent, appelé le manager, décompose ses tâches (de façon prédéterminée) en plusieurs sous-contrats qu'il propose aux autres agents qui sont des contractants potentiels ;

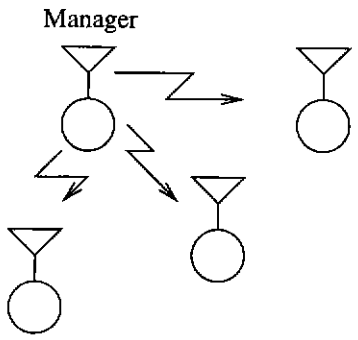


FIG. 2.1 – *Réseau contractuel : proposition de tâche*

(2) Ayant évalué leur charge de travail et estimé le coût d'un sous-contrat en fonction de leur expertise (e. g. en termes de durée), les contractants renvoient leurs offres au manager ;

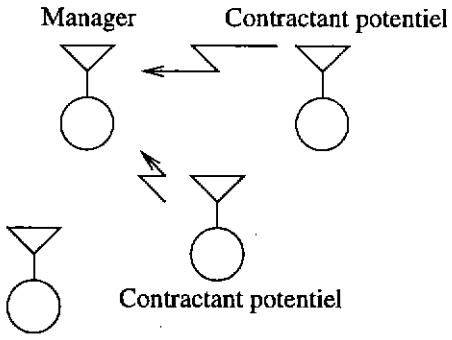


FIG. 2.2 – *Réseau contractuel : offres*

(3) Dans la dernière phase, le manager évalue l'ensemble des offres et attribue le contrat à l'agent lui ayant fait la meilleure proposition. À son tour, le contractant peut devenir manager et proposer une part de son contrat aux autres membres du réseau.

Le «contract-net» fournit donc un cadre général pour l'échange de messages entre agents mais il ne dit pas comment ils doivent négocier. De même, il ne donne aucune indication sur les connaissances nécessaires à un manager pour choisir les tâches à proposer et pour choisir la meilleure proposition. Réciproquement pour le contractant, on doit se poser la question des critères de sélection d'un contrat et d'élaboration d'une proposition.

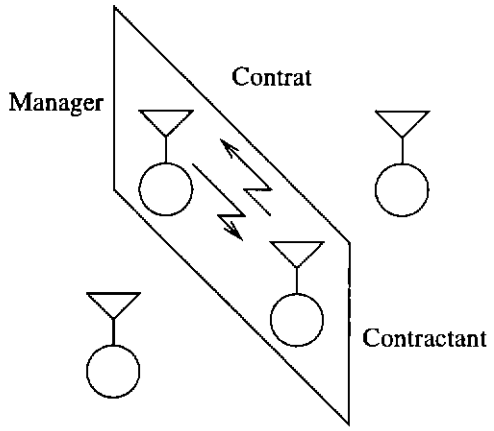


FIG. 2.3 – *Réseau contractuel : évaluation des offres et attribution de la tâche*

2.1.2 La résolution de contraintes distribuées

La négociation par étapes (i. e. «multistage negotiation»), est une extension des réseaux contractuels. Elle s'intéresse, comme le «contract-net», au problème de l'allocation de tâches représenté sous la forme d'un problème de satisfaction de contraintes dans lequel un ensemble d'actions doit être coordonné afin d'atteindre les objectifs du réseau, mais où chaque agent a des ressources limitées pour réaliser ses propres actions et celles qui lui sont assignées par d'autres agents.

S. E. Conry (Conry et al., 1991) a étudié le cas d'un réseau de télécommunications représenté par un graphe dont les nœuds sont les relais de transmission. Un agent est un équipement de surveillance ayant sous sa responsabilité plusieurs nœuds. Lorsqu'une interruption du service se produit (e. g. panne, surcharge), cet équipement peut rediriger le trafic des communications vers d'autres régions ; les agents interagissent donc pour résoudre un problème distribué, qui consiste à évaluer la situation et à rétablir les liaisons point à point. La négociation par étapes permet d'itérer les phases d'offres et d'attributions car un agent cherche à trouver la meilleure configuration satisfaisant ses contraintes (i. e . il n'y a pas de construction d'une vue globale) en tenant compte des effets de ses décisions sur la charge des autres régions.

S. E. Lander et V. R. Lesser (Lander and Lesser, 1993) adoptent une approche semblable, dans laquelle les buts sont spécifiés par des contraintes. Mais, alors que dans le formalisme précédent tous les agents doivent partager une même représentation du problème pour s'échanger des informations, ils s'intéressent au cas d'agents hétérogènes, qui peuvent avoir des paradigmes de résolution différents et qui sont susceptibles de ne pas pouvoir donner des

informations aux autres agents ou comprendre leurs messages. Dans le système TEAM, S. E. Lander et V. R. Lesser font une hypothèse forte car les agents ne partagent *a priori* qu'une vue identique sur l'état de la solution en construction et sur les opérateurs de changements d'état pour la faire évoluer ; lorsque cela est possible, les agents échangent des propositions ou des informations sur les conflits portant sur les attributs de la solution qu'ils ont en commun et une proposition ne devient donc une solution définitive que quand elle est acceptée par tous les agents. Formellement, l'état d'une solution est représenté par trois attributs prenant différentes valeurs :

- Acceptabilité : {*acceptable, inacceptable, infaisable*} ;
- Complétude : {*complet, incomplet*} ;
- État de la recherche : {*initial, terminé*}.

Il y a cinq opérateurs de changements d'état :

- Initier_solution ;
- Critiquer_solution ;
- Étendre_solution ;
- Relaxer_contraintes_sur_solution ;
- Terminer_solution.

On peut représenter le processus de recherche par un graphe d'états qui caractérise l'ensemble des états possibles et les changements d'états admissibles (cf. figure 2.4).

On passe d'un état à l'autre en appliquant l'un des opérateurs. Par exemple, on passe d'une solution incomplète et acceptable à une solution incomplète et non acceptable avec l'opérateur Critiquer_solution.

TEAM met en œuvre un mécanisme de relaxation de contraintes (i. e. Relaxer_contraintes_sur_solution) qui agit suivant trois modes : la relaxation unilatérale intervient lorsqu'un agent ne trouve pas de solution locale à la spécification d'un problème et décide donc de réduire ses exigences ; la relaxation rétroactive se déclenche lorsqu'un agent reçoit des informations lui indiquant un conflit entre ses contraintes et celles d'un autre agent plus prioritaire ; enfin, la relaxation globale est une réaction à l'absence de progression du processus de résolution et oblige l'ensemble des agents à réviser leurs exigences (différentes stratégies sont envisageables comme une relaxation générale au bout d'un certain nombre de cycles de résolution ou par niveau de priorité).

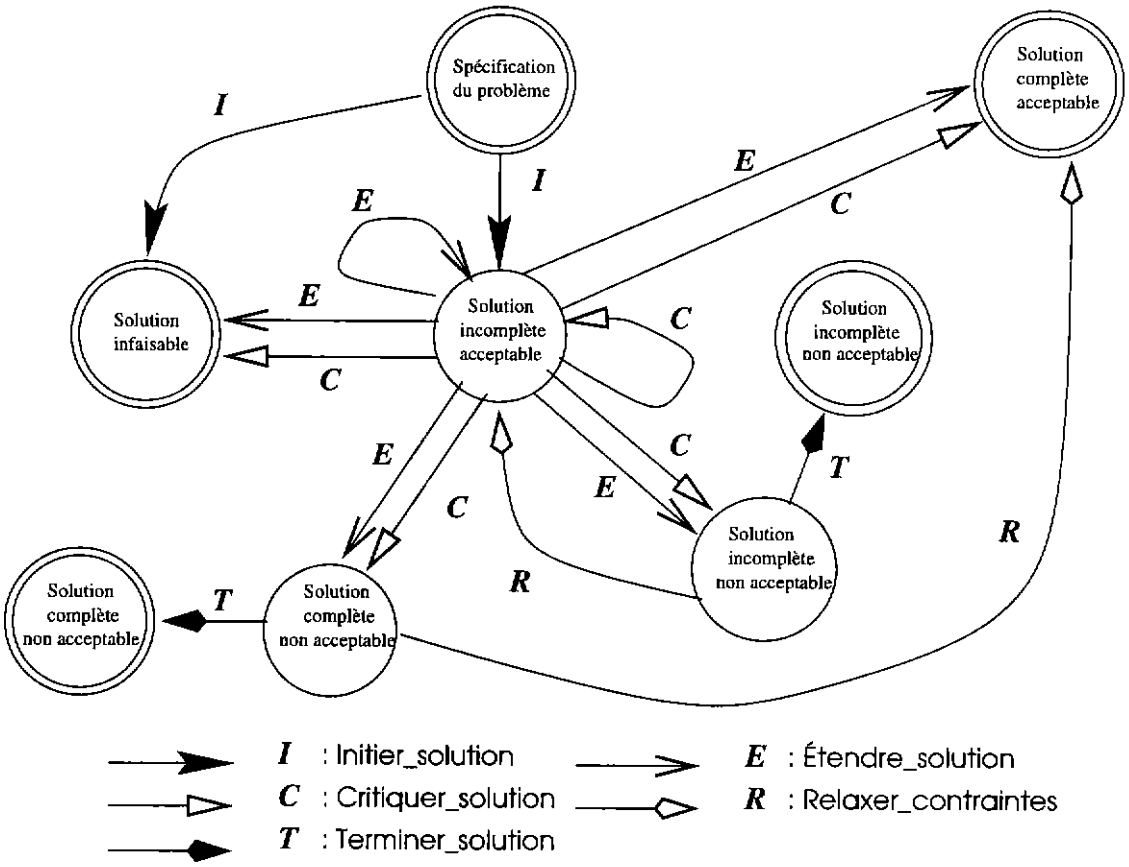


FIG. 2.4 – *Processus de recherche*

2.1.3 Les protocoles de négociation

Les protocoles de négociation sont fondés sur un échange itératif de proposition/contre-proposition qui conduit à un *accord* (i. e. «deal») ; les agents, par un échange structuré d'informations pertinentes, cherchent à améliorer cet accord en leur faveur.

Le terme de protocole ne se réfère pas ici à un ensemble de règles spécifiant des mécanismes de communication comme par exemple un protocole de transfert de fichiers (i. e. «File Transfer Protocol») : il dénote un ensemble de notions, comme un «accord» ou un «compromis», et de propriétés vérifiées par elles. En particulier, J. S. Rosenschein et G. Zlotkin (Rosenschein and Zlotkin, 1994) appliquent les outils de la théorie de jeux pour concevoir et évaluer ces protocoles.

Par exemple, l'«efficacité» d'un protocole se détermine par rapport à la possibilité pour les agents de trouver un accord «Pareto-optimal» : les agents

poursuivent des objectifs individuels et ils négocient le «meilleur» accord possible ; on suppose que les agents sont capables de mesurer le coût d'un accord par une fonction d'utilité. G. Zlotkin et J. S. Rosenschein (Zlotkin and Rosenschein, 1993) définissent un accord «Pareto-optimal», par un accord tel qu'aucun agent ne peut obtenir un gain supérieur à un accord différent sans réduire le gain d'un autre agent (i. e. si A_1 retire la même utilité des accords x et y, mais que A_2 «préfère» y, on fait en sorte que x ne soit pas choisi).

De même, l'accord est «globalement optimal» lorsque la somme des gains de chaque agent est maximale. K. Sycara (Sycara, 1989), dans son système PERSUADER définit un accord «équitable» comme un accord qui maximise les gains individuels et minimise l'écart de gain.

G. Zlotkin et J. S. Rosenschein (Rosenschein and Zlotkin, 1994) s'intéressent à d'autres propriétés :

- la stabilité : le protocole doit être tel qu'aucun agent ne doit avoir intérêt à s'en écarter ;
- la simplicité : les interactions entre agents doivent nécessiter peu de ressources en termes de temps de calcul ou de bande passante ;
- la distribution : les interactions ne doivent pas passer par un coordonnateur ;
- la symétrie : tous les agents doivent être traités de la même façon.

Remarque : certains travaux se sont intéressés à la modélisation de la négociation telle qu'elle existe dans les organisations sociales (Sycara, 1989; Sathi et al., 1986). Par exemple, K. Sycara a étudié la notion d'argumentation. Dans le système PERSUADER (Sycara, 1989), qui simule une négociation entre un syndicat, la direction d'une entreprise et un médiateur, , elle se sert du «raisonnement sur des cas» (i. e. «Case-Based Reasoning») et de la théorie de la décision pour générer des arguments pertinents. Les agents font évoluer les attributs d'un argument (e. g. coût de production(x), salaires(y), productivité(z)) ainsi que l'importance, c'est-à-dire l'évaluation qu'ils accordent aux attributs de cet argument. Pour cela, ils emploient un ensemble de stratégies de négociation telles que la menace et l'appel au *status quo*.

2.2 La coopération

Les systèmes distribués, et en particulier les systèmes multiagents, du fait de l'autonomie des agents, sont non déterministes. La meilleure façon de

résoudre ces incertitudes consisterait à donner, à chaque agent et à chaque instant, une connaissance complète sur les autres agents (i. e. quelles sont leurs tâches? de quelles informations disposent-ils?). Raisonnablement, cet objectif n'est pas réalisable.

Comment, alors, obtenir un comportement cohérent des agents vis-à-vis de leur but commun?

Dans le reste de ce chapitre, nous précisons la notion de cohérence. Puis, nous présentons les trois principales méthodes utilisées en les illustrant par des exemples tirés d'expérimentations ou de systèmes effectifs.

De notre point de vue, les méthodes *ad hoc* réduisant l'incertitude inhérente à l'impossibilité de diffuser une connaissance complète à travers un système sont :

- l'utilisation de connaissances communes ;
- l'échange d'informations de contrôle ;
- l'élaboration d'une conjecture commune.

2.2.1 Notion de cohérence

Les méthodes de coopération sont une façon d'améliorer la *cohérence* du système multiagent, c'est-à-dire sa capacité à se comporter comme une unité par rapport à certains critères d'évaluation (Bond and Gasser, 1988). E. H. Durfee, V. R. Lesser et D. D. Corkill (Durfee et al., 1989) dénombrent quatre buts génériques pour la coopération :

- augmenter la vitesse de résolution des tâches par leur parallélisation ;
- augmenter le nombre ou la portée des tâches réalisables par le partage des ressources ;
- augmenter la possibilité d'achever des tâches par leur duplication sur le groupe d'agents et, si possible, par l'utilisation de méthodes différentes ;
- diminuer les «interférences» entre les tâches.

En somme, la cohérence est essentiellement synonyme d'efficacité et de robustesse du système.

Selon J. Ferber (Ferber, 1995), l'efficacité recherchée dans les systèmes multiagents est dû à un *effet de masse* et à un *effet de catalyse* :

- Effet de masse : il se caractérise par l'assistance que peuvent s'accorder les agents les uns aux autres. Prenons le cas d'un agent qui se trouve dans l'impossibilité de réaliser complètement une tâche par manque de ressources. Celles-ci peuvent être apportées par un autre agent : la réalisation de la tâche est obtenue par l'«addition» des capacités des agents. En particulier, un agent inoccupé peut venir en aide aux autres agents comme c'est, par exemple, le cas dans le système DVMT (Durfee et al., 1987) ;

- Effet de catalyse : une tâche peut ne pas être complètement réalisée par un agent par manque de compétence. Un agent peut donc jouer le rôle de catalyseur en apportant les compétences nécessaires à l'achèvement de la tâche. Ainsi, dans le système multi-expert SMIEG (Bouzouane, 1995), un rapport d'expertise est réalisé par un «expert juridique», un «expert financier» et un «expert en sinistres».

La robustesse est aussi une propriété importante d'un système. Elle se caractérise par la dégradation de performances d'un système lorsqu'il sort de son point de fonctionnement, par exemple en présence de données fortement bruitées ou en cas de défaillances d'agents. On peut compenser le risque de défaillance par des agents redondants ; J. Ferber (Ferber, 1995) définit le *degré de redondance* R par le nombre d'agents ayant les mêmes compétences. Le *degré de spécialisation* H d'un agent pour un problème indique alors le taux de compétence qu'un agent possède par rapport au nombre de compétences nécessaires pour résoudre ce problème. J. Ferber propose ainsi de définir une «fonction de performance organisationnelle» $\mu(R, H)$ qui indique la performance du système dans l'«espace» $R \times H$.

Dans le cas de systèmes intégrant des superviseurs humains, la transparence ou lisibilité du fonctionnement semblent être des critères importants pour évaluer la cohérence (Wittig et al., 1994; Erceau et al., 1994) : ils posent la question du comportement complexe de ces systèmes et de l'intégration de fonctions de supervision et de régulation afin de rendre possibles les tâches de diagnostic et la maintenance par un superviseur humain.

2.2.2 L'utilisation de connaissances communes

L'utilisation de connaissances communes à tous les agents est la première méthode de coopération que nous abordons.

Notion de connaissances communes

Comme le souligne J. P. Dupuy[2], des agents interdépendants ne peuvent en général coordonner leurs actions que s'ils ont recours à des formes collectives de savoir. Ainsi, la théorie des jeux suggère que, pour ajuster leurs stratégies d'actions en fonction des raisonnements que les autres font à leur sujet, les agents doivent partager un même savoir. Les travaux de M. R. Genesereth et J. S. Rosenschein (Genesereth et al., 1986) illustrent ce point de vue.

		K	
		c	d
J	a	1 4	4 2
	b	3 1	2 3

TAB. 2.2 – *Exemple de matrice de coût pour les agents J et K*

Pour interagir «rationnellement», leurs agents partagent la même matrice de coûts portant sur les actions. Cette matrice doit définir les coûts de toutes les interactions possibles : les coûts sont invariants, ne dépendent pas des choix précédents, et les décisions des agents doivent être simultanées. Grâce à ce savoir partagé, ils sont capables de prendre des décisions rationnelles par rapport à des critères de maximisation de gains (e. g. le tableau 2.2 montre que le coût pour J de choisir l'action a lorsque K choisit c est de 4 ; réciproquement le coût pour K de choisir c lorsque J choisit a est de 1).

Partager des connaissances communes permet à un agent de raisonner sur les connaissances des autres agents ; ainsi, les agents doivent non seulement partager un même savoir, mais encore savoir qu'ils le partagent. Le paradoxe de «Conway» (Halpern and Moses, 1990) permet d'illustrer l'importance de ce concept, qui n'est pas trivial :

> Imaginons n enfants jouant ensemble. La mère de ces enfants leur a dit que, s'ils se salissaient, ils seraient sévèrement punis. Donc, chaque enfant veut rester propre, mais tous voudraient voir les autres se salir. Supposons que, pendant le jeu, k enfants se salissent le front. Chaque enfant peut voir la tache des autres mais pas l'état de son propre front. Évidemment, personne ne parle. Peu après, le père dit «au moins l'un d'entre vous a le front taché», exprimant donc un fait que *tous* connaissaient avant qu'il ne parle (si $k > 1$). A

2. J. P. Dupuy et P. Livet, «Rationalité située et savoir collectif», *Le courrier du CNRS*, vol. 79, n. 112, octobre 1992.

priori, son intervention n'apporte rien de neuf. Le père répète alors la question suivante jusqu'à obtenir une réponse : «L'un d'entre vous peut-il prouver qu'il a le front taché ?». En supposant que tous les enfants sont sincères, qu'ils perçoivent tous les membres du groupe et qu'ils répondent simultanément, qu'arrivera-t-il ?

J. Y. Halpern et Y. Moses ont formalisé ce problème (Halpern and Moses, 1990). Nous reprenons ici leurs notations. $K_i p$ dénote que «l'agent i connaît p» et par définition, on pose que :

$$K_i p \supset p$$

ce qui signifie que si un agent i connaît p alors p est vrai.

Pour J. Y. Halpern et Y. Moses, il existe différents «niveaux de connaissances» dans un système distribué :

- $D_G p$ dénote le fait que «*le groupe G a une connaissance distribuée de p*». On dit que p est distribuée sur G si un agent qui sait tout ce que chaque membre de G sait, connaît p.
- $S_G p$ dénote le fait que «*quelqu'un dans G connaît p*» c'est-à-dire :

$$S_G p \equiv \bigvee_{i \in G} K_i p$$

- $E_G p$ dénote que «*tout le monde dans G sait p*» c'est-à-dire :

$$E_G p \equiv \bigwedge_{i \in G} K_i p$$

- $E_G^k p$ dénote que «*p est une connaissance de niveau k dans G*» : $E_G^k p$ est défini par :

$$E_G^1 p = E_G p,$$

$$E_G^{k+1} p = E_G(E_G^k p),$$

  pour $k \geq 1$

- $C_G p$ dénote que «*p est une connaissance commune dans G*» c'est-à-dire :

$$C_G p \equiv E_G p \wedge E_G^2 p \wedge \cdots \wedge E_G^m p \wedge \cdots$$

Un fait p est donc un savoir ou connaissance commune pour deux agents J et K si chacun non seulement sait p, mais encore sait que chacun sait p, cette itération allant à l'infini. Cette définition est équivalente à un «point fixe». D'où l'axiome suivant :

$$C_G p \equiv E_G(p \wedge C_G p)$$

On remarque aussi que la connaissance du groupe est hiérarchisée de sorte que :

$$Cp \supset \cdots \supset E_G^{k+1} p \supset \cdots \supset E_G p \supset Sp \supset Dp \supset p$$

Grâce à ce formalisme, mis au point pour l'étude des propriétés des protocoles de communication dans les systèmes distribués, on peut démontrer que les $k-1$ premières fois que le père pose la question, tous les enfants répondent «non» mais que la ke fois tous les enfants tachés répondent «oui».

En effet, supposons que p dénote le fait «*au moins l'un d'entre vous est taché*». Avant que le père ne parle, $E^{k-1}p$ est vrai mais pas $E^k p$. Considérons le cas $k = 2$ et supposons que α et β sont les deux seuls enfants tachés. Comme chaque enfant voit au moins un autre enfant taché, on a Ep. Mais le seul enfant taché que voit α est β, et comme α ne sait pas s'il est taché, α considère que β est le seul enfant taché. α considère donc possible le fait que β soit le seul enfant taché et que β ne voit pas d'enfant taché. Bien que α et β connaissent p (d'où Ep), α ne sait pas que β connaît p et donc E^2p ne tient pas.

Par conséquent, le rôle du père est d'améliorer l'état de connaissance des enfants sur p de $E^{k-1}p$ à $E^k p$ sans, apparemment, apporter une nouvelle information. Après l'annonce publique par le père du fait p, chaque enfant sait p et que p est connu de tous c'est-à-dire que Cp est vrai. Revenons à notre exemple pour $k = 2$. À la première question, les deux enfants répondent «non» à cause de la tache sur le front qu'ils perçoivent sur l'autre enfant. Mais, lorsque β répond «non», α réalise qu'il doit être taché sans quoi β aurait su que la tache était sur son front et aurait répondu «oui» la première fois. Donc α répond «oui» la deuxième fois de même que β qui a utilisé le même raisonnement.

Connaissances communes dans les systèmes multiagents

Nous avons montré que les agents ont besoin de connaissances communes pour se coordonner. Dans les systèmes multiagents, l'*organisation* du système est une forme de connaissance fondamentale portant sur les relations existant entre les agents. Grâce à la connaissance de l'organisation, un agent peut raisonner sur les activités des autres agents et prendre des décisions rationnelles (e. g. échanger des informations pertinentes).

Pour J. Ferber (Ferber, 1995), il existe trois sortes de relations entre les agents : les relations d'*accointances* indiquent qu'un agent connaît l'existence d'un autre agent ; les relations de *communication* représentent l'existence d'un canal de communication entre deux agents ; les relations de *subordination* indiquent les *rôles* joués par les agents (e. g. maître - esclave), leurs compétences, les responsabilités dans un processus de résolution et éventuellement leurs buts. La structure de contrôle du système peut ainsi être centralisée, hiérarchique.

Dans DVMT, E. H. Durfee et V. R. Lesser (Durfee et al., 1987; Durfee and Lesser, 1987; Durfee and Lesser, 1991) proposent un environnement appelé «PGP - Partial Global Planning» pour la coordination des agents dont le but est de former les trajectoires de véhicules traversant une zone surveillée. L'architecture de DVMT intègre une structure de connaissance commune représentant l'organisation du groupe de deux points de vue, les relations d'accointances et les relations de subordination, c'est-à-dire :

- l'organisation d'un point de vue fonctionnel («domain-level organisation») : elle spécifie le rôle et les compétences des agents. En fonction de ses buts et plans locaux et de sa connaissance du domaine, un agent peut ainsi détecter des interactions ou des informations susceptibles de l'intéresser.

- l'organisation d'un point de vue structurel («metalevel organisation»): contient l'information sur le rôle de coordination de chaque agent, à un instant donné. Un agent peut ainsi décider à quel moment et avec qui échanger des informations (ces décisions peuvent faire l'objet de négociation entre agents). Par exemple, dans une structure centralisée, les résultats locaux sont envoyés à un agent qui en fait la synthèse.

Les *conventions* sont une autre forme de connnaissance collective : ce sont des dispositifs qui épargnent aux individus des efforts dans leur coordination, en particulier dans la gestion des situations d'incertitude. Par exemple, dans le domaine de la circulation routière, les véhicules doivent se déplacer en respectant le code de la route : conduire à droite, respecter la signalisation. La connaissance par tous de ces normes permet aux agents que sont les voitures, les piétons et tous les autres usagers, d'interagir de manière cohérente.

On trouve ce type de mécanisme dans une application du système GRATE (i. e. «Generic Rules and Agent model Testbed Environment») pour la surveillance d'un réseau de distribution d'électricité (Jennings, 1992; Jennings, 1994). Il se compose de trois modules experts, l'agent superviseur du réseau, l'agent d'analyse des alarmes (i. e. identification des éléments du réseau en panne) et l'agent de localisation des zones de «black-out». Lorsque le superviseur est informé d'une panne, il consulte une fiche (i. e. «recipe») qui lui indique

la procédure à suivre et l'existence des deux autres agents. Le superviseur demande alors l'adhésion des agents à la procédure de diagnostic. Le protocole de coopération prévoit que les trois agents doivent s'engager à suivre cette fiche. Ils peuvent aussi se désengager s'ils la trouvent inadéquate (en cas de fausse alerte par exemple), mais ils doivent en informer les autres agents (dans le cas contraire, les agents continueraient inutilement d'élaborer un diagnostic) ; tous les agents adhèrent à cette convention.

2.2.3 Échange d'information de contrôle

Une autre source de connaissance importante concerne les informations sur l'état local de chaque agent (e. g. actif ou inactif) et ses décisions (e. g. choix concernant des buts, prises de décision par rapport aux actions d'autres agents).

E. H. Durfee et V. R. Lesser (Durfee et al., 1987; Durfee and Lesser, 1987) ont intégré l'échange d'information de contrôle dans le «Partial Global Planning» : lorsqu'un agent reçoit des informations sur un plan, ils essaie de les intégrer à son propre plan afin de former un plan plus général (i. e. représentant un plus grand nombres d'activités du groupe d'agents). À cet effet, les agents s'échangent des «Partial Global Plans» qui représentent les buts poursuivis, leurs ordonnancements, les résultats et les temps de réalisation escomptés (i. e. des informations prédictives). Lorsque des activités complémentaires sont détectées entre deux «PGP», ceux-ci sont fusionnés.

Un «PGP» se divise en trois parties :

- l'objectif : contient l'information sur le but du «PGP» (par exemple la trajectoire d'un véhicule donné à construire) et son importance (i. e. un niveau de priorité par rapport à d'autres «PGP») ;
- le plan d'activités («plan-activity-map») : représente les étapes principales du plan (i. e. un ensemble de buts intermédiaires) entreprises parallèlement, leurs coûts et les résultats attendus ainsi que des justifications sur l'ordre des étapes ;
- le plan de construction («solution-construction-graph») : contient de l'information sur comment les agents doivent interagir, quels résultats partiels échanger et à quel moment.

À un instant donné, un «PGP» représente l'état global d'une activité du groupe tel que le perçoit localement un agent. L'ensemble des «PGP» peut donc être incohérent et les agents tentent de converger vers une vue commune de l'activité globale, une mise à jour locale des «PGP» étant effectuée par l'échange

de «PGP» entre agents. Ainsi, les agents essaient de converger vers une orga-

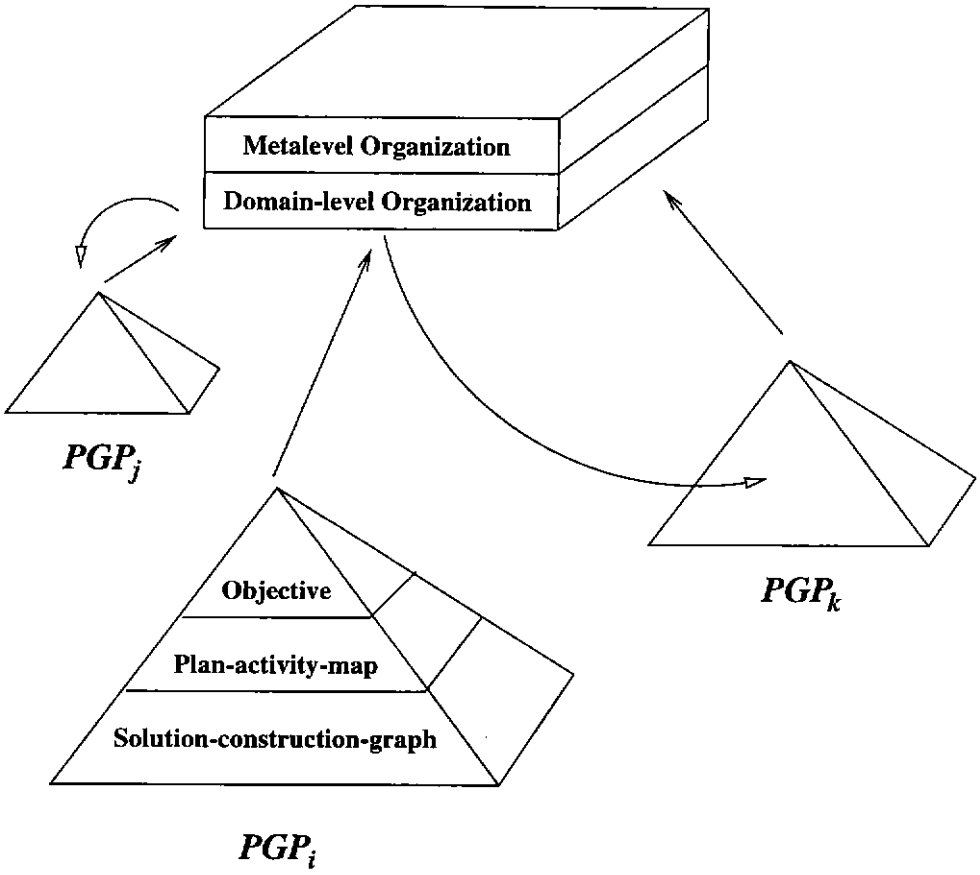


FIG. 2.5 – «*Partial Global Planning*»

nisation collective commune afin de diminuer au mieux l'incertitude sur leurs activités respectives. Ici, une organisation est d'autant plus efficace que les temps effectifs d'élaboration des trajectoires se rapprochent d'un temps théorique. Ce critère d'évaluation de la coopération est élémentaire mais il a permis d'expérimenter l'utilité de l'échange d'information de contrôle pour la coordination.

Exemple : lorsqu'un agent est inactif (e. g. il n'y a pas de véhicule dans sa zone de perception), il le signale aux autres agents qui peuvent ainsi se décharger d'une part de leurs tâches de traitement de données sur lui. Il peut aussi jouer le rôle de module de fusion pour les trajectoires partielles. Un autre intérêt de l'échange d'information sur le contrôle local est que les agents peuvent planifier leurs tâches de façon plus efficace ou choisir les stratégies les mieux adaptées à la situation courante. Par exemple, deux agents élaborant une même trajectoire peuvent décider de traiter en priorité les plots proches

de leur frontière commune, fusionner les fragments ainsi obtenus, puis intégrer les autres points par la suite.

De manière générale les stratégies de prises de décision sont :

- préférer les actions qui réalisent de façon concurrente plusieurs buts ;
- préférer les actions qui requièrent le moins de ressources ;
- préférer les actions qui approuvent ou réfutent la poursuite de certains buts.

Remarques :

- la diffusion par un agent d'informations erronées ou obsolètes peut introduire des perturbations dans le système. Les concepteurs de DVMT ont ainsi généré des points aberrants (i. e. trajectoires «fantômes») dans certaines zones et mesuré à quel point leur incidence peut être néfaste sur les performances du système ; une solution pour résoudre ce problème a consisté à augmenter les capacités individuelles (i. e. discrimination plus efficace des points aberrants). L'amélioration des performances dans ce cas n'est pas due à un meilleur «travail d'équipe» mais plutôt au raffinement des stratégies locales de prise de décision.
- les agents modifient ou intégrent les «PGP» en fonction de leurs places dans la hiérarchie du système.

La figure 2.5 montre l'architecture du «Partial Global Planning» : les agents s'échangent et éventuellement modifient les «PGP» en fonction de leurs rôles et de leurs compétences, qui sont inscrits dans l'organisation du système.

2.2.4 Élaboration d'une conjecture commune

De façon informelle, nous appelons conjecture une structure de données commune à tous les agents. Il peut s'agir d'un plan d'actions, d'un diagnostic ou d'une situation élaborée à partir d'observations.

Ce dernier mode de coopération consiste à délibérer et s'accorder sur le contenu de la conjecture.

On peut comparer l'élaboration d'une conjecture avec les deux méthodes précédentes :

- l'organisation : ce sont des données statiques et prédéfinies (e. g. rôles ou compétences des agents) ;

- les informations de contrôle : ce sont des informations évolutives portant sur les décisions locales ou les buts à «moyen terme» ;
- la conjecture : elle concerne les actions ou les observations des agents c'est-à-dire des données à «court terme». Par exemple, les agents peuvent construire un plan tenant compte de la synchronisation des actions et l'accès concurrent aux ressources.

Remarque : nous avons décidé de regrouper un certain nombre de méthodes de coopération sous le terme d'élaboration d'une conjecture. À notre connaissance, ce choix est inédit. Le chapitre 3 est consacré à la formalisation *a posteriori* des caractéristiques génériques de ces méthodes.

Travaux précurseurs

▷ **Les systèmes ouverts :** en partant de la métaphore de la communauté scientifique (Kornfeld and Hewitt, 1981), C. Hewitt a élaboré le concept des *systèmes ouverts* (Hewitt, 1986) ou de processus d'organisation de la connaissance chez les êtres humains comme chez les machines, pour générer des informations pertinentes et fiables.

Les systèmes ouverts sont des réseaux d'agents qui raisonnent à partir de leur propre base de connaissances ou *micro-théories*. Pour C. Hewitt, une micro-théorie est une théorie mathématique qui décrit le modèle d'un système physique. Par définition, elle doit être cohérente d'un point de vue interne ; mais elle peut être incompatible avec une autre micro-théorie. De plus, dans les systèmes ouverts, l'activité des agents est concurrente et asynchrone, le contrôle est décentralisé et l'information globale est potentiellement incohérente. Comme les agents[3] n'ont pas accès à l'état ni aux opérations internes des autres agents, le problème qui se pose est celui de la prise de décision distribuée dans un contexte incertain, non déterministe et incohérent.

Selon C. Hewitt, la notion de systèmes ouverts est importante pour comprendre les systèmes de traitement de l'information répartis.

Il existe des travaux très proches de cette approche (Fiorino and Tessier, 1994; Tessier and Chaudron, 1996) : la «SuprA-cooperation» met en

3. Il est à noter que le mot agent n'apparaît pas dans l'article de C. Hewitt (Hewitt, 1986) qui utilise les notions de «office work» et de «due process». Pour C. Hewitt, l'entité de base est une organisation, c'est-à-dire un ensemble de connaissances, et ses processus de traitement de l'information qui coordonnent cette connaissance à l'intérieur et à l'extérieur de l'organisation par des communications asynchrones.

œuvre un mécanisme d'approbation/réfutation qui permet à deux agents de construire un espace de recherche commun. Contrairement aux gains habituels qu'apporte la coopération et que l'on peut mesurer quantitativement (e. g. rapidité d'exécution, nombre de solutions), l'objectif de la SuprA-cooperation est de raffiner qualitativement les solutions. Ce gain ne peut être mesuré quantitativement car le résultat est construit pour englober les théories initiales et ne peut donc pas être comparé à elles.

Nous avons développé un exemple de ce mécanisme dans le cadre du diagnostic de pannes (Fiorino and Tessier, 1994): un système physique est représenté par un modèle comportemental et un modèle fonctionnel. Le problème ici n'est pas de fournir un diagnostic à partir de la détection d'un symptôme de panne car les deux modèles le font, mais plutôt de combiner progressivement les deux points de vue pour arriver à une solution qu'aucun des modèles n'aurait «trouvé» autrement.

▷ **Partage de résultats :** dans un contexte distribué, où aucun agent n'a de vue globale sur le problème, les agents essaient par un processus itératif d'identifier les relations implicites entre leurs sous-problèmes : il s'agit d'une approche ascendante où le but est d'intégrer et d'analyser des données distribuées pour générer un modèle sémantique de ces données. HEARSAY-II (Erman et al., 1980), DVMT (Durfee and Lesser, 1991) sont des systèmes qui mettent en œuvre cette approche.

Les interactions négatives sont dues à des données erronées ou incomplètes qui induisent des actions incohérentes parmi les agents (e. g. un agent peut inutilement reproduire le travail d'un autre agent). Le but de la coopération est de converger vers un résultat globalement cohérent malgré des données initiales potentiellement incohérentes.

Pour cela, les agents s'échangent des tentatives de solutions partielles (i. e. hypothèses locales) et éventuellement incohérentes par rapport à une autre solution partielle. Dans de tels systèmes, plusieurs hypothèses incompatibles peuvent donc coexister à un moment donné. En s'échangeant leurs résultats, les agents découvrent ces hypothèses incompatibles, et si l'un d'entre eux dispose des informations nécessaires pour résoudre l'incohérence, l'hypothèse fausse est abandonnée et une hypothèse plus globale est formée à partir de l'ensemble des hypothèses compatibles.

Réciproquement, l'échange de solutions partielles peut empêcher un agent de produire une hypothèse incompatible et élaguer ainsi son espace de recherche. De ce fait, un agent peut prédire les caractéristiques de la

solution partielle et compatible à rechercher et donc augmenter les performances du système en se focalisant sur les pistes les plus prometteuses.

Élaboration d'une conjecture entendue comme planification des actions

Le principe de ces méthodes de planification repose sur l'utilisation d'un agent coordonnateur dont le rôle consiste à recueillir les plans locaux, à vérifier leur cohérence et à construire un plan global. Nous donnons deux exemples :

▷ **ATC - Air-Traffic Control :** S. Cammarata et son équipe se sont intéressés au problème de l'organisation du trafic aérien et plus particulièrement aux systèmes anti-collision (Cammarata et al., 1983). Dans cette approche, on suppose que chaque avion dispose d'un planificateur de trajectoire embarqué capable de communiquer avec celui des autres appareils. Différentes stratégies pour désigner le coordonnateur dans le groupe des agents ont été testées :

- dans la première stratégie, les appareils appliquent une convention commune qui leur permet de choisir le coordonnateur en fonction de certains paramètres (e.g position, vitesse et cap) ;
- la deuxième stratégie a pour but d'attribuer le rôle de coordonnateur à l'appareil le moins contraint : le facteur de contrainte d'un avion est un paramètre qui tient compte du nombre d'avions l'entourant, du carburant restant, de la distance par rapport à sa destination et de la saturation du réseau de communications ;
- dans la troisième stratégie, le coordonnateur est l'appareil le moins contraint et le mieux informé de la situation globale : un facteur de connaissance global est calculé en fonction du nombre de facteurs de contraintes qu'un agent a reçus.

D'après les conclusions de S. Cammarata, cette dernière stratégie est la meilleure dans la plupart des scénarios étudiés.

Remarque : ce résultat est moins intuitif qu'il n'y paraît. En effet, il existe des approches réactives à la coordination (Stirling, 1993; Zeghal and Ferber, 1993) c'est-à-dire que les agents n'ont que des capacités locales de contrôle (i. e. ils ne communiquent pas entre eux) et leurs décisions ne dépendent que des informations qu'ils perçoivent dans leur environnement. Dans ces systèmes, il n'y a ni coordonnateur, ni organisation explicite. Néanmoins, les agents agissent de façon cohérente.

▷ **Insertion incrémentale de plans :** Dans cette approche (Robert, 1996), plusieurs robots partagent une zone d'évolution commune. Un robot en-

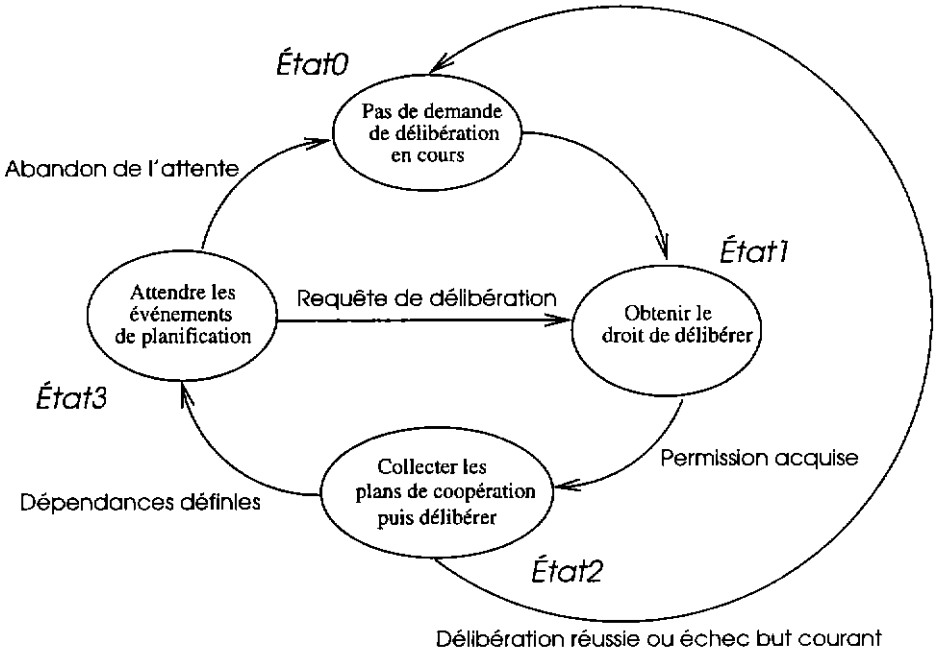


FIG. 2.6 – *Phase de délibération*

tremèle deux tâches, l'exécution de son plan courant et la planification à court terme de ses activités en fonction des activités des autres robots.

Pour cela (cf. figure 2.6 d'après (Robert, 1996)), le robot collecte tous les plans pour élaborer son propre plan (*État2*) : cette phase est appélée «délibération» car le robot tente d'insérer ses propres actions parmi celles des autres robots. La délibération peut aboutir à un échec si le but du robot est irréalisable (*État0*) ou si le robot est en attente d'un événement provenant d'un autre robot (*État3*). Pour entrer dans la phase de délibération, le robot doit s'assurer qu'un autre robot ne délibère pas (*État1*) : cette condition garantit la validité des plans collectés.

Conclusion

Nous avons postulé que les agents coopèrent ou négocient selon qu'ils ont des buts communs ou concurrents. Cette distinction nous a permis de classer les approches multiagents en deux catégories : la négociation et la coopération.

La négociation, fondée sur le paradigme de l'offre et la demande met en relation des agents qui cherchent à satisfaire leurs propres buts.

La coopération caractérise, au contraire, des agents dont le but est d'interagir de façon cohérente. À cause de son autonomie et de son point de vue partiel, un agent cherche à coordonner ses propres activités avec celles des autres agents.

Cela suppose que les agents partagent une connaissance commune relative aux rôles, aux compétences des uns et des autres et à l'organisation structurelle du système : ils peuvent ainsi, dans une certaine mesure «prédire» leurs comportements. Les agents peuvent aussi échanger des informations concernant leurs décisions locales et se coordonner de façon plus précise en élaborant une conjecture commune.

La chapitre 3 est consacré à la formalisation que nous proposons pour l'élaboration d'une conjecture commune par des agents coopérants.

Chapitre 3

Modèle de coopération pour l'élaboration d'une conjecture commune

Introduction

Nous avons vu qu'une des notions fondamentales de l'Intelligence Artificielle Distribuée est la notion de *cohérence* introduite par A. H. Bond et L. Gasser (Bond and Gasser, 1988) : on la définit comme une mesure du bon fonctionnement d'un système par rapport à un critère d'évaluation donné. Il en découle que la cohérence peut être évaluée de différentes façons : par exemple en termes de temps de calcul (Durfee et al., 1987), de qualité de la solution (Lander and Lesser, 1993; Khedro and Genesereth, 1994), d'états mentaux (Cohen and Levesque, 1990; Wooldridge, 1994; Wooldridge and Jennings, 1994).

L'objectif que nous nous sommes fixé est double : il s'agit d'une part de se demander ce que doit être un comportement cohérent pour un groupe de systèmes autonomes, et d'autre part, de proposer un modèle générique de fonctionnement de la coopération fondé sur cette définition de la cohérence et à finalité opératoire.

Le modèle que nous proposons caractérise le fonctionnement cohérent d'un ensemble d'entités autonomes du point de vue d'un superviseur et permet d'en évaluer la validité.

Le modèle ne se définit pas par rapport à une réalité observable : il ne cherche pas, par exemple, à formaliser la coopération dans les organisations sociales humaines : de nombreux travaux ont abordé ce sujet, mais nous ne cherchons pas à les prolonger. En revanche, il constitue une synthèse, c'est-

à-dire une formalisation *a posteriori*, de méthodes de coopération ayant été utilisées dans diverses plates-formes multiagents.

En effet, nous avons choisi de construire un modèle pour :

- la coopération d'agents autonomes devant réaliser une tâche collective : élaborer la situation courante de l'environnement (e. g. les agents sont des capteurs intelligents chargés de la surveillance d'un site sensible comme un aéroport), coordonner des actions (e. g. les agents sont des robots engagés dans une mission commune ou devant partager un environnement commun),
- la supervision de ces agents par un opérateur humain (cf. figure 3.1 d'après le projet CATA - Control Automation and Tasks Allocation de l'US Air Force) ou un système de contrôle, qui doivent pouvoir connaître à chaque instant l'état du groupe d'agents.

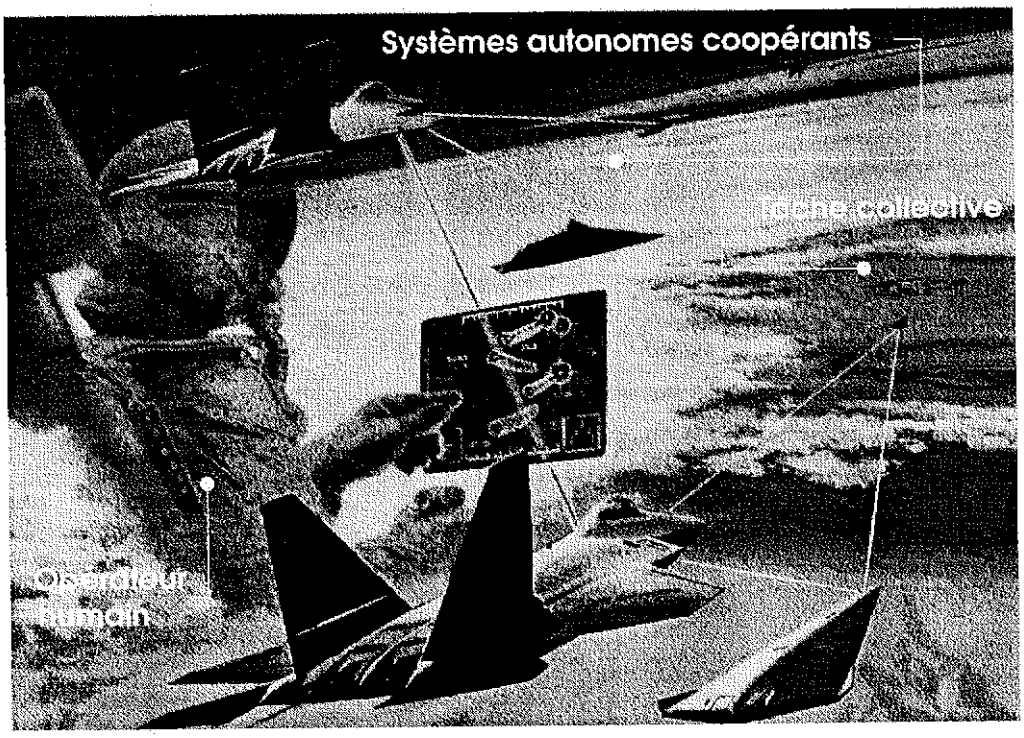


FIG. 3.1 – *Gnoptères en soutien à un avion de chasse*

Par conséquent, coopérer signifie ici délibérer, s'accorder sur la manière d'accomplir cette tâche collective. Le but des agents est de trouver un accord, c'est-à-dire une *conjecture commune* portant sur un plan d'actions à exécuter ou sur la situation courante de leur environnement. Le terme «conjecture» est

ici entendu comme «opinion ou supposition fondée sur des justifications reconnues comme insuffisantes» (Mazur, 1997).

Dans notre modèle, les agents n'ont donc qu'*un seul but*: élaborer conjointement une conjecture commune. Cependant, chaque agent dispose localement de son propre point de vue et le protocole représente la dynamique de la procédure d'élaboration qui permet aux agents de construire la conjecture commune à partir de leurs connaissances.

Par ailleurs, la cohérence de cette procédure d'élaboration est elle-même évaluée d'un point de vue global par une entité que nous avons appelée le *greffier*: l'état global du système se détermine par rapport à l'état de chaque agent. Le modèle définit les composantes de l'état global et identifie précisément les fonctions qui doivent être mises en place pour l'évaluer à partir des états locaux.

Le modèle de coopération se place donc sur deux plans : le premier caractérise les mécanismes internes à l'agent pour mettre à jour son état local et son propre point de vue *sur* la conjecture commune, c'est-à-dire en fonction du point de vue global à un instant donné et des connaissances de l'agent. En effet, la conjecture ne peut pas être envisagée comme la simple concaténation des points de vue locaux et indépendants. Nous avons modélisé une dynamique, qui, à partir de l'échange de points de vue locaux, permet d'élaborer *itérativement* une conjecture commune.
Le second plan caractérise les interactions entre agents et le rôle du greffier.

Pour concevoir ce modèle, nous avons cherché un paradigme permettant de formaliser la coopération d'agents devant réaliser une tâche collective. L'intérêt d'un paradigme est qu'il constitue un modèle de mise en œuvre facilement compréhensible. Cela explique sans doute pourquoi leur utilisation pour caractériser le fonctionnement de machines n'est pas rare : ainsi l'un des plus connus pour la programmation parallèle est le paradigme maître-esclave. Concernant la résolution distribuée de tâches, le protocole *contract net* (Smith and Davis, 1981) est, quant à lui, fondé sur le paradigme du marché dans lequel des agents émettent des appels d'offres représentant des tâches à réaliser tandis que d'autres y répondent.

Pour notre part, nous avons utilisé pour formaliser l'élaboration d'une conjecture commune par des agents coopérants le paradigme de la communauté scientifique et plus spécifiquement le principe de *preuve-réfutation*.

3.1 Le cycle de Lakatos comme paradigme de coopération

Le modèle de coopération s'inspire fortement des idées de I. Lakatos (Lakatos, 1984) pour lequel l'activité scientifique se limite à la création de théories représentant des observations et à leur renforcement ou à leur réfutation : ces processus sont *la conjecture et la réfutation.* Ils sont mis en œuvre par trois types d'agents différents (Kornfeld and Hewitt, 1981), ceux qui font des propositions, ceux qui les complètent et ceux qui les testent. Ce dernier «rôle» est tenu par des agents sceptiques qui cherchent des *anomalies* dans la théorie, c'est-à-dire des conséquences logiques de la théorie non vérifiées ou contredites. La théorie est alors modifiée afin de corriger ces anomalies ; il s'agit du processus d'*ajustement.*

Les travaux de I. Lakatos se situent dans la continuité de ceux de K. Popper (Popper, 1973) : la connaissance scientifique est en évolution permanente et toutes les théories *scientifiques* doivent être sujettes à la réfutation. Sa philosophie des sciences est fondée sur cette observation : il existe une «dissymétrie» entre la démonstration logique d'une théorie et sa réfutation. En effet, les observations ne peuvent pas impliquer logiquement une théorie mais elles peuvent la réfuter. Une illustration classique de ce principe est celui de l'hypothèse que «tous les cygnes sont blancs» ; de toutes les observations de cygnes blancs, il n'est pas possible de conclure que l'hypothèse «tous les cygnes sont blancs» est vraie. En revanche, il suffit d'une et une seule observation d'un cygne noir pour qu'elle soit contredite. Pour K. Popper, l'activité scientifique consiste donc à déduire toutes les implications logiques d'une théorie et à les confronter avec les observations ; l'admissibilité d'une théorie est liée à sa capacité à infirmer les tentatives de réfutation. Ainsi, une conjecture classique dans le domaine de la complexité algorithmique considère que la classe des problèmes admettant un algorithme déterministe de décision[1] avec un coût polynomial n'est pas égale à la classe de problèmes pour lesquels il existe un algorithme non-déterministe avec un coût polynomial ($P \neq NP$). Bien qu'il n'y ait aucune raison théorique de croire en cette conjecture, la communauté scientifique la considère comme vraie parce que beaucoup de personnes ont essayé de la réfuter et ont échoué.

Plus spécifiquement, la thèse de Popper semble s'appliquer à toutes les sciences sauf les mathématiques. En effet, les vérités mathématiques sont considérées comme infaillibles. Les théorèmes découlent d'un ensemble d'axiomes

1. On appelle classe P, la classe des problèmes pour lesquels il existe un polynôme $p(x)$, un algorithme déterministe et une question bien définie tels que pour chaque instance α du problème dont la réponse à cette question appartient à l'ensemble $\{oui, non\}$, l'algorithme tourne en un temps borné par $p(\alpha)$.

a priori «vrais» grâce à un enchaînement de règles formelles d'inférence c'est-à-dire une démonstration. Selon la philosophie formaliste des mathématiques dont D. Hilbert a été l'un des représentants, grâce à nos capacités de déduction, nous pouvons atteindre toutes les vérités mathématiques et savoir que nous les avons atteintes. Irrité par les contradictions du socle logique des mathématiques («*crise des fondements*»), D. Hilbert définit un *programme* de recherches devant répondre entre autres choses à la question suivante : existe-t-il une syntaxe cohérente et non contradictoire, c'est-à-dire un ensemble de règles faisant abstraction de la sémantique des formules et permettant de faire des mathématiques de façon mécanique ? K. Gödel montre qu'il n'existe aucune méthode permettant de distinguer systématiquement toutes les assertions vraies des assertions fausses d'un système axiomatique cohérent, et suffisamment riche pour contenir l'arithmétique (théorème d'incomplétude).

Mais, I. Lakatos se démarque radicalement de l'approche mécanique des mathématiques en se plaçant sur le terrain de l'épistémologie. Son projet est «d'étudier en détail la thèse selon laquelle les mathématiques non formelles, quasi empiriques, ne se développent pas dans un accroissement continu du nombre des théorèmes indubitablement établis, mais dans l'amélioration incessante des conjectures grâce à la spéculation et à la critique, grâce à la logique des preuves et réfutations» (Lakatos, 1984). Il introduit donc la notion de théorie empiriste qui se distingue de la notion «classique» de théorie (au sens de Hilbert ou Euclide) car elle tient compte de l'impossibilité pratique d'énoncer l'ensemble des hypothèses initiales ou axiomes fondant les déductions dans une théorie classique (Sallantin et al., 1991). Finalement, une théorie n'est classique que si on la suppose comme telle. Pour I. Lakatos, une théorie n'est évaluable que comparativement à d'autres théories de même type, et ses travaux ont eu pour but de formaliser les principes de comparaison entre ces théories.

Dans cette approche, un résultat mathématique est considéré comme une conjecture dont la preuve, c'est-à-dire la «démonstration», est soumise, comme dans les sciences empiriques, à une expérience ; elle est testée par la communauté des mathématiciens qui tente de la réfuter. Si l'expérience réussit, la démonstration est acceptée et la conjecture est considérée comme un théorème qui peut être enrichi par de nombreux corollaires et exemples. Si un contre-exemple apparaît, l'expérience échoue et la conjecture doit être ajustée : dans ce cas, soit le contre-exemple devient une exception, soit on recherche dans la théorie le «lemme coupable» de la contradiction et on essaie de le rectifier (Chaudron and Tessier, 1995; Tessier and Chaudron, 1996). La principale contribution de Lakatos est d'avoir conceptualisé ce cycle de conjecture et réfutation (cf. figure 3.2) fondé sur des critères relatifs d'acceptabilité des théories.

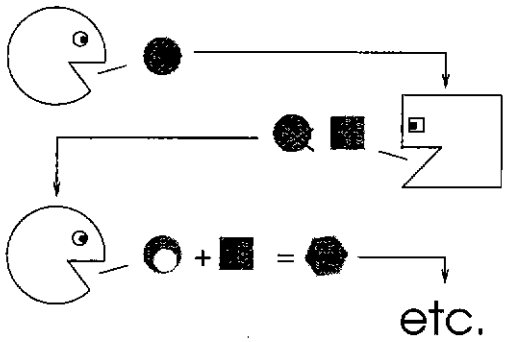


FIG. 3.2 – *Paradigme Preuve - Réfutation*

L'idée d'utiliser la métaphore de la communauté scientifique comme principe d'interaction entre des entités distribuées n'est pas neuve (Kornfeld and Hewitt, 1981). Mais elle semble avoir été abandonnée au profit d'approches fondées sur des mécanismes de négociation. Nous expliquons pourquoi nous avons préféré les principes d'élaboration constructive d'une conjecture à ceux de la recherche d'un accord maximisant la satisfaction de chaque agent. Ces principes sont formalisés afin de constituer un modèle opératoire pour la réalisation de systèmes coopératifs.

3.2 Objectifs du modèle

3.2.1 Classe d'applications visées par le modèle

Notre objectif est de définir un modèle de coopération entre des systèmes autonomes. Les principales caractéristiques des applications visées par ce modèle sont les suivantes :

- plusieurs entités - appelées *agents* - sont physiquement réparties dans l'environnement dont ils ont une perception limitée. Certaines peuvent s'y déplacer (robots transportant des containers, gnoptères en mission de reconnaissance); d'autres sont immobiles (systèmes de surveillance de sites sensibles, relais de télécommunications) ; dans tous les cas, les agents observent leur environnement grâce à des capteurs afin d'élaborer et de mettre à jour leurs représentations du monde réel (Tessier, 1997).

- les agents peuvent éventuellement agir sur l'environnement. Dans ce cas, ils doivent en permanence prendre des décisions concernant l'action adéquate à entreprendre compte tenu de leur connaissance courante de l'état du monde réel.

- l'environnement évolue au cours du temps ; compte tenu de la perception limitée des agents, on peut donc en conclure qu'ils ont une connaissance incertaine voire erronée de leur environnement. Ils sont donc confrontés à un dilemme (Fabiani, 1996; Barrouil et al., 1998) : observer l'environnement pour maintenir une représentation aussi juste que possible du monde réel ou choisir une action en courant le risque de prendre une décision inadaptée.

- les agents ont été conçus pour réaliser une tâche collective comme l'élaboration de situation ou l'élaboration d'un plan commun : les agents ne sont donc pas *a priori* antagonistes ; ils ont le même but qui est d'élaborer une représentation commune. Dans le cas de l'élaboration de situation, cette représentation commune définit la *croyance* du groupe d'agents sur l'état du monde : à partir de leurs connaissances locales, ils construisent une croyance globale et cohérente. Dans le cas de l'élaboration d'un plan commun, les agents se partagent et coordonnent les tâches à réaliser en fonction des capacités et des ressources dont ils disposent. Dans ces deux cas, les agents doivent résoudre des conflits concernant soit des ressources partagées (par exemple, des robots devant simultanément utiliser le même passage), soit des croyances incohérentes. Mais les mécanismes de coopération sont fondés sur le principe de la recherche d'une solution satisfaisante par rapport à des critères locaux *et* globaux.

- les agents ne sont pas utilisés dans le cadre de la résolution «hors ligne» d'un problème (comme par exemple un problème de satisfaction de contraintes (Lemaitre and Verfaillie, 1996; Khedro and Genesereth, 1994; Conry et al., 1991)). Comme ils sont situés dans un environnement dynamique, une représentation commune peut être remise en cause par l'arrivée d'une nouvelle information. Le modèle doit donc spécifier le cycle de construction et de révision de cette représentation. Pour ce type d'applications, la notion de convergence vers une solution ne correspond pas exactement à celle des problèmes de résolution. On peut en fait distinguer deux cycles de traitement des informations pour chaque agent : le premier concerne la mise à jour de sa base de connaissances en fonction des informations issues de ses capteurs ; le second met à jour la représentation commune. Nous ne nous occupons pas du problème de la révision de la base de connaissances d'un agent en fonction des messages envoyés par d'autres agents (Djeffal and Chaib-Draa, 1998; van Eijk and de Boer, 1998) : notre approche est centrée sur le problème de l'élaboration d'une conjecture commune en fonction des connaissances locales.

- le mode de coopération que nous voulons modéliser est celui d'agents élaborant un point de vue commun dans le cadre de systèmes autonomes cherchant à coordonner leurs actions au cours d'une mission, ou de cap-

teurs intelligents élaborant une situation pour un système d'aide à la décision. De tels agents doivent donc être réactifs à tout changement de l'état du monde.

Dans ce contexte, il est difficilement envisageable d'utiliser un coordonnateur pour construire ce point de vue commun. Les principales critiques qui sont faites à ce type d'architectures portent sur le risque d'engorgement des communications, sur le manque de robustesse par rapport aux dysfonctionnements du coordonnateur, ou sur la difficulté d'adapter ce principe de mise en œuvre à des organisations préexistantes et naturellement distribuées (par exemple, une patrouille d'avions de chasse assistée par des gnoptères).

3.2.2 Place du modèle par rapport aux travaux existants

Comme nous l'avons vu dans les chapitre précédents, le thème de la coopération entre agents est au cœur de nombreuses recherches en Intelligence Artificielle Distribuée, parmi lesquelles on peut dégager principalement deux axes : les approches centrées sur les problèmes de résolution distribuée (i. e. les méthodes de résolution) et les approches centrées sur la spécification des interactions (i. e. les modèles de comportement).

Notre modèle de coopération est pertinent pour une certaine classe d'applications. Mais, en plus d'être en adéquation avec cette classe d'applications, nous souhaitons qu'il prescrive des principes de mise en œuvre. Il indique ce que doivent faire les agents dans certaines situations : un agent est donc hétéronome du point de vue de ses interactions avec les autres agents du système.

Notre contribution se situe entre les modèles cognitifs et les méthodes dédiées de coopération (cf. figure 3.3) :

- d'une part, nous ne proposons pas une nouvelle méthode de négociation pour la résolution de conflits ou le partage de tâches. De nombreuses plates-formes de développement de systèmes multiagents intègrent des outils permettant d'implémenter ce type de méthodes (Gauvin, 1995; Gutknecht and Ferber, 1998). Nous avons vu que dans les applications qui nous intéressent, les agents échangent des connaissances locales pour construire un point de vue commun et non un accord entre entités concurrentes. Les protocoles de négociation ne sont adaptés à ces applications. En effet, les interactions se produisant dans ces groupes de systèmes autonomes ne peuvent pas faire l'objet de négociations car ces méthodes

impliquent un rapport de force au sein du groupe (Sycara, 1989) ce qui est contradictoire avec les objectifs du concepteur. On ne peut pas non plus considérer que ces agents sont des modules de résolution d'un problème «hors ligne». Situés dans un environnement dynamique, ces agents reçoivent en permanence de nouvelles informations de leurs capteurs. Par conséquent, si à un instant donné les agents convergent vers un résultat commun, celui-ci peut être remis en cause par la moindre mise à jour de la connaissance locale d'un agent.

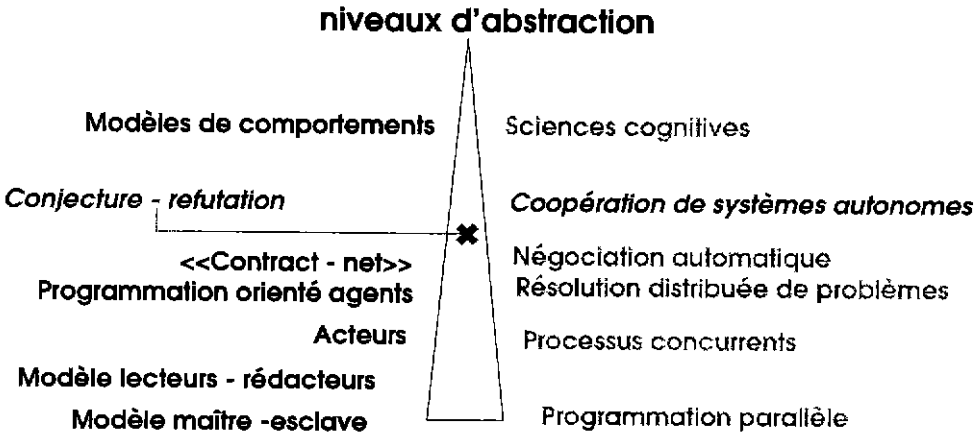


FIG. 3.3 – *Place du modèle par rapport aux paradigmes de l'IAD*

- d'autre part, les modèles de coopération cognitifs reposent sur des concepts trop abstraits pour être mis en œuvre ou difficilement transposables pour des machines. Ces modèles n'ont d'ailleurs pas pour vocation d'être utilisés à court terme par des concepteurs de systèmes multiagents car les notions qu'ils tentent de formaliser (e. g. intention collective de faire quelque chose, croyances mutuelles) ne font pas encore l'objet d'un consensus et restent un sujet de recherche important.

3.2.3 Caractéristiques du modèle

Nous avons fondé le modèle sur l'idée que deux types de fonctionnalités sont nécessaires pour concevoir des agents coopératifs : (1) les fonctionnalités définissant la coopération en elle-même, et (2) les fonctionnalités liées à la mise en œuvre de la coopération.

Les objectifs du modèle sont les suivants :

(1) Garantir les propriétés suivantes :

- propriétés de base :
  - atomicité de la conjecture : à chaque instant, il n'y a qu'une seule instance de la conjecture ;
  - pas d'agent affamé : la conjecture est une ressource à laquelle les agents peuvent tous avoir accès.
- réactivité des agents : les agents réagissent aux changements d'état de l'environnement ;
- observabilité : à chaque instant, une instance de la conjecture et l'état global du système sont accessibles à un opérateur humain.

(2) Spécifier les fonctionnalités coopératives (éléments *génériques* du modèle) :

- notion d'autonomie : le comportement que doit avoir chaque agent localement pour participer à la tâche collective ; son but, les décisions qu'il peut prendre et les actions qui en découlent ;
- notion d'hétéronomie : le comportement de chaque agent vis-à-vis du groupe, ce à quoi il doit s'engager ;
- notion d'interaction : les flux d'information circulant entre les agents ;
- notion de cohérence : un critère global permettant d'évaluer la coopération. Ce critère concerne le résultat courant de la tâche de coopération, son statut vis-à-vis du groupe et le statut du groupe par rapport à la coopération.

(3) Identifier les paramètres (éléments *spécifiques* du modèle) correspondant aux choix de mise en œuvre du système coopératif :

- les critères de décision et les méthodes d'élaboration propres à l'agent ;
- les méthodes de mise à jour des bases de connaissances locales ;
- le protocole de communication utilisé par les agents.

Nous avons choisi de concevoir un modèle de coopération qui ne spécifie que ce qui est directement lié à la coopération, c'est-à-dire *les règles définissant le comportement cohérent d'agents devant réaliser une tâche collective.* Les choix de mise en œuvre sont inclus sous la forme de paramètres à instancier conformément à une application donnée. De ce point de vue, ce modèle n'est ni un nouveau protocole de négociation, ni la caractérisation de comportements sociaux d'agents cognitifs.

Le modèle est donc composé de deux sortes d'éléments :

- les éléments spécifiant la notion de coopération :
- les éléments dépendant de choix de mise en œuvre.

3.3 Modèle de coopération

3.3.1 Composition

D'un point de vue fonctionnel, le modèle comprend différents composants, les agents et un greffier, ayant les caractéristiques suivantes :

- les agents ont le même but qui est d'élaborer un point de vue commun concernant une tâche collective à réaliser : on appelle *conjecture commune* ce point de vue. Concrètement, la conjecture commune est un plan d'actions ou une base de connaissances représentant une situation.
- les agents ont été conçus pour participer à une action de groupe et ils ne poursuivent donc pas d'objectifs individuels : ils doivent se conformer aux règles régissant ce groupe que nous qualifions de *fédération* d'agents. Ces règles ont été choisies afin de garantir un comportement *cohérent* de la fédération.
- les agents doivent annoncer leur décision de poursuivre le but commun ; ils s'engagent donc vis-à-vis de la fédération.
- l'état de la fédération est évaluée par une fonction globale appelée *greffier*, qui enregistre les engagements et évalue le statut de la conjecture commune. Il faut souligner que le terme de greffier dénote une fonction globale alors que celui d'agent indique un ensemble de fonctions locales et génériques : la fonction de greffier est «passive» c'est-à-dire qu'elle n'implique aucune procédure de décision ; elle peut donc être remplie par un seul agent ou être répartie sur plusieurs agents, en fonction des stratégies de mise en œuvre.

3.3.2 Formalisme de représentation

Le formalisme retenu pour représenter notre modèle est celui des réseaux de Petri coloré (cf. annexe D). Ce choix des réseaux de Petri a été motivé par différentes considérations :

- les réseaux de Petri permettent de facilement prendre en compte et visualiser le parallélisme ou le séquencement d'activités à l'intérieur d'un

système. De même, la synchronisation entre les différentes activités (les états et les événements, ainsi que leurs relations) est modélisée de façon explicite ;

- ils permettent de définir un modèle conforme aux caractéristiques attendues pour la coopération, car elles peuvent être démontrées (marquages atteignables, vivacité) ;
- les logiciels de réseaux de Petri permettent de mettre en œuvre le modèle sans avoir à le traduire dans un autre formalisme ;
- ce formalisme est utilisé pour la commande de processus (métro (Dhalluin et al., 1987), chaînes de production (Zhou and Dicesare, 1989)).

3.3.3 Définition

Nous introduisons à présent les définitions et les postulats du modèle de coopération.

Soit $F = \{\alpha_1, \ldots, \alpha_m\}$ une fédération de m agents devant réaliser une tâche collective.

Ces agents élaborent une conjecture commune à partir d'un langage commun.

Formellement,

Définition 3.1 *Soit $\mathcal{V}$ un alphabet. On note $\mathcal{P}(\mathcal{V})$ l'ensemble des parties de $\mathcal{V}$. Soit $\mathcal{L} \subseteq \mathcal{P}(\mathcal{V})$ le langage de la fédération F et $Conject = \mathcal{P}(\mathcal{L})$ l'ensemble de toutes les conjectures pouvant être formées à partir de $\mathcal{L}$.*

La $\boxed{\textit{conjecture } c_n}$ *est un élément de $Conject$: n est un indice sur la suite des conjectures qui sont générées par les agents.*

Remarque : précisons que les agents n'utilisent pas nécessairement tous le même langage local (e. g. un agent peut raisonner avec un langage logique alors qu'un autre manipulera des Grafcets (Fiorino and Tessier, 1994)) ; en revanche, les agents doivent avoir un langage $\mathcal{L}$ en commun afin d'élaborer une conjecture commune ($\mathcal{L}$ doit donc être traduit localement par les agents).

Par ailleurs,

Postulat 3.1 *La conjecture est une ressource non partageable : lorsqu'un agent y accède, il exclut tous les autres.*

Le modèle de coopération se définit alors de la manière suivante :

Définition 3.2 *Le* $\boxed{\textit{modèle de coopération}}$ *est un triplet (F,$\mathcal{L}$,CooPN), où CooPN est un réseau de Petri coloré et interprété ayant pour ensemble de couleurs :*

$$Col = F \times \{acceptable, unacceptable\} \cup F \times \{changed, unchanged\} \cup F \times \{committed, uncommitted\} \cup F \cup \{\bullet\}$$

• *représente le jeton Logic, c'est-à-dire le jeton classique des réseaux de Petri (cf. annexe D).*

La sémantique des couleurs et des variables est définie dans la suite de ce chapitre.

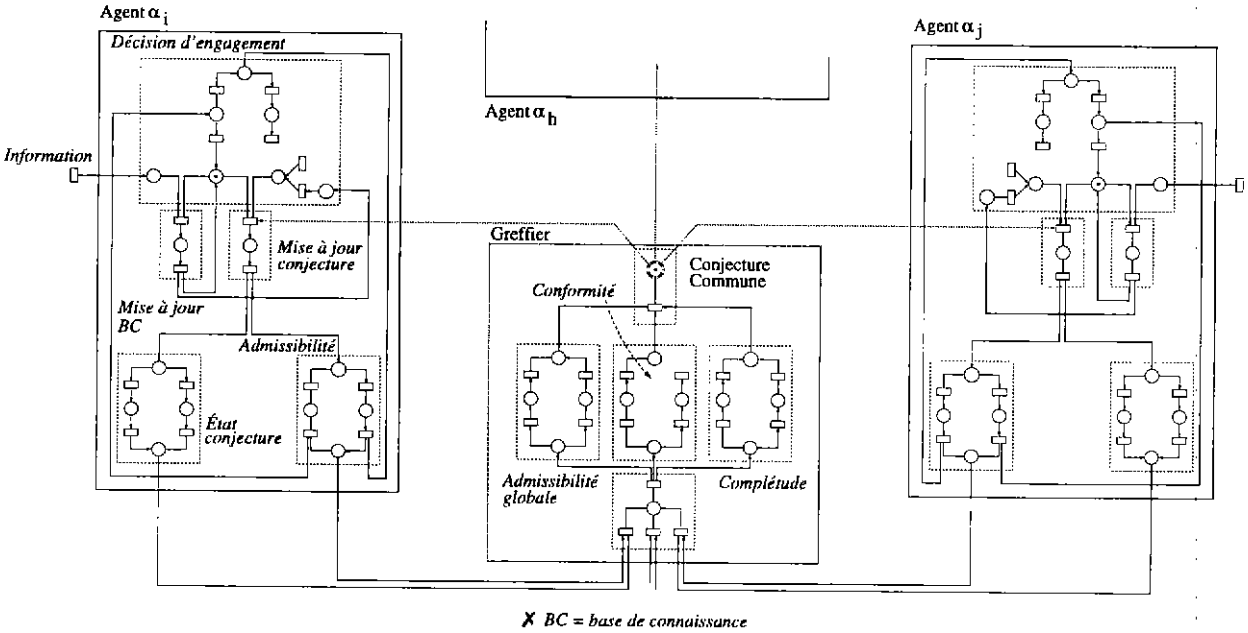


FIG. 3.4 – *Vue fonctionnelle du modèle*

La figure 3.4 donne un aperçu du modèle de coopération : il est modulaire car on peut ajouter ou retirer des agents sans modifier globalement sa structure globale (cf. figure 3.5). La conjecture est représentée comme une ressource partagée qui est disponible lorsque la place correspondante est marquée avec un jeton Logic (i. e. •).

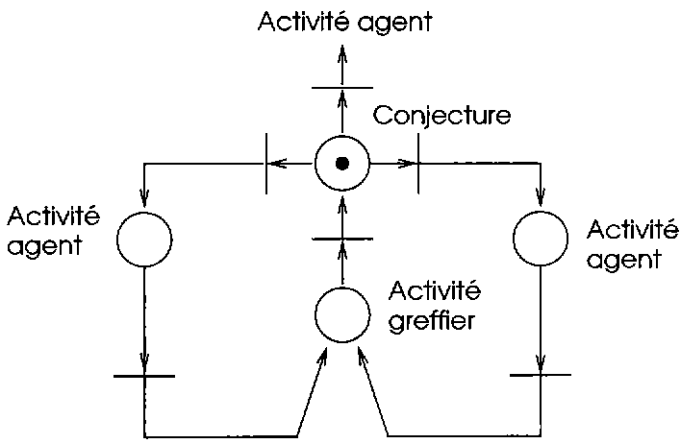


FIG. 3.5 – *Structure globale du modèle*

3.4 L'agent coopératif

3.4.1 Principe

Un agent reçoit trois types d'informations provenant de l'extérieur : les informations issues de ses propres capteurs, la conjecture courante et des informations envoyées par le greffier concernant l'état de la fédération. Les sorties de l'agent sont son état local renvoyé au greffier et la conjecture mise à jour le cas échéant.

Du point de vue de l'élaboration coopérative d'une conjecture commune, les activités internes de l'agent concernent d'une part, la mise à jour de sa base de connaissances, et d'autre part, la mise à jour de la conjecture à partir de ces connaissances. L'activité de mise à jour de la conjecture est un cycle comportant trois étapes élémentaires (cf. figure 3.6) :

- **Choix de l'activité** : l'agent choisit soit de (1) mettre à jour sa base de connaissances, soit de (2) mettre à jour la conjecture courante. Ce choix dépend du type d'information que l'agent a reçue.

  (1) lorsqu'il reçoit une information de ses capteurs, il met à jour sa base de connaissances, qui peut contenir des croyances sur l'état de l'environnement ou sur les croyances des autres agents. Toutes ces croyances peuvent évoluer suivant deux principes classiques de mise à jour des bases de connaissances : maintien de la cohérence et changement minimal (Garcia, 1993; Katsuno and Mendelzon, 1991a). En revanche, on ne retient pas le principe qui consiste à privilégier l'information par rapport à la base de connaissances : l'agent garde la

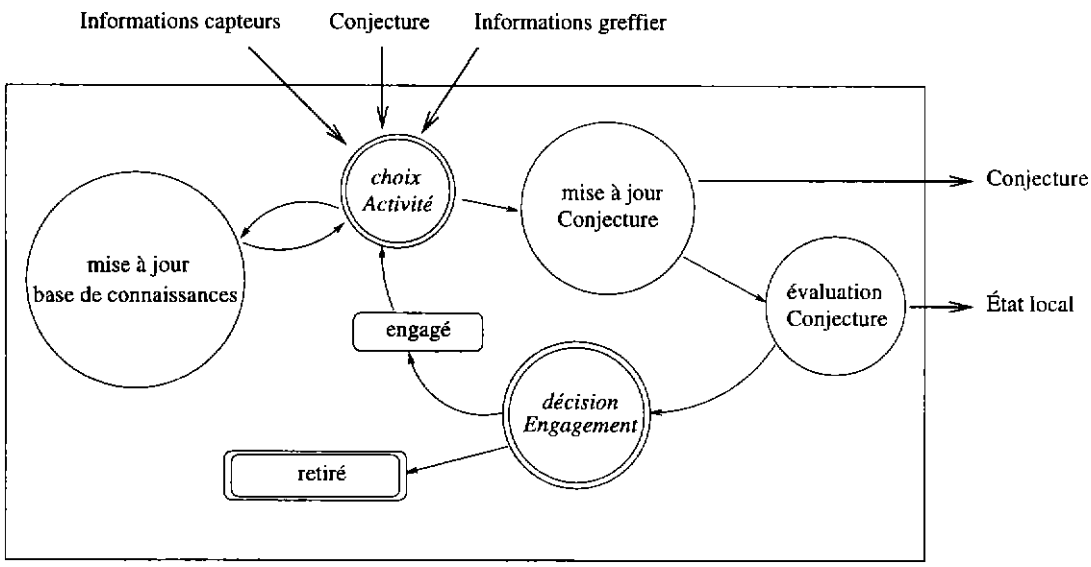


FIG. 3.6 – *Vue fonctionnelle des activités locales de l'agent*

possibilité de rejeter une information qu'il juge erronée.

Les lois physiques (par exemple : non ubiquité des objets, valeurs admissibles des variables d'état), le modèle d'évolution de l'environnement, et les capacités intrinsèques d'un agent lui permettant de raisonner sur ce qu'il est capable de faire, sont représentés dans la base de connaissances sous la forme de *contraintes du domaine* qui ne sont pas sujettes à la mise à jour.

(2) la mise à jour de la conjecture courante n'est possible que si au moins l'une des conditions suivantes est vérifiée : l'agent a mis à jour sa base de connaissances ou la conjecture a été modifiée par un autre agent (cette information est délivrée par le greffier). La mise à jour de la conjecture par l'agent est fondée sur un cycle de preuve/réfutation dont le seul but est de rendre la conjecture *admissible* par rapport aux critères de l'agent. La conjecture n'est modifiée qu'à cette seule condition.

- **Évaluation de l'état de la conjecture** : l'état local est constitué par (1) l'évaluation de l'état de la conjecture représenté par les variables booléennes LocalAcceptability et LocalChanged, et par (2) la décision de l'agent concernant son engagement dans la fédération représenté par la variable LocalCommitment. Ces variables sont mises à jour chaque fois que l'agent accède à la conjecture. L'état local de chaque agent est accessible au greffier, mais aucun agent n'a accès directement à l'état d'un autre agent.

L'agent évalue l'admissibilité de la conjecture par rapport à ses critères propres, qui dépendent de l'application. Le modèle spécifie que, lorsque la conjecture a été modifiée par un agent, tous les autres agents doivent évaluer l'admissibilité de la conjecture ainsi modifiée ; si l'un d'eux considère qu'elle est *non admissible* (LocalAcceptability=*faux*), il a la possibilité de la changer à son tour à condition que cette modification la rende *admissible* (LocalAcceptability=*vrai*). De même, si la conjecture est de son point de vue admissible, il peut la modifier dans la mesure où elle le demeure.

D'autre part, l'agent détermine le statut de la conjecture c'est-à-dire qu'il enregistre dans son état si la conjecture a été *changée* par lui (LocalChanged=*vrai*) ou si elle est restée *inchangée* (LocalChanged=*faux*).

La possibilité donnée aux agents de modifier la conjecture chaque fois qu'elle a été changée par un autre membre de la fédération pose le problème de la convergence vers une conjecture stable. Cette question ne peut être que partiellement abordée par le modèle de coopération parce qu'elle dépend des critères locaux et du flux des informations provenant des capteurs. Cependant, le modèle traite le problème du fonctionnement *cohérent* de la fédération pour l'élaboration d'une conjecture commune.

- **Décision sur le maintien de l'engagement** : comme il est impossible de garantir un comportement coordonné de la fédération sans la connaissance des agents fédérés et de leurs capacités, la notion d'engagement des agents a été introduite dans le modèle. Un agent doit décider de son adhésion à la fédération chaque fois qu'il accède à la conjecture. Il ne peut donc pas se retirer de façon intempestive. De plus, un agent ne peut abandonner la fédération que s'il lui est impossible de rendre admissible une conjecture qu'il considérait comme non admissible. Par conséquent, il ne peut pas se retirer si la conjecture est jugée admissible. Par ailleurs, nous avons choisi de rendre le retrait d'un agent irrévocable.

  L'agent est donc, selon les circonstances, *engagé* ou *retiré* : ainsi, les conditions de convergence d'une instance du modèle (i. e. pour une application donnée), se limitent, si on suppose qu'aucune information n'entre dans le système, à l'étude de paramètres déterministes (i. e. les critères de décision, l'état de la base de connaissances à un instant donné, les méthodes propres à chaque agent).

Le tableau 3.1 récapitule les variables d'état locales.

Variables	Sémantiques
LocalAcceptability	conjecture *admissible* ou *non admissible*
LocalChanged	conjecture *changée* ou *inchangée*
LocalCommitment	agent *engagé* ou *retiré*

TAB. 3.1 – *Variables d'état locales*

L'état initial d'un agent est *engagé, admissible* et *inchangé.*

Les trois fonctions génériques, mise à jour de la conjecture, mise à jour de la base de connaissances et décision d'engagement sont successivement détaillées.

3.4.2 Mise à jour de la conjecture commune

Ce module est au cœur du modèle car il définit le mécanisme de preuve-réfutation au niveau de chaque agent.

Avant de voir comment il fonctionne, nous définissons les opérateurs élémentaires qui sont utilisés.

Remarque : tous les opérateurs devraient être indicés par α_i car ils sont propres à chaque agent. Cependant, nous ne les faisons pas apparaître afin de ne pas surcharger la présentation.

Soit $K_t \subseteq Conject$ la base de connaissances à l'instant t de l'agent α_i : t représente le temps fourni par une horloge commune à tous les agents.

Cet agent applique sur la conjecture des opérateurs d'évaluation et de transformation.

Définition 3.3 *Soit* $\boxed{accept}$ *l'opérateur déterminant l'admissibilité des éléments de la conjecture.*

$$accept : \ Conject \rightarrow \{vrai, faux\}$$

et, pour toute conjecture c_n*, qui, par définition, est un élément de l'ensemble des parties du langage :*

$$\boxed{Admissible(c_n)} = \{w \mid \forall w \in c_n, \ accept(w) = vrai\}$$

est l'ensemble des éléments de conjecture admissibles du point de vue de α_i.

Définition 3.4 *Soit* $\boxed{reject}$ *l'opérateur déterminant la non-admissibilité des éléments de la conjecture.*

$$reject : \ Conject \rightarrow \{vrai, faux\}$$

et, pour toute conjecture c_n,

$$\boxed{Non-Admissible(c_n)} = \{w | \forall w \in c_n,\ reject(w) = vrai\}$$

est l'ensemble des éléments de conjecture non admissibles du point de vue de α_i.

Le fait que les éléments w de la conjecture c_n soient localement admissibles ou non admissibles dépend de l'application et des critères de l'agent. Ainsi,

$$\Gamma = c_n \backslash Admissible(c_n) \cup Non-Admissible(c_n)$$

n'est pas nécessairement vide (cf. figure 3.7), ce qui signifie que l'agent peut être *indifférent* à certains éléments de la conjecture : en effet, il semble raisonnable de penser que les agents n'ont pas une opinion à chaque instant sur tous les éléments de la conjecture courante ; il est possible qu'un agent ne dispose pas à un instant donné des connaissances suffisantes pour décider de l'admissibilité d'un élément de la conjecture.

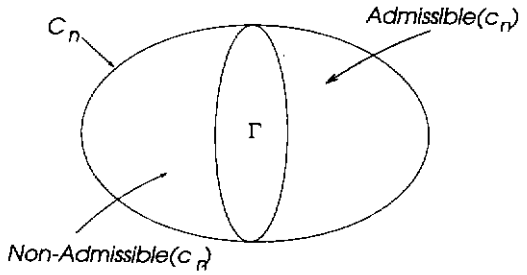


FIG. 3.7 – *Admissibilité d'une conjecture*

Exemple : supposons que $\mathcal{L}$ soit un langage propositionnel ; un critère possible pour déterminer si un élément w de c_n est admissible ou non est le suivant :

- un élément de la conjecture est admissible si l'agent est capable de le déduire de ses connaissances :

$$\forall w \in Admissible(c_n) \Rightarrow K_t \vdash w$$

- *a contrario*, un élément n'est pas admissible s'il est contradictoire avec elles :

$$\forall w \in Non-Admissible(c_n) \Rightarrow K_t \vdash \neg w$$

- l'agent est indifférent à un élément de la conjecture s'il ne peut pas le déduire de ses connaissances et s'il n'est pas contradictoire avec ses connaissances :

$$\forall w \in \Gamma \Rightarrow \begin{cases} K_t \cup \{w\} \text{ peut être satisfaite au sens usuel de la logique} \\ K_t \not\vdash w \end{cases}$$

Définition 3.5 *Soit* $\boxed{accept_conjecture}$ *la variable booléenne déterminant l'admissibilité de la conjecture :*

$$accept_conjecture : \ Conject \rightarrow \{vrai, faux\}$$

On impose que :

$$accept_conjecture(c_n) = vrai \Rightarrow Non-Admissible(c_n) = \emptyset$$

Dans le cas contraire $\boxed{reject_conjecture(c_n) = vrai}$.

Les opérateurs *accept* et *reject* permettent à un agent de décider s'il doit ou non modifier la conjecture courante. Il reste donc à définir les opérateurs de mise à jour. Pour cela, on utilise les opérateurs *add*, *retract* et *adjust*. On note c_{n+1} la conjecture mise à jour.

Postulat 3.2 *Nous postulons qu'un agent ne peut modifier la conjecture que si cette modification ne rend pas la conjecture non admissible.*

D'où les définitions suivantes :

Définition 3.6 *Soit* $\boxed{add}$ *l'opérateur qui permet à un agent* α_i *d'ajouter un élément à la conjecture.*

$$add : \ Conject \times \mathcal{P}(K_t) \rightarrow Conject$$

tel que $\forall c_n \in Conject$ *et* $\forall v \in \mathcal{P}(K_t)$,

$$add(c_n, v) = c_n \cup \{v\}$$

On impose que :

$$c_{n+1} = add(c_n, v) \Rightarrow Non-Admissible(c_{n+1}) = \emptyset$$

Définition 3.7 *Soit* $\boxed{retract}$ *l'opérateur qui permet à un agent* α_i *de retirer un élément de la conjecture.*

$$retract : \ Conject \times Conject \rightarrow Conject$$

tel que $\forall c_n \in Conject$ *et* $\forall w \in Conject$,

$$retract(c_n, w) = c_n \backslash \{w\}$$

Définition 3.8 *Soit* $\boxed{adjust}$ *l'opérateur qui permet à un agent* α_i *de modifier la conjecture.*

$$adjust : \ Conject \times Conject \rightarrow Conject$$

tel que $\forall c_n \in Conject$, $\forall w \in \mathcal{P}(c_n)$ *et* $\forall v \in \mathcal{P}(K_t)$,

$$c_{n+1} = adjust(c_n) = add(retract(c_n, w), v)$$

On impose que :

$$c_{n+1} = adjust(c_n) \Rightarrow Non-Admissible(c_{n+1}) = \emptyset$$

À partir de ces opérateurs, nous définissons les variables booléennes qui caractérisent la mise à jour locale de la conjecture.

Définition 3.9 *Soit* $\boxed{extend_conjecture}$, *la variable booléenne caractérisant l'enrichissement de la conjecture par l'agent* α_i *:*

$$extend_conjecture : \ Conject \rightarrow \{vrai, faux\}$$

telle que :

$$extend_conjecture(c_n) = vrai \Leftrightarrow \exists v \in \mathcal{P}(K_t), \ c_{n+1} = add(c_n, v)$$

Définition 3.10 *Soit* $\boxed{adjust_conjecture}$, *la variable booléenne caractérisant l'ajustement de la conjecture par l'agent* α_i *:*

$$adjust_conjecture : \ Conject \rightarrow \{vrai, faux\}$$

telle que :

$$adjust_conjecture(c_n) = vrai \Leftrightarrow \exists v \in \mathcal{P}(K_t), \ \exists w \in \mathcal{P}(c_n), \ c_{n+1} = adjust(c_n)$$

Voyons à présent comment fonctionne le module de mise à jour de la conjecture (cf. figure 3.8).

Lorsque l'agent accède à la conjecture, il évalue son admissibilité (place P0). S'il la trouve admissible ($accept_conjecture(c_n) = vrai$ et la place P2 est marquée), il peut soit l'améliorer ($extend_conjecture(c_n) = vrai$ et la place P3 est marquée), soit la laisser telle quelle. Les variables d'état LocalAcceptability et LocalChanged prennent respectivement les valeurs *vrai* et *vrai* dans le premier cas, et *vrai* et *faux* dans le second.

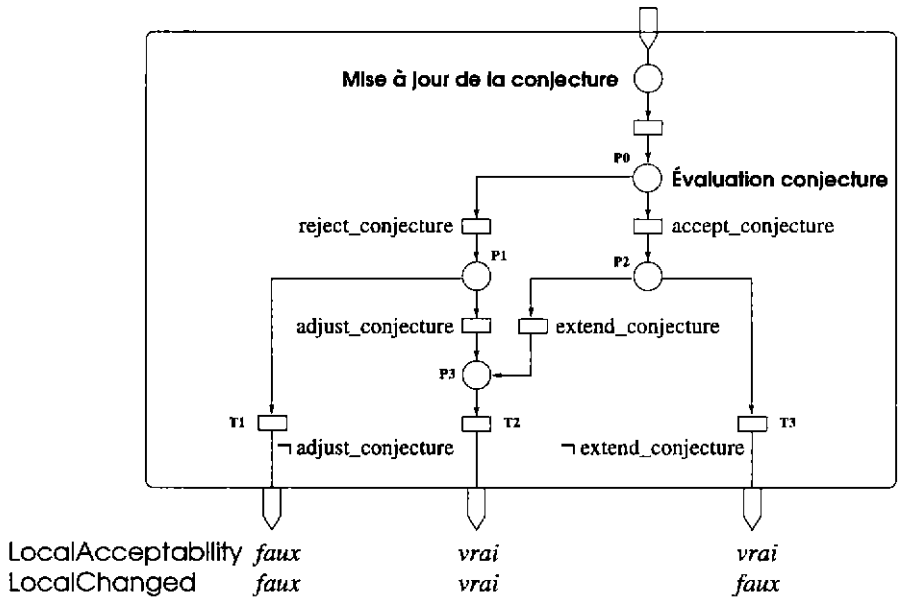


FIG. 3.8 – *Module de mise à jour*

Si l'agent trouve la conjecture non admissible ($reject_conjecture(c_n) = vrai$ et la place P1 est marquée), il peut soit la laisser telle quelle, soit la modifier à condition de la rendre admissible ($adjust_conjecture(c_n) = vrai$ et la place P3 est marquée). Les variables d'état LocalAcceptability et LocalChanged prennent respectivement les valeurs *faux* et *faux* dans le premier cas, et *vrai* et *vrai* dans le second. Le cas où LocalAcceptability = *faux* et LocalChanged = *vrai* a été exclu : en effet, nous avons défini les opérateurs de sorte que la modification de la conjecture ne soit possible que si elle conduit à une nouvelle conjecture admissible (cf. postulat 3.2).

3.4.3 Mise à jour de la base de connaissances

Les agents reçoivent de façon asynchrone des informations en provenance de leurs capteurs. Au niveau du modèle, celles-ci sont représentées comme des jetons stockés dans une place servant de tampon et provenant d'une transition source. L'environnement dans lequel évoluent les agents change constamment. Les agents doivent donc mettre régulièrement à jour leur base de connaissances locale à partir des informations envoyées par les capteurs.

Remarque : nous définissons une information comme un élément du langage. Mais on peut considérer qu'une information est aussi estampillée par une date (date d'observation, date d'envoi, date d'arrivée au niveau de l'agent). On peut ainsi découpler la réception des informations de leurs traitements (mise

à jour de la base de connaissances et mise à jour de la conjecture) par l'agent : la conjecture et la base de connaissances de l'agent sont une représentation d'un état de l'environnement à un instant t donné. Mais il est possible que le tampon d'informations contienne des informations concernant la date t' avec $t' > t$; il y donc un décalage entre l'élaboration d'une conjecture concernant un instant t et les informations dont disposent potentiellement les agents.
Nous n'abordons pas ce problème temporel dans le cadre de ces travaux : comme les agents élaborent conjointement une conjecture concernant un instant t, on supposera que la capacité de traitement de l'information de la fédération est bornée par «le plus lent» des agents. Implicitement, nous faisons l'hypothèse que cette capacité est «grande» devant la période de temps séparant deux changements d'état de l'environnement.

Différentes approches ont été proposées pour formaliser les mécanismes de mise à jour d'une base de connaissances ; on peut les classer dans deux catégories :

- d'une part, les approches qui considèrent qu'une nouvelle information ne peut jamais être rejetée même si elle est en contradiction avec la base de connaissances (Katsuno and Mendelzon, 1991a). L'information doit nécessairement y être intégrée. Il faut alors retirer certains éléments de la base afin de préserver sa cohérence logique.

- d'autre part les approches (de Saint-Cyr, 1996; Boutilier et al., 1998) qui remettent en cause ce postulat parce qu'il ne correspond pas aux besoins applicatifs. En effet, la chaîne de traitement de l'information peut comporter une succession de fonctions (identification, pistage, classification) et il n'est pas rare d'aboutir à une information aberrante ou erronée. Cette information doit donc pouvoir être rejetée en fonction de différents critères relatifs par exemple à l'historique des observations ou la vraisemblance d'apparition d'un événement.

C'est cette dernière approche que nous adoptons pour notre modèle de coopération. L'agent dispose donc de deux variables mutuellement exclusifs, $accept_info$ et $reject_info$: $accept_info$ est la variable qui détermine si une information est acceptée. $reject_info$ est la variable qui détermine si une information est rejetée (la base de connaissances n'évolue donc pas). L'opérateur *update* de mise à jour de la base de connaissances, est un paramètre du modèle, c'est-à-dire que nous n'imposons pas de type spécifique de mise à jour.

Définition 3.11 *Une* $\boxed{information\ I_t}$ *est un élément du langage :* $I_t \in Conject$.

Définition 3.12 *Soit* $\boxed{accept_info}$ *la variable déterminant l'admissibilité des éléments d'une information :*

$$accept_info : \ Conject \rightarrow \{vrai, faux\}$$

et, pour toute information I_t,

$$\boxed{Admissible_Info(I_t)} = \{\ w \mid \forall w \in I_t,\ accept_info(w) = vrai\}$$

est l'ensemble des éléments de I_t admissibles du point de vue de α_i.

Contrairement à ce qui se produit avec la conjecture, on impose à l'agent de partager une information en deux parties, les éléments qu'il estime corrects et les éléments qu'il estime erronés. En effet, lorsqu'un agent reçoit la conjecture, il peut ne pas être apte à évaluer l'admissibilité de tous ses éléments car certains peuvent dépendre de capacités de traitement spécifiques à d'autres agents. En revanche, il semble raisonnable de considérer qu'un agent est capable d'évaluer l'admissibilité des informations fournies par ses propres capteurs.

Par conséquent, les éléments d'une information I_t sont soit admissibles, soit non-admissibles (cf. figure 3.9).

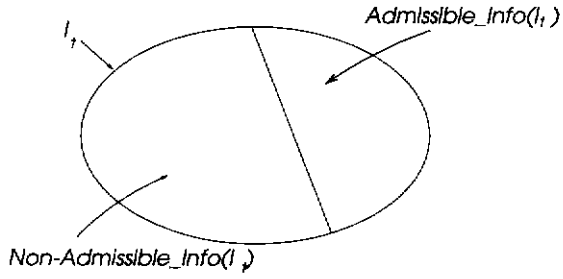


FIG. 3.9 – *Information issue d'un capteur*

Définition 3.13

$$\boxed{Non-Admissible_Info(I_t)} = I_t \setminus Admissible_Info(I_t)$$

Cela suppose, en particulier, que :

$$accept_info(I_t) = vrai \Rightarrow Non-Admissible_Info(I_t) \neq I_t$$

On en déduit la définition de la variable duale $reject_info(I_t)$:

Définition 3.14

$$reject_info:\ Conject \rightarrow \{vrai, faux\}$$

et, pour toute information I_t,

$$reject_info(I_t) = vrai \Rightarrow Non-Admissible_Info(I_t) = I_t$$

L'agent effectue donc un tri entre les éléments qu'il juge admissibles et ceux qu'il juge non-admissibles. Les éléments admissibles sont alors intégrés à la base de connaissances :

Définition 3.15 *Soit update l'opérateur de mise à jour de la base de connaissances de l'agent α_i :*

$$update : Conject \rightarrow Conject$$

On note $K_t = update(I_t)$ la base de connaissances mise à jour. On impose que :

- $Admissible_Info(I_t) \subseteq K_t$,
- $\forall w \in K_t \Rightarrow w \notin Non-Admissible_Info(I_t)$,
- *K_t soit cohérente au sens du langage de l'agent.*

Voyons à présent comment fonctionne le module de mise à jour de la base de connaissances (cf. figure 3.10) : lorsque l'agent reçoit une information, il peut choisir de mettre à jour sa base de connaissances (la place P1 est marquée). Si l'information ne contient aucun élément admissible, elle est rejetée (la place P3 est marquée) et la base de connaissances n'évolue pas.

Une autre possibilité est celle où l'information contient certains éléments admissibles : ces éléments sont extraits de l'information (la place P2 est marquée) puis intégrés à la base de connaissances (la place P4 est marquée).

3.4.4 Décision d'engagement

L'engagement de l'agent implique de sa part une décision sur son maintien dans la fédération et sur les activités à entreprendre le cas échéant.

Du point de vue de la coopération, les deux seules activités que peut mener un agent sont la mise à jour de sa base de connaissances et la mise à jour de la conjecture commune ; toutes les autres activités soit en découlent (e. g. actions sur l'environnement, communications), soit n'ont aucune pertinence vis-à-vis des interactions (e. g. perception).

Par ailleurs, nous postulons que :

Postulat 3.3 *un agent ne peut choisir de mettre à jour la conjecture courante que si elle a été modifiée par un autre agent ou s'il a mis à jour sa base de connaissances.*

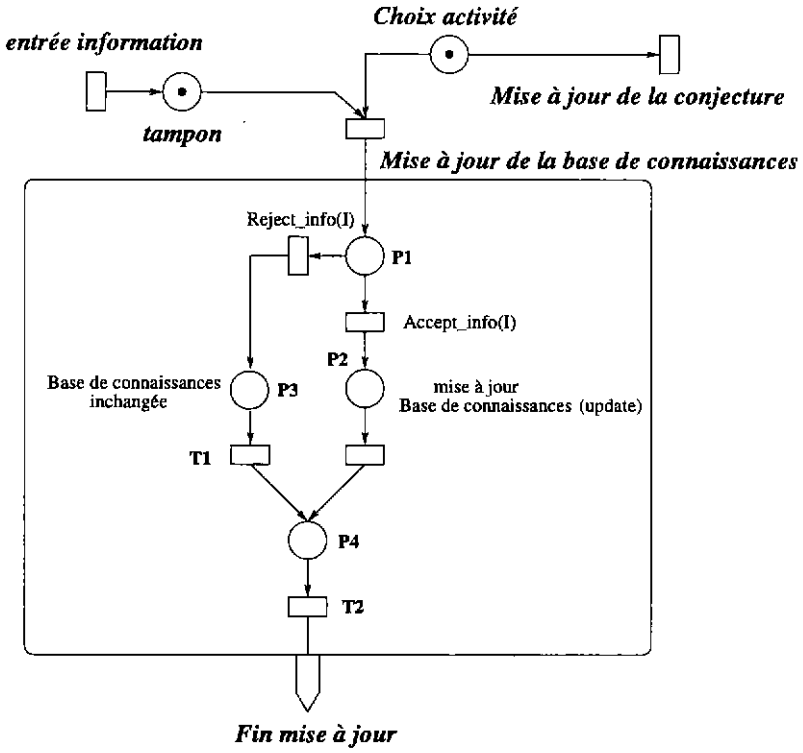


FIG. 3.10 – *Module de mise à jour de la base de connaissances*

Définition 3.16 *Soit* $\boxed{drop(K_t)}$, $\boxed{hold(K_t)}$, $\boxed{KB-update(K_t)}$ *et* $\boxed{C-update(K_t)}$ *les variables de décision suivantes :*

- $drop(K_t) = vrai$ *lorsque l'agent* α_i *décide de se retirer de la fédération ;*
- $hold(K_t) = vrai$ *dans le cas contraire ;*
- KB–$update(K_t) = vrai$ *et* C–$update(K_t) = vrai$ *respectivement lorsque l'agent* α_i *décide de mettre à jour sa base de connaissances locale ou la conjecture courante.*

drop et *hold* d'une part, et $KB-update$ et $C-update$ d'autre part, sont mutuellement exclusifs.

Les variables de décision dépendent de stratégies codées dans la base de connaissances de l'agent, mais celles-ci sont en partie contraintes par le modèle de la façon suivante :

Postulat 3.4 *Le retrait est irrévocable :*

$$drop(K_t) = vrai \Rightarrow drop(K_{t'}) = vrai \quad pour\ tout\ t' > t.$$

Postulat 3.5 *un agent ne peut choisir de se retirer que s'il ne peut pas rendre la conjecture admissible :*

$$(drop(K_t) = vrai) \Rightarrow Non-Admissible(c_n) \neq \emptyset$$

Nous avons imposé ces contraintes sur les variables de décision afin de garantir une certaine cohésion de la fédération : les agents ayant tous le même but, rien ne justifie qu'un agent décide de se retirer du groupe alors qu'il trouve la conjecture admissible. Dans le cas contraire, un agent peut choisir de se retirer pour des raisons qui sont représentées dans sa base de connaissances : il peut, par exemple, considérer que la tâche collective n'est plus réalisable ou pertinente, ou qu'il ne dispose pas des ressources nécessaires pour y participer. Il peut ou non communiquer aux autres agents les raisons de son retrait (cette caractéristique n'est pas prise en compte par le modèle), mais dans tous les cas, sa décision devient une connaissance globale par le biais du greffier qui peut alors déterminer l'incidence de ce retrait sur le fonctionnement de la fédération.

Toutes les variables du module ayant été définies, voyons son fonctionnement (cf. figure 3.11) : son marquage initial est tel que l'agent est engagé (la place P10 est marquée). L'agent ne peut en même temps mettre à jour la conjecture courante et sa base de connaissances (il n'y a qu'un seul jeton dans la place P2) : ceci permet de garantir que l'état de la base de connaissances n'évoluera pas pendant la mise à jour de la conjecture. D'autres conditions «cablées» par le réseau contraignent ce choix : il ne peut y avoir de mise à jour de la base de connaissances que si une nouvelle information est donnée par un capteur (la place P1 est marquée) - il ne peut donc pas y avoir d'auto-enrichissement de la base. De plus, l'agent prend une décision concernant son engagement dans la fédération chaque fois que la conjecture est non admissible de son point de vue (la place P9 est marquée). S'il décide de se retirer, son retrait est irrévocable (la place P6 est marquée).

Par ailleurs, nous avons postulé que l'agent ne peut choisir de mettre à jour la conjecture courante que si elle a été modifiée par un autre agent (la place CM est marquée) ou s'il a mis à jour sa base de connaissances (la place P4 est marquée). Ceci garantit que la conjecture commune, qui est une ressource partagée, est utilisée rationnellement ; mais elle peut ne pas être disponible au moment où l'agent veut y accéder.

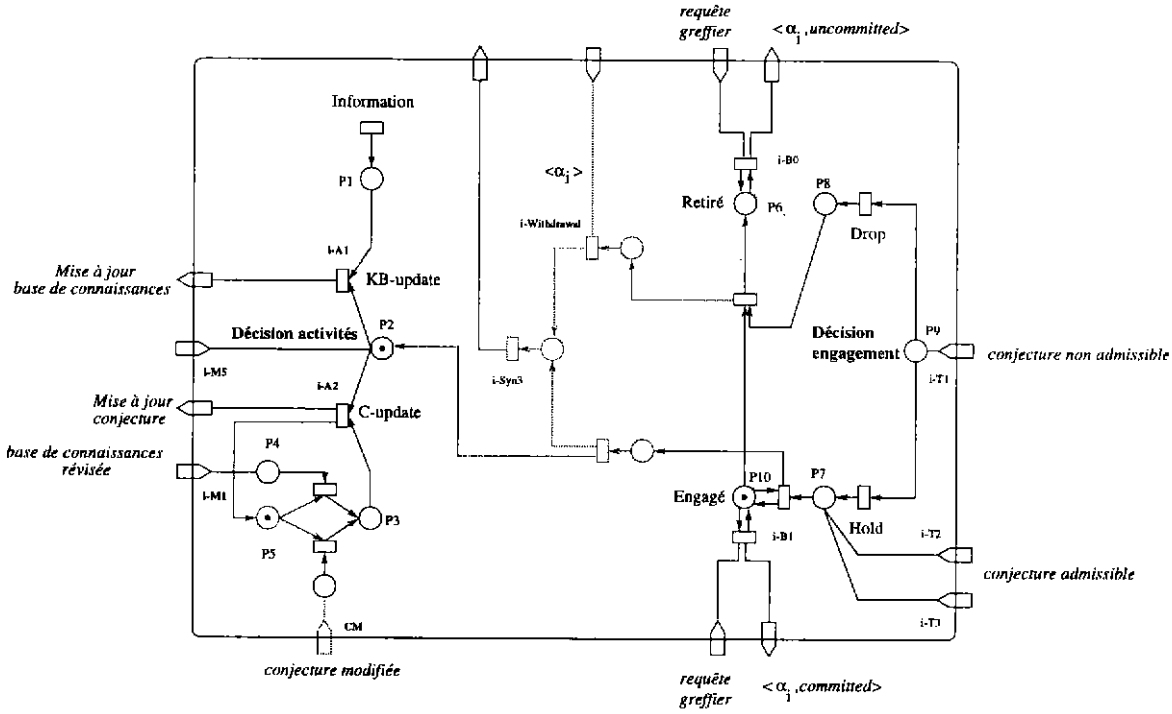


FIG. 3.11 – *Module de décision*

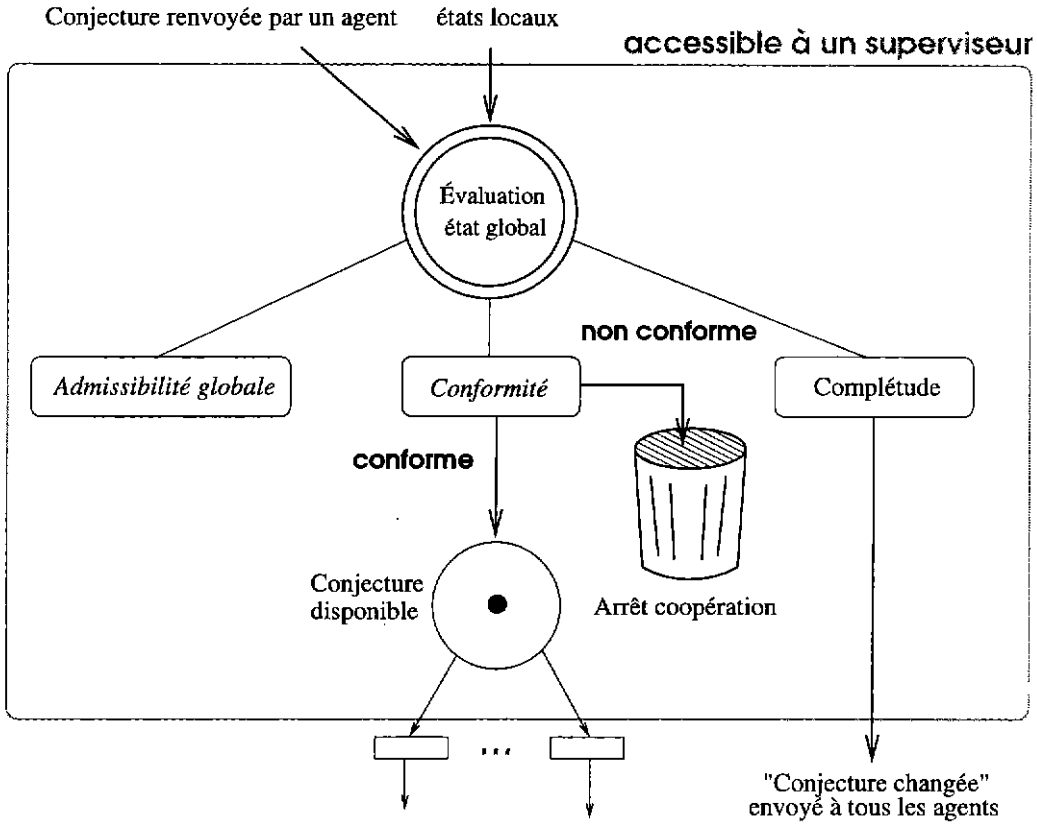
3.5 Le greffier

3.5.1 Aspects fonctionnels

Le greffier (cf. figure 3.12) est une fonctionnalité globale de la fédération dont le rôle est :

- d'évaluer l'état global de la fédération d'agents à travers un indicateur de coopération permettant d'évaluer sa cohérence quant à la tâche d'élaboration collective d'une conjecture commune.

- de maintenir à jour une copie de la conjecture courante. Un superviseur, humain ou non, (e. g. un pilote de chasse utilisant un essaim de gnoptères) peut ainsi consulter à chaque instant une conjecture «partielle», évaluer son «degré d'incomplétude» et éventuellement prendre une décision en fonction de ces paramètres. Ces informations sont disponibles à chaque instant, bien que les agents reçoivent des informations et fassent évoluer la conjecture en permanence.



✘ *les fonctions dépendant de choix d'implémentation apparaissent en italiques*

FIG. 3.12 – *Vue fonctionnelle du greffier*

3.5.2 État global de la fédération

L'état global est évalué par le greffier à partir des états locaux de chaque agent qui sont pour lui des données publiques. Cette évaluation est fondée sur trois critères :

- la **conformité** détermine si les agents fédérés sont capables de réaliser la tâche collective. En effet, les capacités des agents restants peuvent ne plus être en adéquation avec leur objectif dans le cas de retrait d'agents. La tâche collective doit alors être réadaptée, ce qui revient à créer une nouvelle fédération d'agents.

Nous postulons que :

Postulat 3.6 *Le critère de conformité est le seul critère d'arrêt fourni par le modèle.*

Le critère de conformité dépend de l'application. Prenons un exemple : un groupe de gnoptères effectue une mission de reconnaissance. Supposons que l'un d'eux soit détruit : le critère de conformité permet de déterminer à partir de quel moment la mission n'est plus réalisable. Chaque agent devant dire s'il est engagé dans la fédération, il permet aussi de détecter tout problème survenant à l'un de ses membres.

- la **complétude** caractérise le fait, pour l'ensemble des agents fédérés, d'avoir évalué une instance donnée de la conjecture : chaque fois que la conjecture est modifiée par un agent, elle doit être évaluée par les autres agents qui peuvent à leur tour la modifier. L'évaluation d'une conjecture n'est donc *complète* que si elle a été évaluée par l'ensemble des agents sans avoir évolué.

  Ce critère est indépendant du point de vue de chaque agent sur la conjecture (i. e. de son admissibilité), ainsi que de l'application.

- le critère d'**admissibilité globale** détermine la validité de la conjecture du point de vue de la fédération des agents. Ce critère dépend de l'application : par exemple, il peut s'agir d'une procédure de vote. On peut, par exemple, considérer que la conjecture est *admissible* si elle l'est du point de vue de tous les agents.

De manière formelle, l'état global de la fédération des agents est caractérisée par trois variables d'états booléennes : Compliance, Completeness et GlobalAcceptability.

Variables	Sémantiques
Compliance	groupe *conforme* ou *non conforme*
Completeness	évaluation *complète* ou *incomplète*
GlobalAcceptability	conjecture *admissible* ou *non admissible*

TAB. 3.2 – *Variables d'état globales*

On appelle *States* (cf. annexe D) l'ensemble de variables booléennes associées aux places du réseaux de Petri.

Définition 3.17 *Soit I_C une fonction définie de States vers $\{vrai, faux\}$. La variable* $\boxed{\mathsf{Compliance}_n}$ *est définie par :*

$$\mathsf{Compliance}_n \equiv I_C(\mathsf{LocalCommitment}_n^{\alpha_1}, \ldots, \mathsf{LocalCommitment}_n^{\alpha_m})$$

avec $\alpha_i \in F$.
La fédération est $\boxed{conforme}$ *si et seulement si* $\mathsf{Compliance}_n = vrai$.

Exemple :

$$\mathsf{Compliance}_n \equiv \bigwedge_{\alpha_i \in F} (\mathsf{LocalCommitment}_n^{\alpha_i})$$

Dans ce cas, la fédération n'est conforme que si chaque agent est engagé. On peut aussi envisager d'établir un critère en fonction de caractéristiques propres à chaque agent.

Définition 3.18 *Soit I_G une fonction définie de States vers $\{vrai, faux\}$. La variable* $\boxed{\mathit{GlobalAcceptability}(c_n)}$ *est définie par :*

$$\mathit{GlobalAcceptability}(c_n) \equiv I_G(\mathit{LocalAcceptability}_n^{\alpha_1}, \ldots, \mathit{LocalAcceptability}_n^{\alpha_m})$$

avec $\alpha_i \in F$.
La conjecture courante c_n est $\boxed{\mathit{admissible}}$ *si et seulement si* $\mathit{GlobalAcceptability}(c_n) = vrai$.

Exemple : soit $Th_k(M)$ une fonction booléenne à seuil (Sallantin et al., 1991). Par définition, $Th_k(M)$ est vraie si au moins k éléments de M sont vrais. Posons :

$$M = \{\mathsf{LocalAcceptability}_n^{\alpha_1}, \ldots, \mathsf{LocalAcceptability}_n^{\alpha_m}\}$$

$k = n/2+1$ et $\mathsf{GlobalAcceptability}(c_n) \equiv Th_k(M)$: la conjecture n'est admissible que si la majorité des agents la juge admissible.

Définition 3.19 *La variable* $\boxed{\mathit{Completeness}(c_n)}$ *est définie par :*

$$\mathit{Completeness}(c_n) \equiv \bigwedge_{\alpha_i \in F} (\neg \mathit{LocalChanged}_n^{\alpha_i})$$

pour une instance c_n de la conjecture. Ce critère est indépendant de l'application.
La conjecture courante c_n est $\boxed{\mathit{complète}}$ *si et seulement si* $\mathit{Completeness}(c_n) = vrai$.

Nous disposons à présent des éléments permettant de définir la coopération : on souhaite en particulier pouvoir évaluer à chaque instant si la fédération forme un ensemble *cohérent* c'est-à-dire si les agents qui coopèrent sont parvenus ou non à s'accorder sur une conjecture.

Définition 3.20 *Un système de décision composé de plusieurs agents est* $\boxed{\mathit{coopératif}}$ *si et seulement si :*

(1) *les agents ont pour but d'élaborer une conjecture commune ;*

(2) *le système est* $\boxed{\textit{cohérent}}$, *i. e. la fédération des agents est* <u>*conforme*</u> *à la tâche, et la conjecture courante est* <u>*admissible*</u> *et* <u>*complète*</u>*;*

(3) *les critères de conformité, d'admissibilité et de complétude sont évalués par la fonction de greffier.*

<u>Exemple</u> : supposons qu'un agent propose comme conjecture la proposition logique a. Un second agent examine cette conjecture et la remplace par $\neg a$. Le premier agent apprenant par le greffier que la conjecture a été modifiée, décide de l'évaluer... sa base de connaissances n'ayant pas évolué, il remplace à son tour $\neg a$ par a... on voit que le système peut osciller indéfiniment sans jamais converger vers une conjecture stable.
Précisément, la convergence du système dépend des stratégies utilisées par les agents pour élaborer la conjecture (dans cet exemple les agents sont «bornés»). Comme elles dépendent elles-mêmes de l'application et des choix de mise en œuvre, il n'est pas possible au niveau de ce modèle de coopération de caractériser les conditions de convergence. Il est cependant possible d'évaluer la *cohérence* de la fédération. Dans cet exemple, la conjecture n'est jamais *complète*, la fédération n'est donc pas cohérente, donc non coopérative.

3.5.3 Greffier et mise en œuvre

Le premier des postulats du modèle stipule que la conjecture est une ressource partagée, et qu'il ne doit y avoir qu'une seule instance de la conjecture. Par conséquent, elle est traitée par le greffier comme une ressource exclusive, afin de garantir que deux conjectures mutuellement incohérentes ne seront jamais générées par les agents. Du point de vue de la mise en œuvre, ce postulat n'implique pas nécessairement une architecture centralisée : supposons que la conjecture commune représente une situation. Les agents surveillant des zones géographiques différentes peuvent effectuer leurs traitements en parallèle ; dans le cas de zones se chevauchant, on peut choisir une architecture distribuée dans laquelle les agents fusionnent itérativement leurs données.

De notre point de vue, le mode de communication entre agents fait partie des choix de mise en œuvre, c'est pourquoi il n'apparaît pas dans le modèle : les informations nécessaires à la coopération circulent des agents vers le greffier, du greffier vers les agents et entre les agents par l'intermédiaire de la conjecture. Par conséquent, le modèle caractérise les flux d'informations nécessaires à un fonctionnement cohérent de la fédération mais n'impose pas un type particulier de communication (cette démarche diffère donc de celles qui s'appuient sur la théorie des actes de langages pour spécifier la coopération (Shoham, 1993; Werner, 1989; Georgeff, 1983; Seguin, 1992)).

Le greffier n'est pas un coordonnateur : en effet, il ne contrôle pas les interactions et ne prend aucune décision concernant l'élaboration de la conjecture ; il peut être mis en œuvre comme une tâche répartie ou dupliquée au sein de la fédération d'agents. Cependant, l'état global doit être unique (tout comme la conjecture).

3.6 Fonctionnement dynamique du modèle

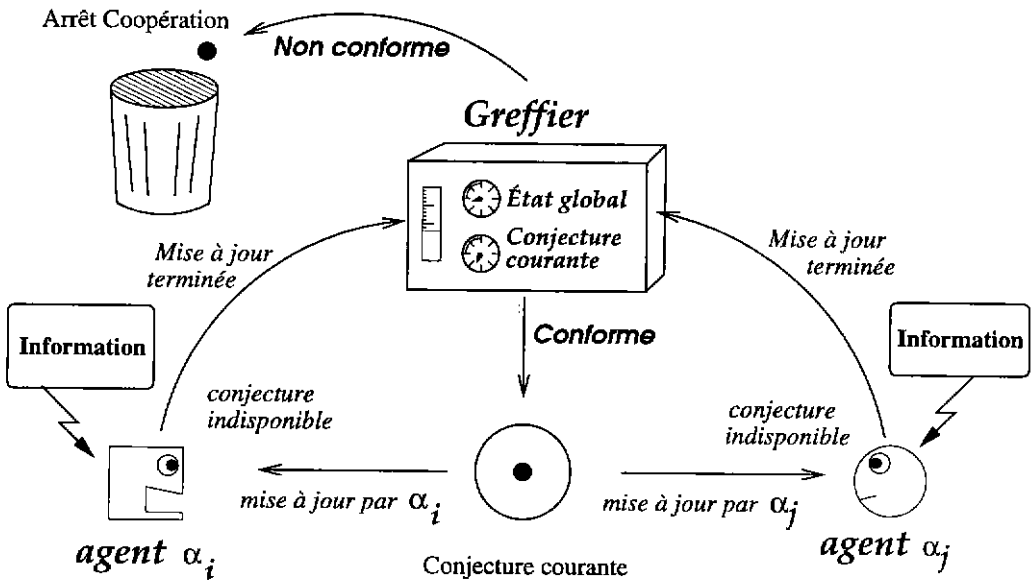


FIG. 3.13 – *Fonctionnement dynamique du modèle*

Les interactions à l'intérieur de la fédération sont fondées sur un cycle d'échanges continus entre le greffier et les agents (cf. figure 3.13) : l'un des agents accède à la conjecture, l'évalue de son point de vue et éventuellement la modifie. L'agent renvoie alors la conjecture et son état local au greffier.

Le greffier agrège les états locaux : si le groupe n'est plus *conforme* vis-à-vis de la tâche commune, le cycle s'arrête. Dans le cas contraire, la conjecture est rendue disponible.

Remarque : le fonctionnement dynamique du modèle de coopération est illustré au chapitre 6.

Conclusion

Les définitions proposées spécifient l'ensemble «minimal» des éléments qui caractérisent un groupe coopératif de systèmes autonomes, et donnent un cadre formel à la réalisation de tels systèmes (Fiorino, 1998; Fiorino and Tessier, 1998a; Fiorino and Tessier, 1998c; Fiorino and Tessier, 1998b).

Ce modèle (cf. figure 3.14) a été testé à l'aide d'un logiciel d'analyse de réseaux de Petri (des, 1993) : ses propriétés sont conformes à celles qui étaient attendues. En particulier, tous les états définis par le modèle sont atteignables et il n'est apparu aucun blocage (excepté le cas de non conformité du la fédération), ni cycle infini.

Les différents fonctions correspondant au modèle tel qu'il a été testé sont détaillées dans l'annexe B. Seuls les réseaux les plus importants ont été présentés dans ce chapitre. Les tableaux suivants récapitulent tous les éléments du modèle de coopération :

Fonctions du greffier
- accéder aux états locaux des agents
- rendre accessible la conjecture
- évaluer la variable Compliance
- évaluer la variable GlobalAcceptability
- évaluer la variable Completeness

TAB. 3.3 – *Récapitulatif des fonctions du greffier*

Variables	**Sémantiques**
Compliance	groupe *conforme* ou *non conforme*
Completeness	évaluation *complète* ou *incomplète*
GlobalAcceptability	conjecture *admissible* ou *non admissible* (global)
LocalAcceptability	conjecture *admissible* ou *non admissible* (local)
LocalChanged	conjecture *changée* ou *inchangée*
LocalCommitment	agent *engagé* ou *retiré*

TAB. 3.4 – *Récapitulatif des variables d'état du modèle*

Postulats du modèle
1 - atomicité de la conjecture
2 - Modification de la conjecture $\Rightarrow$ conjecture admissible
3 - Retrait irrévocable
4 - Retrait si et seulement si conjecture *non-admissible*
5 - Accès conjecture $\Rightarrow$ conjecture changée ou KB mise à jour
6 - Critère d'arrêt = Conformité

TAB. 3.5 – *Récapitulatif des postulats du modèle*

Variables de décision	**Sémantiques**
KB-update	choisir de mettre à jour la base de connaissances
C-update	choisir de mettre à jour la conjecture
drop	décision de retrait
hold	décision d'engagement
accept_conjecture	accepter la conjecture
reject_conjecture	rejeter la conjecture
adjust_conjecture	rendre la conjecture acceptable
extend_conjecture	enrichir un conjecture acceptée
accept_info	accepter des éléments d'information
reject_info	rejeter des éléments d'information

TAB. 3.6 – *Récapitulatif des variables de décision d'un agent*

Fonctions d'un agent
- prendre une décision concernant l'engagement
- accéder à la conjecture
- recevoir les informations provenant des capteurs
- évaluer la variable LocalCommitment
- évaluer la variable LocalAcceptability
- évaluer la variable LocalChanged

TAB. 3.7 – *Récapitulatif des fonctions d'un agent*

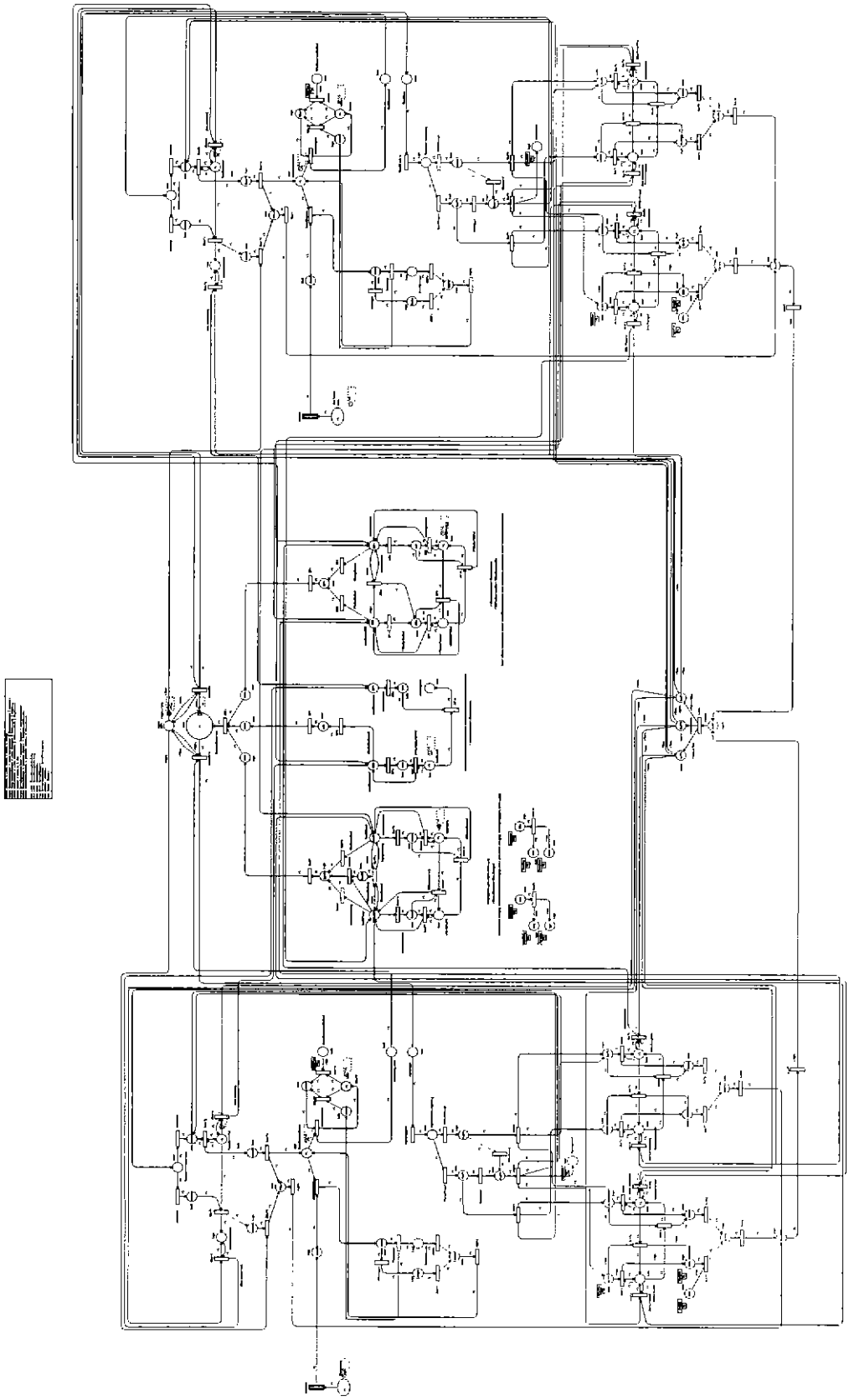


FIG. 3.14 – *Mise en œuvre du modèle*

Chapitre 4

La conjecture commune entendue comme la «croyance» commune d'une fédération d'agents

Introduction

Les applications visées par notre modèle mettent en œuvre des agents *conçus dans le but de coopérer* et l'élaboration d'une conjecture commune est leur seul but : nous avons présenté un modèle de coopération adapté à cette classe d'applications.

Dans ce chapitre, nous nous intéressons à un dispositif d'élaboration de situation dans lequel la conjecture porte sur l'état de l'environnement : l'objectif des agents est ici *d'estimer* cet état à partir de leurs différents points de vue. On s'intéresse donc *au contenu* de cette conjecture en fonction des connaissances des agents et des informations qu'ils reçoivent.

L'établissement d'une conjecture sur l'état de l'environnement est proposé comme cadre formel pour différents types d'applications :

- la coordination de systèmes autonomes : un système autonome doit pour réaliser ses buts, accomplir une boucle de contrôle qui comprend trois grandes étapes ; il doit observer l'environnement et mettre à jour ses connaissances (i. e. étape de perception) ; puis, raisonnant sur ses connaissances (e. g. changement de l'état du monde, prédiction réalisée, échec d'une action), il construit ou met à jour son plan d'actions (i. e. étape de décision) ; la dernière étape est la mise en œuvre du plan (e. g. as-

servissement des effecteurs). Nous postulons qu'un groupe de systèmes autonomes doit procéder de la même manière : les agents doivent d'abord élaborer conjointement une «vision» commune avant de pouvoir coordonner efficacement leurs actions.

- l'aide à la décision : notre objectif est de fournir à un superviseur (humain ou automate) la capacité de prendre une décision adéquate sur la base des observations d'un ensemble d'entités géographiquement réparties ou percevant un même environnement avec différents capteurs. Dans un contexte dynamique, cette caractéristique nous paraît particulièrement importante pour la conduite des systèmes autonomes impliquant des superviseurs humains : ceux-ci peuvent, par exemple, reconfigurer la mission du groupe ou prendre une décision le concernant directement - continuer ou abandonner la mission.

La notion de conjecture commune se trouve donc à l'intersection de deux boucles de contrôle : la première est interne au groupe et coordonne ses éléments constituants en fonction des consignes du superviseur ; la seconde est externe au groupe d'agents et concerne l'assignation de tâches par un superviseur.

Les agents faisant des propositions *différentes*, fondées sur des connaissances locales et incomplètes voire erronées, des conflits peuvent surgir. Cependant, les agents ne sont pas en compétition et ne défendent pas des intérêts divergents : ils mettent à jour leurs connaissances locales en intégrant les informations provenant de leurs capteurs et élaborent conjointement une conjecture.

Intuitivement, l'état de l'environnement que doivent estimer les agents est le «barycentre» des *croyances* des agents, c'est-à-dire de leurs connaissances incertaines. Pour construire cette *croyance commune* (i. e. une conjecture commune portant sur ce que les agents croient être l'état de l'environnement), les agents échangent des propositions et des contre-propositions : la croyance commune est itérativement élaborée à partir des croyances de chaque agent et donc de ses observations. Si les agents ont des croyances mutuellement incohérentes, ils doivent être capables de converger vers une croyance commune représentant au moins partiellement l'état «réel» de l'environnement.

Remarquons que la résolution de conflits est un thème central de l'Intelligence Artificielle Distribuée : en particulier, la recherche porte sur l'étude de méthodes permettant d'éviter des comportements incohérents par rapport à la notion de coopération : un conflit peut se produire lorsque deux agents cherchent à utiliser la même ressource au même instant ; par exemple, lorsque deux robots se rencontrent à un passage séparant deux pièces.

Mais ce n'est pas ce type de conflits qui nous intéresse ici : les agents doivent délibérer pour aboutir à un accord que nous appelons croyance commune et qui représente le point de vue global de la fédération des agents sur l'état de leur environnement. Ce type de coopération paraît particulièrement pertinent lorsque des agents doivent rester coordonnés au cours d'une mission.

Reprenons l'exemple du chapitre 3 où un agent *croit* que la propriété a est vraie alors qu'un autre agent croit en $\neg a$: de notre point de vue, la croyance commune ne doit pas être un choix entre a et $\neg a$. En effet, ces agents n'ont pas un problème à résoudre (Durfee et al., 1987) mais sont plongés dans un environnement dynamique, et de nouvelles informations leur parviennent régulièrement. Par conséquent, ne pas choisir entre des hypothèses alternatives revient à rendre compte de *l'incertitude du groupe* à un instant donné. Désigner l'une d'entre elles comme l'état de l'environnement, c'est courir un risque inutile puisque, dans un contexte dynamique, les nouvelles informations peuvent permettre d'élaguer les hypothèses. Les agents doivent donc trouver un compromis entre attendre et décider : c'est le principe de *l'engagement différé* (Ghose and Goebel, 1998) et le problème consiste à garder *aussi peu d'hypothèses que possible* à chaque instant.

Ce chapitre est organisé de la façon suivante : dans un premier temps, nous abordons le problème de la représentation des états de l'environnement par un langage logique ; nous définissons les notions de *croyances* et de *préférences* d'un agent ; nous introduisons un cadre algébrique pour la fusion des préférences ; nous définissons les notions de *conflit* et de *croyance commune*, en fondant cette dernière sur la fusion des préférences ; puis, nous donnons un exemple d'opérateur de fusion dont nous étudions les propriétés.

Le chapitre suivant sera consacré à la définition des opérateurs du modèle de coopération afin d'élaborer cette croyance commune.

4.1 Représentation logique de la notion d'état de l'environnement et de croyance d'un agent

4.1.1 Exemple

Nous allons illustrer les aspects formels de cet exposé avec un «exemple jouet» inspiré par les problèmes de surveillance d'un site : deux agents α et β observent le même environnement (cf. figure 4.1.1) composé de deux pièces,

$S1$ et $S2$, d'un chat et d'une souris. Les sources d'informations des agents sont

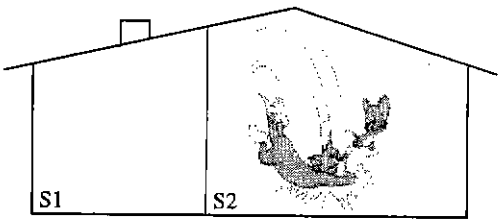


FIG. 4.1 – *environnement observé par les agents*

indépendantes les unes des autres et les agents ne reçoivent donc pas nécessairement les mêmes informations. Ces informations portent sur la position du chat et de la souris et sont incomplètes, c'est-à-dire que les agents ne sont pas nécessairement informés de la position du chat et de la position de la souris : les informations reçues sont un sous-ensemble de ces attributs, qui peut être différent à chaque instant t[1]. Ainsi, l'agent α peut à un instant donné observer que «le chat est en $S1$» puis à un autre instant que «la souris est en $S1$». De même, l'agent β peut percevoir à un instant donné que «la souris est en $S1$» et ne rien détecter pendant que α détecte quelque chose. Par ailleurs, la souris est effrayée lorsqu'elle se trouve dans la même pièce que le chat. Cette propriété n'est pas observable mais elle peut être *déduite* par les agents.

Les agents mettent à jour leurs bases de connaissances locales en intégrant les informations qui leur parviennent. Afin de maintenir leur base de connaissances cohérente, ils doivent parfois retirer un certain nombre de faits.

4.1.2 Représentation du monde par la logique prédicative

Afin de construire une représentation commune de l'environnement, les agents doivent utiliser le même langage $\mathcal{L}$ pour décrire l'ensemble des états possibles. **Nous appelons *monde* un état de l'environnement.**

Une approche classique (Garcia, 1993) consiste à décrire tous les mondes possibles avec un nombre limité de propriétés élémentaires. Dans notre exemple, ces propriétés sont $\mathcal{P} = \{a, b, c, d, e\}$ où chaque lettre dénote respectivement :

- «le chat est en $S1$»,
- «le chat est en $S2$»,

1. Rappelons que nous avons défini t comme un indice évoluant à chaque mise à jour de la conjecture.

- «la souris est en $S1$»,
- «la souris est en $S2$»,
- et «la souris est effrayée».

Par conséquent, un sous-ensemble $\mathcal{E} \subseteq \mathcal{P}$ correspond à un monde dans lequel toutes les propriétés de $\mathcal{E}$ sont vraies et toutes les propriétés de $\mathcal{P} \backslash \mathcal{E}$ sont fausses. Avec n propriétés, on peut donc décrire 2^n états. Toutefois, une représentation logique des états est souvent préférée à cette représentation ensembliste ; il est en effet plus aisé de rendre compte des relations entre les propriétés avec les connecteurs de la logique $\wedge, \vee, \rightarrow$ et $\neg$ qu'avec les opérateurs ensemblistes $\cup, \cap, \subset$ et $\backslash$. Par ailleurs, la logique permet de mettre en œuvre des mécanismes d'inférence.

En particulier, la logique des prédicats du premier ordre restreinte aux univers de Herbrand finis (pour une présentation formelle de la logique prédicative et des interprétations de Herbrand se reporter à l'annexe C) est un langage suffisamment expressif pour représenter le monde (Garcia, 1993).

La base de connaissances K_t d'un agent est une théorie logique qui se compose de deux parties : $K_t = D_t \cup C$, telles que :

- C sont des contraintes sur les états possibles d'un domaine donné. Dans notre exemple, le chat se trouve nécessairement dans l'une des deux pièces mais pas dans les deux en même temps. Il en va de même pour la souris. Par ailleurs, si le chat et la souris sont simultanément dans la même pièce, alors la souris est effrayée. Ces *contraintes du domaine* sont représentées par des formules logiques (le symbole $\equiv$ représente l'équivalence logique) :

$$C = \begin{cases} \forall x, y\ Dans(x,y) \rightarrow (x = S1 \vee x = S2) \wedge (y = Chat \vee y = Souris) \\ \forall x\ Peur(x) \rightarrow x = Souris \\ \forall x, y, z\ Dans(x,y) \wedge Dans(z,y) \rightarrow x = z \\ \exists x, y\ Dans(x, Chat) \wedge Dans(y, Souris) \\ \forall x\ Dans(x, Chat) \wedge Dans(x, Souris) \rightarrow Peur(Souris) \end{cases}$$

Ainsi, avec *H={S1,S2,Chat,Souris}* comme univers de Herbrand, le vocabulaire des agents est composé des formules atomiques suivantes Ξ = {*Dans(S1,Chat), Dans(S2,Chat), Peur(Souris), Dans(S1,Souris), Dans-(S2,Souris)*} qui sont associées à chaque propriété.

Nous faisons l'hypothèse que **tous les agents possèdent le même ensemble de contraintes** ;

- D_t est un ensemble de formules atomiques représentant des faits observés, comme par exemple $Dans(S2, Chat)$, et des faits déduits à partir des contraintes, comme par exemple $Peur(Souris)$. Par conséquent, à un instant donné, seule leur connaissance du monde représenté par D_t différencie les agents.

Un état de l'environnement est donc décrit par un ensemble de formules: à un instant donné, il peut être tel que «le chat est en $S1$» *et* «la souris est en $S2$» :

$$K_t = \left\{ \begin{array}{l} Dans(S1, Chat) \\ Dans(S2, Souris) \end{array} \right. \cup C$$

On associe à cette théorie des modèles logiques qui représentent des états complets de l'environnement. Dans le cas de notre exemple, il en existe deux :

$$\omega_1 = \left\{ \begin{array}{l} Dans(S1, Chat) \\ Dans(S2, Souris) \\ \neg Dans(S1, Souris) \\ \neg Dans(S2, Chat) \\ \mathbf{Peur(Souris)} \end{array} \right.$$

$$\omega_2 = \left\{ \begin{array}{l} Dans(S1, Chat) \\ Dans(S2, Souris) \\ \neg Dans(S1, Souris) \\ \neg Dans(S2, Chat) \\ \neg\mathbf{Peur(Souris)} \end{array} \right.$$

Le langage $\mathcal{L}$ permet de décrire $2^5 = 32$ mondes (nombre d'interprétations associées au langage). Il faut cependant remarquer que certains de ces états ne peuvent jamais se produire : par exemple, un monde dans lequel le chat serait en même temps en $S1$ et en $S2$ (problème d'ubiquité). Ces mondes sont éliminés par les contraintes du domaine.

Supposons, par exemple, qu'un agent n'a fait aucune observation et ne possède comme connaissances que les contraintes du domaine. À une telle

représentation de l'environnement, on peut associer six modèles logiques :

$$\omega_1 = \begin{cases} Dans(S1, Chat) \\ Dans(S2, Souris) \\ \neg Dans(S1, Souris) \\ \neg Dans(S2, Chat) \\ Peur(Souris) \end{cases} \quad \omega_2 = \begin{cases} Dans(S1, Chat) \\ Dans(S1, Souris) \\ \neg Dans(S2, Souris) \\ \neg Dans(S2, Chat) \\ Peur(Souris) \end{cases}$$

$$\omega_3 = \begin{cases} Dans(S2, Chat) \\ Dans(S1, Souris) \\ \neg Dans(S2, Souris) \\ \neg Dans(S1, Chat) \\ Peur(Souris) \end{cases} \quad \omega_4 = \begin{cases} Dans(S2, Chat) \\ Dans(S2, Souris) \\ \neg Dans(S1, Souris) \\ \neg Dans(S1, Chat) \\ Peur(Souris) \end{cases}$$

$$\omega_5 = \begin{cases} Dans(S1, Chat) \\ Dans(S2, Souris) \\ \neg Dans(S1, Souris) \\ \neg Dans(S2, Chat) \\ \neg Peur(Souris) \end{cases} \quad \omega_7 = \begin{cases} Dans(S2, Chat) \\ Dans(S1, Souris) \\ \neg Dans(S2, Souris) \\ \neg Dans(S1, Chat) \\ \neg Peur(Souris) \end{cases}$$

Ces modèles rendent compte de l'*incertitude* de l'agent relative à l'état réel du monde : en effet, l'état de l'environnement est unique ; il correspond à l'un de ces modèles mais l'agent n'est pas capable de dire lequel. Les interprétations de la base de connaissances représentent donc des *hypothèses* mutuellement exclusives sur l'état de l'environnement : du point de vue de l'agent, l'environnement est soit dans l'état ω_1, soit dans l'état ω_2, etc. Lorsque l'agent reçoit des informations, il diminue son incertitude car le nombre des hypothèses admissibles (i. e. les interprétations qui satisfont sa base de connaissances) diminue. Par exemple, si l'agent observe que «le chat est en $S1$» et que «la souris est en $S2$», les deux seuls mondes possibles sont ω_1 et ω_2.

4.2 Notion de croyances et de préférences

Dans le reste ce chapitre, nous considérerons que $\mathcal{L}$ **est un langage de la logique propositionnelle.**

Ξ dénote l'ensemble des variables propositionnelles de $\mathcal{L}$:

Définition 4.1 *Une* $\boxed{\textit{interprétation}}$ *est une fonction I de l'ensemble des variables propositionnelles Ξ vers $\{vrai, faux\}$. Ω est l'ensemble de toutes les interprétations :*

$$\Omega = \{vrai, faux\}^n \quad avec\ n = card(\Xi)$$

Notation : soit $\Xi = \{a, b, c\}$, on note $\omega =< a >$ l'interprétation I telle que $I(a) = vrai$ et $I(b) = I(c) = faux$.

De plus, une interprétation ω est un *modèle* d'une formule ψ si elle assigne la valeur *vrai* à cette formule au sens usuel (cf. annexe C).

Pour revenir à notre exemple, considérons que $\Xi = \{a, b, c\}$ où :

- a est *vrai* lorsque le chat est en $S1$ et *faux* dans le cas contraire ;
- b est *vrai* lorsque la souris est en $S1$ et *faux* dans le cas contraire ;
- c est *vrai* lorsque la souris est effrayée, et *faux* sinon.

Soit Ω l'ensemble de toutes les interprétations du langage. Dans notre exemple, elles sont au nombre de huit (cf. figure 4.2).

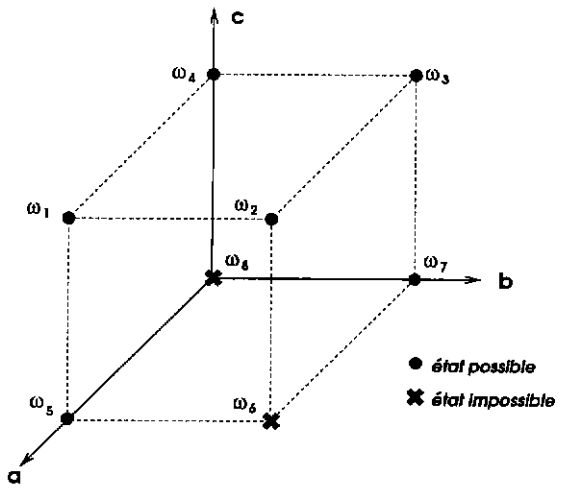


FIG. 4.2 – *Ensemble d'interprétations et états possibles de l'environnement*

Les interprétations ω_6 et ω_8 correspondent à des états impossibles de l'environnement (i. e. le chat et la souris sont dans la même pièce et la souris n'a pas peur). Ce ne sont donc pas des modèles de la base de connaissances de l'agent car elles ne satisfont pas ses contraintes du domaine.

Il est habituel (Gärdenfors, 1988; Boutilier et al., 1998) de distinguer une base de croyances d'une base de connaissances :

Définition 4.2 *On appelle* $\boxed{\textit{fermeture logique}}$ *de la base de connaissances* K_t *l'ensemble de toutes les formules conséquences logiques de* K_t *et on note* $Cn(K_t)$ *cet ensemble :* $Cn(K_t)$ *est la* $\boxed{\textit{base de croyances}}$ *de l'agent. On note* ψ *la conjonction de tous les éléments de* $Cn(K_t)$.

L'état de l'environnement est caractérisé par un ensemble de propriétés qui sont formellement représentées par des variables propositionnelles. On doit cependant distinguer les croyances d'un agent concernant l'état de l'environnement de l'état «réel» de l'environnement. Comme l'agent a une connaissance incomplète du monde, ses «croyances» (i. e. les modèles de sa base de croyances) sont un ensemble d'états possibles. Par ailleurs, cet ensemble ne peut pas être vide : lorsque l'agent ne sait rien sur l'environnement, ses «croyances» correspondent à l'ensemble des états possibles de l'environnement. Comme il ne peut pas déterminer lequel de ces états représente l'état réel, ils sont tous «équivalents» de son point de vue.

Intuitivement, on aimerait cependant pouvoir rendre compte de la notion de *préférence* entre les différentes hypothèses possibles. Cela supposerait que les agents aient des *croyances graduelles* (Gärdenfors, 1988), c'est-à-dire que certains états soient *plus crus* que d'autres. À cet effet, nous introduisons la notion d'*état de croyance* de l'agent : cet état est représenté par une relation d'ordre total[2] sur une partition de l'ensemble des interprétations Ω.

Nous nous plaçons dans le cas où K_t, la base de connaissances d'un agent, est constituée d'un ensemble de formules atomiques représentant les faits (observés ou déduits) relatifs à l'environnement, et des contraintes sur le domaine.

Définition 4.3 *Soit $P = \{P_1, \ldots, P_n\}$ une partition de Ω (i. e. $\bigcup_{i\in[1,\ldots,n]} P_i = \Omega$ et $P_i \cap P_j = \emptyset$ avec $i \neq j$). On note $\leq_\psi$ la relation d'ordre total établie par un agent ψ[3] sur P. On déploie la relation d'ordre $\leq_\psi$ sur Ω par :*

$$\forall \omega_i \in P_i,\ \forall \omega_j \in P_j,\ P_i \leq_\psi P_j \Leftrightarrow \omega_i \leq_\psi \omega_j$$

Définition 4.4 *On définit la relation d'équivalence $\approx_\psi$ sur Ω par :*

$$\forall \omega_i,\ \forall \omega_j \in \Omega,\ \omega_i \approx_\psi \omega_j \Leftrightarrow \begin{cases} \omega_i \leq_\psi \omega_j \\ \omega_j \leq_\psi \omega_i \end{cases}$$

On pose que :

$$\forall \omega_i,\ \forall \omega_j \in P_i,\ \omega_i \approx_\psi \omega_j$$

Définition 4.5 *L'* $\boxed{\textit{état de croyance}}$ *d'un agent ψ à un instant t est le triplet $(\Omega, \leq_\psi, t)$.*

2. Une relation d'ordre (on dit aussi un ordre partiel) est une relation binaire $\leq$ sur un ensemble E qui est reflexive, antisymétique et transitive. Cet ordre est *total* si tout couple d'éléments de E est comparable.

3. Dans la mesure où il n'y a pas de confusion possible, nous utilisons désormais la même notation pour le nom de l'agent et le nom de sa base de croyances.

Définition 4.6 *On dit que :*

- *l'agent* $\boxed{\textit{préfère}}$ ω_i *à* ω_j *si* $\omega_i \leq_\psi \omega_j$ *;*
- ω_i *et* ω_j *sont* $\boxed{\textit{également préférées}}$ *par l'agent si* $\omega_i \approx_\psi \omega_j$.

À un instant donné, plusieurs agents peuvent donc avoir des préférences différentes, c'est-à-dire un ordre différent sur les interprétations. Par ailleurs, nous voulons que les agents soient *rationnels* :

Définition 4.7 *On note* $[\![\psi]\!]$ *l'ensemble des modèles de* ψ. *Un agent est* $\boxed{\textit{rationnel}}$ *si et seulement si :*

- *sa base de croyances n'est pas contradictoire, c'est-à-dire si elle admet au moins un modèle :*

$$[\![\psi]\!] \neq \emptyset$$

- *les modèles de la base de croyances sont préférés à toute autre interprétation ne la satisfaisant pas :*

$$\forall \omega \in [\![\psi]\!],\ \forall \omega' \in \Omega \setminus [\![\psi]\!],\ \omega \leq_\psi \omega'$$

Soit $\mathcal{M}$ un sous-ensemble de Ω. Une interprétation ω est *minimale* dans $\mathcal{M}$ par rapport à la relation de préférence $\leq_\psi$ si ω est un élément de $\mathcal{M}$ et qu'il n'existe pas d'interprétation ω' dans $\mathcal{M}$ préférée à ω. Formellement,

Définition 4.8 *On appelle* $\boxed{\textit{interprétations minimales}}$ *d'un ensemble* $\mathcal{M}$ *par rapport à* $\leq_\psi$, *l'ensemble noté* $min(\mathcal{M}, \leq_\psi)$ *et défini par :*

$$min(\mathcal{M}, \leq_\psi) = \{\omega \in \mathcal{M} \mid (\forall \omega' \in \mathcal{M}),\ \omega \leq_\psi \omega'\}$$

Par définition, un agent rationnel préfère ses modèles à toute autre interprétation. Mais il faut souligner que ce critère de rationalité n'implique pas que tous les modèles de ψ soient également préférés : nous appelons *convictions* les modèles les plus préférés par l'agent, c'est-à-dire ses mondes les plus préférés :

Définition 4.9 *Soit* $\widehat{K}_t^\psi$ *l'ensemble des* $\boxed{\textit{convictions}}$ *de l'agent* ψ *à l'instant* t *:*

$$\widehat{K}_t^\psi = \{\phi \in \mathcal{L} \mid [\![\phi]\!] = min(\Omega, \leq_\psi)\}$$

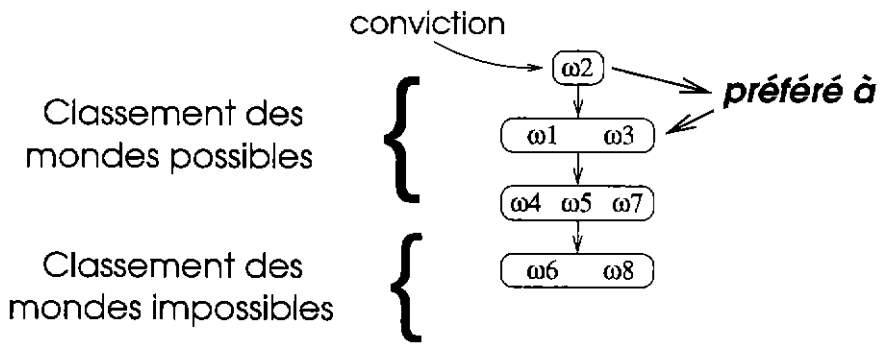


FIG. 4.3 – *Préférences d'un agent*

Par définition, $[\![\phi]\!] \subseteq [\![\psi]\!]$. Nous avons introduit cette notion pour montrer que la croyance commune que doivent construire les agents n'est, d'un point de vue sémantique, ni l'intersection des convictions de chaque agent, ni leur union : nous préférons définir, comme nous le verrons par la suite (cf. paragraphe 4.4.2), la croyance commune en termes de *persistance des préférences.*

Exemple : les préférences d'un agent disposant des informations «le chat est en $S1$» et «la souris est en $S1$» peuvent être celles illustrées par la figure 4.3.

Sur cet exemple, l'agent préfère croire que toutes ses informations sont exactes : ω_2 est une conviction ; les interprétations suivantes correspondent au cas où l'une des informations est fausse et au cas où les deux informations sont fausses. Les interprétations qui ne correspondent pas à des états possibles de l'environnement (i. e. ω_6 et ω_8 ne satisfont pas les contraintes du domaine) sont rejetées en bout de classement. On a donc d'une part les états irréalisables de l'environnement, et d'autre part une classification des mondes possibles dont le préféré correspond au modèle de la base de croyances. On peut en particulier remarquer que l'ensemble des états irréalisables est connu *a priori* et statique à cause des contraintes du domaine, alors que le classement des mondes possibles évolue en fonction des changements d'état de l'environnement.

Par ailleurs, le graphe des préférences est un sous-ensemble du produit cartésien $\Omega \times \Omega$. On note $\mathcal{O}_\psi \subseteq \Omega \times \Omega$ le graphe bi-partie défini par $\leq_\psi$:

$$\omega \leq_\psi \omega' \Leftrightarrow (\omega, \omega') \in \mathcal{O}_\psi$$

On peut représenter ce graphe sous la forme d'un tableau. Si on reprend l'exemple de la figure 4.3, $\mathcal{O}_\psi$ est donnée par le tableau 4.1.

$\leq_\psi$	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6	ω_7	ω_8
ω_1	•		•	•	•	•	•	•
ω_2	•	•	•	•	•	•	•	•
ω_3	•		•	•	•	•	•	•
ω_4				•	•	•	•	•
ω_5				•	•	•	•	•
ω_6						•		•
ω_7				•	•	•	•	•
ω_8						•		•

TAB. 4.1 – *Graphe de préférences de* $\leq_\psi$

4.3 Cadre algébrique pour la fusion des préférences

Notre but est de définir l'élaboration d'une croyance commune à partir des croyances de chaque agent. Nous présentons un cadre algébrique pour la fusion des préférences.

4.3.1 Principe de la fusion endogène

La fusion des préférences est fondée sur la structure des treillis.

Étant donné tout couple d'éléments x et y, la structure de treillis garantit l'existence d'une borne inférieure $x \sqcap y$ et d'une borne supérieure $x \sqcup y$ encadrant les deux éléments.

D'après (Chaudron, 1995b) :

Définition 4.10 *Soit E un ensemble, $\sqcap$ et $\sqcup$ deux lois internes définies de $E \times E$ vers E. Le triplet $(E, \sqcap, \sqcup)$ est un* $\boxed{\textit{treillis}}$ *si pour tout x, pour tout y et pour tout z dans E :*

(Idempotence) $x \sqcap x = x,\ x \sqcup x = x$;

(Commutativité) $x \sqcap y = y \sqcap x,\ x \sqcup y = y \sqcup x$;

(Associativité) $x \sqcap (y \sqcap z) = (x \sqcap y) \sqcap z,\ x \sqcup (y \sqcup z) = (x \sqcup y) \sqcup z$;

(Absorption) $x \sqcap (x \sqcup y) = x \sqcup (x \sqcap y) = x$.

Exemple : l'ensemble $\mathcal{P}(\Omega)$ des parties de l'ensemble Ω muni des opérateurs d'intersection et de réunion est un treillis noté $(\mathcal{P}(\Omega), \cap, \cup)$.

De même (Chaudron, 1995b) :

Propriété 4.1 *Tout treillis* $(E, \sqcap, \sqcup)$ *est un ensemble ordonné* $(E, \preceq, \sqcap, \sqcup)$ *où la relation* $\preceq$ *est définie par :*

$$x \preceq y \text{ si et seulement si } x \sqcap y = x$$

Par commodité, lorsqu'on désigne un treillis, il est souvent judicieux d'indiquer la relation d'ordre lorsqu'on la connaît : on a, par exemple, $(\mathcal{P}(\Omega), \subseteq, \cap, \cup)$.

Un processus de fusion endogène de deux éléments de connaissances provenant de deux agents différents n'introduit pas de connaissances externes, et est défini comme une fonction f (Maille, 1997) telle que :

$$f : E \times E \to E$$

$$x \sqcap y \preceq f(x, y) \preceq x \sqcup y$$

Ce principe permet, pour un opérateur de fusion donné, de caractériser un ensemble de résultats possibles.

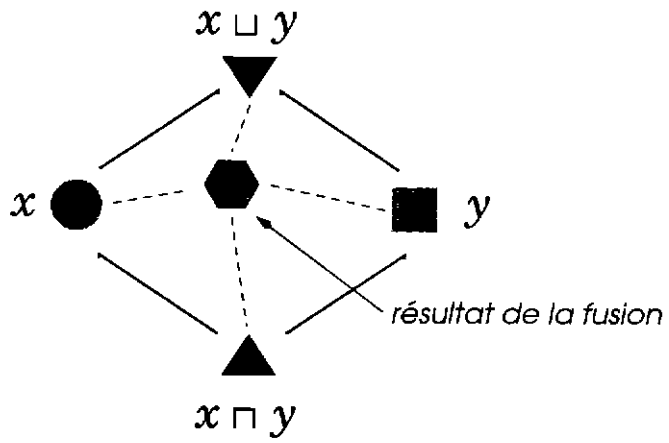


FIG. 4.4 – *Principe de fusion*

4.3.2 Treillis de graphes de préférences

Il est trivial de montrer que :

Propriété 4.2 $(\mathcal{P}(\Omega \times \Omega), \subseteq, \cap, \cup)$ *est un treillis.*

Il est donc possible de construire un treillis à partir des relations de préférences de deux agents, qui caractérise l'*intérêt commun* (Chaudron et al., 1997c; et C. Tessier, 1998) des deux agents. Cet intérêt commun est le cadre formel et indépendant de l'opérateur de fusion, à l'intérieur duquel les agents doivent trouver un résultat commun (cf. figure 4.4), c'est-à-dire une relation de *préférence commune*; **nous verrons, par la suite, comment calculer cette préférence commune et en déduire la croyance commune.**

Exemple : soit deux agents ψ et μ ayant respectivement pour croyances «la souris est en $S2$ et elle n'est pas effrayée» et «le chat est en $S1$ et la souris est en $S2$». Leurs relations d'ordre sur les interprétations sont données par la figure 4.5.

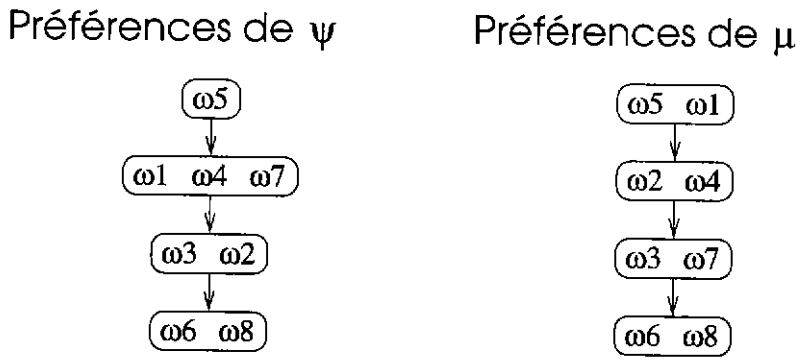


FIG. 4.5 – *Préférences des agents*

La structure de treillis résultante est montrée par la figure 4.6. On appelle $\mathcal{O}_\psi$ le graphe de préférences de ψ et $\mathcal{O}_\mu$ le graphe de préférences de μ. La borne inférieure $\mathcal{O}_\psi \cap \mathcal{O}_\mu$ représente les préférences communes et la borne supérieure $\mathcal{O}_\psi \cup \mathcal{O}_\mu$ représente l'ensemble de leurs préférences.

On constate que la borne supérieure n'est pas nécessairement une relation d'ordre (contrairement à la borne inférieure (Chaudron, 1995b)).

Exemple : $\mathcal{O}_\psi \cup \mathcal{O}_\mu$ n'est pas une relation de préférences, c'est-à-dire une relation d'ordre : ω_3 est préféré ω_2, ω_2 est préféré ω_4, mais ω_3 n'est pas préféré à ω_4. Par conséquent, la transitivité n'est pas vérifiée.

Certaines relations à l'intérieur de la structure de treillis (e. g. elles sont au nombre[4] de 2053 à l'intérieur de la structure de treillis de la figure 4.6) sont

4. Il y a une différence de 11 relations élémentaires entre la borne supérieure et la borne inférieure : il y donc 11 façons différentes d'ajouter une relation élémentaire à la borne

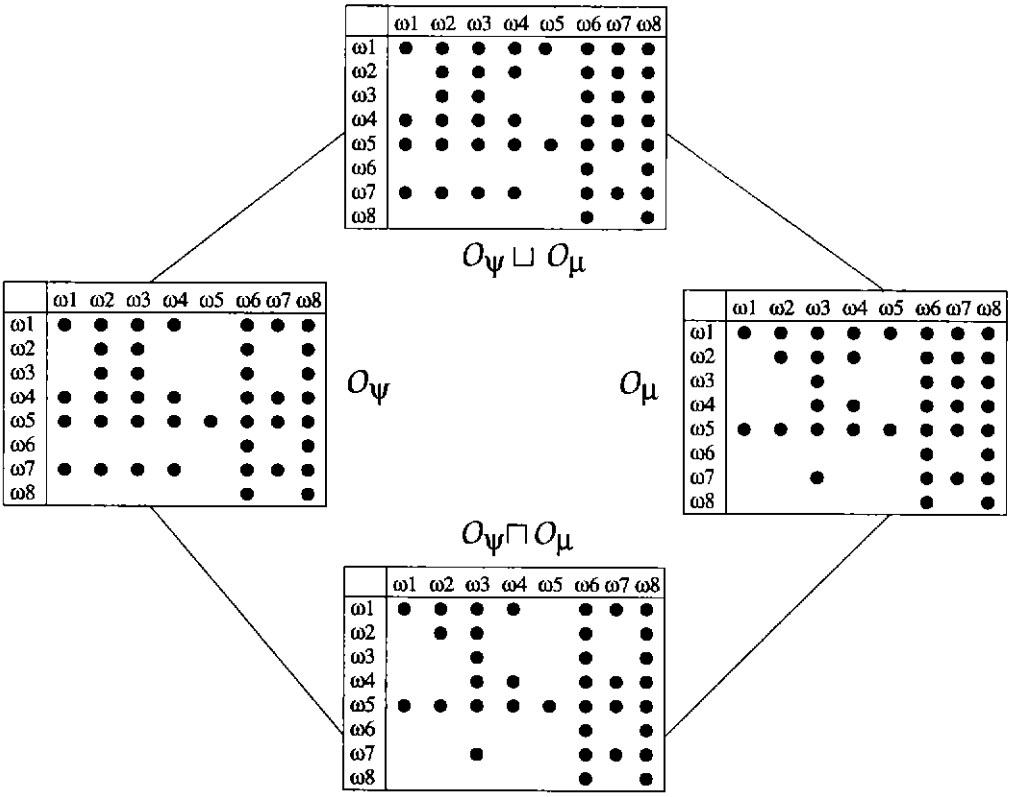


FIG. 4.6 – *Intérêt commun des agents*

des relations d'ordre. Par ailleurs, ces relations vérifient toutes les relations élémentaires (i. e. entre deux interprétations) qui sont communes aux deux relations d'ordre des agents. De plus, elles ne contiennent pas de relations élémentaires autres que celles déjà présentes dans les relations d'ordre des agents : cela signifie qu'un opérateur de fusion qui choisit un résultat à l'intérieur du treillis, n'introduit pas d'information supplémentaire.

4.4 La notion de conflit et de croyance commune

4.4.1 Notion de conflit

Nous allons d'abord introduire cette notion de façon intuitive.

inférieure, 55 façons différentes d'ajouter deux relations élémentaires etc. Le nombre total de relations est égal à $\sum_{i=1}^{i=11} C_{11}^{i} = 2053$.

Contradiction

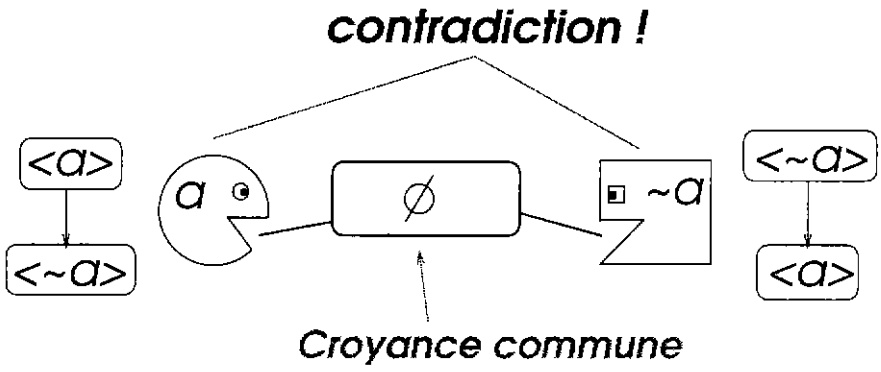


FIG. 4.7 – *Exemple de conflit : connaissances contradictoires*

Supposons qu'un agent croit a tandis que l'autre croit $\neg a$: il s'agit d'une *contradiction*. En effet, l'agent «rond» est convaincu que a est *vrai* alors que l'agent «carré» est convaincu du contraire (i. e. $[\![\widehat{K}_t^{\bigcirc}]\!] \cap [\![\widehat{K}_t^{\square}]\!] = \emptyset$).

Les protocoles de négociation permettent de résoudre de tels conflits : ils sont fondés sur l'idée que les agents doivent abandonner certaines connaissances afin de trouver un accord qui satisfait l'ensemble des agents. Ces approches conviennent à la résolution distribuée de problèmes, et notre propos n'est pas d'en proposer une nouvelle ; notre démarche s'appuie sur le principe qui consiste à ne pas choisir, en l'absence d'information discriminante, entre deux propositions contradictoires venant de deux agents : dans l'exemple de la figure 4.7, la croyance commune devrait être «vide», ce qui représente l'incertitude de la fédération relative à l'attribut a. Nous postulons donc que :

Postulat 4.1 *les agents doivent suspendre leur jugement en cas d'informations contradictoires.*

Plus précisément, l'acceptation d'une information équivaut à une procédure de vote majoritaire : lorsqu'il y a autant d'agents qui croient en une information que d'agents qui n'y croient pas, cette information est laissée en suspens. En effet, les applications qui nous intéressent n'imposent pas nécessairement de prendre une décision immédiate : cela dépend du temps dont disposent les agents pour décider. Implicitement, nous supposons que le flux régulier d'informations peut permettre de sélectionner les hypothèses sur l'état de l'environnement et que les agents doivent différer leur décision tant que cela est possible par rapport aux contraintes de leur mission. Réciproquement, choisir une hypothèse, c'est courir le risque de se tromper et de propager cette erreur à chaque nouvelle mise à jour des croyances.

La principale difficulté de cette stratégie de l'engagement différé (Ghose and Goebel, 1998) consiste à gérer «au mieux» l'incertitude des agents, c'est-à-dire l'ensemble des hypothèses alternatives courantes : la croyance commune est une conjecture globale concernant les croyances des agents : dans un contexte dynamique et distribué, sa fonction est de donner les états de l'environnement *les plus préférés* par les agents.

Différence

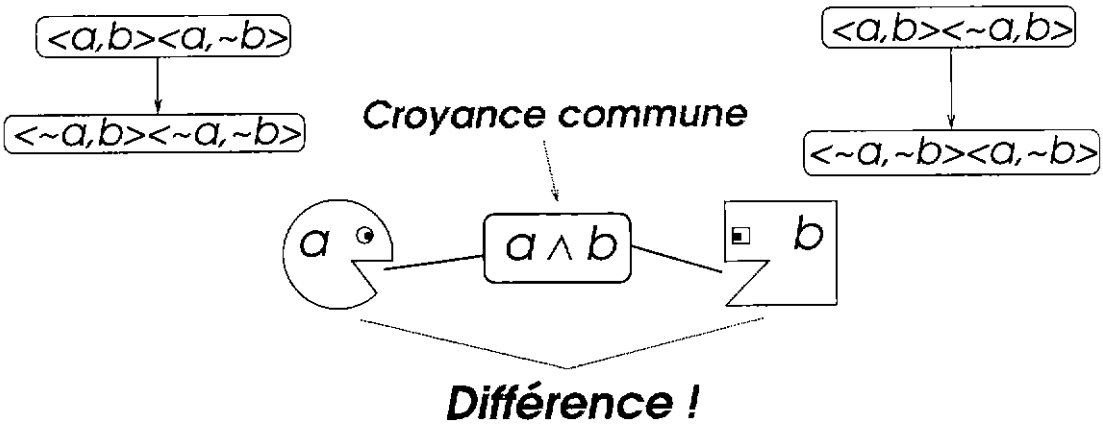


FIG. 4.8 – *Exemple de conflit : connaissances différentes*

Un autre cas de conflit peut se produire (cf. figure 4.7). Reprenons notre exemple de deux agents cherchant à estimer l'état courant de leur environnement **commun** : ils observent un même objet mais ne détectent pas les mêmes attributs. Un agent peut, par exemple, détecter l'attribut a et un autre agent l'attribut b (cf. figure 4.8). L'agent «rond» préfère donc les interprétations dans lesquelles a est $vrai$ à toutes celles dans lesquelles a est $faux$; le même raisonnement s'applique pour l'agent «carré» qui préfère les interprétations dans lesquelles b est $vrai$ à toutes celles dans lesquelles b est $faux$: leurs croyances, qui portent sur un même objet, sont donc *différentes*. Cette différence peut devenir une contradiction lorsque une contrainte du domaine exprime le fait que cet objet est soit dans l'état a, soit dans l'état b. En l'absence de cette contrainte, on peut tout de même parler de conflit car l'agent «rond» (cf. figure 4.8) admet comme possible un monde dans lequel $\neg b$ est $vrai$ alors que l'agent «carré» est convaincu du contraire (i. e. il est impossible que b ne soit pas $vrai$). Réciproquement, l'agent «carré» admet comme possible un monde dans lequel $\neg a$ est $vrai$ alors que l'agent «rond» est convaincu du contraire (i. e. $[\![\widehat{K}_t^{\bigcirc}]\!] \cap [\![\widehat{K}_t^{\square}]\!] \neq \emptyset$ et $[\![\widehat{K}_t^{\bigcirc}]\!] \neq [\![\widehat{K}_t^{\square}]\!]$).

Définitions

De façon formelle :

Définition 4.11 *Soit $\mathcal{O}_\psi$ et $\mathcal{O}_\mu$ les graphes de préférences de deux agents ψ et μ, tous deux éléments de $\mathcal{P}(\Omega \times \Omega)$:*

(Conflit) *Les agents observant le même environnement ont des* $\boxed{\textit{croyances conflictuelles}}$ *si et seulement si :*

$$\mathcal{O}_\psi \neq \mathcal{O}_\mu$$

(Accord) *Les agents sont* $\boxed{\textit{accordés}}$ *si et seulement si :*

$$\mathcal{O}_\psi = \mathcal{O}_\mu$$

(Contradiction) *Soit $\mathcal{O}_\psi^{-1}$ la* $\boxed{\textit{relation duale}}$ *de $\mathcal{O}_\psi$ telle que :*

$$(\omega, \omega') \in \mathcal{O}_\psi^{-1} \text{ si et seulement si } (\omega', \omega) \in \mathcal{O}_\psi$$

Les agents sont en $\boxed{\textit{contradiction}}$ *si et seulement si :*

$$\mathcal{O}_\mu = \mathcal{O}_\psi^{-1}$$

(Ignorance) *Un agent est* $\boxed{\textit{ignorant}}$ *si et seulement si :*

$$\mathcal{O}_\psi = \Omega \times \Omega$$

(Être mieux informé que) *Un agent ψ, ayant pour graphe de préférences $\mathcal{O}_\psi$, est* moins ignorant *ou* $\boxed{\textit{mieux informé}}$ *qu'un autre agent μ ayant pour graphe de préférences $\mathcal{O}_\mu$ si et seulement si :*

$$\mathcal{O}_\psi \subset \mathcal{O}_\mu \subseteq \Omega \times \Omega$$

Remarque : un agent mieux informé qu'un autre agent est nécessairement en conflit avec celui-ci, mais la réciproque n'est pas vraie : deux agents peuvent être en conflit sans que leurs graphes de préférences soient comparables du point de vue de la notion d'ignorance (i. e. la relation d'inclusion). Il suffit, en effet que le premier agent préfère ω à ω' tandis que le second préfère ω' à ω.

Il est évident qu'on ne peut pas considérer qu'un conflit portant sur des croyances différentes est de même nature que celui portant sur des croyances contradictoires. Afin de les différencier, nous proposons une distance permettant de mesurer l'intensité du conflit. Plus précisément, les états de croyances

sont représentés par une relation d'ordre et il est possible de mesurer leur *concordance* c'est-à-dire leur «propension» à classer les interprétations dans le même ordre. Nous proposons de mesurer la concordance par la distance suivante :

Définition 4.12 *Soit $n = card(\Omega)$ le nombre total d'interprétations, $i \in [1, n]$, $j \in [1, n]$, $\mathcal{O}_\psi$ et $\mathcal{O}_\mu$ deux graphes de préférences :*

- *Soit $\mathbb{1}_\psi$ la fonction indicatrice définie par :*

$$\mathbb{1}_\psi \ : \ \Omega \times \Omega \rightarrow [0, 1]$$

$$\forall(\omega_i, \omega_j) \in \Omega \times \Omega, \ \mathbb{1}_\psi(\omega_i, \omega_j) = \begin{cases} 1 & si\ (\omega_i, \omega_j) \in \mathcal{O}_\psi, \\ 0 & sinon. \end{cases}$$

- *On appelle $c(\mathcal{O}_\psi, \mathcal{O}_\mu)$ la* $\boxed{\textit{distance entre deux graphes de préférences}}$ *mesurant le nombre total de relations élémentaires désaccordées. Cette distance est définie par :*

$$c(\mathcal{O}_\psi, \mathcal{O}_\mu) = \frac{1}{n(n-1)} \sum_{ij} \delta_{ij} \quad avec\ i \neq j$$

$$et \quad \forall(\omega_i, \omega_j) \in \Omega \times \Omega, \ \delta_{ij} = \begin{cases} 0 & si\ \mathbb{1}_\psi(\omega_i, \omega_j) = \mathbb{1}_\mu(\omega_i, \omega_j), \\ 1 & sinon. \end{cases}$$

On constate que :

$$0 \leq c(\mathcal{O}_\psi, \mathcal{O}_\mu) \leq 1$$

Définition 4.13 *Les croyances des deux agents sont* $\boxed{\textit{accordées}}$ *si $c(\mathcal{O}_\psi, \mathcal{O}_\mu) = 0$ et* $\boxed{\textit{en contradiction}}$ *si $c(\mathcal{O}_\psi, \mathcal{O}_\mu) = 1$.*

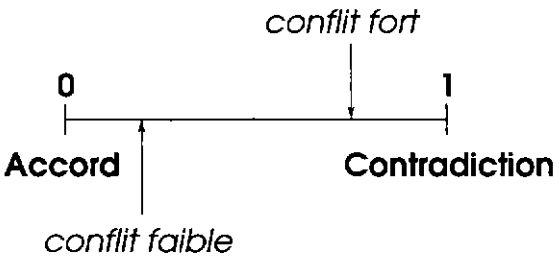


FIG. 4.9 – *Échelle des conflits*

La figure 4.9 montre l'échelle des conflits en fonction de $c(\mathcal{O}_\psi, \mathcal{O}_\mu)$.

Preuve 4.1 *Démontrons que $c(\mathcal{O}_\psi, \mathcal{O}_\mu)$ est une distance. Il est clair que :*

$$\begin{cases} c(\mathcal{O}_\psi, \mathcal{O}_\psi) = 0 \\ c(\mathcal{O}_\psi, \mathcal{O}_\mu) = c(\mathcal{O}_\mu, \mathcal{O}_\psi) \end{cases}$$

Démontrons l'inégalité triangulaire, c'est-à-dire que :

$$c(\mathcal{O}_\psi, \mathcal{O}_\mu) + c(\mathcal{O}_\psi, \mathcal{O}_\varphi) \geq c(\mathcal{O}_\mu, \mathcal{O}_\varphi)$$

Soit $\Gamma = \Omega^2 \backslash \Delta$ avec $\Delta = \{x \in \Omega^2 \mid x = (\omega_i, \omega_i)\}$. On peut toujours définir une partition de $\Gamma = \mathcal{P} \cup \mathcal{Q} \cup \mathcal{R} \cup \mathcal{S}$ telle que :

$$\begin{array}{ll} \mathcal{P} = \{x \in \Gamma \mid \mathbb{I}_\psi(x) = \mathbb{I}_\mu(x) = \mathbb{I}_\varphi(x)\} & card(\mathcal{P}) = p \\ \mathcal{Q} = \{x \in \Gamma \mid \mathbb{I}_\psi(x) = \mathbb{I}_\mu(x) \text{ et } \mathbb{I}_\psi(x) \neq \mathbb{I}_\varphi(x)\} & card(\mathcal{Q}) = q \\ \mathcal{R} = \{x \in \Gamma \mid \mathbb{I}_\psi(x) \neq \mathbb{I}_\mu(x) \text{ et } \mathbb{I}_\psi(x) = \mathbb{I}_\varphi(x)\} & card(\mathcal{R}) = r \\ \mathcal{S} = \{x \in \Gamma \mid \mathbb{I}_\psi(x) \neq \mathbb{I}_\mu(x) \text{ et } \mathbb{I}_\psi(x) \neq \mathbb{I}_\varphi(x)\} & card(\mathcal{S}) = s \end{array}$$

Remarquons que, dans les partitions $\mathcal{R}$ et $\mathcal{Q}$, pour tout $x \in \Gamma$, $\mathbb{I}_\varphi(x) \neq \mathbb{I}_\mu(x)$. On en déduit que :

$$\begin{cases} c(\mathcal{O}_\psi, \mathcal{O}_\mu) = k(r+s) \\ c(\mathcal{O}_\psi, \mathcal{O}_\varphi) = k(q+s) \\ c(\mathcal{O}_\mu, \mathcal{O}_\varphi) = k(q+r) \end{cases} \quad \text{avec } k = \frac{1}{n(n-1)}$$

Par conséquent :

$$c(\mathcal{O}_\psi, \mathcal{O}_\mu) + c(\mathcal{O}_\psi, \mathcal{O}_\varphi) = k[r+q+2s]$$

$$c(\mathcal{O}_\mu, \mathcal{O}_\varphi) = k[r+q]$$

Donc :

$$c(\mathcal{O}_\psi, \mathcal{O}_\mu) + c(\mathcal{O}_\psi, \mathcal{O}_\varphi) \geq c(\mathcal{O}_\mu, \mathcal{O}_\varphi)$$

□

4.4.2 Notion de croyance commune

Nous avons présenté de façon informelle ce que devait être, de notre point de vue, la croyance commune en fonction des différents états de croyances des agents. Nous avons montré que cette notion est liée aux préférences des agents et, en particulier, à leurs convictions. Par conséquent, il semble naturel

de définir la croyance commune à partir d'une *préférence commune* rendant compte des préférences de chaque agent.

Définition 4.14 *On appelle* $\boxed{\textit{préférence commune}}$ *la relation d'ordre* $\leq_{\psi\perp\mu}$ *vérifiant les postulats suivants :*

(P1) *On note* $\mathcal{O}_{\psi\perp\mu}$ *le graphe de préférences associé à* $\leq_{\psi\perp\mu}$. $\mathcal{O}_{\psi\perp\mu}$ *respecte le principe de fusion endogène défini au paragraphe 4.3.1 :*

$$\mathcal{O}_\psi \cap \mathcal{O}_\mu \subseteq \mathcal{O}_{\psi\perp\mu} \subseteq \mathcal{O}_\psi \cup \mathcal{O}_\mu$$

(P2) $\leq_{\psi\perp\mu}$ *est une relation d'ordre total sur* Ω.

(P3) *Les préférences concordantes sont conservées :*

$$\left.\begin{array}{l}\omega \leq_\psi \omega' \\ \omega \leq_\mu \omega'\end{array}\right\} \Rightarrow \omega \leq_{\psi\perp\mu} \omega'$$

Remarque :

- Il découle de **(P1)** et de **(P3)** que, pour toute interprétation ω et ω' de Ω, si $\omega \leq_\psi \omega'$ et $\omega' \leq_\mu \omega$, alors $\omega \leq_{\psi\perp\mu} \omega'$, $\omega' \leq_{\psi\perp\mu} \omega$ et $\omega \approx_{\psi\perp\mu} \omega'$ sont des résultats possibles.
- On peut discuter de l'opportunité d'imposer le postulat suivant : deux agents qui ont les mêmes croyances sont dans le même état de croyances. Formellement :

$$\psi \equiv \mu \Rightarrow \forall \omega, \omega' \in \Omega, \left\{\begin{array}{l}\omega \leq_\psi \omega' \\ \omega \leq_\mu \omega'\end{array}\right.$$

  Si on ne considère pas ce postulat, deux agents, qui ont la même base de faits, peuvent classer les interprétations de façons différentes. Au contraire, si on impose ce postulat, les seuls conflits que nous considérons sont ceux liés à des croyances divergentes sur l'état de l'environnement et non à la façon de classer les interprétations.

Nous définissons à présent les opérateurs de fusion permettant d'élaborer la croyance commune à partir de la préférence commune.

Définition 4.15 *On définit la* $\boxed{\textit{croyance commune}}$ *comme le résultat de la fusion des bases de croyances* ψ *et* μ *par l'opérateur de fusion* $\perp$ *:*

$$\perp : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$$

$$(\psi, \mu) \mapsto \psi \perp \mu$$

tel que les modèles de $\psi \perp \mu$ correspondent aux mondes les plus préférés selon $\leq_{\psi\perp\mu}$:

$$[\![\psi \perp \mu]\!] = min(\Omega, \leq_{\psi\perp\mu})$$

On impose, par ailleurs, que la croyance commune soit indépendante de la syntaxe de ψ et de μ :

$$\left.\begin{array}{l}\psi \equiv \psi' \\ \mu \equiv \mu'\end{array}\right\} \Rightarrow \psi \perp \mu \equiv \psi' \perp \mu'$$

Remarque : On peut toujours construire au moins une relation d'ordre total $\leq_{\psi\perp\mu}$ à partir de $\leq_\psi$ et de $\leq_\mu$: d'après (Chaudron, 1995b), la borne inférieure du treillis des graphes de préférences, c'est-à-dire $\mathcal{O}_\psi \cap \mathcal{O}_\mu$, est une relation d'ordre. Or, à partir d'une relation d'ordre, on peut toujours construire une relation d'ordre total $\mathcal{O}_{\psi\perp\mu}$ en complétant le graphe $\mathcal{O}_\psi \cap \mathcal{O}_\mu$. D'ou la propriété : $\mathcal{O}_\psi \cap \mathcal{O}_\mu \subseteq \mathcal{O}_{\psi\perp\mu}$. Il faut aussi montrer qu'il est possible de construire une telle relation d'ordre total sans contredire le principe de fusion, c'est-à-dire que la propriété $\mathcal{O}_{\psi\perp\mu} \subseteq \mathcal{O}_\psi \cup \mathcal{O}_\mu$ doit être vérifiée.

Nous apportons la preuve de cela en donnant un opérateur de fusion permettant de calculer une préférence commune et dont on démontre qu'il respecte les postulats **(P1-P3)**.

4.5 Un opérateur de fusion sur les préférences

Nous allons voir comment choisir une relation d'ordre total respectant les postulats énoncés pour la préférence commune. Par définition, le rôle de l'opérateur de fusion est de calculer la relation $\leq_{\psi\perp\mu}$ à partir des états de croyances $(\Omega, \leq_\psi, t)$ et $(\Omega, \leq_\mu, t)$. À cet effet, nous proposons une méthode appelée *procédure additive.*

4.5.1 Procédure additive

Définition 4.16 *Soit κ_ψ une fonction qui associe à chaque élément ω de Ω son rang dans $\leq_\psi$. On suppose que $\leq_\psi$ est codée sous la forme d'une partition $P = \{P_0, \ldots, P_n\}$ de Ω où 0 est le rang des interprétations les plus préférées et $n \in \mathbb{N}$ est le rang le plus élevé :*

$$\kappa_\psi : \Omega \to \mathbb{N}$$

$$\forall \omega \in \Omega, \ si\ \omega \in P_i\ alors\ \kappa_\psi(\omega) = i \quad (i \in [0, n])$$

Remarque : on peut construire P à partir de $\mathcal{O}_\psi$ en utilisant l'algorithme suivant :

Algorithme Construire_$P(\mathcal{O}_\psi)$
$n \leftarrow 0$
$\Lambda \leftarrow \Omega$
$P_n \leftarrow min(\Omega, \leq_\psi)$
$\Lambda \leftarrow \Lambda \setminus P_n$
Tant que $\Lambda \neq \emptyset$ **Faire**
$n \leftarrow n + 1$
$P_n \leftarrow min(\Lambda, \leq_\psi)$
$\Lambda \leftarrow \Lambda \setminus P_n$

Exemple : la figure 4.10 illustre le calcul des rangs à partir d'une relation d'ordre total donnée.

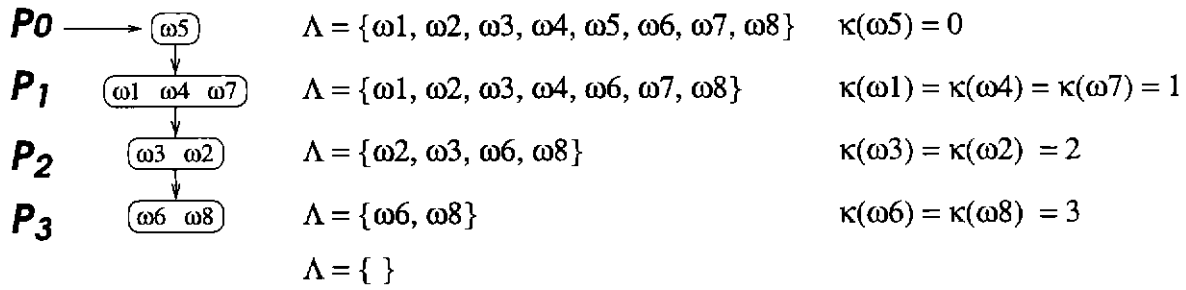


FIG. 4.10 – *Calcul des rangs*

Définition 4.17 *La* $\boxed{\textit{procédure additive}}$, *notée* $\oplus$, *telle que :*

$$\oplus : \ \Omega^2 \times \Omega^2 \rightarrow \Omega^2$$

est définie par l'algorithme suivant :

Algorithme $\oplus$
Pour tout $\omega \in \Omega$ **Faire**
$\boxed{n \leftarrow \kappa_\psi(\omega) + \kappa_\mu(\omega)}$
$Liste \leftarrow (\omega, n)$
$P \leftarrow$ *Trier(Liste)*

On note $\mathcal{O}_\psi \oplus \mathcal{O}_\mu$ la préférence commune obtenue par la procédure additive.

La fonction Trier construit la partition P caractérisant la préférence commune $\leq_{\psi\perp\mu}$. La fonction $\kappa_{\psi\perp\mu}$ donne le rang des interprétations à partir de P.

Exemple : la figure 4.11 montre le calcul de la préférence commune. On constate que l'agent μ croit que l'environnement est dans l'état ω_5 *ou* dans l'état ω_1. L'agent ψ préfère l'état ω_5 à l'état ω_1. Il résulte de la procédure additive que l'état le plus préféré est ω_5. On constate aussi que la somme des rangs d'une interprétation ne correspond pas à son rang dans la préférence commune. C'est, par exemple, le cas de ω_8 dont le rang dans la préférence commune vaut 5 alors que la somme des rangs vaut 6.

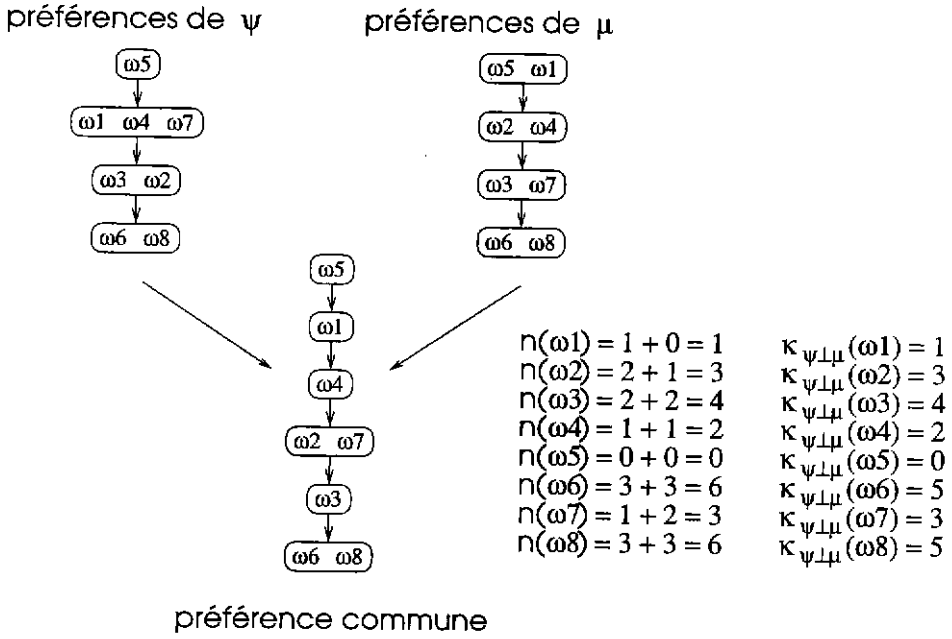


FIG. 4.11 – *Procédure additive*

4.5.2 Propriétés de la procédure additive

Adéquation de la procédure additive

Propriété 4.3 *La relation d'ordre calculée par la procédure additive vérifie les postulats* **(P1-P3)**.

Preuve 4.2 **(P2-P3)** *se vérifient aisément.*

Nous allons démontrer **(P1)** *c'est-à-dire que la procédure additive conserve les relations élémentaires communes à $\mathcal{O}_\psi$ et $\mathcal{O}_\mu$ (i. e. $\mathcal{O}_\psi \cap \mathcal{O}_\mu \subseteq \mathcal{O}_\psi \oplus \mathcal{O}_\mu$), et qu'elle n'ajoute aucune nouvelle relation (i. e. $\mathcal{O}_\psi \oplus \mathcal{O}_\mu \subseteq \mathcal{O}_\psi \cup \mathcal{O}_\mu$). Rappelons que (ω, ω') signifie que ω est préférée à ω' :*

$$\forall(\omega,\omega') \in \mathcal{O}_\psi \cap \mathcal{O}_\mu, \begin{cases} \kappa_\psi(\omega) \leq \kappa_\psi(\omega') \\ \kappa_\mu(\omega) \leq \kappa_\mu(\omega') \end{cases}$$

$$\kappa_\psi(\omega) + \kappa_\mu(\omega) \leq \kappa_\psi(\omega') + \kappa_\mu(\omega')$$

$$\kappa_{\psi\perp\mu}(\omega) \leq \kappa_{\psi\perp\mu}(\omega')$$

Par conséquent :

$$(\omega,\omega') \in \mathcal{O}_\psi \oplus \mathcal{O}_\mu$$

$$\mathcal{O}_\psi \cap \mathcal{O}_\mu \subseteq \mathcal{O}_\psi \oplus \mathcal{O}_\mu$$

D'autre part, les préférences sont des relations d'ordre total, c'est-à-dire que tout couple d'élément de l'ensemble des interprétations est comparable : cela suppose que, soit (ω,ω'), soit (ω',ω), soit les deux sont éléments des graphes de préférences. Supposons que :

$$\begin{cases} (\omega,\omega') \in \mathcal{O}_\psi \cap \mathcal{O}_\mu \\ (\omega',\omega) \notin \mathcal{O}_\psi \cap \mathcal{O}_\mu \end{cases}$$

On en déduit que :

$$\begin{cases} \kappa_\psi(\omega) < \kappa_\psi(\omega') \\ \kappa_\mu(\omega) < \kappa_\mu(\omega') \end{cases} \quad \textit{donc } \kappa_{\psi\perp\mu}(\omega) < \kappa_{\psi\perp\mu}(\omega')$$

parce que, lorsque $(\omega,\omega') \in \mathcal{O}_\psi$ **et** *$(\omega',\omega) \in \mathcal{O}_\psi$, $\kappa_\psi(\omega) = \kappa_\psi(\omega')$. Par conséquent :*

$$\begin{cases} (\omega,\omega') \in \mathcal{O}_\psi \oplus \mathcal{O}_\mu \\ (\omega',\omega) \notin \mathcal{O}_\psi \oplus \mathcal{O}_\mu \end{cases}$$

La relation élémentaire (ω',ω) qui n'était ni dans $\mathcal{O}_\psi$, ni dans $\mathcal{O}_\mu$ n'apparaît pas non plus dans $\mathcal{O}_{\psi\perp\mu}$. Donc :

$$\mathcal{O}_\psi \oplus \mathcal{O}_\mu \subseteq \mathcal{O}_\psi \cup \mathcal{O}_\mu$$

Le cas où (ω,ω') et (ω',ω) font partie des graphes de préférences n'apporte rien à la démonstration car nécessairement $\mathcal{O}_\psi \oplus \mathcal{O}_\mu \subseteq \mathcal{O}_\psi \cup \mathcal{O}_\mu$. □

Notion d'équité

La procédure additive retient toutes les préférences accordées des agents. Implicitement, elle est fondée sur l'idée de l'élaboration d'une croyance commune qui corresponde à un *changement minimal* par rapport aux états de croyances $(\Omega, \leq_\psi, t)$ et $(\Omega, \leq_\mu, t)$. Contrairement aux travaux sur la mise à jour des croyances (Katsuno and Mendelzon, 1991b; Katsuno and Mendelzon, 1991a), ce changement minimal ne se justifie pas par rapport au postulat d' «inertie» de l'environnement. Nous considérons, dans cette approche, qu'aucun agent n'est plus fiable qu'un autre (d'autres approches sont, au contraire, fondées sur des relations d'ordre sur les agents (Cholvy, 1993; Cholvy, 1998) ou sur la notion de crédibilité des agents (Djeffal and Chaib-Draa, 1998)). Par conséquent, tous les états de croyances ont le même poids dans la procédure d'élaboration de la préférence commune.

Une propriété souhaitable pour l'opérateur de fusion serait de produire une relation d'ordre «équitable» c'est-à-dire telle que la concordance qu'un agent obtient avec la préférence commune soit, dans le pire des cas, égale à la concordance qu'il avait avec l'autre agent :

Définition 4.18 *Un opérateur de fusion $\perp$ est* $\boxed{\textit{équitable}}$ *si et seulement si :*

$$\begin{cases} c(\mathcal{O}_{\psi\perp\mu}, \mathcal{O}_\psi) \leq c(\mathcal{O}_\psi, \mathcal{O}_\mu) \\ c(\mathcal{O}_{\psi\perp\mu}, \mathcal{O}_\mu) \leq c(\mathcal{O}_\psi, \mathcal{O}_\mu) \end{cases}$$

Propriété 4.4 *La procédure additive est équitable.*

Preuve 4.3 *Soit $\Gamma = \Omega^2 \backslash \Delta$ avec $\Delta = \{x \in \Omega^2 \mid x = (\omega_i, \omega_i)\}$. On peut toujours définir une partition de $\Gamma = \mathcal{P} \cup \mathcal{Q} \cup \mathcal{R} \cup \mathcal{S}$ telle que :*

$$\begin{array}{ll} \mathcal{P} = \{x \in \Gamma \mid \mathbb{I}_{\psi\perp\mu}(x) = \mathbb{I}_\psi(x) = \mathbb{I}_\mu(x)\} & card(\mathcal{P}) = p \\ \mathcal{Q} = \{x \in \Gamma \mid \mathbb{I}_{\psi\perp\mu}(x) = \mathbb{I}_\psi(x) \text{ et } \mathbb{I}_{\psi\perp\mu}(x) \neq \mathbb{I}_\mu(x)\} & card(\mathcal{Q}) = q \\ \mathcal{R} = \{x \in \Gamma \mid \mathbb{I}_{\psi\perp\mu}(x) \neq \mathbb{I}_\psi(x) \text{ et } \mathbb{I}_{\psi\perp\mu}(x) = \mathbb{I}_\mu(x)\} & card(\mathcal{R}) = r \\ \mathcal{S} = \{x \in \Gamma \mid \mathbb{I}_{\psi\perp\mu}(x) \neq \mathbb{I}_\psi(x) \text{ et } \mathbb{I}_{\psi\perp\mu}(x) \neq \mathbb{I}_\mu(x)\} & card(\mathcal{S}) = s \end{array}$$

Il en découle que :

$$\begin{cases} c(\mathcal{O}_{\psi\perp\mu}, \mathcal{O}_\psi) = k(r+s) \\ c(\mathcal{O}_{\psi\perp\mu}, \mathcal{O}_\mu) = k(q+s) \\ c(\mathcal{O}_\psi, \mathcal{O}_\mu) = k(q+r) \end{cases} \quad \text{avec } k = \frac{1}{n(n-1)}$$

Les éléments x de $\mathcal{S}$ sont tels que : soit $x \in \mathcal{O}_{\psi\perp\mu}$, mais $x \notin \mathcal{O}_\psi$ et $x \notin \mathcal{O}_\mu$; soit $x \notin \mathcal{O}_{\psi\perp\mu}$, mais $x \in \mathcal{O}_\psi$ et $x \in \mathcal{O}_\mu$. Or, d'après la propriété 4.3, la procédure additive n'introduit pas de relation élémentaire externe aux deux graphes de préférences et conserve les préférences accordées. Par conséquent :

$$s = 0$$

$$\begin{cases} c(\mathcal{O}_\psi \oplus \mathcal{O}_\mu, \mathcal{O}_\psi) = k(r) \\ c(\mathcal{O}_\psi \oplus \mathcal{O}_\mu, \mathcal{O}_\mu) = k(q) \\ c(\mathcal{O}_\psi, \mathcal{O}_\mu) = k(q+r) \end{cases} \quad \textit{avec } k = \frac{1}{n(n-1)}$$

D'où :

$$\begin{cases} c(\mathcal{O}_\psi \oplus \mathcal{O}_\mu, \mathcal{O}_\psi) \leq c(\mathcal{O}_\psi, \mathcal{O}_\mu) \\ c(\mathcal{O}_\psi \oplus \mathcal{O}_\mu, \mathcal{O}_\mu) \leq c(\mathcal{O}_\psi, \mathcal{O}_\mu) \end{cases}$$

□

Exemple: Supposons que $\Omega = \{\omega, \omega'\}$: l'agent ψ préfère ω à ω' et l'agent μ préfère ω' à ω. On en déduit que $\mathcal{O}_\psi = \{(\omega,\omega'), (\omega,\omega), (\omega',\omega')\}$, $\mathcal{O}_\mu = \{(\omega',\omega), (\omega,\omega), (\omega',\omega')\}$ et $\mathcal{O}_\psi \oplus \mathcal{O}_\mu = \{(\omega,\omega'), (\omega',\omega), (\omega,\omega), (\omega',\omega')\}$:

$$c(\mathcal{O}_\psi, \mathcal{O}_\mu) = 1$$

$$c(\mathcal{O}_\psi \oplus \mathcal{O}_\mu, \mathcal{O}_\psi) = c(\mathcal{O}_\psi \oplus \mathcal{O}_\mu, \mathcal{O}_\mu) = \frac{1}{2} \times 1 = \frac{1}{2}$$

Par définition, deux agents ayant les mêmes préférences auront une préférence commune totalement accordée avec elles. Il faut vérifier que d'autres cas singuliers ne produisent pas une préférence commune contraire à l'intuition.

Effet d'entraînement

Lorsqu'un agent est moins informé qu'un autre agent, on constate que la préférence commune «suit» les préférences de l'agent le moins ignorant. Cet effet est illustré par la figure 4.12 :

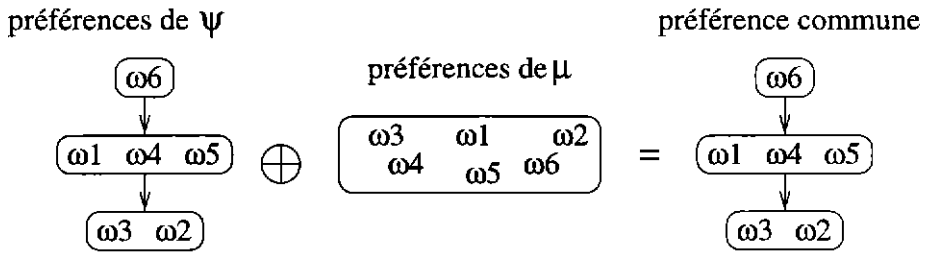


FIG. 4.12 – *Effet d'entraînement*

On en déduit que Ω^2 est l'élément neutre pour la procédure additive :

Propriété 4.5

$$\mathcal{O}_\psi \oplus \Omega^2 = \Omega^2 \oplus \mathcal{O}_\psi = \mathcal{O}_\psi$$

Preuve 4.4

$$\mathcal{O}_\mu = \Omega^2 \text{ si et seulement si } \forall\omega \in \Omega,\ \kappa_\mu(\omega) = 0$$

Par conséquent :

$$\forall\omega \in \Omega,\ \kappa_{\psi\perp\mu}(\omega) = \kappa_\psi(\omega) \text{ et } \mathcal{O}_\psi \oplus \mathcal{O}_\mu = \mathcal{O}_\psi$$

□

Effet de contradiction

Lorsque deux agents sont en total désaccord, la préférence commune est équivalente à l'ignorance. Cet effet est illustré par la figure 4.13 :

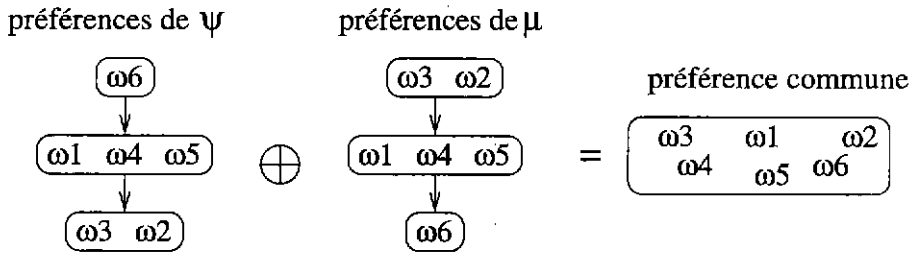


FIG. 4.13 – *Effet de contradiction*

On constate que l'opposé de $\mathcal{O}_\psi$ pour la procédure additive est sa relation duale $\mathcal{O}_\psi^{-1}$:

Propriété 4.6

$$\mathcal{O}_\psi \oplus \mathcal{O}_\psi^{-1} = \mathcal{O}_\psi^{-1} \oplus \mathcal{O}_\psi = \Omega^2$$

Preuve 4.5 *Soit $n_\psi,\ n_\mu \in \mathbb{N}$ les rangs maximaux des interprétations dans $\mathcal{O}_\psi$ et dans $\mathcal{O}_\mu$:*

$$\forall\omega \in \Omega,\ \begin{cases} \kappa_\psi(\omega) \leq n_\psi \\ \kappa_\mu(\omega) \leq n_\mu \end{cases}$$

$$\text{si } \mathcal{O}_\psi = \mathcal{O}_\mu^{-1} \text{ alors } n_\psi = n_\mu = n \text{ et } \forall\omega \in \Omega,\ \kappa_\psi(\omega) = n - \kappa_\mu(\omega)$$

On en déduit que :

$$\forall\omega \in \Omega,\ \kappa_\psi(\omega) + \kappa_\mu(\omega) = n \text{ donc } \kappa_{\psi\perp\mu}(\omega) = 0 \text{ et } \mathcal{O}_\psi \oplus \mathcal{O}_\mu = \Omega^2$$

□

Conclusion

Dans ce chapitre, nous nous sommes intéressé au problème de la représentation de l'état de l'environnement par un agent au moyen d'un langage logique. Puis, nous avons distingué cet état, qui est unique, de l'état de croyances d'un agent, qui est un ensemble ordonné par une relation de préférences d'états possibles de l'environnement : ce mode de représentation de préférences rend compte de l'incertitude de l'agent.

Nous avons ensuite introduit la notion de croyance commune comme un cas particulier de conjecture commune représentant la croyance d'une fédération d'agents sur l'état de l'environnement obtenue par la fusion de leurs préférences locales. À cet effet, nous avons défini un cadre formel délimitant ce qui, de notre point de vue, caractérise la notion de croyance commune (Fiorino and Maille, 1998).

En particulier, nous avons fondé cette notion sur plusieurs principes :

- la définition de la croyance commune comme la représentation des mondes les plus préférés par les agents : grâce à la procédure additive, le choix des mondes équivaut à un vote majoritaire. Ainsi, les croyances d'un agent peuvent venir renforcer ou, au contraire, infirmer celles d'un autre agent.
- la persistance des préférences accordées ;
- l'utilisation d'un opérateur de fusion n'introduisant pas d'éléments externe à la fédération.

La procédure additive permet de calculer la croyance commune quelles que soient les préférences de chaque agent : cette approche, fondée sur les interprétations, est classique (Gärdenfors, 1988; Katsuno and Mendelzon, 1991a; Dalal, 1989) car elle permet de caractériser la sémantique des opérations effectuées sur les interprétations (e. g. la fusion de préférences). Du point de vue de la mise en œuvre, il peut être difficile de faire interagir les agents par le biais d'interprétations. En effet, calculer la croyance commune nécessite de classer 2^n interprétations pour un langage comportant n formules atomiques.

Dans le chapitre suivant, nous introduisons une procédure permettant de calculer la croyance commune directement à partir des bases de croyances de chaque agent et de la distance de Hamming comme relation de préférences : ainsi, en associant une métrique particulière aux bases de croyances, il n'est plus nécessaire de manipuler les interprétations. Nous définissons ensuite les opérateurs de mise à jour du modèle de coopération permettant d'élaborer la croyance commune dans le cadre d'une fédération d'agents.

Chapitre 5

Un cas particulier de préférences fondées sur la distance de Hamming

Introduction

Jusqu'à présent, nous avons défini de manière formelle un modèle de coopération, la notion de conflit et la notion de croyance commune. Nous avons supposé que les agents ont des préférences sur les interprétations et nous avons donné des postulats permettant de définir la croyance commune à partir de ces préférences.

Dans la première partie de ce chapitre, nous proposons un cas particulier de préférences fondées sur la distance de Hamming. Puis nous étudions leurs propriétés relativement à la procédure additive.

5.1 Un exemple de relation de préférences : la relation de Hamming

5.1.1 Définitions

Jusqu'à présent, nous avons supposé que les agents avaient des préférences mais nous n'avons pas imposé de méthode constructive pour les représenter : la seule contrainte que nous avons introduite concerne la rationalité des agents ; elle exige de préférer les modèles à toute autre interprétation. Ainsi, la répartition la plus simple consiste à séparer les interprétations en deux classes d'équivalence : d'une part les modèles de la base de croyances, d'autre part, toutes les interprétations qui ne satisfont pas la base de croyances.

Nous proposons ici une relation de préférences plus fine fondée sur la distance de Hamming.

Soit Ξ un ensemble de variables propositionnelles et $n = card(\Xi)$; soit ω et ω' deux interprétations de Ω, $x_i \in \omega$ et $x'_i \in \omega'$ les valeurs de vérité assignées aux variables respectivement dans ω et ω'. On note $h(\omega, \omega')$ la distance de Hamming (Simon, 1984; Garcia, 1993) entre deux interprétations ω et ω', définie par :

$$h(\omega, \omega') = \sum_{i=1}^{i=n} \delta_i$$

avec

$$\delta_i = \begin{cases} 1 & si\ x_i \neq x'_i, \\ 0 & sinon. \end{cases}$$

Exemple: supposons que $\Xi = \{abc\}$, $\omega =< faux, vrai, vrai >$ et $\omega' =< vrai, faux, vrai >$, alors $h(\omega, \omega') = 2$.

On étend cette définition pour évaluer la distance de Hamming entre deux ensembles d'interprétations $\mathcal{M}$ et $\mathcal{M}'$:

$$h(\mathcal{M}, \mathcal{M}') = min_{\omega \in \mathcal{M},\ \omega' \in \mathcal{M}'} h(\omega, \omega')$$

c'est-à-dire la distance minimale entre $\mathcal{M}$ et $\mathcal{M}'$.

Il est maintenant possible de définir une relation de préférences :

Définition 5.1 *Soit $P = \{P_0, \ldots, P_n\}$ une partition de Ω et $\leq_\psi$ la relation de préférences de l'agent ψ définie à partir de P. $\leq_\psi$ est une* $\boxed{\textit{relation de Hamming}}$ *si et seulement si :*

$$P_0 = [\![\psi]\!]$$

et

$$\forall \omega \in P_i,\ h(\omega, P_0) = i$$

$i \in [0, n]$ est le $\boxed{\textit{rang}}$ *de ω dans $\leq_\psi$.*

Conformément aux notations du chapitre précédent :

$$\kappa_\psi(\omega) = h(\omega, [\![\psi]\!]) = i$$

La relation de Hamming est fondée sur l'idée qu'une interprétation est d'autant plus préférée qu'elle est «proche» des modèles de la base de croyances. La notion de proximité est ici définie en termes du nombre de variables prenant la même valeur entre deux interprétations.

Rappelons que P_i est une classe d'équivalence au sens de la relation de préférences et que les interprétations de P_{i+1} sont moins préférées que celles de P_i.

Exemple : considérons un langage propositionnel $\mathcal{L}$ composé de n variables (e. g. $n = 3$). On peut représenter une interprétation par un point dans $\mathbb{R}^n$; l'ensemble des interprétations forme alors un hypercube de dimension n (Simon, 1984; Carvallo, 1962) tel que, si ω_i et ω_j sont les extrémités d'une arête alors $h(\omega_i, \omega_j) = 1$ (cf. figure 5.1). Ainsi, $h(\omega_4, \omega_1) = 1$, car ces deux interprétations diffèrent exactement d'une valeur de vérité : $\omega_1 =< vrai, faux, vrai >=< a, c >$ et $\omega_4 =< faux, faux, vrai >=< c >$:

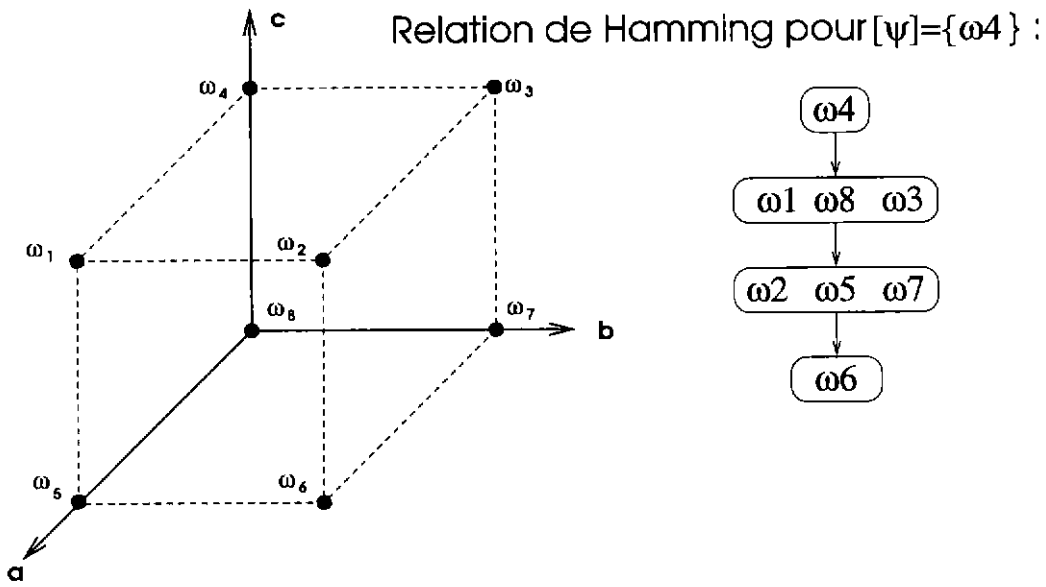


FIG. 5.1 – *Exemple d'une relation de Hamming*

$$P_0 = \{\omega_4\}$$

$$P_1 = \{\omega_1, \omega_3, \omega_8\}$$

$$P_2 = \{\omega_2, \omega_5, \omega_7\}$$

$$P_3 = \{\omega_6\}$$

5.1.2 Procédure additive et relation de Hamming

Soit $n(\omega)$ la somme des rangs d'une interprétation dans les relations de Hamming de deux agents ψ et μ. La procédure additive s'écrit :

$$\kappa_\psi(\omega) + \kappa_\mu(\omega) = n(\omega)$$

Par conséquent :

$$h(\omega, [\![\psi]\!]) + h(\omega, [\![\mu]\!]) = n(\omega)$$

Par définition, les modèles de la croyance commune, c'est-à-dire les interprétations les plus préférées par rapport à $\leq_{\psi\perp\mu}$, sont les interprétations qui minimisent $n(\omega)$. Par conséquent :

Propriété 5.1

$$\forall\omega \in [\![\psi \perp \mu]\!],\ h(\omega, [\![\psi]\!]) + h(\omega, [\![\mu]\!]) = h([\![\psi]\!], [\![\mu]\!])$$

Preuve 5.1 *Ce résultat est obtenu en appliquant l'inégalité triangulaire. Soit $\omega_\psi \in [\![\psi]\!]$ et $\omega_\mu \in [\![\mu]\!]$ deux interprétations telles que :*

$$h(\omega, [\![\psi]\!]) = h(\omega, \omega_\psi) \text{ et } h(\omega, [\![\mu]\!]) = h(\omega, \omega_\mu)$$

On en déduit que :

$$h(\omega, \omega_\psi) + h(\omega, \omega_\mu) \geq h(\omega_\psi, \omega_\mu)$$

Par ailleurs, la distance entre ω_ψ et ω_μ est, par définition, supérieure ou égale à la distance entre $[\![\psi]\!]$ et $[\![\mu]\!]$:

$$h(\omega_\psi, \omega_\mu) \geq h([\![\psi]\!], [\![\mu]\!])$$

On en déduit que :

$$h(\omega, \omega_\psi) + h(\omega, \omega_\mu) \geq h(\omega_\psi, \omega_\mu) \geq h([\![\psi]\!], [\![\mu]\!])$$

Comme la distance de Hamming est un entier naturel, on peut écrire que :

$$h(\omega, \omega_\psi) + h(\omega, \omega_\mu) = h([\![\psi]\!], [\![\mu]\!]) + k$$

Par ailleurs :

$$n(\omega) = h(\omega, [\![\psi]\!]) + h(\omega, [\![\mu]\!]) = h(\omega, \omega_\psi) + h(\omega, \omega_\mu) = h([\![\psi]\!], [\![\mu]\!]) + k$$

Quel que soit $\omega \in [\![\psi]\!]$, on a $h(\omega, [\![\psi]\!]) = 0$ et :

$$n(\omega) = h(\omega, [\![\mu]\!])$$

Comme la distance minimale entre $[\![\psi]\!]$ et $[\![\mu]\!]$ est $h([\![\psi]\!], [\![\mu]\!])$, on en déduit qu'il existe au moins un élément $\omega \in [\![\psi]\!]$ tel que $n(\omega) = h([\![\psi]\!], [\![\mu]\!])$ et donc que $n(\omega)$ est minimale lorsque $k = 0$.

□

Par ailleurs, on peut faire l'analogie avec une distribution en forme d'ellipse car le classement d'une interprétation ω par rapport à la relation d'ordre $\leq_{\psi \perp \mu}$ est déterminé par la somme des distances de Hamming de ω à $[\![\psi]\!]$ et à $[\![\mu]\!]$: les modèles de la croyance commune sont une classe d'équivalence (i. e. tous les modèles sont également préférés) qui représente l'incertitude de la fédération des agents sur l'état de l'environnement.

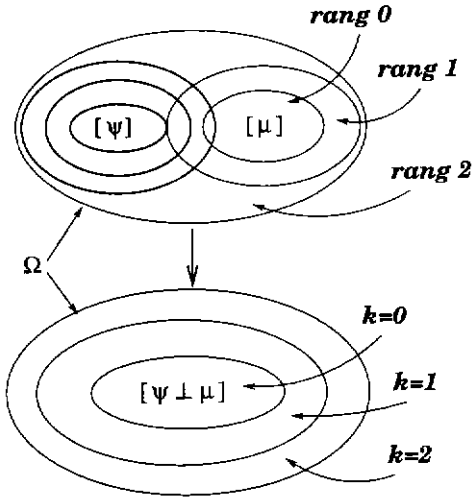


FIG. 5.2 – «*Distribution*» *des interprétations*

5.2 Notion de cube

5.2.1 Définitions

Nous introduisons à présent des définitions qui seront utiles pour donner les propriétés de la procédure additive opérant sur des relations de Hamming.

Définition 5.2 *Soit $\xi \subseteq \Xi$, un sous-ensemble des variables propositionnelles du langage $\mathcal{L}$ et $\mathcal{V}_\xi$ une* $\boxed{\textit{valuation}}$ *telle que :*

$$\mathcal{V}_\xi \ : \ \xi \rightarrow \{vrai, faux\}$$

On appelle ξ le $\boxed{\textit{support}}$ *de la valuation $\mathcal{V}_\xi$.*

Rappelons qu'une interprétation est une fonction I de l'ensemble des variables propositionnelles Ξ vers $\{vrai, faux\}$ (cf. paragraphe 4.2).

Définition 5.3 *Soit $\xi \subseteq \Xi$ et $m = card(\Xi \backslash \xi)$. On appelle* $\boxed{\textit{cube}}$ *l'ensemble des 2^m interprétations, noté $C_{\mathcal{V}_\xi}$, telles que :*

$$\forall x \in \xi, \ I(x) = \mathcal{V}_\xi(x)$$

m est la $\boxed{\textit{dimension}}$ *du cube.*

Exemple : soit $\Xi = \{a, b, c\}$, $\xi = \{c\}$ et $\mathcal{V}_\xi(c) = \{faux\}$. $\{\omega_5, \omega_6, \omega_7, \omega_8\}$ est un cube (cf. figure 5.1).

Notations : on note x les variables d'une valuation $\mathcal{V}_\xi$ prenant la valeur *vrai* et $\bar{x}$ toutes les autres. Par convention, $\bar{\bar{x}} = x$. On note $\mathcal{E}$ l'ensemble des variables valuées. Dans l'exemple précédent, $\mathcal{E} = \{\bar{c}\}$. On appelle *variables libres* les variables de $\Xi \backslash \xi$ et on note $\mathcal{F}$ cet ensemble. Dans le reste de ce chapitre, nous raisonnons sur $\mathcal{E}$ et $\mathcal{F}$ afin de ne pas surcharger les définitions et les démonstrations : il est, en effet, toujours possible de se ramener de façon univoque à la notion de support et de valuation à partir de $\mathcal{E}$ et $\mathcal{F}$.

Pour représenter un cube, on écrit ses variables valuées x ou $\bar{x}$ et on note les variables libres entre parenthèses. Ainsi, le cube de l'exemple est noté $C_{\mathcal{V}_\xi} = \bar{c}(a)(b)$ ou, de façon équivalente, $C_{\mathcal{V}_\xi} = \bar{c}(ab)$.

Le passage d'un cube à la syntaxe est immédiat : en effet, $\bar{c}(ab)$ sont les modèles de la proposition $\neg c$. Nous préférons cependant raisonner sur les cubes.

Remarque : il existe un lien fort entre ces définitions et l'algèbre de Boole (Carvallo, 1962). Ainsi, $\bar{c}(ab)$ est une forme contractée de $ab\bar{c} + \bar{a}b\bar{c} + a\bar{b}\bar{c} + \bar{a}\bar{b}\bar{c}$.

Définition 5.4 *Soit $C_{\mathcal{V}_\xi}$ et $C_{\mathcal{V}'_{\xi'}}$ deux cubes et, $\mathcal{E}$ et $\mathcal{E}'$, leurs ensembles de variables valuées respectifs :*

- *$C_{\mathcal{V}_\xi}$ et $C_{\mathcal{V}'_{\xi'}}$ sont* $\boxed{\textit{orthogonaux}}$ *si et seulement si :*

$$\forall x \in \mathcal{E}, \bar{x} \notin \mathcal{E}' \text{ et } \forall x \in \mathcal{E}', \bar{x} \notin \mathcal{E}$$

- *$C_{\mathcal{V}_\xi}$ et $C_{\mathcal{V}'_{\xi'}}$ sont* $\boxed{\textit{parallèles}}$ *si et seulement si :*

$$\exists x \in \mathcal{E}, \bar{x} \in \mathcal{E}' \text{ ou } \exists x \in \mathcal{E}', \bar{x} \in \mathcal{E}$$

Soit $\mathcal{C}$ l'ensemble des cubes pouvant être définis sur Ξ. Nous allons à présent définir deux opérateurs binaires sur les cubes :

Définition 5.5 *Soit $C_{\mathcal{V}_\xi}$, $C_{\mathcal{V}'_{\xi'}} \in \mathcal{C}$ et, $\mathcal{E}$ et $\mathcal{E}'$, leurs ensembles de variables valuées respectifs. On appelle* $\boxed{\textit{plus petit cube englobant}}$ *de $C_{\mathcal{V}_\xi}$ et $C_{\mathcal{V}'_{\xi'}}$, noté $ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}})$, le cube tel que son ensemble de variables valuées est $\mathcal{E}_{ppce} = \mathcal{E} \cap \mathcal{E}'$.*

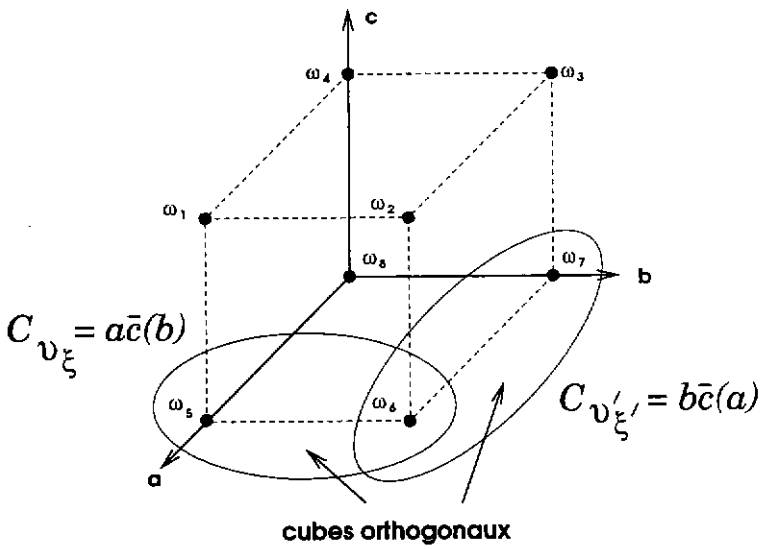


FIG. 5.3 – *Exemple de cubes orthogonaux*

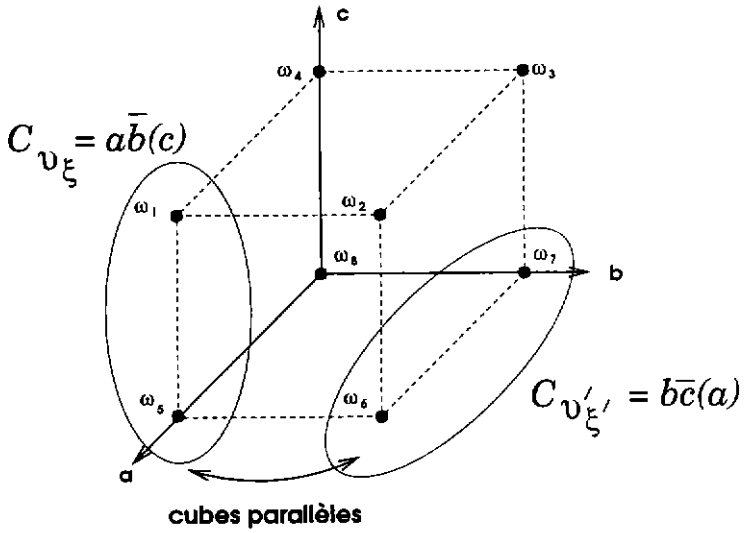


FIG. 5.4 – *Exemple de cubes parallèles*

Exemple: soit $C_{\mathcal{V}_\xi} = a\bar{b}(c)$ et $C_{\mathcal{V}'_{\xi'}} = ab\bar{c}$. Le plus petit cube englobant $ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}}) = a(bc)$ (cf. figure 5.5).

Définition 5.6 *Soit $C_{\mathcal{V}_\xi}$, $C_{\mathcal{V}'_{\xi'}} \in \mathcal{C}$ et, $\mathcal{E}$ et $\mathcal{E}'$, leurs ensembles de variables valuées respectifs. On appelle* $\boxed{\textit{somme de deux cubes}}$*, noté $\dotplus$, un opérateur binaire défini par:*

$$\dotplus \ : \ \mathcal{C} \times \mathcal{C} \rightarrow \mathcal{C}$$

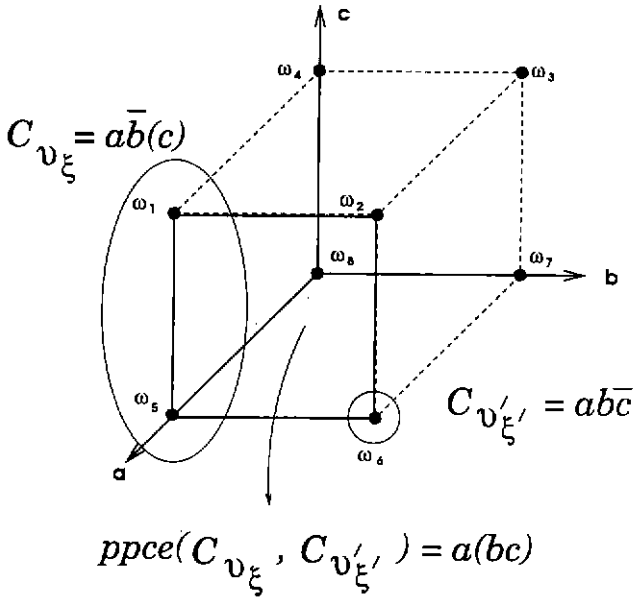


FIG. 5.5 – *Plus petit cube englobant*

$$(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}}) \mapsto C_{\mathcal{V}_\xi} \dot{+} C_{\mathcal{V}'_{\xi'}}$$

tel que $C_{\mathcal{V}_\xi} \dot{+} C_{\mathcal{V}'_{\xi'}}$ *est un cube dont l'ensemble des variables valuées* $\mathcal{E}_{\dot{+}}$ *est tel que :*

$$\mathcal{E}_{\dot{+}} = \Gamma \cup \Gamma' \quad avec \begin{cases} \Gamma = \{x \mid \forall x \in \mathcal{E},\ \bar{x} \notin \mathcal{E}'\} \\ \Gamma' = \{x \mid \forall x \in \mathcal{E}',\ \bar{x} \notin \mathcal{E}\} \end{cases}$$

Exemple : soit $C_{\mathcal{V}_\xi} = a\bar{b}(c)$ et $C_{\mathcal{V}'_{\xi'}} = ab\bar{c}$. $\Gamma = \{a\}$, $\Gamma' = \{a, \bar{c}\}$ et $C_{\mathcal{V}_\xi} \dot{+} C_{\mathcal{V}'_{\xi'}} = a\bar{c}(b)$.

5.2.2 Propriétés

Les relations de Hamming ont des propriétés intéressantes pour la procédure additive. Toutes les propriétés présentées ici sont vérifiées lorsque les convictions des agents sont des cubes.

Propriété 5.2 *Soit* $C_{\mathcal{V}_\xi}$ *et* $C_{\mathcal{V}'_{\xi'}}$ *deux cubes et,* $\mathcal{E}$ *et* $\mathcal{E}'$*, leurs ensembles de variables valuées respectifs :*

- *si* $C_{\mathcal{V}_\xi}$ *et* $C_{\mathcal{V}'_{\xi'}}$ *sont parallèles alors* $C_{\mathcal{V}_\xi} \cap C_{\mathcal{V}'_{\xi'}} = \emptyset$ *;*
- *si* $C_{\mathcal{V}_\xi}$ *et* $C_{\mathcal{V}'_{\xi'}}$ *sont orthogonaux alors* $C_{\mathcal{V}_\xi} \cap C_{\mathcal{V}'_{\xi'}}$ *est un cube tel que son ensemble de variables valuées* $\mathcal{E}_\cap = \mathcal{E} \cup \mathcal{E}'$.

Propriété 5.3 *Soit $C_{\mathcal{V}_\xi}$, $C_{\mathcal{V}'_{\xi'}}$ deux cubes et, $\mathcal{E}$ et $\mathcal{E}'$, leurs ensembles de variables valuées respectifs :*

$$C_{\mathcal{V}_\xi} \subseteq C_{\mathcal{V}'_{\xi'}} \text{ si et seulement si } \mathcal{E}' \subseteq \mathcal{E}$$

On constate, par ailleurs, que l'ensemble des cubes $\mathcal{C}$ muni des opérateurs $\cap$ et *ppce* est un treillis (cf. paragraphe 4.3.1).

Propriété 5.4 $(\mathcal{C}, \subseteq, \cap, ppce)$ *est un treillis.*

Preuve 5.2 *La démonstration a été placée dans l'annexe A.*

Dans le chapitre précédent, nous avons souligné le fait que les modèles de la croyance commune n'étaient pas nécessairement l'intersection ou l'union des convictions des agents : cette caractéristique apparaît explicitement avec l'utilisation de la relation de Hamming.

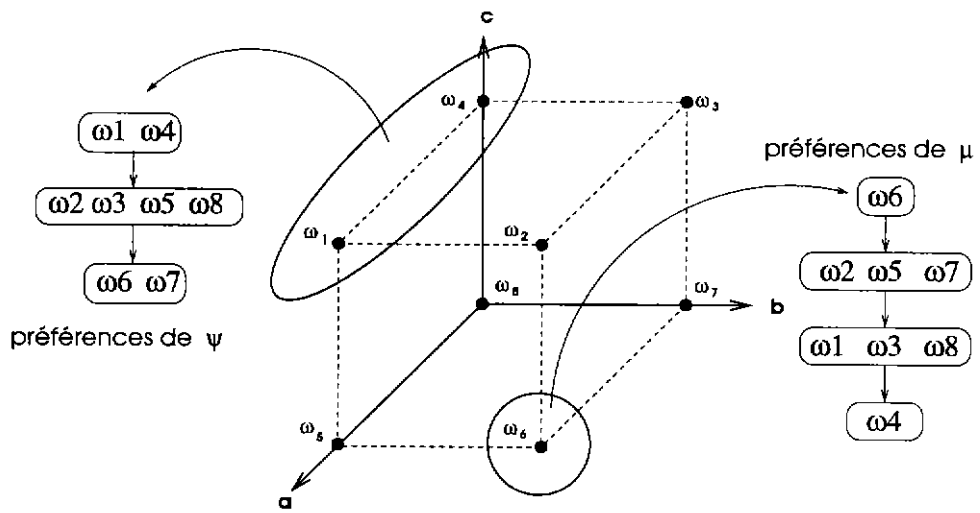


FIG. 5.6 – *Convictions des agents*

La figure 5.6 montre les préférences de deux agents ψ et μ. Soit $[\![\widehat{K}_t^\psi]\!]$ et $[\![\widehat{K}_t^\mu]\!]$ leurs convictions respectives :

$$[\![\widehat{K}_t^\psi]\!] = \{\omega 1, \omega 4\}$$

et

$$[\![\widehat{K}_t^\mu]\!] = \{\omega 6\}$$

On en déduit que :

$$[\![\widehat{K}_t^{\psi}]\!] \cup [\![\widehat{K}_t^{\mu}]\!] = \{\omega 1, \omega 4, \omega 6\}$$

$$[\![\widehat{K}_t^{\psi}]\!] \cap [\![\widehat{K}_t^{\mu}]\!] = \emptyset$$

On constate que :

$$[\![\psi \perp \mu]\!] = \{\omega_1, \omega_2, \omega_5, \omega_6\}$$

et par conséquent, dans le cas général :

$$\begin{cases} [\![\psi \perp \mu]\!] \neq [\![\widehat{K}_t^{\psi}]\!] \cup [\![\widehat{K}_t^{\mu}]\!] \\ [\![\psi \perp \mu]\!] \neq [\![\widehat{K}_t^{\psi}]\!] \cap [\![\widehat{K}_t^{\mu}]\!] \end{cases}$$

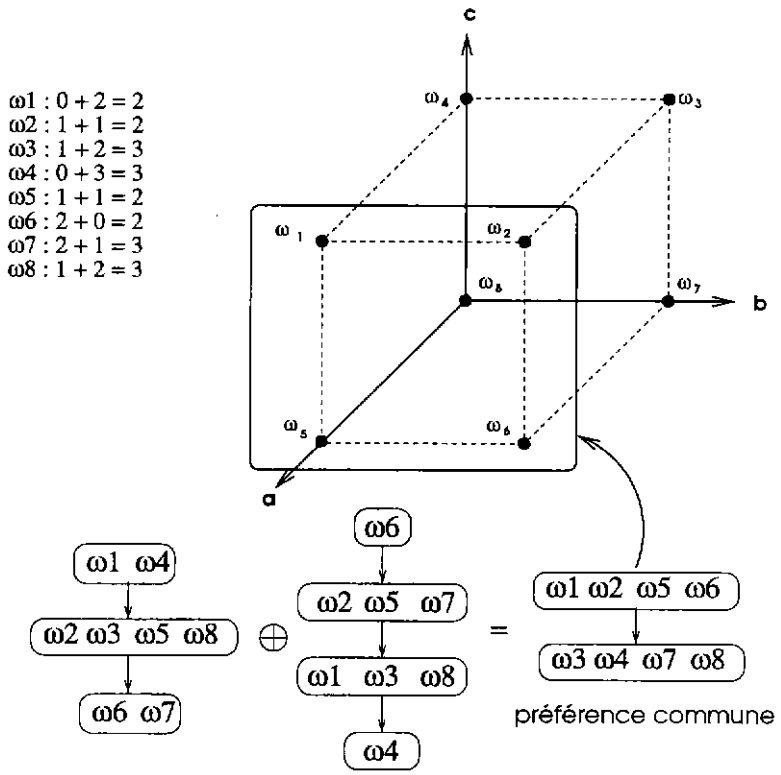


FIG. 5.7 – *Mondes les plus préférés*

Remarque : dans le cas de la relation de Hamming, tous les modèles d'une base de croyances φ sont également préférés ; ils sont tous des éléments de P_0. Par conséquent, les notations suivantes sont équivalentes :

$$[\![\varphi]\!] = [\![\widehat{K}_t^{\varphi}]\!] = P_0$$

Dans le cas général, les modèles de la croyance commune sont donnés par la propriété suivante :

Propriété 5.5 *Soit $[\![\psi]\!]$ et $[\![\mu]\!]$ les convictions de deux agents ψ et μ dont les préférences sont des relations de Hamming. Les modèles de la croyance commune sont tels que :*

$$[\![\psi \perp \mu]\!] = [\![\psi]\!] \dotplus [\![\mu]\!]$$

Exemple : on constate, sur l'exemple de la figure 5.6, que les interprétations de $[\![\psi]\!]$ et $[\![\mu]\!]$ les plus proches sont ω_1 et ω_6 et qu'elles font partie de $[\![\psi \perp \mu]\!]$. Il en est de même pour les interprétations du plus petit cube englobant ces deux interprétations, c'est-à-dire $[\![\psi \perp \mu]\!] = \{\omega_1, \omega_2, \omega_5, \omega_6\}$ (cf. figure 5.7). Vérifions qu'on trouve le même résultat par l'opérateur $\dotplus$: $[\![\psi]\!] = \bar{b}c(a)$, $[\![\mu]\!] = ab\bar{c}$ et $[\![\psi]\!] \dotplus [\![\mu]\!] = a(bc)$, or $a(bc) = \{\omega_1, \omega_2, \omega_5, \omega_6\}$.

Nous avons déjà montré que le passage des cubes à la syntaxe est immédiat. Par conséquent, à partir de ψ et de μ, on peut facilement construire $\psi \perp \mu$. En effet, d'après l'opérateur $\dotplus$, la croyance commune de deux agents est la conjonction leurs croyances à l'exception des croyances mutuellement contradictoires.

Exemple : reprenons l'exemple précédent. Soit $[\![\psi]\!] = \bar{b}c(a)$ et $[\![\mu]\!] = ab\bar{c}$, on en déduit que $\psi = \neg b \wedge c$, $\mu = a \wedge b \wedge \neg c$ et $\psi \perp \mu = a$.

Preuve 5.3 *La démonstration a été placée dans l'annexe A. Son principe est le suivant : à partir de $[\![\psi]\!]$ et $[\![\mu]\!]$, nous construisons deux cubes appelés $\widehat{C}_\psi$ et $\widehat{C}_\mu$. Nous montrons que $ppce(\widehat{C}_\psi, \widehat{C}_\mu) = [\![\psi \perp \mu]\!]$, puis que $ppce(\widehat{C}_\psi, \widehat{C}_\mu) = [\![\psi]\!] \dotplus [\![\mu]\!]$. On en conclut que $[\![\psi]\!] \dotplus [\![\mu]\!] = [\![\psi \perp \mu]\!]$*

On constate, par ailleurs, que, par rapport au treillis des cubes, les modèles de la croyance commune respectent le principe de fusion endogène :

Propriété 5.6

$$[\![\psi]\!] \cap [\![\mu]\!] \subseteq [\![\psi \perp \mu]\!] \subseteq ppce([\![\psi]\!], [\![\mu]\!])$$

Preuve 5.4 *La démonstration a été placée dans l'annexe A.*

Remarque : rappelons que la croyance commune que nous construisons ici est obtenue par la procédure additive appliquée aux relations de Hamming. Par conséquent, elle vérifie nécessairement les postulats (P1-P3) la caractérisant (cf. paragraphe 4.4.2) et, en particulier, le principe de fusion endogène par rapport aux préférences des agents.

Conclusion

Dans ce chapitre, nous avons introduit une relation de préférences fondée sur la distance de Hamming : tous les modèles des croyances d'un agent sont alors également préférés. La procédure additive s'écrit alors comme la somme des distances d'une interprétation aux convictions des agents et les modèles de la croyance commune sont toutes les interprétations telles que cette somme est égale à la distance entre les convictions des agents.

Nous avons ensuite défini la notion de cube, puis nous nous sommes placé dans le cas où les modèles des croyances sont un cube : nous avons montré qu'on pouvait alors construire de façon simple la croyance commune tout en respectant le principe de fusion endogène par rapport au treillis des cubes.

Tous ces résultats sont utilisés dans le chapitre suivant pour illustrer le fonctionnement du modèle de coopération.

Chapitre 6

Synthèse : exemple de reconnaissance d'activités

Introduction

Dans ce chapitre, nous illustrons le fonctionnement du modèle de coopération par un exemple concret.

Nous avons choisi celui de la reconnaissance d'activités dans l'environnement : les agents sont des observateurs intelligents capables de transformer les informations issues de leurs capteurs en propriétés symboliques caractérisant des activités menées par des objets dans l'environnement. Ensemble, ils construisent une croyance commune sur les activités en cours, en confrontant leurs différentes observations qui, parfois, peuvent être conflictuelles.

Pour cela, nous commençons par présenter le dispositif de reconnaissance. Nous reprenons toutes les définitions du chapitre 3 en les instanciant par rapport à cette application. Puis, nous donnons un exemple d'élaboration de croyance commune.

6.1 Dispositif de reconnaissance d'activités

Dans cet exemple, l'objectif est de construire et de mettre à jour une représentation symbolique des activités se déroulant dans un environnement qui inclut des objets en mouvement et est observé par différents types de capteurs (e. g. caméras noir et blanc, couleurs et infrarouges, radars). Des traitements numériques sont chargés de reconnaître et de pister ces objets en délivrant des propriétés symboliques (e. g. type de l'objet : piéton, véhicule) et des attributs numériques (e. g. position, vitesse). Des traitements au niveau symbolique

émettent des hypothèses relatives à ce que *font* les objets et aux situations en cours (e. g. «le piéton va prendre une voiture et quitter le parking»). À partir de ce type d'information, le niveau décisionnel (par exemple des opérateurs humains) peut déclencher des actions adéquates (e. g. des alarmes). On peut envisager ce type de dispositif pour la surveillance de sites sensibles (Castel et al., 1996), la surveillance de circulation au sol dans un aéroport et la détection de risque de collision (Alquier, 1998), ou encore des systèmes d'information et de renseignement militaire pouvant impliquer des observateurs humains (Chaudron, 1995a).

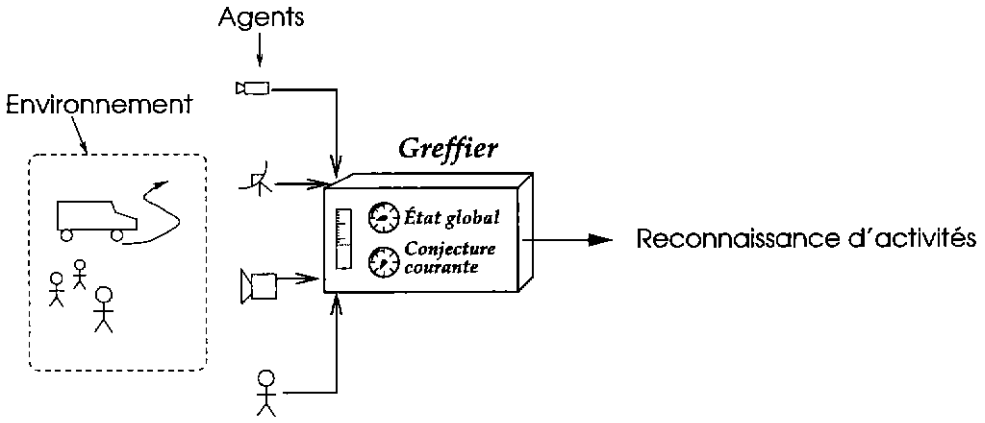


FIG. 6.1 – *Dispositif de reconnaissance d'activités*

La figure 6.1 illustre un dispositif de reconnaissance d'activités : les agents font des *observations* telles que «l'objet x est de type y» ou «la vitesse de x est v» et mettent à jour leurs croyances. Puis, ils élaborent coopérativement une croyance commune à partir de leurs observations. On définit une *activité* (Chaudron et al., 1997a; Castel et al., 1996) comme un ensemble de propriétés caractéristiques. Ainsi, l'activité «piéton se dirigeant vers un véhicule» peut être définie par « l'objet x est de type piéton», «l'objet de type y est de type véhicule», «la vitesse de y est 0» et « x se dirige vers y». Par conséquent, une activité est reconnue lorsque la croyance commune peut être mise en correspondance avec un prototype d'activité donné.

On peut, par ailleurs, envisager d'associer ce dispositif à un système d'élaboration de situation tel qu'il a été étudié dans le projet PERCEPTION (Tessier, 1997) : ainsi, la fédération transmettrait la croyance commune par l'intermédiaire du greffier, au système d'élaboration de situation. Le suivi de situation est assuré par la validation de prototypes de plans qui caractérisent les relations temporelles entre les différentes activités : ainsi le plan «départ en véhicule» est une succession d'activités comme «piéton se dirigeant vers un véhicule»,

« véhicule stationné occupé» et «déplacement véhicule vers sortie». À un instant donné, une situation est caractérisée par un ensemble de plans candidats pouvant expliquer les activités détectées.

6.2 Définitions des éléments du modèle de coopération

Nous supposons que la fédération F est constituée de trois agents que nous appelons ψ, μ et φ où :

- ψ est doté d'une caméra noir et blanc «champ restreint» ;
- μ est doté d'une caméra noir et blanc «champ large» ;
- φ est doté d'une caméra infrarouge.

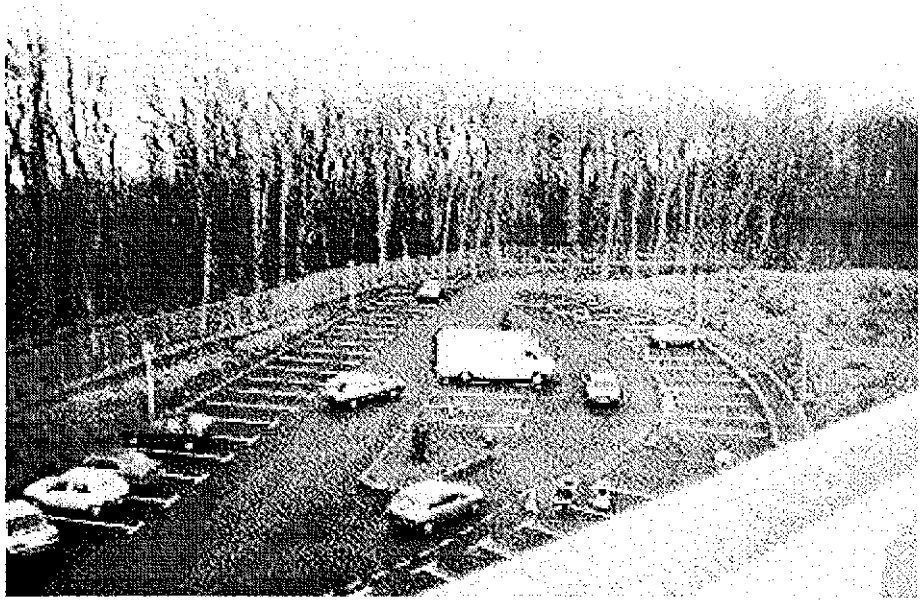


FIG. 6.2 – *Parking surveillé par les agents*

L'environnement des agents est un parking (cf. figure 6.2) : les objets considérés sont les véhicules et les piétons. Le but des agents est de construire et de mettre à jour une représentation symbolique des activités s'y déroulant, comme le départ et l'arrivée de véhicules ou le déplacement de piétons.

Nous supposons par ailleurs que :

- les agents connaissent le décor et les objets immobiles (véhicules stationnés par exemple) avant le démarrage effectif du processus de surveillance ;

- les agents disposent d'une horloge commune ;
- les agents attribuent les mêmes noms aux mêmes objets. Cela suppose qu'ils sont capables de pister les objets dans un référentiel commun. Ainsi, l'objet portant le nom $P1$ désigne de façon univoque un objet ayant les coordonnées (x_1, y_1) par rapport à ce référentiel à la date t ;
- les agents élaborent des croyances communes relatives à l'état de l'environnement à une date donnée. La périodicité Δt du processus d'élaboration est prédéfinie et constante.

Nous allons à présent introduire les définitions des éléments du modèle de coopération adaptées à l'exemple de reconnaissance d'activités : pour cela, nous reprenons les définitions du chapitre 3 en les indexant par les initiales RAC.

Définition$_{RAC}$ 6.1 *Soit $\mathcal{L}$ un langage de la logique des prédicats restreint à un univers de Herbrand fini*[1]*. Soit Ξ l'ensemble des atomes de Herbrand : $Conject = \mathcal{P}(\Xi)$ est l'ensemble de toute les conjectures pouvant être formées.*

On note c_n^t la n^e conjecture construite par les agents relative à l'état de l'environnement à la date t.

6.3 L'agent de reconnaissance d'activités

Les agents du dispositif de reconnaissance d'activités sont capables de transformer les données issues de leurs caméras en propriétés symboliques et de pister les objets de l'environnement. Du point de vue de la coopération, leurs fonctionnalités sont celles définies au chapitre 3, c'est-à-dire la mise à jour de la base de connaissances, la mise à jour de la croyance commune et la prise de décision sur l'engagement.

6.3.1 Mise à jour de la base de connaissances

La base de connaissances K_t d'un agent de reconnaissance d'activités représente l'état de l'environnement du point de vue de cet agent à un instant t donné. Elle se compose de deux parties D_t et C telles que D_t est un ensemble de formules atomiques représentant des faits observés ou déduits et C est l'ensemble des contraintes du domaine, commun à tous les agents du dispositif ; C n'est pas soumis à révision.

1. Formellement, $\mathcal{L}$ est équivalent à un langage propositionnel.

Dans l'exemple de la reconnaissance d'activités dans un parking, les connaissances manipulées par les agents sont les suivantes :

(Identification des objets) les objets reconnus sont soit des véhicules, soit des piétons : la propriété «l'objet x est un y» est codée par le prédicat *type(x,y)* ;

(Vitesse des objets) les agents peuvent évaluer la vitesse d'un objet : cette propriété est codée par le prédicat *vitesse(x,v)* ;

(Direction et proximité) les agents peuvent mettre en relation deux objets et énoncer les propriétés suivantes : «l'objet x est proche de l'objet y» et «l'objet x se rapproche de l'objet y». Ces propriétés sont codées par les prédicats *proche-de(x,y)* et *se-rapproche-de(x,y)*.

Les contraintes du domaine sont les suivantes :

$$C = \begin{cases} \forall x,\ \exists y,\ type(x,y) \\ \forall x,y,z\ type(x,y) \wedge type(x,z) \rightarrow (y = z) \\ \forall x,\ \exists v,\ vitesse(x,v) \\ \forall x,v,w\ vitesse(x,v) \wedge vitesse(x,w) \rightarrow v = w \\ \forall x,y\ se-rapproche-de(x,y) \equiv \neg proche-de(x,y) \end{cases}$$

Les contraintes du domaine donnent une représentation des différents états possibles de l'environnement. En particulier, elles caractérisent le fait que chaque objet a nécessairement un type et une vitesse, et que ce type et cette vitesse sont uniques. Une autre contrainte caractérise le fait que deux objets qui se rapprochent ne sont pas proches.

On peut à présent préciser la nature des informations reçues par les agents :

Définition$_{RAC}$ 6.2 *Une information I_t est un élément du langage : $I_t \in Conject$.*

Une information est donc un ensemble de formules atomiques caractéristiques de l'état de l'environnement à un instant donné. Un agent peut, par exemple, recevoir l'information I_t ={*type(O1,piéton), vitesse(O1, petite)*}.

Remarque : les agents ne peuvent pas recevoir des informations négatives. En effet, ils ne peuvent pas détecter des propriétés comme ¬*type(O1, piéton)*. En revanche, ces propriétés peuvent être inférées à partir des contraintes du domaine.

Nous considérons ici que toutes les informations provenant des capteurs ne peuvent pas être rejetées par l'agent. Ainsi, dans certains cas, l'agent doit retirer des faits afin de rendre K_t *cohérente.*

Définition$_{RAC}$ **6.3** *Soit accept_info la variable déterminant l'admissibilité des éléments d'une information. Comme toutes les informations sont considérées comme* a priori *vraies :*

$$accept_info : \ Conject \rightarrow \{vrai, faux\}$$

$$\boxed{\forall w \in I_t, \ accept_info(w) = vrai}$$

et, pour toute information I_t,

$$Admissible_Info(I_t) = \{ \ w \mid \forall w \in I, \ accept_info(w) = vrai\}$$

est l'ensemble des éléments de I_t *admissibles du point de vue d'un agent.*

Il en découle que :

Définition$_{RAC}$ **6.4**

$$Non-Admissible_Info(I_t) = \emptyset$$

La variable duale de $accept_info(I_t)$, c'est-à-dire $reject_info(I_t)$ est telle que :

Définition$_{RAC}$ **6.5**

$$reject_info : \ Conject \rightarrow \{vrai, faux\}$$

et, pour toute information I_t : $reject_info(I_t) = faux$

La définition de l'opérateur de mise à jour *update* dans le contexte de la reconnaissance d'activités devient :

Définition$_{RAC}$ **6.6** *Soit update l'opérateur de mise à jour de la base de connaissances d'un agent :*

$$update : \ Conject \rightarrow Conject$$

On note $K_t = update(I_t)$ *la base de connaissances mise à jour. On impose que :*

- $I_t \subseteq K_t$,

– K_t *est cohérente.*

Remarque : tous les éléments de I_t sont ajoutés et certains éléments de K_{t-1} sont retirés.

Exemple : soit K_{t-1} ={*type(O1, voiture), vitesse(O1, nulle)*} $\cup$ C et I_t ={*vitesse-(O1, petite)*} : $K_t = update(I_t)$ ={*type(O1, voiture), vitesse(O1, petite)*} $\cup$ C. En effet, à cause d'une contrainte du domaine – $\forall x, v, w\ vitesse(x, v) \wedge vitesse(x, w) \rightarrow v = w$ – les faits *vitesse(O1, nulle)* et *vitesse(O1, petite)* provoquent une contradiction qui est résolue en retirant *vitesse(O1, nulle)* : priorité est donnée aux informations les plus récentes.

6.3.2 Mise à jour de la croyance commune

La croyance commune n'est pas ici le résultat de la fusion des croyances de deux agents, mais, conformément au modèle de coopération, la mise à jour de la croyance commune courante par rapport aux croyances[2] d'un agent. Nous supposons que K_t **est fermée par la déduction** et que les préférences des agents du dispositif sont des relations de Hamming.

Discussion

Jusqu'à présent, nous n'avons pas étudié les conséquences des contraintes du domaine sur l'élaboration de la croyance commune. Or, nous devons distinguer différents cas :

▷ les convictions d'un agent sont un cube : ce cas se produit, par exemple, lorsque les croyances d'un agent sont un monde unique.

La figure 6.3 illustre ce cas : supposons que les variables a, b et c représentent respectivement *type(O1,voiture)*, *type(O1,piéton)* et *type(O1,moto)*. Les états possibles, représentés par les contraintes «un objet n'a qu'un seul type» et «tout objet a nécessairement un type», sont ω_4, ω_5 et ω_7.

Supposons à présent qu'un agent croit K_t ={*type(O1,voiture),¬type(O-1,piéton),¬type(O1,moto)*} c'est-à-dire ω_5 (¬*type(O1,piéton)* et ¬*type-(O1,moto)* sont des faits déduits) et que la croyance commune est c^{t-1} =

2. Rappelons qu'une croyance est un élément de la base de croyances d'un agent, c'est-à-dire de la fermeture de sa base de connaissances par la déduction : les modèles de la base de croyances sont donc les mêmes que ceux de la base de connaissances. Dans le cas de préférences fondées sur une relation de Hamming, tous les modèles sont également préférés et, par conséquent, les convictions d'un agent sont les modèles de sa base de connaissances.

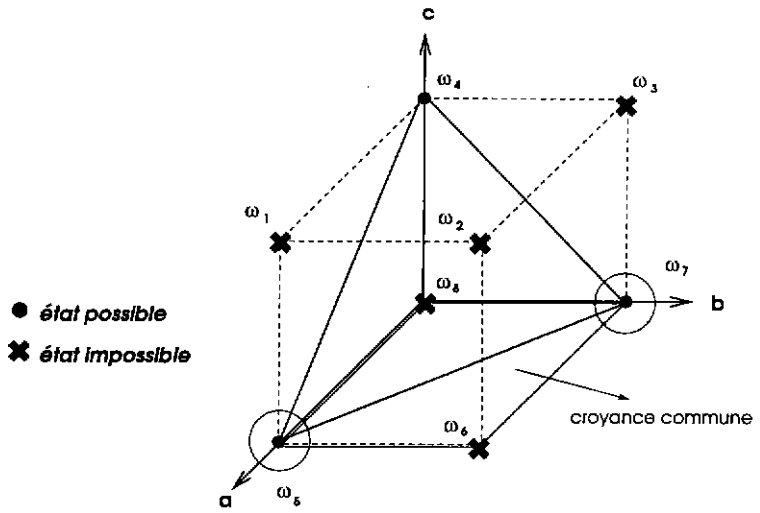


FIG. 6.3 – *Contraintes du domaine*

{*type(O1,piéton)*,¬*type(O1,voiture)*,¬*type(O1,moto)*} (i. e. ω_7). D'après la procédure additive, la nouvelle croyance commune c^t a pour modèles $\{\omega_5, \omega_6, \omega_7, \omega_8\}$ c'est-à-dire c^t ={¬*type(O1,moto)*} : en effet, les faits de la conjecture qui sont contradictoires avec K_t, c'est-à-dire *type(O1,piéton)* et ¬*type(O1,voiture)* sont retirés.

On constate alors que l'intersection des modèles de la croyance commune avec ceux des contraintes du domaine, c'est-à-dire $\{\omega_5, \omega_7\}$, n'est pas un cube. On peut alors soit conserver la croyance commune en l'état, soit envisager une stratégie plus fine pour continuer le processus d'élaboration : la croyance commune est divisée en plusieurs cubes (dans l'exemple, on aurait d'une part ω_5 et d'autre part ω_7) représentant différentes possibilités de l'état de l'environnement : la mise à jour de la croyance commune se ferait alors sur chacune de ses parties. L'intérêt de cette stratégie réside dans la gestion par blocs des hypothèses candidates pour représenter l'état de l'environnement : ainsi la disjonction des blocs permet une représention symbolique de l'incertitude.

Il semble que l'étude des mécanismes de gestion de ces blocs puisse constituer un prolongement intéressant des travaux que nous avons développés (en particulier, nous avons le sentiment que, dans un processus continu de mise à jour des connaissances des agents, ces blocs devraient pouvoir être fusionnés).

D'autre part, rien ne garantit *a priori* que l'intersection des modèles de la

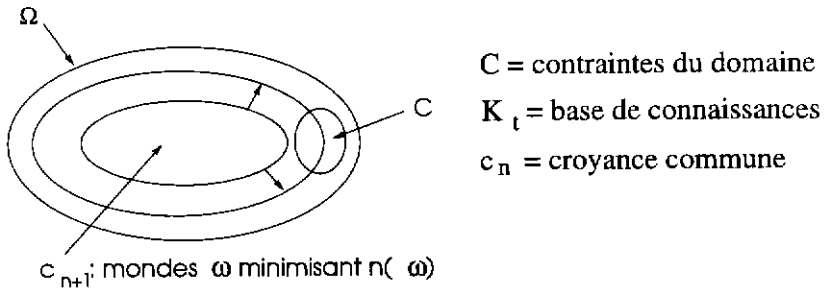


FIG. 6.4 – *Intersection vide*

croyance commune avec ceux des contraintes du domaine ne soit pas vide, c'est-à-dire que la conjecture produite ne corresponde pas à un état possible. Dans ce cas, on peut envisager d'étendre les modèles de la croyance commune aux mondes des classes d'équivalence moins préférées jusqu'à ce que l'intersection ne soit plus vide. La figure 6.4 illustre ce cas : rappelons que les modèles de la croyance commune sont les interprétations qui minimisent $n(\omega)$ c'est-à-dire, lorsqu'on fusionne les croyances de l'agent et la conjecture courante, telles que $n(\omega) = h(\omega, [\![K_t]\!]) + h(\omega, [\![c_n^t]\!]) = h([\![K_t]\!], [\![c_n^t]\!])$.

Il faut cependant souligner que, conformément au modèle de coopération, les agents modifient la conjecture dans la mesure où il sont capables de la rendre admissible. Par conséquent, si les agents ont pour critère d'admissibilité la cohérence, la conjecture produite satisfait nécessairement leurs contraintes communes.

▷ les convictions d'un agent ne sont pas un cube : une possibilité consiste à mettre à jour la croyance commune uniquement à partir de la base de faits D_t, c'est-à-dire à partir d'un cube qui englobe les convictions de l'agent. On est ainsi ramené au cas précédent. La figure 6.5 reprend l'exemple du paragraphe 4.1.2. Supposons qu'un agent reçoive l'information «le Chat est en $S1$» c'est-à-dire $I_t = \{a\}$. On constate que ses convictions, c'est-à-dire $\{\omega_1, \omega_2, \omega_5\}$, ne sont pas un cube et les modèles de I_t, c'est-à-dire $\{\omega_1, \omega_2, \omega_5, \omega_6\}$, sont un cube qui contient les convictions de l'agent.

Définitions des opérateurs

Nous avons constaté au chapitre précédent qu'on peut exprimer aisément un cube sous la forme d'une conjonction de littéraux et nous avons supposé que les préférences des agents étaient fondées sur des relations de Hamming. Ainsi, les modèles de la croyance commune sont les mondes qui minimisent leur distance de Hamming par rapport aux convictions de chaque agent. Nous

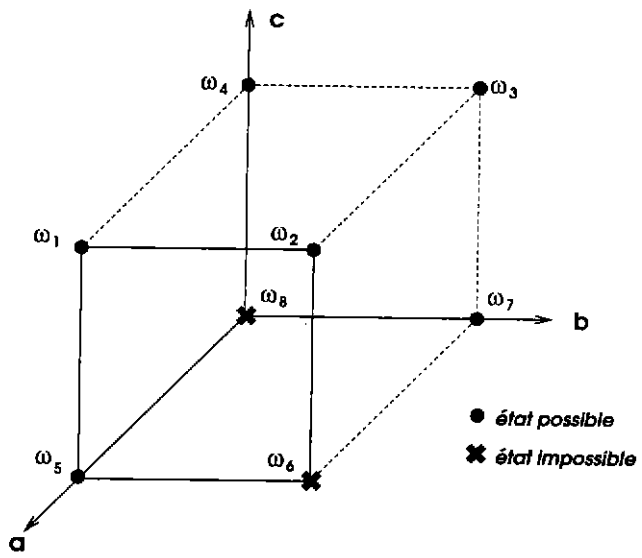


FIG. 6.5 – *Convictions et contraintes*

avons aussi montré que l'ensemble des mondes qui vérifient cette propriété est un cube qu'on peut construire directement à partir de leurs convictions.

Nous allons nous servir de cette procédure et du lien immédiat entre un cube et une conjonction de littéraux pour préciser les définitions des opérateurs de mise à jour du modèle de coopération.

Le modèle de coopération propose cinq opérateurs pour la mise à jour de la conjecture :

▷ **accept** : le critère d'admissibilité d'un élément de la croyance commune est sa cohérence logique avec les croyances de l'agent : comme tous les agents ont les mêmes contraintes et que leurs bases de connaissances sont déductivement fermées, l'admissibilité de la croyance commune se détermine par rapport à la base de faits.

Définition$_{RAC}$ **6.7** *Soit accept l'opérateur déterminant l'admissibilité des éléments de la conjecture.*

$$accept : \ Conject \rightarrow \{vrai, faux\}$$

$$\boxed{\forall p \in c_n^t, \ accept(p) = vrai \Leftrightarrow D_t \cup \{p\} \ cohérent}$$

et, pour toute conjecture c_n^t, qui, par définition, est un élément de l'ensemble des parties du langage :

$$Admissible(c_n^t) = \{w \mid \forall w \in c_n^t,\ accept(w) = vrai\}$$

est l'ensemble des éléments de conjecture admissibles du point de vue de l'agent.

Exemple : supposons que $c_n^t = \{\neg b, c\}$ et que $D_t = \{a, b, \neg c\}$, alors $Admissible(c_n^t) = \emptyset$.

▷ **reject** : sa définition découle directement de celle de l'opérateur *accept* :

Définition$_{RAC}$ 6.8 *Soit reject l'opérateur déterminant la non-admissibilité des éléments de la conjecture.*

$$reject :\ Conject \rightarrow \{vrai, faux\}$$

$$\boxed{\forall p \in c_n^t,\ reject(p) = vrai \Leftrightarrow D_t \cup \{p\}\ incohérent}$$

et, pour toute conjecture c_n^t,

$$Non-Admissible(c_n^t) = \{w | \forall w \in c_n^t,\ reject(w) = vrai\}$$

est l'ensemble des éléments de conjecture non admissibles du point de vue de l'agent.

Exemple : supposons que $c_n^t = \{\neg b, c\}$ et que $D_t = \{a, b, \neg c\}$, alors $Non-Admissible(c_n^t) = \{\neg b, c\}$.

Avec ces définitions $Non-Admissible(c_n^t) \cap Admissible(c_n^t)$ est nécessairement vide : cela signifie que l'agent n'est indifférent à aucune des informations contenues dans la croyance commune.

Les définitions des variables décidant de l'admissibilité ne changent pas :

Définition$_{RAC}$ 6.9 *Soit accept_conjecture la variable booléenne déterminant l'admissibilité de la conjecture :*

$$accept_conjecture :\ Conject \rightarrow \{vrai, faux\}$$

telle que :

$$accept_conjecture(c_n^t) = vrai \Rightarrow Non-Admissible(c_n^t) = \emptyset$$

Dans le cas contraire $reject_conjecture(c_n^t) = vrai$.

▷ **add** : les seules croyances susceptibles d'enrichir la croyance commune sont celles issues de la base de faits D_t. Par conséquent :

Définition$_{RAC}$ **6.10** *Soit add l'opérateur qui permet à un agent d'ajouter un élément à la croyance commune.*

$$add : \ Conject \times \mathcal{P}(D_t) \rightarrow Conject$$

tel que

$$\boxed{\forall c_n^t \in Conject \text{ et } \forall v \in \mathcal{P}(D_t),\ add(c_n^t, v) = c_n^t \cup \{v\}}$$

▷ **retract** : sa définition ne change pas :

Définition$_{RAC}$ **6.11** *Soit retract l'opérateur qui permet à un agent de retirer un élément de la croyance commune.*

$$retract : \ Conject \times Conject \rightarrow Conject$$

tel que :

$$\boxed{\forall c_n^t \in Conject \text{ et } \forall w \in Conject,\ retract(c_n^t, w) = c_n^t \backslash \{w\}}$$

▷ **adjust** : il s'agit de l'opérateur de mise à jour de la croyance commune. Conformément aux résultats du chapitre précédent, la fusion de deux cubes exprimés sous la forme d'une conjonction de littéraux est l'union des cubes à l'exclusion des éléments mutuellement contradictoires. Les éléments non admissibles de la croyance commune sont donnés par l'ensemble $Non-Admissible(c_n^t)$. On en déduit que les éléments à ajouter dans la croyance commune sont :

$$\Delta = D_t \setminus \{p \mid \forall q \in Non-Admissible(c_n^t),\ p = \neg q\}$$

Exemple : supposons que $c_n^t = \{\neg b, c\}$ et que $D_t = \{a, b, \neg c\}$, alors $Non-Admissible(c_n^t) = \{\neg b, c\}$ et $\Delta = \{a\}$.

Définition$_{RAC}$ **6.12** *Soit adjust l'opérateur qui permet à un agent de modifier la conjecture.*

$$adjust : \ Conject \times Conject \rightarrow Conject$$

tel que $\forall c_n^t \in Conject$,

$$\boxed{c_{n+1} = adjust(c_n^t) = add(retract(c_n^t, Non-Admissible(c_n^t)), \Delta)}$$

On impose que :

$$c_{n+1}^t = adjust(c_n^t) \Rightarrow Non-Admissible(c_{n+1}^t) = \emptyset$$

Exemple : supposons que $c_n^t = \{\neg b, c\}$ et que $D_t = \{a, b, \neg c\}$, alors $Non-Admissible(c_n^t) = \{\neg b, c\}$, $\Delta = \{a\}$ et $c_{n+1}^t = \{a\}$. Ainsi, la croyance commune mise à jour par l'agent ne dit rien sur les propriétés b et c qui sont laissées en suspens. Tous les mondes dans lesquels a est vraie, quelle que soit la valeur de vérité de b et de c, sont également préférés.

Remarque : la procédure de mise à jour de la croyance commune ainsi définie n'est pas associative. Soit $adjust_\psi$, l'opérateur de mise à jour de l'agent ψ et $adjust_\mu$, l'opérateur de mise à jour de l'agent μ :

$$adjust_\psi[adjust_\mu(c_n^t)] \neq adjust_\mu[adjust_\psi(c_n^t)]$$

Supposons, par exemple, que les bases de faits de ψ et μ soient telles que $D_t^\psi = \{a, b, \neg c\}$ et $D_t^\mu = \{a, \neg b\}$, et que $c_n^t = \{\neg b, c\}$:

$$adjust_\mu[adjust_\psi(c_n^t)] = adjust_\mu[\{a\}] = \{a, \neg b\}$$

$$adjust_\psi[adjust_\mu(c_n^t)] = adjust_\psi[\{a, \neg b, c\}] = \{a\}$$

Avec cette procédure de mise à jour, le contenu de la conjecture, c'est-à-dire la croyance commune, dépend de l'ordre dans lequel les agents interviennent. Cette caractéristique de la procédure de mise à jour ne contredit pas l'intuition : dans un dialogue, chaque intervenant fait évoluer la conjecture courante en fonction de ses croyances, et, par conséquent, le contenu de la conjecture est tout autant lié aux croyances des intervenants qu'à l'ordre dans lequel ils prennent la parole (et donc font part de leurs connaissances).

Bien évidemment, le dialogue entre personnes met en œuvre des mécanismes cognitifs beaucoup plus complexes que l'ajustement de la conjecture commune, comme par exemple l'implicature, c'est-à-dire le processus d'inférence interne à l'agent en fonction de ce qui vient d'être énoncé

(Maudet and Evrard, 1998).

Notre modèle de coopération ne s'oppose pas à l'utilisation de tels mécanismes. Il semble que leur intégration explicite dans le modèle pourrait constituer un prolongement intéressant.

Afin de garantir la convergence vers une conjecture commune quel que soit l'ordre d'intervention des agents, nous ajoutons des contraintes supplémentaires au modèle de coopération :

- entre deux mises à jour successives de sa base de connaissances, un agent ne peut ajouter ou retirer une proposition p donnée qu'une seule fois. Les agents ne sont donc pas bornés ;
- un agent n'ajoute une propriété p à la conjecture que si elle ne s'y trouve pas déjà (i. e. il n'y a pas d'«intoxication» de la conjecture).

On constate que ces contraintes impliquent nécessairement la convergence vers une conjecture unique. En effet, à cause de la première contrainte, tout se passe comme si les agents «vidaient» leurs bases de faits : le processus d'élaboration de la conjecture s'arrête donc nécessairement. La seconde contrainte, quant à elle, induit un comptage implicite sur les propriétés. Ainsi, si la propriété p est majoritaire parmi les croyances des agents par rapport à $\neg p$, on est assuré de la retrouver dans la conjecture commune.

Remarque : dans une application en prise constante avec un environnement évolutif, la convergence vers une solution unique (au sens de la résolution de problèmes) ne se justifie pas nécessairement : nous introduisons cependant ces contraintes pour montrer que le modèle de coopération ne s'oppose pas, par nature, à la convergence vers une solution unique.

Les définitions des variables de décision *extend_conjecture* et *adjust_conjecture* caractérisant le type de mise à jour (i. e. enrichissement ou ajustement de la croyance commune) ne changent pas (cf. chapitre 3).

6.3.3 Décision d'engagement

Nous considérons qu'un agent se retire de la fédération s'il juge non admissibles un certain nombre de conjectures consécutives.

Comme le critère de non admissibilité est fondé sur la contradiction logique entre la conjecture et la base de connaissances de l'agent, son retrait

n'est pas motivé par la non détection d'une propriété, qui se traduit par l'absence de celle-ci dans la base de connaissances, mais par la détection d'une propriété contradictoire ; par exemple, un agent peut croire *type(x,voiture)* et en déduire $\neg$*type(x,piéton)* alors que c_n^t ={*type(x,piéton)*}. On peut, par conséquent, considérer la contradiction logique comme un symptôme de panne de l'agent. Ainsi, on admet qu'un agent puisse parfois se tromper, mais lorsqu'il est, de façon durable, en contradiction avec les autres agents, on estime qu'il est défaillant : **fixons de façon arbitraire ce seuil à 2**.

Les définitions des variables de décision concernant l'engagement sont donc :

Définition$_{RAC}$ **6.13** *Soit drop(K_t) et hold(K_t) les variables de décision suivantes :*

- *drop(K_t) = vrai lorsque l'agent juge la croyance commune* non admissible *2 fois consécutives ;*
- *hold(K_t) = vrai dans le cas contraire ;*

6.4 Le greffier

Rappelons que le rôle du greffier est d'évaluer l'état global de la fédération : ses trois critères sont la conformité de la fédération, la complétude et l'admissibilité globale de la croyance commune.

Soit $Th_k(M)$ une fonction booléenne à seuil (Sallantin et al., 1991). Par définition, $Th_k(M)$ est vraie si au moins k éléments de M sont vrais.

▷ **Conformité** : dans cet exemple de reconnaissance d'activités, nous supposons que la fédération est conforme si et seulement si la majorité des agents sont engagés.

Définition$_{RAC}$ **6.14** *La variable* $\mathsf{Compliance}_n$ *caractérisant la conformité de la fédération est définie par :*

$$\mathsf{Compliance}_n \equiv Th_2\Big(\mathsf{LocalCommitment}_n^{\psi}, \mathsf{LocalCommitment}_n^{\mu}, \mathsf{LocalCommitment}_n^{\varphi}\Big)$$

où $\mathsf{LocalCommitment}_n^{\alpha_i}$ *avec* $\alpha_i \in \{\psi, \mu, \varphi\}$ *sont respectivement les variables d'état locales des agents* ψ, μ *et* φ *caractérisant leur engagement*

après avoir évalué c_n^t.

La fédération est conforme *si et seulement si* $\mathsf{Compliance}_n = vrai$.

▷ **Admissibilité globale** : dans cet exemple de reconnaissance d'activités, nous supposons que la fédération est conforme si et seulement si la majorité des agents sont engagés.

Définition$_{RAC}$ **6.15** *La variable* $\mathsf{GlobalAcceptability}(c_n^t)$ *caractérisant l'admissibilité globale de la croyance commune est définie par :*

$$\mathsf{GlobalAcceptability}(c_n^t) \equiv Th_2\Big(\mathsf{LocalAcceptability}_n^{\psi}, \mathsf{LocalAcceptability}_n^{\mu}, \mathsf{LocalAcceptability}_n^{\varphi}\Big)$$

où $\mathsf{LocalAcceptability}_n^{\alpha_i}$ *avec* $\alpha_i \in \{\psi, \mu, \varphi\}$ *sont respectivement les variables d'état locales des agents* ψ, μ *et* φ *caractérisant l'admissibilité de* c_n^t.

La croyance commune est admissible *si et seulement si* $\mathsf{GlobalAcceptability}(c_n^t) = vrai$.

▷ **Complétude** : le critère de complétude est indépendant de l'application considérée. Par conséquent, sa définition ne change pas :

Définition$_{RAC}$ **6.16** *La variable* $\mathsf{Completeness}(c_n^t)$ *est définie par :*

$$\mathsf{Completeness}(c_n^t) \equiv \bigwedge_{\alpha_i \in \{\psi,\mu,\varphi\}} (\neg \mathsf{LocalChanged}_n^{\alpha_i})$$

où $\mathsf{LocalChanged}_n^{\alpha_i}$ *est la variable d'état locale de l'agent* α_i *signalant que* c_n^t *a été modifiée.*

La conjecture courante c_n^t *est* complète *si et seulement si* $\mathsf{Completeness}(c_n^t) = vrai$.

6.5 Fonctionnement dynamique du modèle

Nous supposons qu'il n'y a pas de protocole de communication particulier : les agents accèdent à la croyance commune dans la mesure où celle-ci est disponible. Le rôle du greffier est joué par l'agent φ. La croyance commune initiale $c_0^{t_1}$ est vide. L'état de la fédération est $\{complète, admissible, conforme\}$. L'état initial de tous les agents est $\{admissible, inchangé, engagé\}$. Les bases de connaissances initiales sont réduites aux contraintes du domaine C.

6.5.1 Exemple

- À la date t_1, les agents ont détecté un objet $O1$ en mouvement et disposent des informations suivantes :

	I_{t_1}
ψ	*type(O1, piéton), vitesse(O1, faible), se-rapproche-de(O1, V1)*
μ	*type(O1, piéton), vitesse(O1, faible), se-rapproche-de(O1, V1)*
φ	*type(O1, voiture), vitesse(O1, faible), se-rapproche-de(O1, V1)*

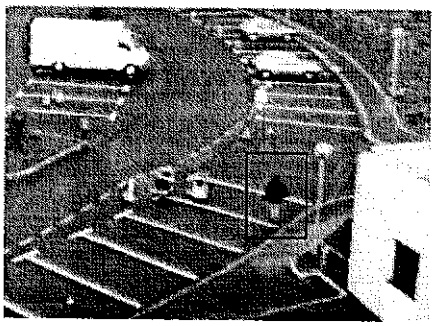


FIG. 6.6 – *Image perçue par l'agent ψ à la date t_1*

Les agents ψ, μ et φ intègrent leurs informations respectives et mettent à jour leurs bases de connaissances de la manière suivante :

$$D_{t_1}^{\psi} = D_{t_1}^{\mu} = \begin{cases} type(O1, piéton) \\ \neg type(O1, voiture) \\ \neg type(O1, moto) \\ vitesse(O1, faible) \\ type(V1, voiture) \\ vitesse(V1, nulle) \\ se-rapproche-de(O1, V1) \\ \ldots \end{cases}$$

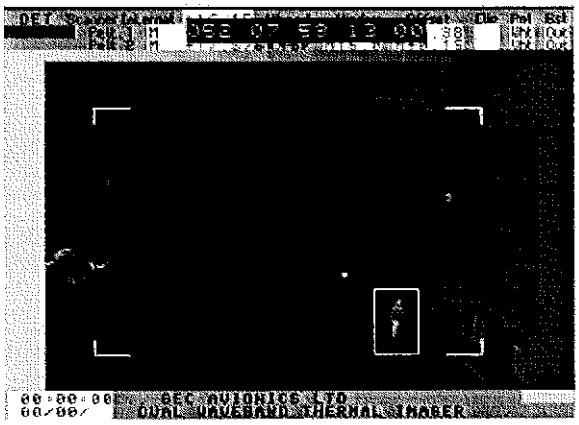


FIG. 6.7 – *Image perçue par l'agent φ à la date $t1$*

$$D^{\varphi}_{t_1} = \begin{cases} type(O1, voiture) \\ \neg type(O1, piéton) \\ \neg type(O1, moto) \\ vitesse(O1, faible) \\ type(V1, voiture) \\ vitesse(V1, nulle) \\ se-rapproche-de(O1, V1) \\ \ldots \end{cases}$$

On constate que φ a identifié $O1$ comme une voiture alors que les deux autres agents croient que $O1$ est un piéton.

Les agents connaissent aussi la présence sur le parking d'un objet $V1$ de type véhicule et qui est immobile.

Les agents ψ et μ ont déduit du fait observé *type(O1,piéton)* que $O1$ n'est ni une voiture, ni une moto. Réciproquement, l'agent φ a déduit du fait observé *type(O1,voiture)* que $O1$ n'est ni un piéton, ni une moto.

Remarque : nous avons fait l'hypothèse que les bases de connaissances sont déductivement fermées : ainsi, les pointillés indiquent la présence d'autres faits déduits dans la base. Cependant, du point de vue de la mise en œuvre, on ne peut pas supposer que les agents déduisent toutes les conséquences logiques de leurs observations : ils doivent disposer de mécanismes externes à la base de connaissances leur permettant de produire des inférences de façon «raisonnable» et de détecter les contradictions.

Supposons que l'agent ψ mette à jour la conjecture : comme $c_0^{t_1} = \{\}$, il *étend* la conjecture, c'est-à-dire qu'il ajoute à la conjecture, grâce à

l'opérateur *add*, tous les éléments de sa base de faits :

$$c_1^{t_1} = add(c_0^{t_1}, D_{t_1}^{\psi}) = \begin{cases} type(O1, piéton) \\ \neg type(O1, voiture) \\ \neg type(O1, moto) \\ vitesse(O1, faible) \\ type(V1, voiture) \\ vitesse(V1, nulle) \\ se-rapproche-de(O1, V1) \\ \ldots \end{cases}$$

L'état de la conjecture devient {***incomplète***,*admissible, conforme*}. Supposons que l'agent φ accède à la conjecture et l'évalue : il la juge alors non admissible parce qu'il croit que $O1$ est une voiture. Il applique l'opérateur *retract* pour retirer les faits *type(O1, piéton)* et ¬*type(O1,voiture)*. En revanche, conformément à la définition de l'opérateur *adjust*, il n'ajoute pas le fait *type(O1,voiture)* qui est contradictoire avec le fait ¬*type(O1,voiture)* de la conjecture. La conjecture devient alors :

$$c_2^{t_1} = adjust(c_1^{t_1}) = \begin{cases} \neg type(O1, moto) \\ vitesse(O1, faible) \\ type(V1, voiture) \\ vitesse(V1, nulle) \\ se-rapproche-de(O1, V1) \\ \ldots \end{cases}$$

Il ressort de la conjecture que $O1$ n'est pas une moto. ¬*type(O1,moto)* est un fait *déduit* par les agents. Mais son statut dans la conjecture est différent : il s'agit d'une inférence produite grâce à la coopération de ψ et φ. En effet, il représente ce sur quoi les agents sont d'accord, c'est-à-dire que $O1$ n'est pas une moto ; aucun agent n'aurait pu déduire cela par les règles d'inférence classiques (i. e. le *modus ponens* et le *modus tollens*) car D_{t1}^{ψ} et D_{t1}^{μ} sont contradictoires. En d'autres termes, si ψ avait simplement communiqué ces croyances à φ, φ n'aurait pas pu déduire ¬*type(O1,moto)* et réciproquement. On constate, par conséquent, que l'opérateur *adjust* est un mécanisme d'inférence différent du *modus ponens* et du *modus tollens* ; comme nous l'avons montré au chapitre précédent, on ne peut pas non plus le réduire à l'intersection ou à l'union des convictions des agents.

L'agent μ accède à son tour à la conjecture : celle-ci reste *admissible* car aucun élément ne vient contredire sa base de connaissances. En revanche, il ajoute les éléments qui avaient été retirés par φ et la conjecture devient :

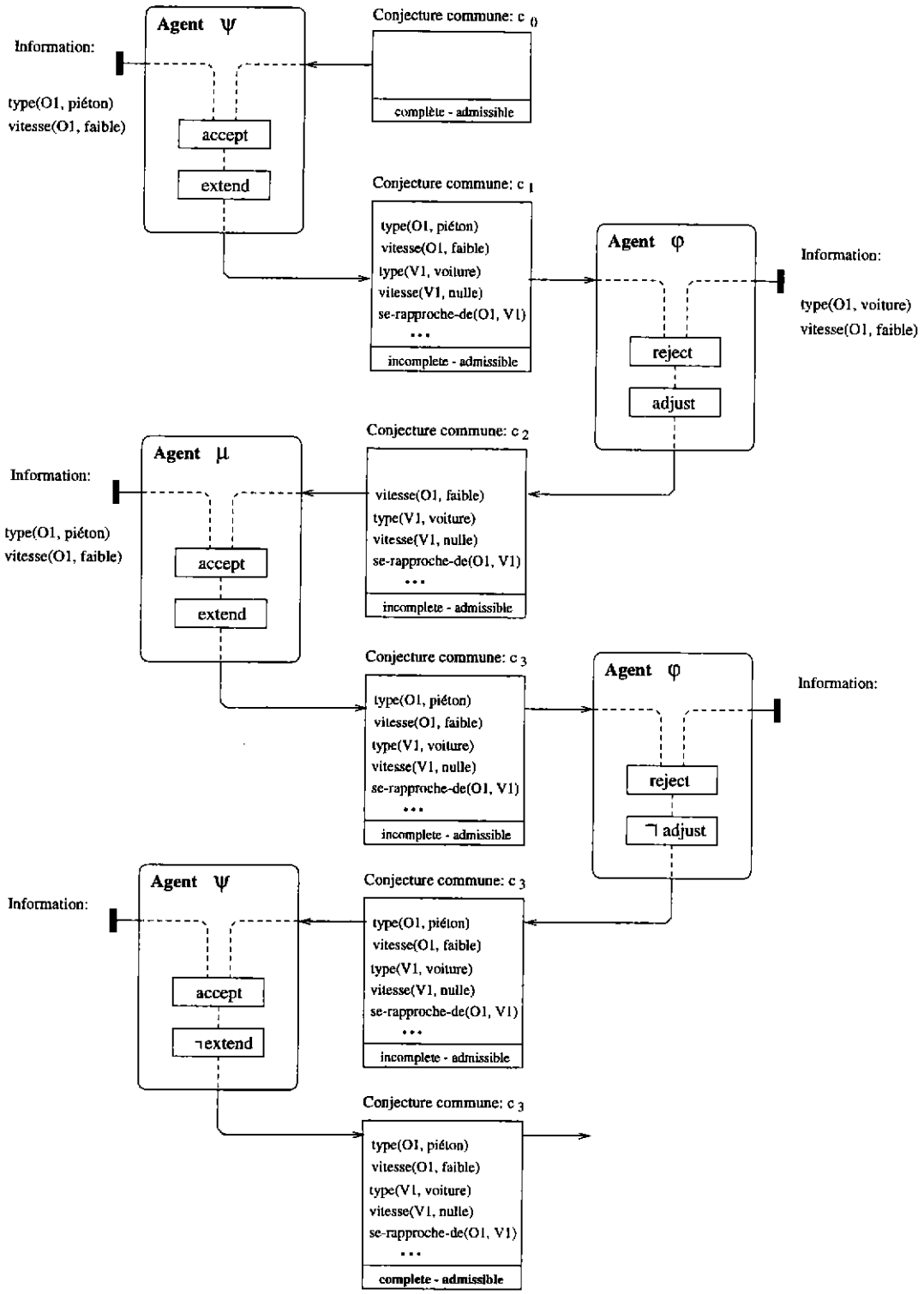


FIG. 6.8 – *Exemple d'élaboration d'une croyance commune*

$$c_3^{t_1} = add(c_2^{t_1}, D_{t_1}^{\mu}) = \begin{cases} type(O1, piéton) \\ \neg type(O1, voiture) \\ \neg type(O1, moto) \\ vitesse(O1, faible) \\ type(V1, voiture) \\ vitesse(V1, nulle) \\ se-rapproche-de(O1, V1) \\ \ldots \end{cases}$$

L'état de la conjecture est alors {*incomplète, admissible, conforme*}. Supposons que l'agent φ évalue une nouvelle fois la conjecture : de son point de vue, elle est non admissible car le fait *type(O1,piéton)* est contradictoire avec ses croyances, mais il ne peut plus la modifier (il a déjà retiré *type(O1,piéton)* de $c_1^{t_1}$). L'état de la conjecture reste alors {*incomplète, admissible, conforme*} car elle n'a été évaluée que par deux agents sur trois sans être modifiée.

L'agent ψ accède à son tour à la conjecture et la juge admissible : il ne la modifie pas car il ne peut pas apporter d'éléments nouveaux. L'état de la conjecture devient alors {*complète, admissible, conforme*} car la majorité des agents la trouve admissible et tous l'ont évaluée sans la modifier. Elle peut alors être transmise par le greffier au processus d'élaboration de situation (la figure 6.8 récapitule l'élaboration de cette croyance commune ; les faits niés ne sont pas représentés).

- À la date $t_2 = t_1 + \Delta t$, les bases de connaissances des agents ont évolué : les trois agents ont reçu l'information *proche-de(O1, V1)*.

	I_{t_2}
ψ	*proche-de(O1, V1)*
μ	*proche-de(O1, V1)*
φ	*proche-de(O1, V1)*

ψ et μ continuent de croire *type(O1, piéton)* tandis que φ croit *type(O1, voiture)*. De plus, le fait *proche-de(O1, V1)* est préféré à *se-rapproche-de(O1, V1)* : priorité est donnée aux faits les plus récents et *se-rapproche-*

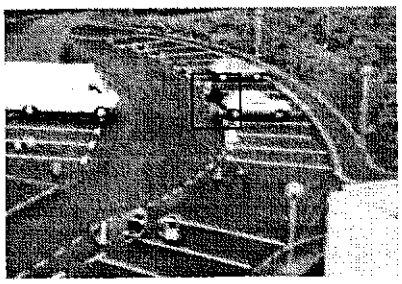


FIG. 6.9 – *Image perçue par l'agent ψ à la date t2*

de(O1,V1) est donc retiré des bases de connaissances.

$$D^{\psi}_{t_2} = D^{\mu}_{t_2} = \begin{cases} type(O1, piéton) \\ \neg type(O1, voiture) \\ \neg type(O1, moto) \\ vitesse(O1, faible) \\ type(V1, voiture) \\ vitesse(V1, nulle) \\ proche-de(O1, V1) \\ \ldots \end{cases}$$

$$D^{\varphi}_{t_2} = \begin{cases} type(O1, voiture) \\ \neg type(O1, piéton) \\ \neg type(O1, moto) \\ vitesse(O1, faible) \\ type(V1, voiture) \\ vitesse(V1, nulle) \\ proche-de(O1, V1) \\ \ldots \end{cases}$$

Dans le cycle précédent, *type(O1, voiture)* a été rejeté par ψ et par μ. Par conséquent, la nouvelle conjecture commune est identique à la précédente à l'exception du fait *se-rapproche-de(O1,V1)* qui est remplacé par *proche-de(O1,V1)*:

$$c_3^{t_2} = \begin{cases} type(O1, piéton) \\ \neg type(O1, voiture) \\ \neg type(O1, moto) \\ vitesse(O1, faible) \\ type(V1, voiture) \\ vitesse(V1, nulle) \\ proche-de(O1, V1) \\ \ldots \end{cases}$$

De plus, l'agent φ se retire de la fédération car, de son point de vue, cette conjecture est, pour la deuxième fois consécutive, non admissible et non modifiable. Le système de surveillance continue cependant de fonctionner car deux agents sur trois restent engagés. Le retrait de φ est motivé par la mise en cause des réglages de ses capteurs comme explication de ses désaccords fréquents avec les autres agents. En effet, les tests effectués pour le projet PERCEPTION (Gaillet, 1996) ont montré que les piétons pouvaient être confondus avec les moteurs des voitures sur les images infrarouges. φ peut cependant continuer à jouer le rôle de greffier.

- À la date $t_3 = t_1 + 2\Delta t$, les agents n'ont pas reçu de nouvelles informations et les bases de connaissances n'évoluent pas. Comme, ni la conjecture, ni les bases de connaissances ne sont mises à jour, les agents n'ont aucune raison d'accéder à la conjecture courante : la conjecture relative à la date t_3 est donc la même que la précédente.

FIG. 6.10 – *Image perçue par l'agent ψ à la date t_3*

$$c_n^{t_3} = \begin{cases} type(O1, piéton) \\ \neg type(O1, voiture) \\ \neg type(O1, moto) \\ vitesse(O1, faible) \\ type(V1, voiture) \\ vitesse(V1, nulle) \\ proche-de(O1, V1) \\ \ldots \end{cases}$$

- À la date $t_4 = t_1 + 3\Delta t$, les agents reçoivent de nouvelles informations :

	I_{t4}
ψ	*type(V1, voiture), vitesse(V1, faible)*
μ	*type(V1, voiture), vitesse(V1, faible), type(V2, voiture), vitesse(V2, grande)*

Ainsi, l'agent ψ n'a pas détecté l'objet $V2$ contrairement à μ qui dispose d'une caméra «champ large» (cf. figure 6.11).

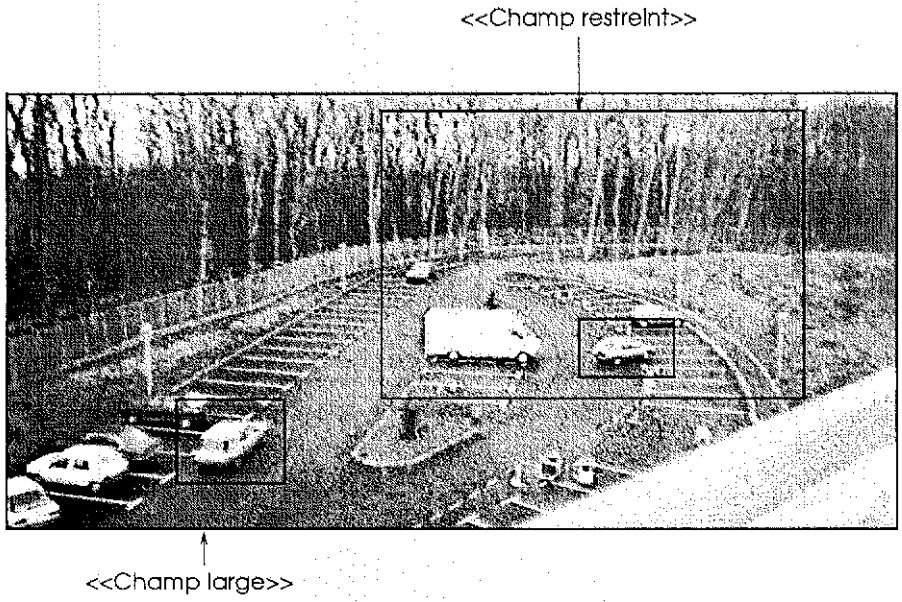


FIG. 6.11 – *Images perçues par les agents ψ et μ à la date t_4*

En revanche, les deux agents ont détecté le démarrage de $V1$: le fait *vitesse(V1, nulle)* est retiré au profit *vitesse(V1, faible)* qui est un fait plus récent. De même, l'objet $O1$ n'apparaît plus sur les images : tous les faits liés à $O1$ sont donc retirés[3]. Les bases de connaissances sont les suivantes :

$$D^{\psi}_{t_4} = \begin{cases} type(V1, voiture) \\ vitesse(V1, faible) \\ \dots \end{cases}$$

$$D^{\mu}_{t_4} = \begin{cases} type(V1, voiture) \\ vitesse(V1, faible) \\ type(V2, voiture) \\ vitesse(V2, grande) \\ \dots \end{cases}$$

3. Seul un mécanisme de mise à jour ne dépendant pas du critère de cohérence des bases de connaissances permet de retirer tous les faits liés à $O1$: la gestion des objets dans la représentation de l'environnement a été étudiée dans le cadre du projet Perception (Perception, 1996) : nous n'abordons pas ce problème ici.

Comme l'agent ψ ne dispose d'aucune information sur $V2$ contradictoire avec celles de μ, il accepte les faits *type(V2, véhicule)* et *vitesse(V2, grande)* lorsqu'il évalue la conjecture. La nouvelle conjecture est donc :

$$c_n^{t_4} = \begin{cases} type(V2, voiture) \\ type(V1, voiture) \\ vitesse(V1, faible) \\ vitesse(V2, grande) \\ \dots \end{cases}$$

6.5.2 Discussion

Dans l'exemple que nous venons d'étudier, les agents trouvent une croyance commune globalement admissible ; ce n'est pas toujours le cas.

Notons a, b et c les faits suivants : «$O1$ est un piéton», «$O1$ est une voiture» et «$O1$ est une moto». Supposons que ψ croit $\{a, \neg b, \neg c\}$, μ croit $\{\neg a, b, \neg c\}$ et φ croit $\{\neg a, \neg b, c\}$. D'après ce que nous savons des contraintes de convergence imposées aux agents, nous pouvons déduire que la croyance commune devrait tendre vers $\{\neg a, \neg b, \neg c\}$ car aucun de ces faits n'est majoritaire (par ailleurs, cette croyance commune ne satisfait pas les contraintes du domaine qui impliquent que chaque objet a nécessairement un type). Pourtant, cette conjecture ne peut pas être élaborée car elle est contradictoire avec les croyances de tous les agents et, par conséquent, non admissible.

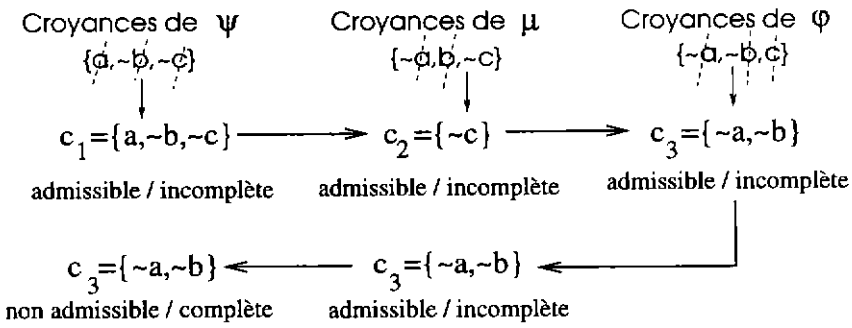


FIG. 6.12 – *Conjecture non admissible*

La figure 6.12 illustre cet exemple :

Étape 1 l'agent ψ produit la conjecture $\{a, \neg b, \neg c\}$. Celle-ci est alors globalement admissible mais incomplète. Par ailleurs, conformément aux contraintes de convergence, ψ n'a plus le droit d'utiliser a, $\neg b$ et $\neg c$;

Étape 2 l'agent μ modifie la conjecture pour la rendre admissible de son point de vue : en particulier, il rejète a et $\neg b$ et la conjecture devient $\{\neg c\}$. Par ailleurs, il ne peut plus utiliser $\neg a$ et b dans sa base de connaissances, car il les a utilisés pour nier a et $\neg b$. L'état de la conjecture ne change pas ;

Étape 3 φ change à son tour la conjecture qui devient $\{\neg a, \neg b\}$ en niant le fait $\neg c$ par c dans sa base de connaissances et en ajoutant $\neg a$ et $\neg b$. L'état de la conjecture ne change pas ;

Étape 4 μ évalue à nouveau la conjecture : il la juge non admissible et ne peut plus la modifier de façon à la rendre admissible de son point de vue. Cependant la conjecture reste globalement admissible car elle a été jugé admissible par ψ et φ ;

Étape 5 ψ accède à la conjecture pour l'évaluer après les modifications de φ : comme il ne peut pas la rendre admissible de son point de vue, il ne la modifie pas. La conjecture devient non admissible parce qu'elle est été jugée non admissible par deux agents, ψ et μ.

Si on avait donné la possibilité aux agents de modifier la conjecture sans la rendre admissible de leur point de vue, mais en utilisant l'opérateur *adjust* tel qu'il a été précédemment défini, la conjecture aurait convergée vers $\{\neg a, \neg b, \neg c\}$. Ici, on vérifie aisément que la conjecture converge vers $\{\neg a, \neg b\}$, $\{\neg b, \neg c\}$ ou $\{\neg a, \neg c\}$; le point important étant que, dans tous les cas de figure, les agents convergent vers une conjecture unique qui est jugée globalement non admissible.

Conclusion

Nous avons montré que le modèle de coopération pouvait s'appliquer pour l'élaboration d'une croyance commune dans le cadre de la reconnaissance d'activités.

Pour cela, nous avons précisé les définitions des éléments du modèle de coopération. Puis, nous avons illustré son fonctionnement dynamique avec un exemple de reconnaissance d'activités.

Conclusion générale

Dans ce mémoire, nous avons abordé le problème de l'élaboration de conjectures par des agents coopérants : dans un environnement évolutif, un groupe d'agents partageant le même objectif font une synthèse de leurs différents points de vue. La conjecture commune ainsi construite est le point de départ de la prise de décision collective ou la source d'information pour des systèmes de renseignement, de surveillance et de diagnostic.

Au cours de cette étude, nous avons été confronté à une grande variété de thèmes dont les plus importants sont la coopération, la cohérence des interactions, les conflits et la fusion de connaissances incertaines et contradictoires.

Contributions

L'étude bibliographique a montré qu'il existe une grande quantité de travaux sur les systèmes multiagents, principalement axés sur la recherche de méthodes de coopération et de négociation et la mise au point de plates-formes de développement et de techniques de programmation innovantes. Il semble, en revanche, que ce domaine de l'Intelligence Artificielle Distribuée manque de bases formelles à finalité opératoire. En particulier, nous avons montré que les modèles cognitifs de coopération ne pouvaient pas constituer un cadre théorique pour la conception de systèmes autonomes.

Notre objectif a donc été de proposer une approche qui se démarque des notions cognitives et est centrée sur la cohérence comme critère d'évaluation de la coopération. Pour cela, nous avons fait une synthèse de nombreuses méthodes visant à organiser les interactions entre agents. Cette formalisation *a posteriori* nous a conduit à un modèle de coopération pour l'élaboration d'une conjecture commune. Le paradigme de ce modèle est celui du cycle de preuves et réfutations initialement proposé par I. Lakatos pour rendre compte de la production de théorèmes : dans notre modèle, les agents interagissent par le biais de mise à jour constante d'une structure de données commune, la conjecture. Les agents sont autonomes car ils décident seuls, à partir de critères qui leur sont propres, des modifications sur la conjecture. Cependant, afin d'assu-

rer un fonctionnement globalement cohérent, les agents observent un certain nombre de règles concernant leur retrait du groupe, les modifications effectuées ou l'accès à la conjecture. Nous avons donc considéré que le comportement des agents doit être encadré, ce qui explique que nous utilisions le terme de fédération ; nous avons ainsi cherché à étudier quels devaient être les facteurs limitant l'autonomie des agents.

Nous nous sommes ensuite intéressé à un cas particulier d'une conjecture représentant l'état de l'environnement du point de vue de la fédération des agents, c'est-à-dire une croyance commune : chaque agent a des préférences sur les états possibles de l'environnement. Nous avons formalisé la notion de conflit portant sur les croyances comme des préférences différentes. Ainsi, il est possible de distinguer divers types de conflits selon que les agents ont des croyances partiellement ou entièrement contradictoires. Afin de rendre compte de l'intensité du conflit, nous avons défini une distance entre les préférences.

La notion de croyance commune découle de celle de conflit : nous nous sommes demandé comment représenter aussi justement que possible l'opinion d'agents qui ne partagent pas le même point de vue. Lorsque ces points de vue sont entièrement contradictoires, nous avons postulé que la croyance commune devait refléter cette contradiction. Au contraire, lorsque les points de vue ne sont que partiellement contradictoires, la croyance commune doit rendre compte des éléments sur lesquels les agents sont d'accord.

Nous avons ensuite proposé une méthode pour construire cette croyance commune et étudié le cas particulier de préférences fondées sur la distance de Hamming.

Enfin, nous avons appliqué ces résultats à la reconnaissance d'activités par un groupe d'agents interagissant conformément à notre modèle de coopération.

Perspectives

L'une de nos ambitions est de susciter l'élaboration de nouvelles méthodes de coopération et la mise au point d'applications fondées sur notre modèle afin de valider expérimentalement les principes que nous avons proposés.

Du point de vue théorique, nous avons cependant laissé en suspens plusieurs problèmes susceptibles de constituer des projets de recherches intéressants :

- il faudrait étudier l'élaboration d'un plan partagé d'actions comme l'élaboration d'une conjecture commune. On pourrait, en particulier, s'inté-

resser au couplage de la phase de planification avec celle de l'élaboration de l'état de l'environnement ;

- le cadre formel de la croyance commune pourrait être étendu au cas où les préférences des agents sont partiellement ordonnées et où leurs croyances sont un ensemble quelconque d'interprétations.

De façon plus générale, nos travaux peuvent être la base de l'étude et de la résolution des incertitudes liées à la coopération de plusieurs systèmes autonomes évoluant dans un environnement dynamique.

Annexe A

Démonstrations des propriétés

Démonstration de la propriété 5.4

On veut prouver que $(\mathcal{C}, \subseteq, \cap, ppce)$ est un treillis.

Nous devons démontrer les propriétés d'idempotence, de commutativité, d'associativité et d'absorption de $\cap$ et *ppce*.

On vérifie aisément que $\cap$ est idempotent, commutatif et associatif. De même, on vérifie aisément que *ppce* est idempotent et commutatif.

Démontrons que *ppce* est associatif. Soit trois cubes $C_{\mathcal{V}_\xi}$, $C_{\mathcal{V}'_{\xi'}}$ et $C_{\mathcal{V}''_{\xi''}}$ ayant, respectivement, pour ensembles de variables valuées, $\mathcal{E}$, $\mathcal{E}'$ et $\mathcal{E}''$. Nous devons démontrer que :

$$ppce\Big(ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}}), C_{\mathcal{V}''_{\xi''}}\Big) = ppce\Big(C_{\mathcal{V}_\xi}, ppce(C_{\mathcal{V}'_{\xi'}}, C_{\mathcal{V}''_{\xi''}})\Big)$$

$ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}})$ est un cube tel que son ensemble de variables valuées est $\mathcal{E} \cap \mathcal{E}'$. On en déduit que $ppce\Big(ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}}), C_{\mathcal{V}''_{\xi''}}\Big)$ est un cube tel que son ensemble de variables valuées est $\mathcal{E} \cap \mathcal{E}' \cap \mathcal{E}''$. Par le même raisonnement, on conclut rapidement que $ppce\Big(C_{\mathcal{V}_\xi}, ppce(C_{\mathcal{V}'_{\xi'}}, C_{\mathcal{V}''_{\xi''}})\Big)$ est un cube tel que sont ensemble de variables valuées est $\mathcal{E} \cap \mathcal{E}' \cap \mathcal{E}''$. Par conséquent :

$$ppce\Big(ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}}), C_{\mathcal{V}''_{\xi''}}\Big) = ppce\Big(C_{\mathcal{V}_\xi}, ppce(C_{\mathcal{V}'_{\xi'}}, C_{\mathcal{V}''_{\xi''}})\Big)$$

Démontrons la propriété d'absorption, c'est-à-dire :

$$ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}_\xi} \cap C_{\mathcal{V}'_{\xi'}}) = C_{\mathcal{V}_\xi} \cap ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}}) = C_{\mathcal{V}_\xi}$$

Nous étudions deux cas :

- $C_{\mathcal{V}_\xi}$ et $C_{\mathcal{V}'_{\xi'}}$ sont parallèles :
  - Démontrons que $ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}_\xi} \cap C_{\mathcal{V}'_{\xi'}}) = C_{\mathcal{V}_\xi}$:
    $C_{\mathcal{V}_\xi} \cap C_{\mathcal{V}'_{\xi'}} = \emptyset$; donc $ppce(C_{\mathcal{V}_\xi}, \emptyset) = C_{\mathcal{V}_\xi}$;
  - Démontrons que $C_{\mathcal{V}_\xi} \cap ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}}) = C_{\mathcal{V}_\xi}$:
    soit $ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}}) = C_{\mathcal{V}''_{\xi''}}$. $C_{\mathcal{V}''_{\xi''}}$ a pour ensemble de variables valuées $\mathcal{E}'' = \mathcal{E} \cap \mathcal{E}'$. Par conséquent, quel que soit $x \in \mathcal{E}''$, $x \in \mathcal{E}$: on en déduit que $C_{\mathcal{V}_\xi}$ et $C_{\mathcal{V}''_{\xi''}}$ sont orthogonaux. D'après la propriété 5.2, $C_{\mathcal{V}_\xi} \cap C_{\mathcal{V}''_{\xi''}}$ est donc un cube dont l'ensemble de variables valuées est $\mathcal{E}_\cap = \mathcal{E} \cup \mathcal{E}''$. Par conséquent, $C_{\mathcal{V}_\xi} \cap ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}})$ est un cube ayant pour ensemble des variables valuées $\mathcal{E} \cup (\mathcal{E} \cap \mathcal{E}') = \mathcal{E}$. Cela entraîne que :

$$C_{\mathcal{V}_\xi} \cap ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}}) = C_{\mathcal{V}_\xi}$$

- $C_{\mathcal{V}_\xi}$ et $C_{\mathcal{V}'_{\xi'}}$ sont orthogonaux :
  - Démontrons que $ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}_\xi} \cap C_{\mathcal{V}'_{\xi'}}) = C_{\mathcal{V}_\xi}$:
    soit $C_{\mathcal{V}''_{\xi''}} = C_{\mathcal{V}_\xi} \cap C_{\mathcal{V}'_{\xi'}}$. D'après la propriété 5.2, ce cube a pour ensemble de variables valuées $\mathcal{E}'' = \mathcal{E} \cup \mathcal{E}'$. On en déduit que $ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}_\xi} \cap C_{\mathcal{V}'_{\xi'}})$ a pour ensemble de variables valuées $\mathcal{E} \cap \mathcal{E}'' = \mathcal{E} \cap (\mathcal{E} \cup \mathcal{E}') = \mathcal{E}$. Par conséquent :

$$ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}_\xi} \cap C_{\mathcal{V}'_{\xi'}}) = C_{\mathcal{V}_\xi}$$

  - $C_{\mathcal{V}_\xi} \cap ppce(C_{\mathcal{V}_\xi}, C_{\mathcal{V}'_{\xi'}}) = C_{\mathcal{V}_\xi}$ se démontre de la même façon que dans le cas des cubes parallèles.

□

Démonstration de la propriété 5.5

Soit $[\![\psi]\!]$ et $[\![\mu]\!]$ les convictions de deux agents ψ et μ dont les préférences sont des relations de Hamming. On veut démontrer que les modèles de la croyance commune sont tels que :

$$[\![\psi \perp \mu]\!] = [\![\psi]\!] \dotplus [\![\mu]\!]$$

Soit $[\![\psi]\!]$ et $[\![\mu]\!]$ deux cubes ayant respectivement $\mathcal{E}_\psi$ et $\mathcal{E}_\mu$ pour ensembles de variables valuées et, $\mathcal{F}_\psi$ et $\mathcal{F}_\mu$ pour ensembles de variables libres. On définit

cinq ensembles $\mathcal{P}$, $\mathcal{Q}$, $\mathcal{R}$, $\mathcal{S}$ et $\mathcal{T}$ tels que :

$$\begin{aligned}
\mathcal{P} &= \mathcal{E}_\psi \cap \mathcal{E}_\mu \\
\mathcal{Q} &= \{x \mid \forall x \in \mathcal{E}_\psi, \bar{x} \in \mathcal{E}_\mu\} \\
\mathcal{R} &= \{x \mid \forall x \in \mathcal{E}_\psi, x \ et \ \bar{x} \notin \mathcal{E}_\mu\} \\
\mathcal{S} &= \{x \mid \forall x \in \mathcal{E}_\mu, x \ et \ \bar{x} \notin \mathcal{E}_\psi\} \\
\mathcal{T} &= \mathcal{F}_\psi \cap \mathcal{F}_\mu
\end{aligned}$$

Remarquons que $\mathcal{P} \cup \mathcal{Q} \cup \mathcal{R}$ est une partition de $\mathcal{E}_\psi$, $\mathcal{S} \cup \mathcal{T}$ est une partition de $\mathcal{F}_\psi$, et que $card(\mathcal{Q}) = h([\![\psi]\!], [\![\mu]\!])$.

Déterminons à présent l'ensemble $\mathcal{M}$ des interprétations de $[\![\psi]\!]$ tel que $h(\mathcal{M}, [\![\mu]\!]) = h([\![\psi]\!], [\![\mu]\!])$ (i. e. les convictions de l'agent ψ les plus proches de celles de μ). Cet ensemble est un cube, noté $\widehat{C}_\psi$, et dont l'ensemble des variables valuées, noté $\widehat{\mathcal{E}}_\psi$, est tel que :

$$\widehat{\mathcal{E}}_\psi = \mathcal{P} \cup \mathcal{Q} \cup \mathcal{R} \cup \mathcal{S}$$

En effet, comme $\widehat{C}_\psi \subseteq [\![\psi]\!]$, d'après la propriété 5.3, $\mathcal{P} \cup \mathcal{Q} \cup \mathcal{R} \subseteq \widehat{\mathcal{E}}_\psi$. Par ailleurs, parmi les variables libres de $[\![\psi]\!]$, certaines sont valuées dans $[\![\mu]\!]$: assignons à ces variables libres les mêmes valeurs que dans $[\![\mu]\!]$. Par définition, la distance de Hamming entre le cube ainsi défini et $[\![\mu]\!]$ est la même qu'entre $[\![\psi]\!]$ et $[\![\mu]\!]$: ce cube est donc tel que l'ensemble de ses variables valuées est $\mathcal{P} \cup \mathcal{Q} \cup \mathcal{R} \cup \mathcal{S}$.

De la même façon, on définit cinq ensembles $\mathcal{P}'$, $\mathcal{Q}'$, $\mathcal{R}'$, $\mathcal{S}'$ et $\mathcal{T}'$ tels que :

$$\begin{aligned}
\mathcal{P}' &= \mathcal{E}_\psi \cap \mathcal{E}_\mu \\
\mathcal{Q}' &= \{x \mid \forall x \in \mathcal{E}_\mu, \bar{x} \in \mathcal{E}_\psi\} \\
\mathcal{R}' &= \{x \mid \forall x \in \mathcal{E}_\mu, x \ et \ \bar{x} \notin \mathcal{E}_\psi\} \\
\mathcal{S}' &= \{x \mid \forall x \in \mathcal{E}_\psi, x \ et \ \bar{x} \notin \mathcal{E}_\mu\} \\
\mathcal{T}' &= \mathcal{F}_\psi \cap \mathcal{F}_\mu
\end{aligned}$$

Remarquons que :

- $\mathcal{P}' \cup \mathcal{Q}' \cup \mathcal{R}'$ est une partition de $\mathcal{E}_\mu$, et $\mathcal{S}' \cup \mathcal{T}'$ est une partition de $\mathcal{F}_\mu$;
- $\mathcal{P} = \mathcal{P}'$, $\mathcal{R}' = \mathcal{S}$, $\mathcal{S}' = \mathcal{R}$ et $\mathcal{T}' = \mathcal{T}$;
- $\forall x \in \mathcal{Q}$, $\bar{x} \in \mathcal{Q}'$ et $card(\mathcal{Q}') = h([\![\psi]\!], [\![\mu]\!])$.

On conduit le même raisonnement que dans le cas précédent : $\widehat{C}_\mu \subseteq [\![\mu]\!]$ et $h(\widehat{C}_\mu, [\![\psi]\!]) = h([\![\psi]\!], [\![\mu]\!])$. L'ensemble $\widehat{\mathcal{E}}_\mu$ des variables valuées de $\widehat{C}_\mu$ est tel que $\widehat{\mathcal{E}}_\mu = \mathcal{P}' \cup \mathcal{Q}' \cup \mathcal{R}' \cup \mathcal{S}'$ c'est-à-dire que $\widehat{\mathcal{E}}_\mu = \mathcal{P} \cup \mathcal{Q}' \cup \mathcal{R} \cup \mathcal{S}$.

Nous allons montrer que $ppce(\widehat{C}_\psi, \widehat{C}_\mu) = [\![\psi \perp \mu]\!]$, c'est-à-dire que $\forall \omega \in ppce(\widehat{C}_\psi, \widehat{C}_\mu),\ h(\omega, [\![\psi]\!]) + h(\omega, [\![\mu]\!]) = h([\![\psi]\!], [\![\mu]\!])$.

Comme $h(\widehat{C}_\psi, [\![\mu]\!]) = h(\widehat{C}_\mu, [\![\psi]\!]) = h([\![\psi]\!], [\![\mu]\!])$, on en déduit que :

$$\widehat{C}_\psi \subseteq [\![\psi \perp \mu]\!] \ et\ \widehat{C}_\mu \subseteq [\![\psi \perp \mu]\!]$$

$ppce(\widehat{C}_\psi, \widehat{C}_\mu)$ a pour ensemble de variables valuées $\mathcal{E}_{ppce} = \widehat{\mathcal{E}}_\psi \cap \widehat{\mathcal{E}}_\mu = \mathcal{P} \cup \mathcal{R} \cup \mathcal{S}$. D'après la propriété 5.3, comme $\mathcal{E}_{ppce} \subseteq \widehat{\mathcal{E}}_\psi$ et $\mathcal{E}_{ppce} \subseteq \widehat{\mathcal{E}}_\mu$, on a $\widehat{C}_\psi \subseteq ppce(\widehat{C}_\psi, \widehat{C}_\mu)$ et $\widehat{C}_\mu \subseteq ppce(\widehat{C}_\psi, \widehat{C}_\mu)$.

Soit, à présent, une interprétation $\omega \in ppce(\widehat{C}_\psi, \widehat{C}_\mu)$ telle que $\omega \notin \widehat{C}_\psi$ et $\omega \notin \widehat{C}_\mu$. Soit $n = card(\mathcal{Q}) = card(\mathcal{Q}')$. ω différe de toutes les interprétations de $\widehat{C}_\psi$ par certaines des valeurs de vérité de ses variables : ces variables ne sont pas celles de $\mathcal{P}$, $\mathcal{R}$ ou $\mathcal{S}$ car, par définition, ω n'appartiendrait pas à $ppce(\widehat{C}_\psi, \widehat{C}_\mu)$. Par conséquent, ω se différencie par l'assignation de différentes valeurs de vérité sur les variables de $\mathcal{Q}$. Supposons que k valeurs de vérité changent parmi n. Comme, par ailleurs, pour tout $x \in \mathcal{Q}$, $\bar{x} \in \mathcal{Q}'$, on a :

$$\left.\begin{array}{r} h(\omega, \widehat{C}_\psi) = k \\ h(\omega, \widehat{C}_\mu) = h(\widehat{C}_\psi, \widehat{C}_\mu) - k \end{array}\right\} \ et\ h(\omega, \widehat{C}_\psi) + h(\omega, \widehat{C}_\mu) = h(\widehat{C}_\psi, \widehat{C}_\mu)$$

Par ailleurs, $h(\widehat{C}_\psi, \widehat{C}_\mu) = card(\mathcal{Q}) = card(\mathcal{Q}')$. Comme nous avons montré que $h([\![\psi]\!], [\![\mu]\!]) = card(\mathcal{Q})$, on en déduit que $h(\widehat{C}_\psi, \widehat{C}_\mu) = h([\![\psi]\!], [\![\mu]\!])$. D'autre part, comme l'ensemble des variables valuées de $[\![\psi]\!]$ est $\mathcal{P} \cup \mathcal{Q} \cup \mathcal{R}$ et celui de $\widehat{C}_\psi$ est $\mathcal{P} \cup \mathcal{Q} \cup \mathcal{R} \cup \mathcal{S}$, on a $h(\omega, \widehat{C}_\psi) = h(\omega, [\![\psi]\!])$: en effet, les valeurs de vérité qui changent entre ω et $\widehat{C}_\psi$, et entre ω et $[\![\psi]\!]$, sont les mêmes, et ne dépendent que de $\mathcal{Q}$ (on conduit le même raisonnement avec $\widehat{C}_\mu$ et $[\![\mu]\!]$). Par conséquent :

$$h(\omega, \widehat{C}_\psi) + h(\omega, \widehat{C}_\mu) = h(\omega, [\![\psi]\!]) + h(\omega, [\![\mu]\!])$$

Comme :

$$h(\omega, \widehat{C}_\psi) + h(\omega, \widehat{C}_\mu) = h(\widehat{C}_\psi, \widehat{C}_\mu) = h([\![\psi]\!], [\![\mu]\!])$$

On en déduit que :

$$\forall x \in ppce(\widehat{C}_\psi, \widehat{C}_\mu),\ h(\omega, [\![\psi]\!]) + h(\omega, [\![\mu]\!]) = h([\![\psi]\!], [\![\mu]\!])$$

Soit $\omega' \notin ppce(\widehat{C}_\psi, \widehat{C}_\mu)$. Rappelons que $\mathcal{E}_{ppce} = \mathcal{P} \cup \mathcal{R} \cup \mathcal{S}$ est l'ensemble des variables valuées de ω, $\mathcal{E}_\psi = \mathcal{P} \cup \mathcal{Q} \cup \mathcal{R}$ et $\mathcal{E}_\mu = \mathcal{P} \cup \mathcal{Q}' \cup \mathcal{S}$. Or, pour passer de ω à ω' on change au moins l'une des valeurs de vérité de $\mathcal{P}$ ou $\mathcal{R}$ ou $\mathcal{S}$: si on change la valeur de vérité de l'une des variables de $\mathcal{P}$, on s'éloigne à la fois

de $[\![\psi]\!]$ et de $[\![\mu]\!]$; si on change la valeur de vérité de l'une des variables de $\mathcal{R}$, on s'éloigne de $[\![\mu]\!]$; si on change la valeur de vérité de l'une des variables de $\mathcal{S}$, on s'éloigne de $[\![\mu]\!]$. Par conséquent, dans tous les cas :

$$h(\omega', [\![\psi]\!]) + h(\omega', [\![\mu]\!]) \geq h([\![\psi]\!], [\![\mu]\!])$$

On en déduit que :

$$ppce(\widehat{C}_\psi, \widehat{C}_\mu) = [\![\psi \perp \mu]\!]$$

Montrons, pour finir, que $ppce(\widehat{C}_\psi, \widehat{C}_\mu) = [\![\psi]\!] \dot{+} [\![\mu]\!]$: rappelons que $\widehat{\mathcal{E}}_\psi = \mathcal{P} \cup \mathcal{Q} \cup \mathcal{R} \cup \mathcal{S}$ et $\widehat{\mathcal{E}}_\mu = \mathcal{P} \cup \mathcal{Q}' \cup \mathcal{R} \cup \mathcal{S}$. D'après la définition de $\dot{+}$, on en déduit que $\Gamma = \Gamma' = \mathcal{P} \cup \mathcal{R} \cup \mathcal{S}$. Par conséquent, $\mathcal{E}_{\dot{+}} = \mathcal{P} \cup \mathcal{R} \cup \mathcal{S}$ et donc :

$$ppce(\widehat{C}_\psi, \widehat{C}_\mu) = [\![\psi]\!] \dot{+} [\![\mu]\!]$$

Comme, par ailleurs, $ppce(\widehat{C}_\psi, \widehat{C}_\mu) = [\![\psi \perp \mu]\!]$, on en conclut que :

$$[\![\psi \perp \mu]\!] = [\![\psi]\!] \dot{+} [\![\mu]\!]$$

□

Démonstration de la propriété 5.6

On veut démontrer que :

$$[\![\psi]\!] \cap [\![\mu]\!] \subseteq [\![\psi \perp \mu]\!] \subseteq ppce([\![\psi]\!], [\![\mu]\!])$$

Soit $[\![\psi]\!]$ et $[\![\mu]\!]$ deux cubes dont les ensembles de variables valuées sont respectivement $\mathcal{E}$ et $\mathcal{E}'$. Cette démonstration comprend deux parties :

- Démontrons que $[\![\psi]\!] \cap [\![\mu]\!] \subseteq [\![\psi \perp \mu]\!]$:
  - supposons que $[\![\psi]\!]$ et $[\![\mu]\!]$ soient parallèles. On en déduit que $[\![\psi]\!] \cap [\![\mu]\!] = \emptyset$ et, par conséquent, $[\![\psi]\!] \cap [\![\mu]\!] \subseteq [\![\psi \perp \mu]\!]$.
  - supposons que $[\![\psi]\!]$ et $[\![\mu]\!]$ soient orthogonaux. $[\![\psi]\!] \cap [\![\mu]\!]$ est un cube tel que $\mathcal{E}_\cap = \mathcal{E} \cup \mathcal{E}'$.

    D'autre part, $[\![\psi \perp \mu]\!] = [\![\psi]\!] \dot{+} [\![\mu]\!]$ et, de façon générale, $[\![\psi]\!] \dot{+} [\![\mu]\!]$ est un cube dont les variables valuées $\mathcal{E}_{\dot{+}}$ sont telles que $\mathcal{E}_{\dot{+}} = \Gamma \cup \Gamma'$ avec :

$$\begin{cases} \Gamma = \{x \mid \forall x \in \mathcal{E},\ \bar{x} \notin \mathcal{E}'\} \\ \Gamma' = \{x \mid \forall x \in \mathcal{E}',\ \bar{x} \notin \mathcal{E}\} \end{cases}$$

Par conséquent :

$$\Gamma \subseteq \mathcal{E} \ et \ \Gamma' \subseteq \mathcal{E}'$$

On en déduit que :

$$\Gamma \cup \Gamma' \subseteq \mathcal{E} \cup \mathcal{E}'$$

et, d'après la propriété 5.3, on peut conclure que :

$$[\![\psi]\!] \cap [\![\mu]\!] \subseteq [\![\psi \perp \mu]\!]$$

– Démontrons que $[\![\psi \perp \mu]\!] \subseteq ppce([\![\psi]\!], [\![\mu]\!])$.

$ppce([\![\psi]\!], [\![\mu]\!])$ est un cube dont l'ensemble des variables valuées $\mathcal{E}_{ppce}$ est tel que $\mathcal{E}_{ppce} = \mathcal{E} \cap \mathcal{E}'$.

D'autre part, $[\![\psi \perp \mu]\!] = [\![\psi]\!] \dot{+} [\![\mu]\!]$ et, de façon générale, $[\![\psi]\!] \dot{+} [\![\mu]\!]$ est un cube dont les variables valuées $\mathcal{E}_{\dot{+}}$ sont telles que $\mathcal{E}_{\dot{+}} = \Gamma \cup \Gamma'$ avec :

$$\begin{cases} \Gamma = \{x \mid \forall x \in \mathcal{E},\ \bar{x} \notin \mathcal{E}'\} \\ \Gamma' = \{x \mid \forall x \in \mathcal{E}',\ \bar{x} \notin \mathcal{E}\} \end{cases}$$

De façon générale, on peut écrire que :

$$\mathcal{E} \cap \mathcal{E}' \subseteq \mathcal{E} \cup \mathcal{E}' \tag{A.1}$$

Comme :

$$\Gamma \subseteq \mathcal{E} \ et \ \Gamma' \subseteq \mathcal{E}'$$

On peut donc écrire que :

$$\Gamma = \mathcal{E} \backslash \delta \ et \ \Gamma' = \mathcal{E}' \backslash \delta'$$

où δ et δ' sont tels que :

$$\begin{cases} \forall x \in \delta,\ \bar{x} \in \delta' \\ \forall x \in \delta',\ \bar{x} \in \delta \end{cases}$$

On en déduit que :

$$\mathcal{E} \cap \mathcal{E}' = \Gamma \cap \Gamma' \tag{A.2}$$

D'après (A.1) et (A.2), on conclut que :

$$\Gamma \cap \Gamma' \subseteq \Gamma \cup \Gamma$$

$$\mathcal{E} \cap \mathcal{E}' \subseteq \Gamma \cup \Gamma'$$

Et, d'après la propriété 5.3, cela entraîne que :

$$[\![\psi \perp \mu]\!] \subseteq ppce([\![\psi]\!], [\![\mu]\!])$$

□

Annexe B

Modélisation de la coopération par des réseaux de Petri

```
color Agents = with Alpha | Beta | NopAg;
color Acceptability = with Acceptable | Unacceptable | NopAcc;
color ConjectureChange = with Changed | Unchanged | NopConj;
color Commitment = with Committed | Uncommitted | NopCommit;
color Logic = with L | CM;
color LocalAcceptability = product Agents * Acceptability;
color LocalChange = product Agents * ConjectureChange;
color LocalCommitment = product Agents * Commitment;

var acc1: LocalAcceptability;
var acc2: LocalAcceptability;
var cha1: LocalChange;
var cha2: LocalChange;
var localcommitment: LocalCommitment;
var agents: Agents;
var logic: Logic;
```

FIG. B.1 – *Déclaration des variables et des couleurs du réseau*

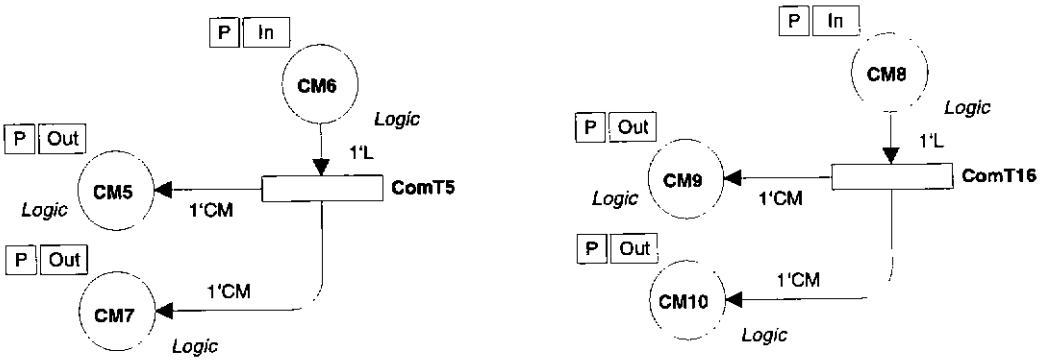
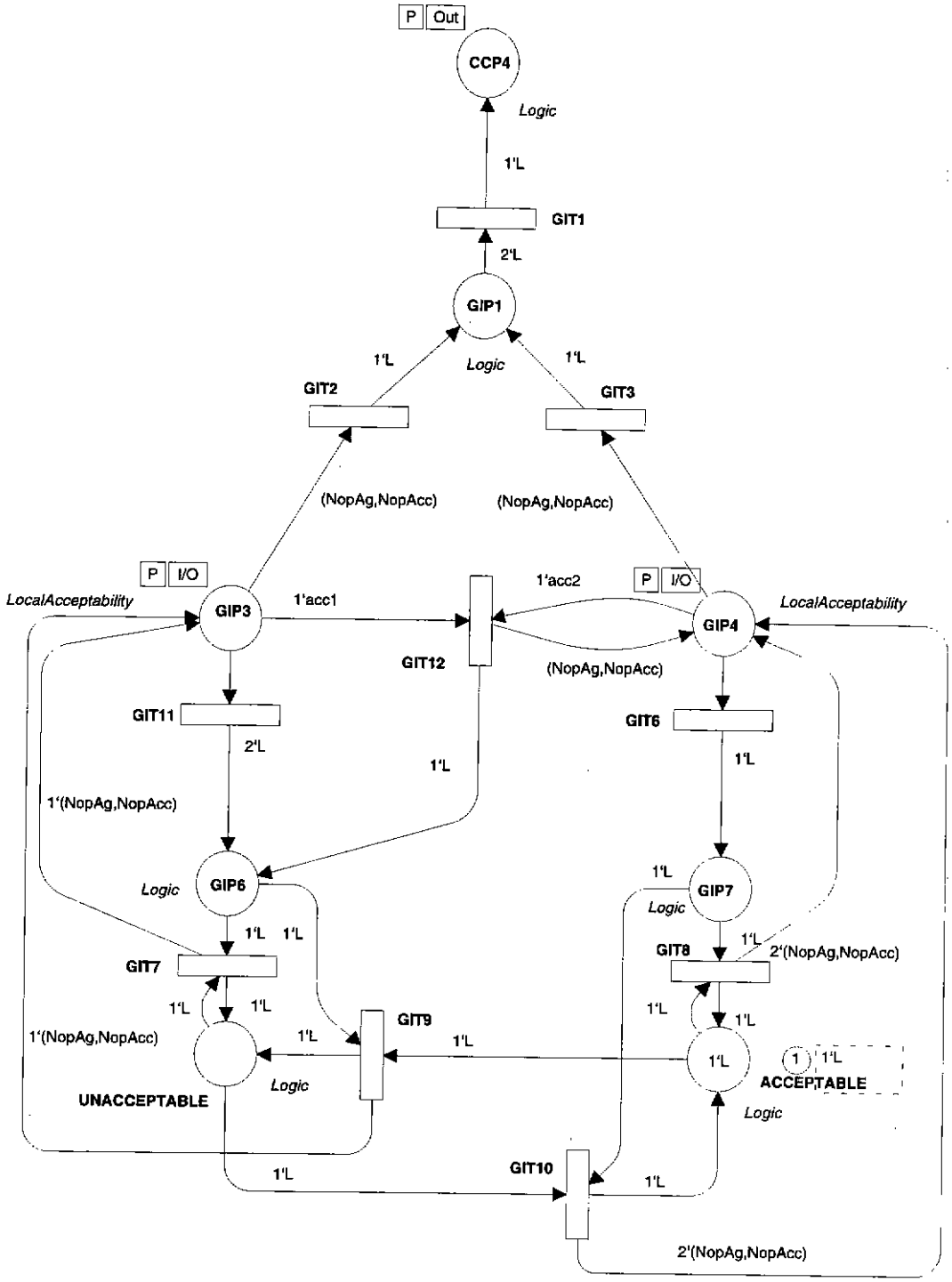


FIG. B.2 – *Greffier : conjecture modifiée*

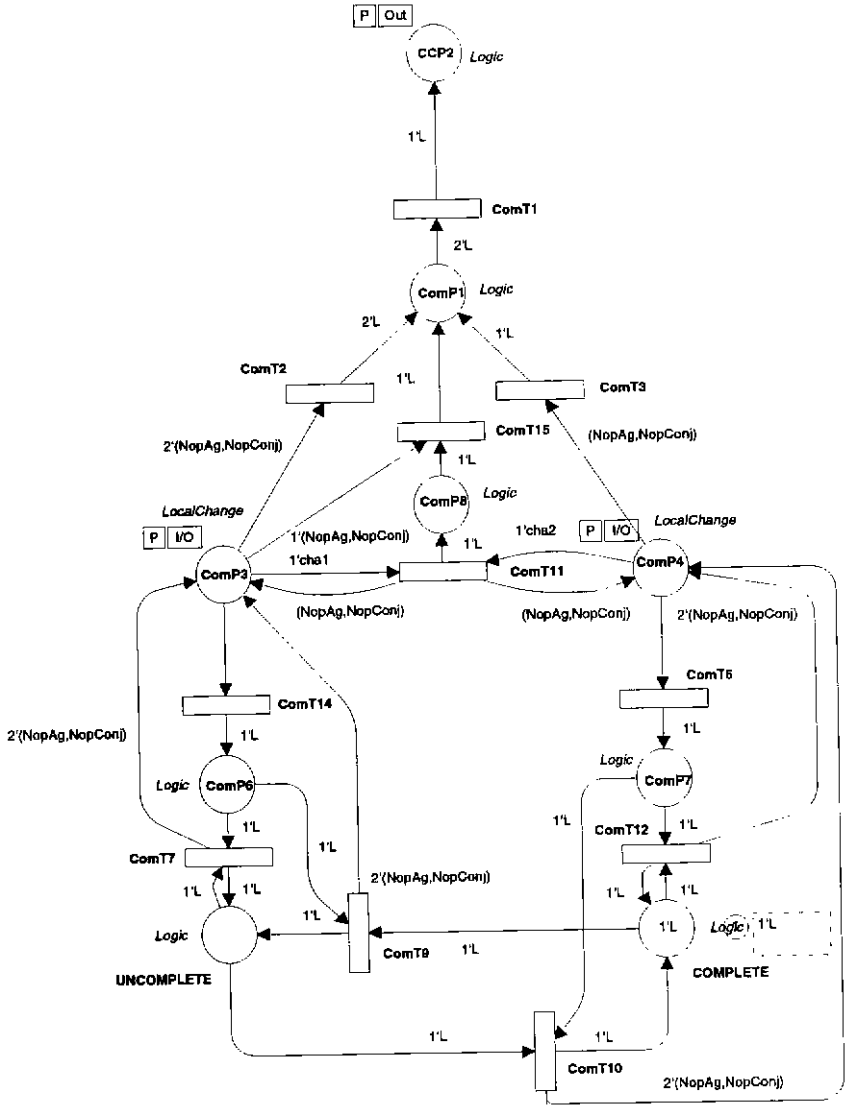


1'(Alpha,Acceptable)+1'(Beta,Acceptable)

1'(Alpha,Unacceptable)+1'(Beta,Unacceptable)

[((acc1=(Alpha,Unacceptable)) andalso (acc2=(Beta,Acceptable))) orelse ((acc1=(Beta,Unacceptable)) andalso (acc2=(Alpha,Acceptable)))]

FIG. B.3 – *Greffier : admissibilité globale*



1'(Alpha,Changed)+1'(Beta,Changed)

1'(Alpha,Unchanged)+1'(Beta,Unchanged)

[((cha1=(Alpha,Changed)) andalso (cha2=(Beta,Unchanged))) orelse ((cha1=(Beta,Changed)) andalso (cha2=(Alpha,Unchanged)))]

FIG. B.4 – *Greffier : complétude*

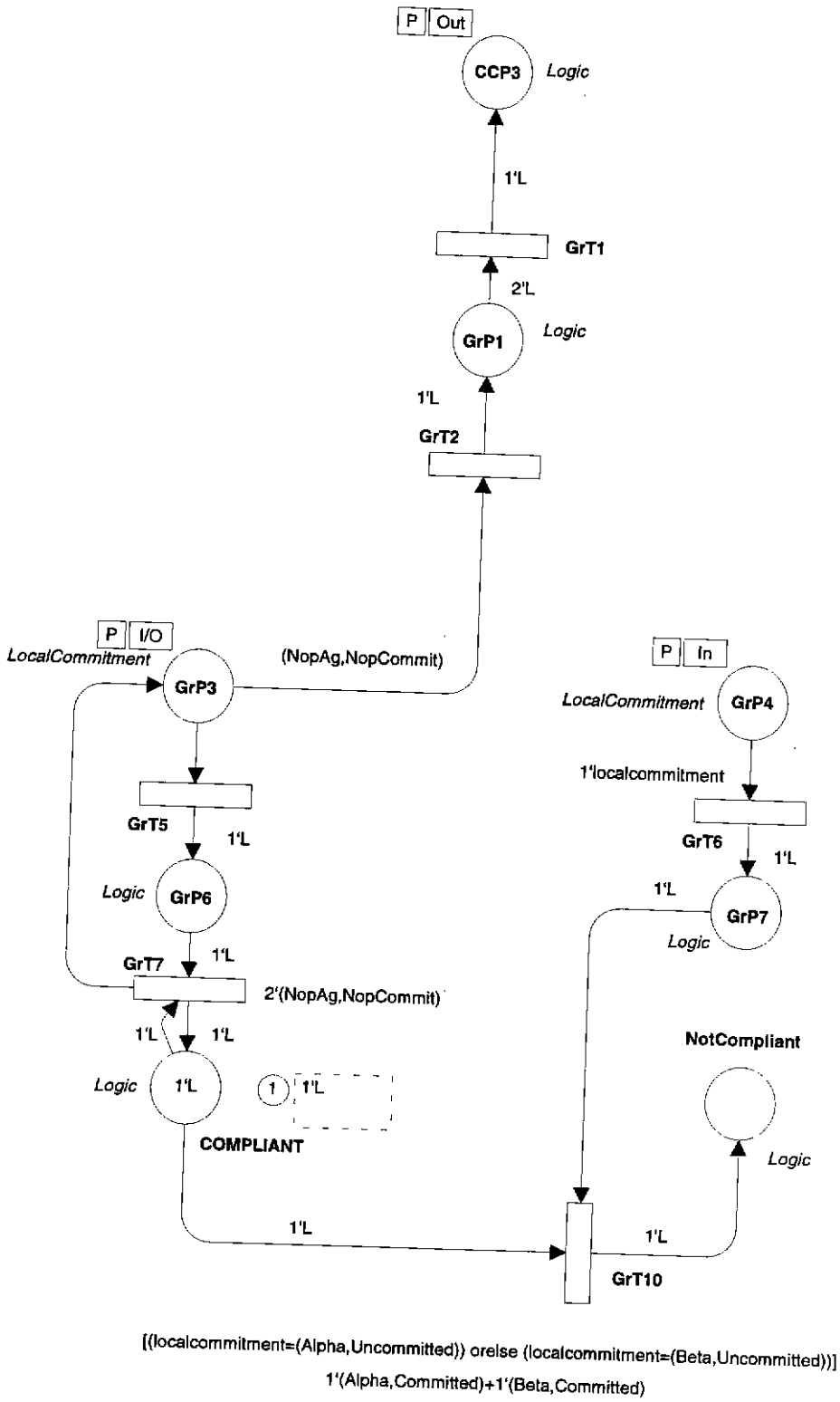


FIG. B.5 – *Greffier : conformité*

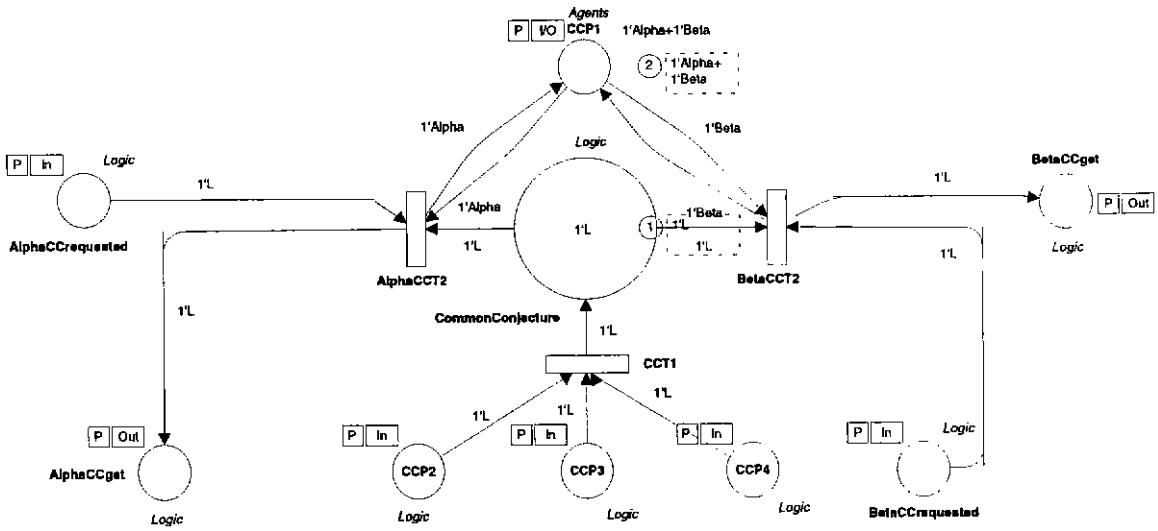


FIG. B.6 – *Greffier : conjecture disponible*

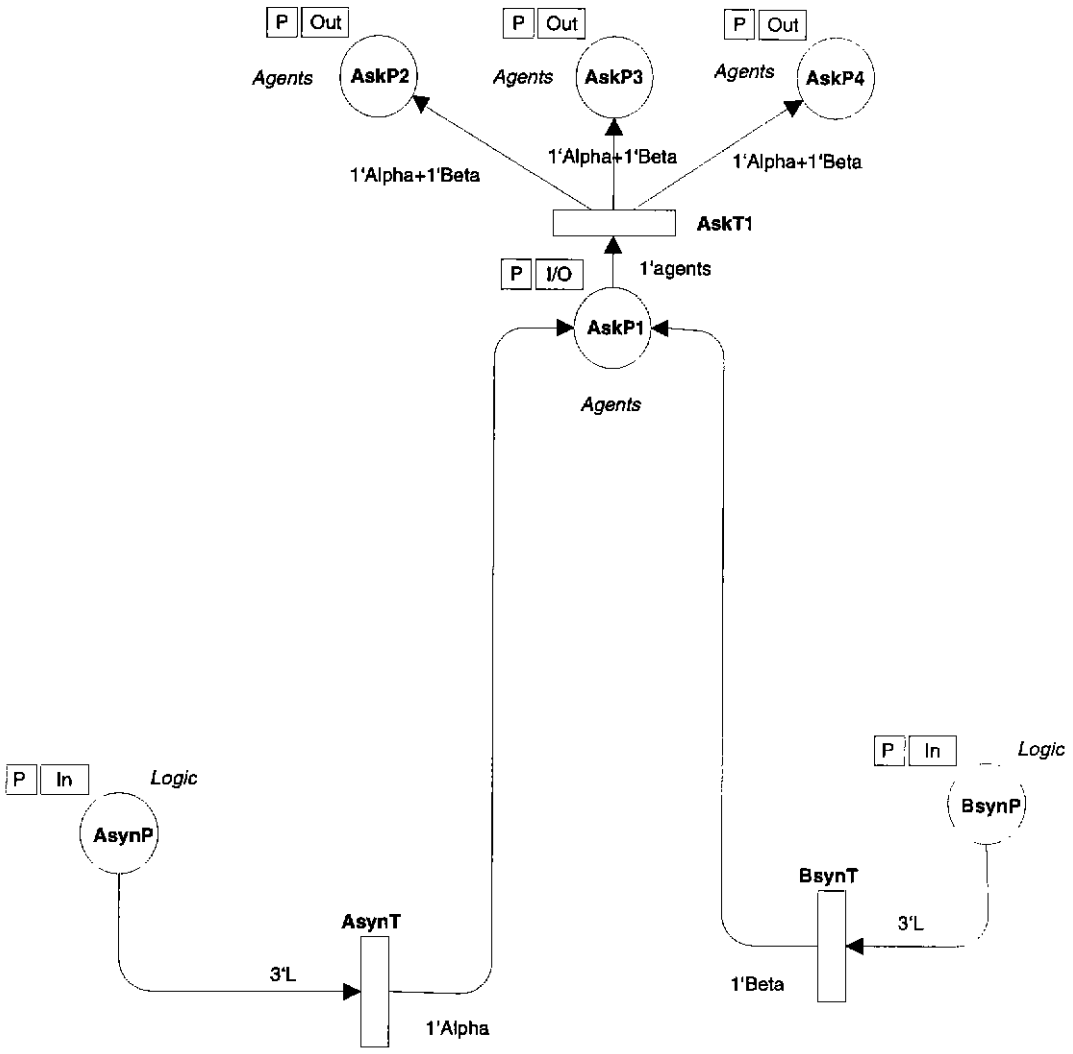


FIG. B.7 – *Greffier : accès aux états locaux*

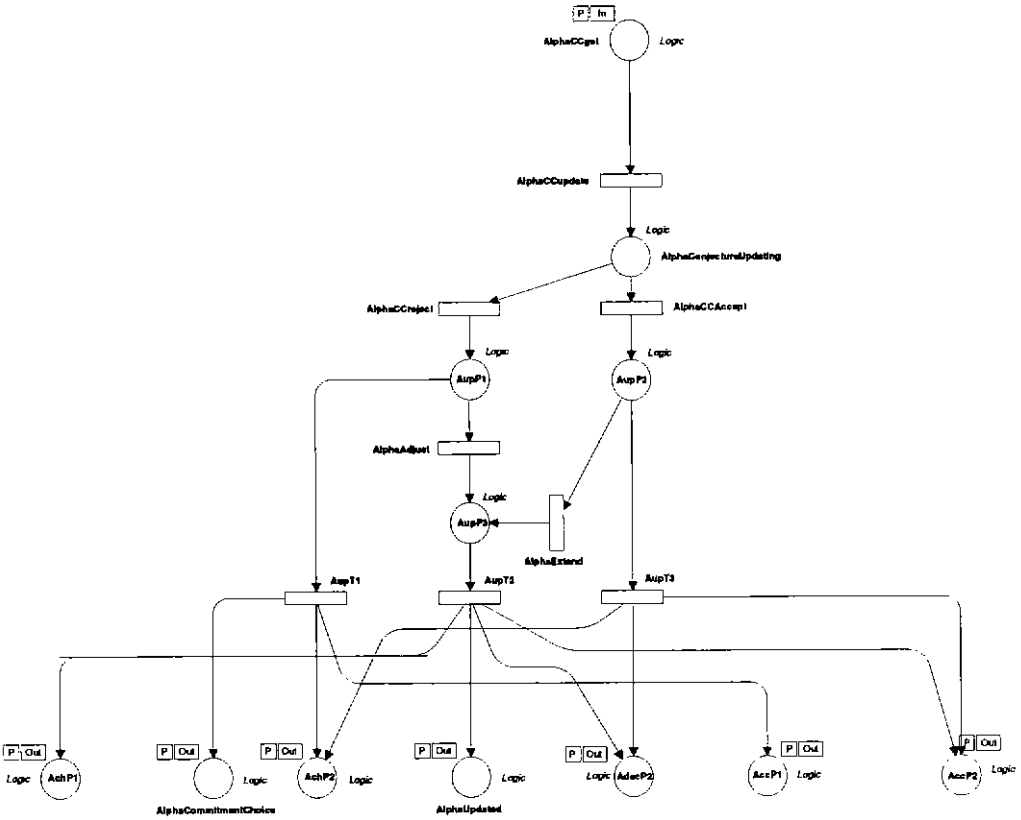


FIG. B.8 – *Agent α : module de mise à jour de la conjecture*

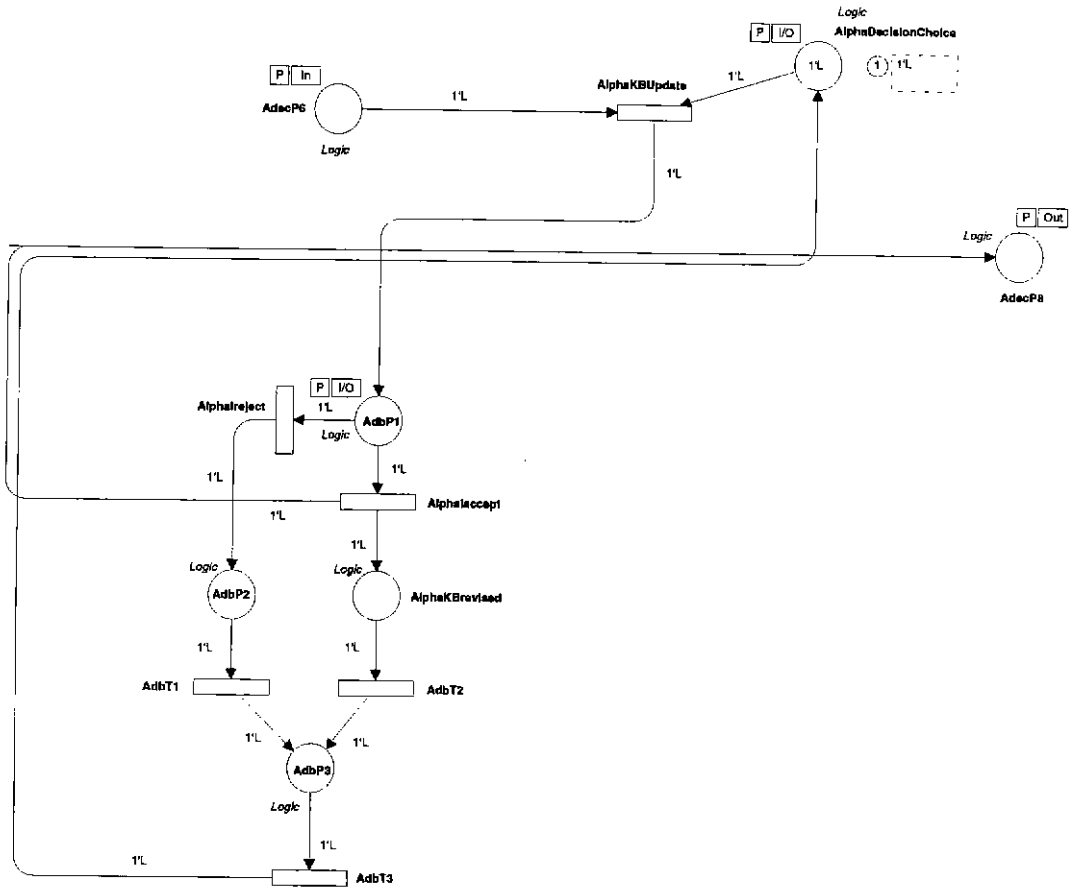


FIG. B.9 – *Agent α : module de mise à jour de la base de connaissances*

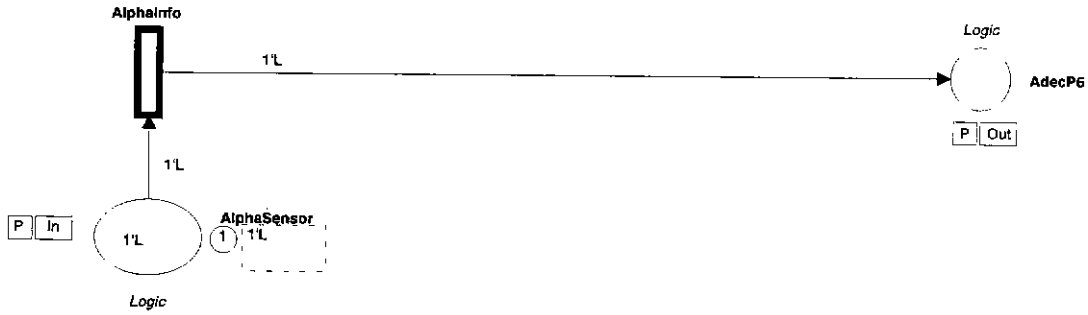


FIG. B.10 – *Agent α : réception des informations*

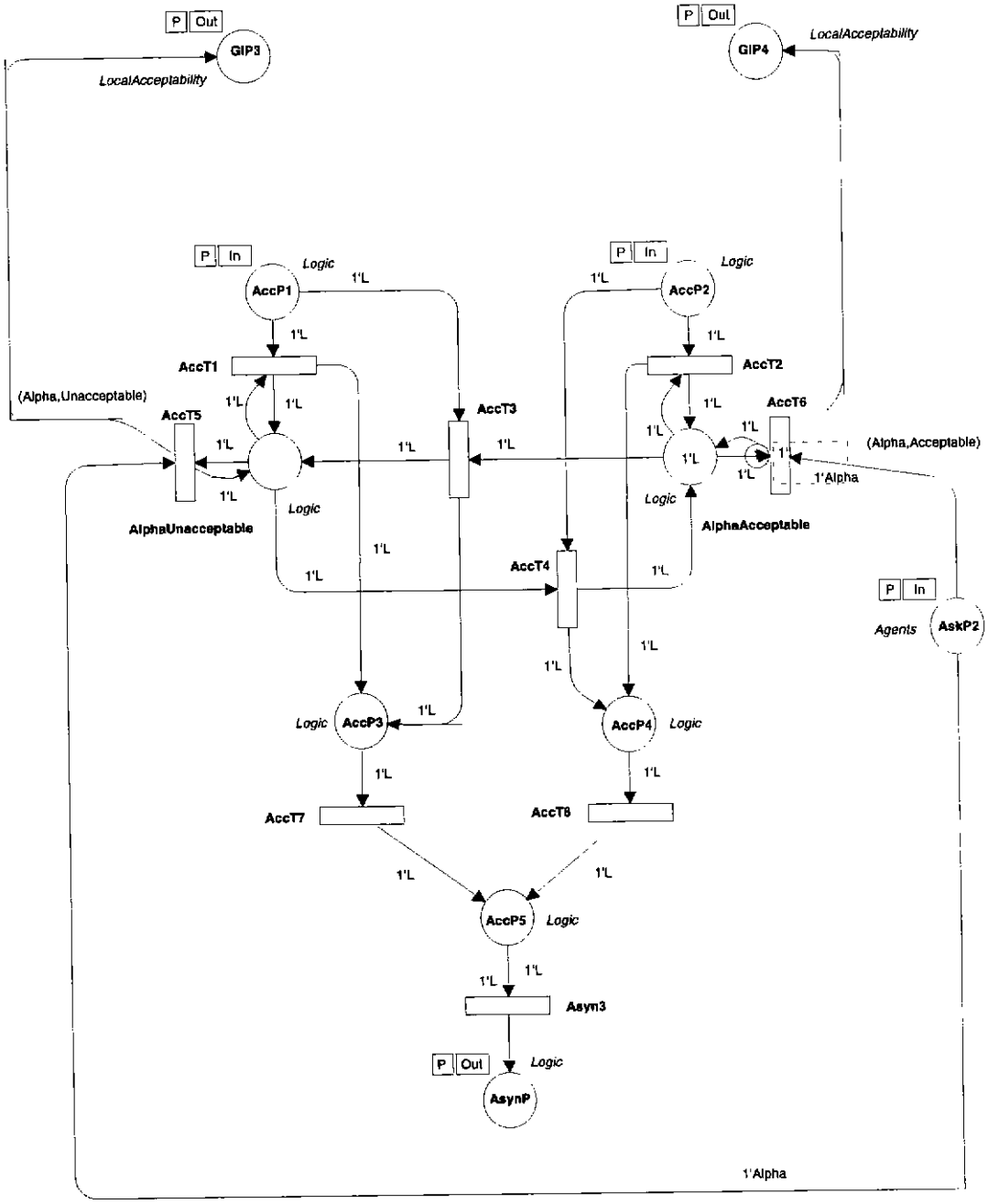


FIG. B.11 – *Agent α : admissibilité*

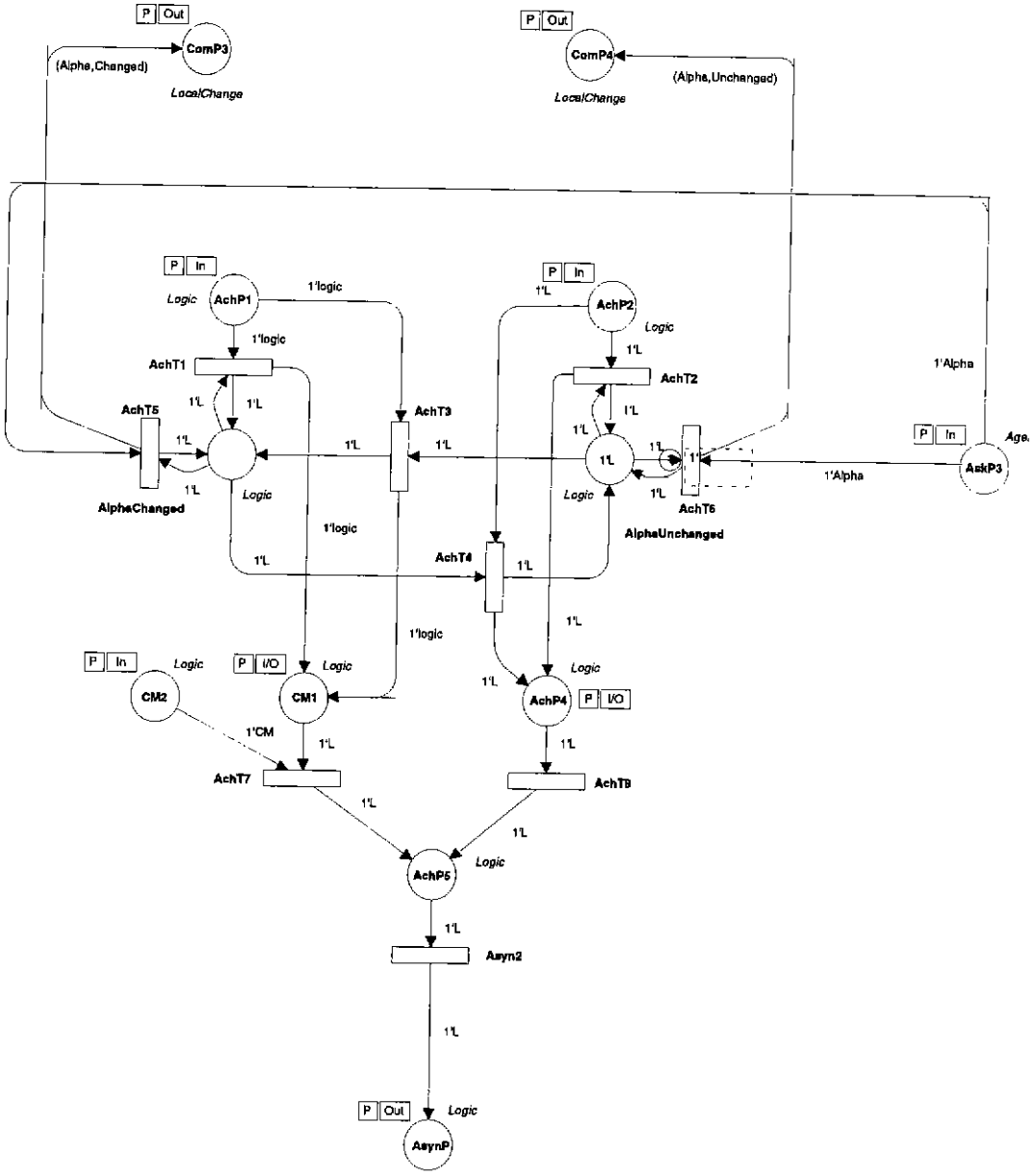


FIG. B.12 – *Agent α : conjecture modifiée*

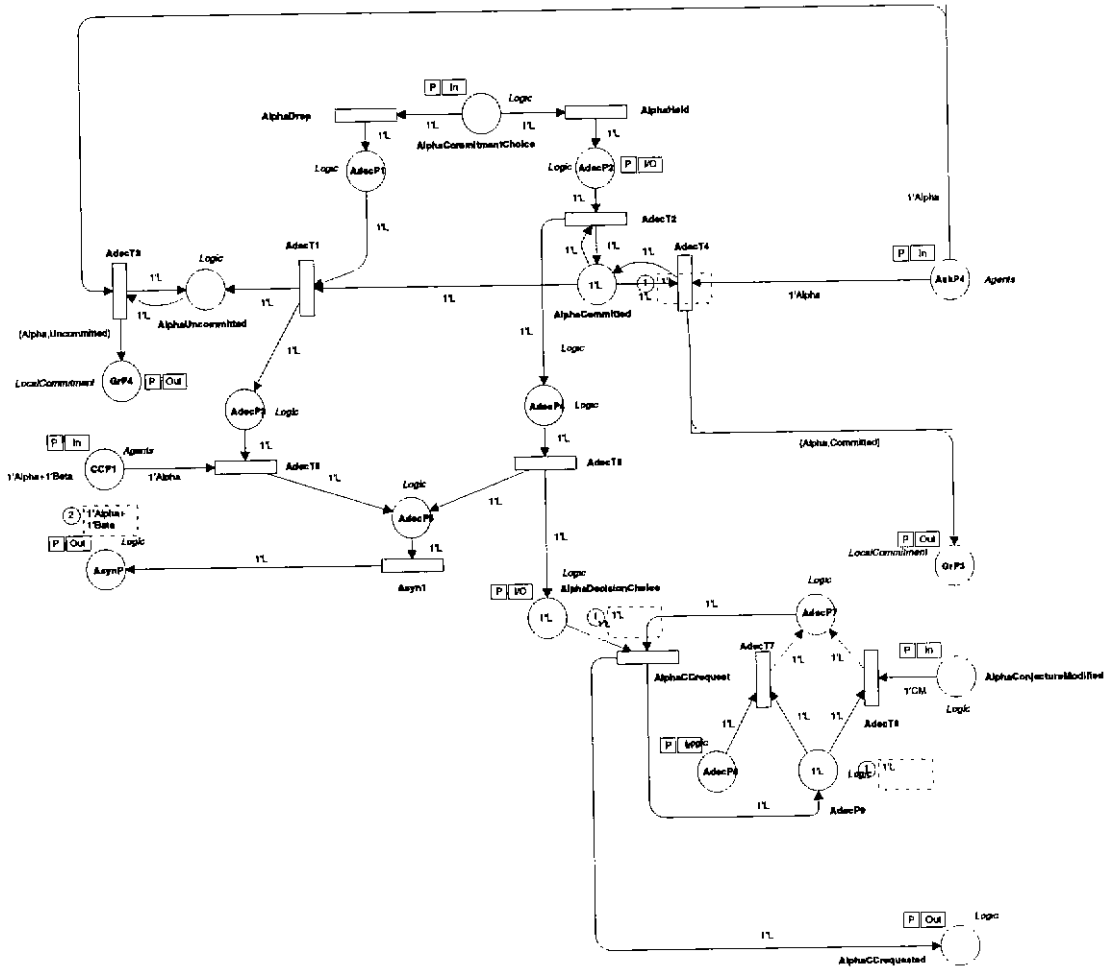


FIG. B.13 – *Agent α : décision d'engagement*

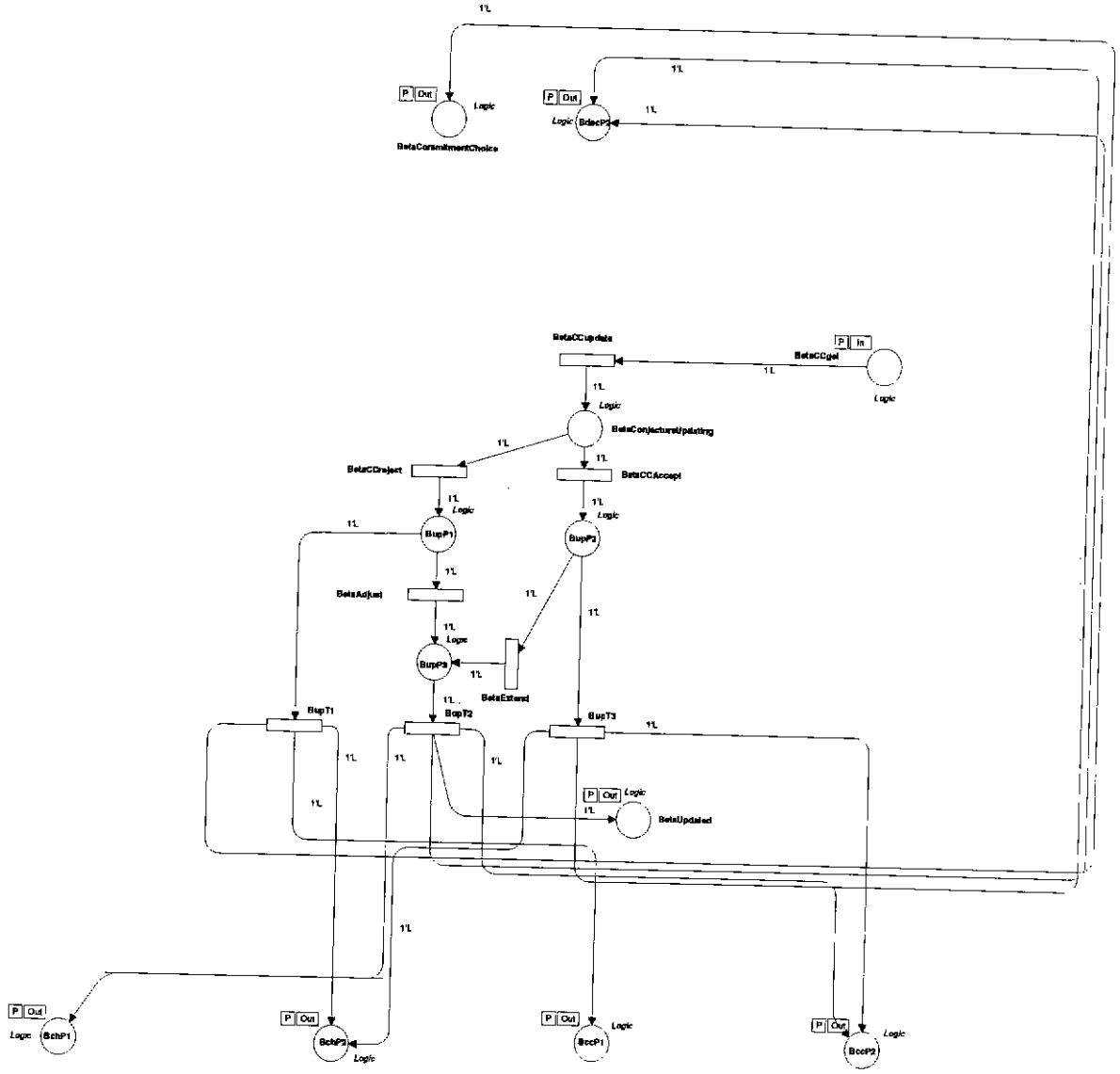


FIG. B.14 – *Agent β : module de mise à jour de la conjecture*

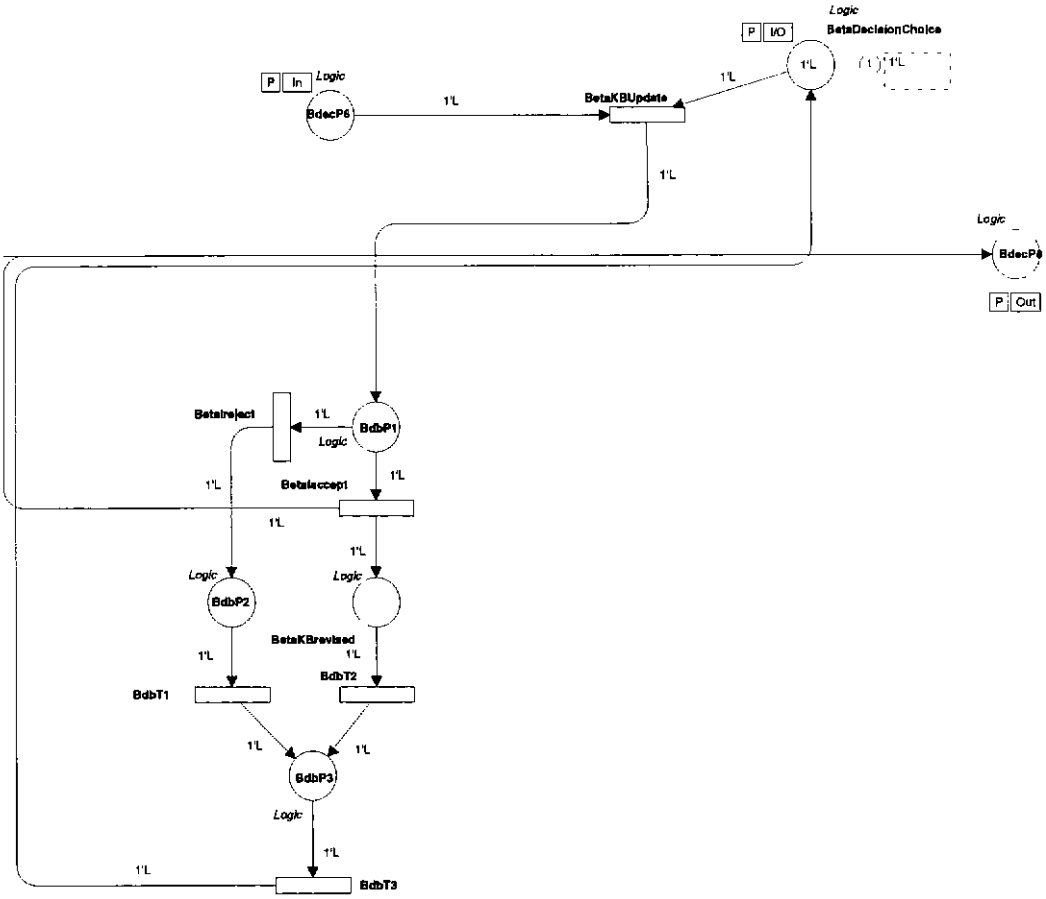


FIG. B.15 – *Agent β : module de mise à jour de la base de connaissances*

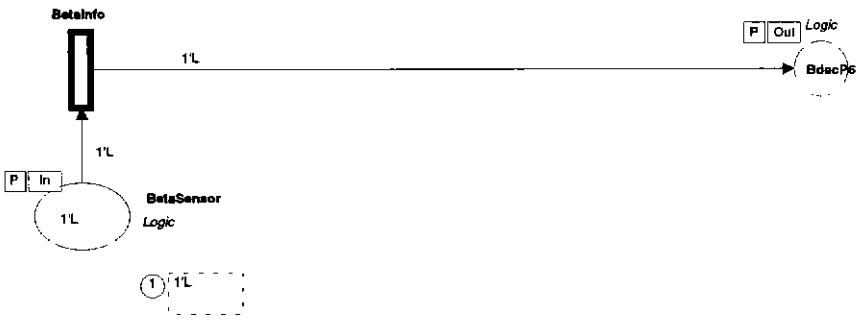


FIG. B.16 – *Agent β : réception des informations*

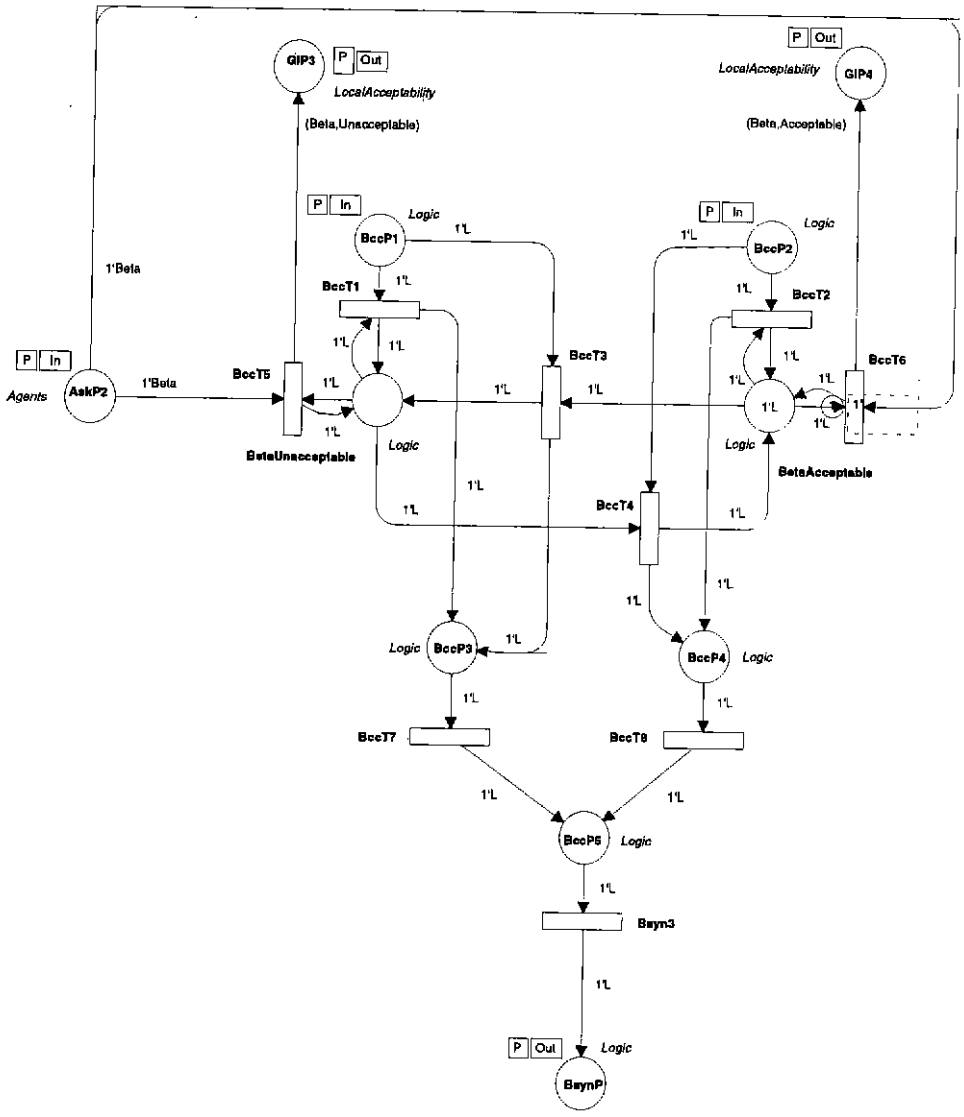


FIG. B.17 – *Agent β : admissibilité*

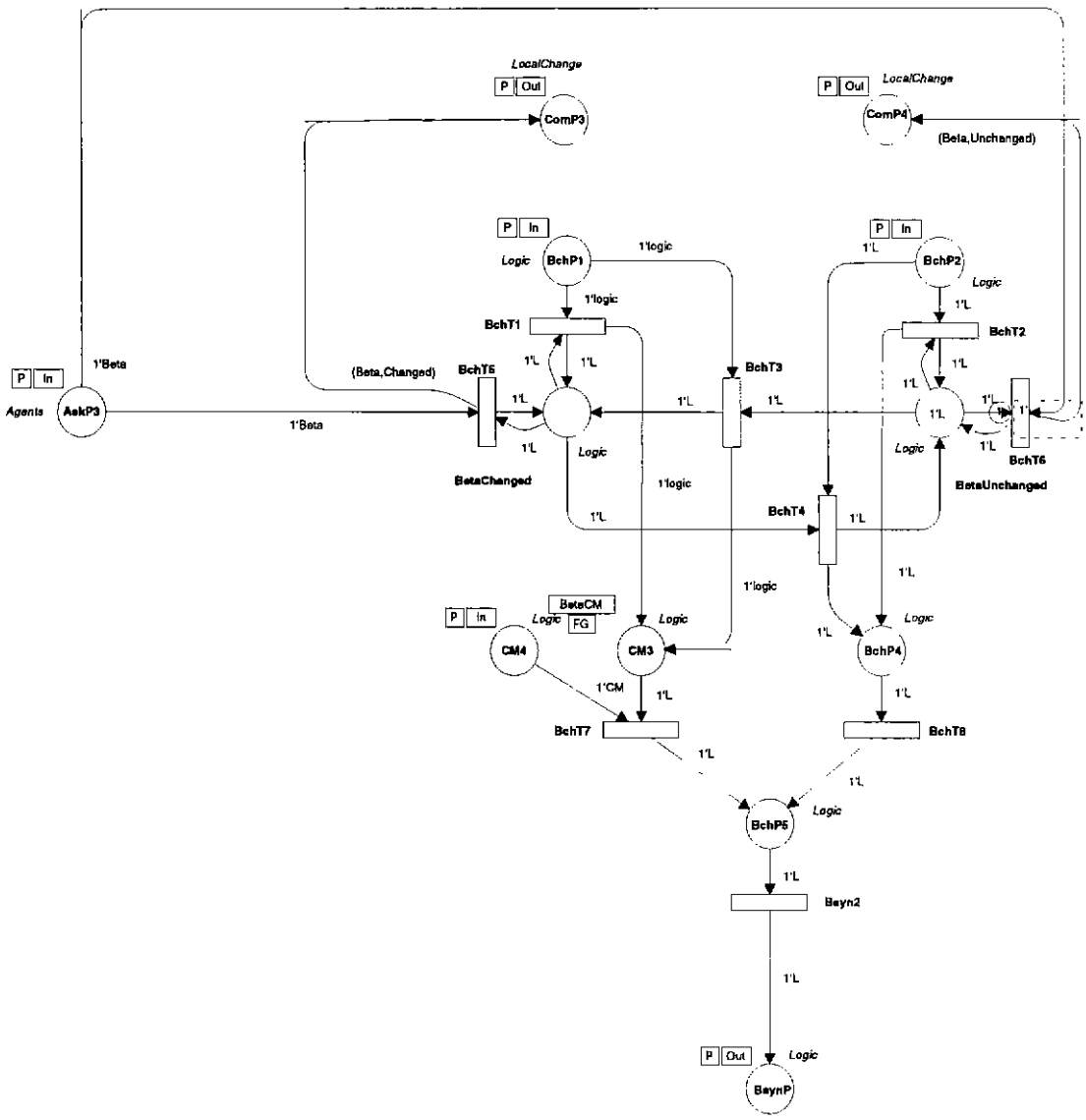


FIG. B.18 – *Agent β : conjecture modifiée*

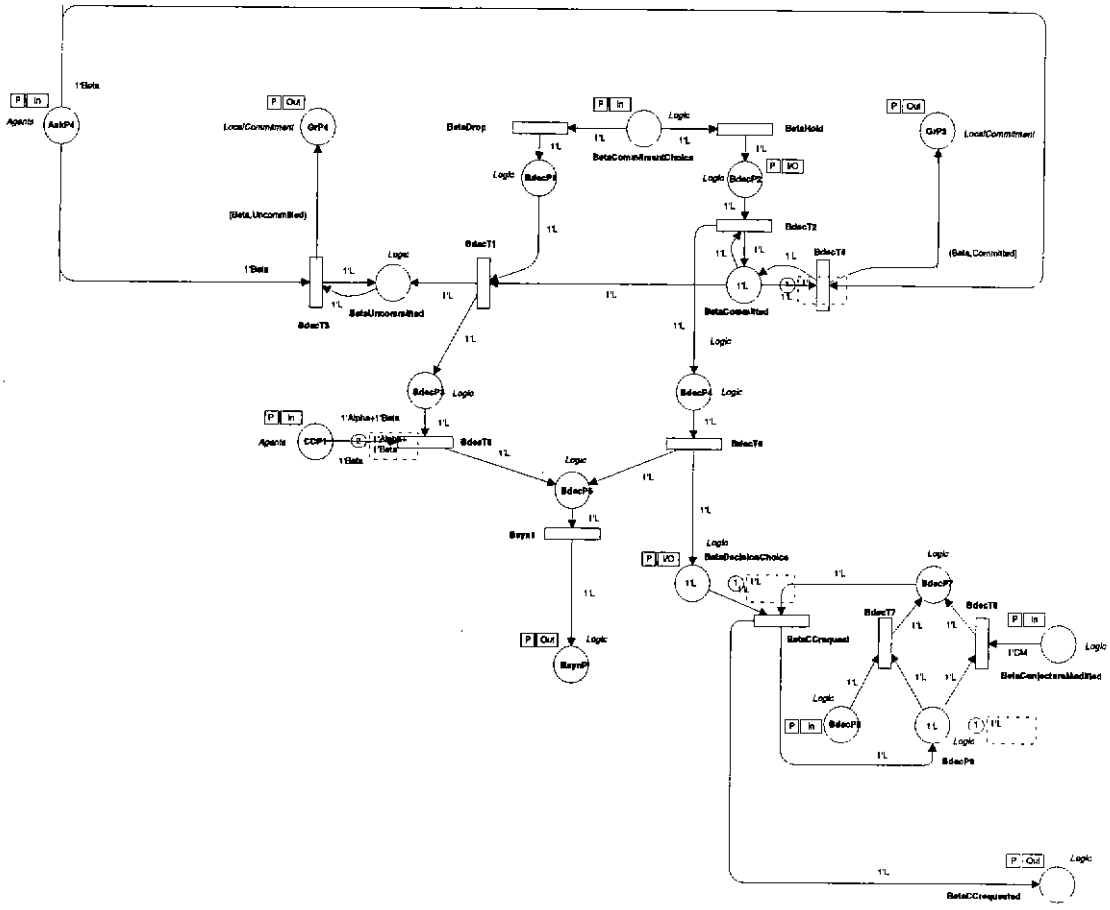


FIG. B.19 – *Agent β : décision d'engagement*

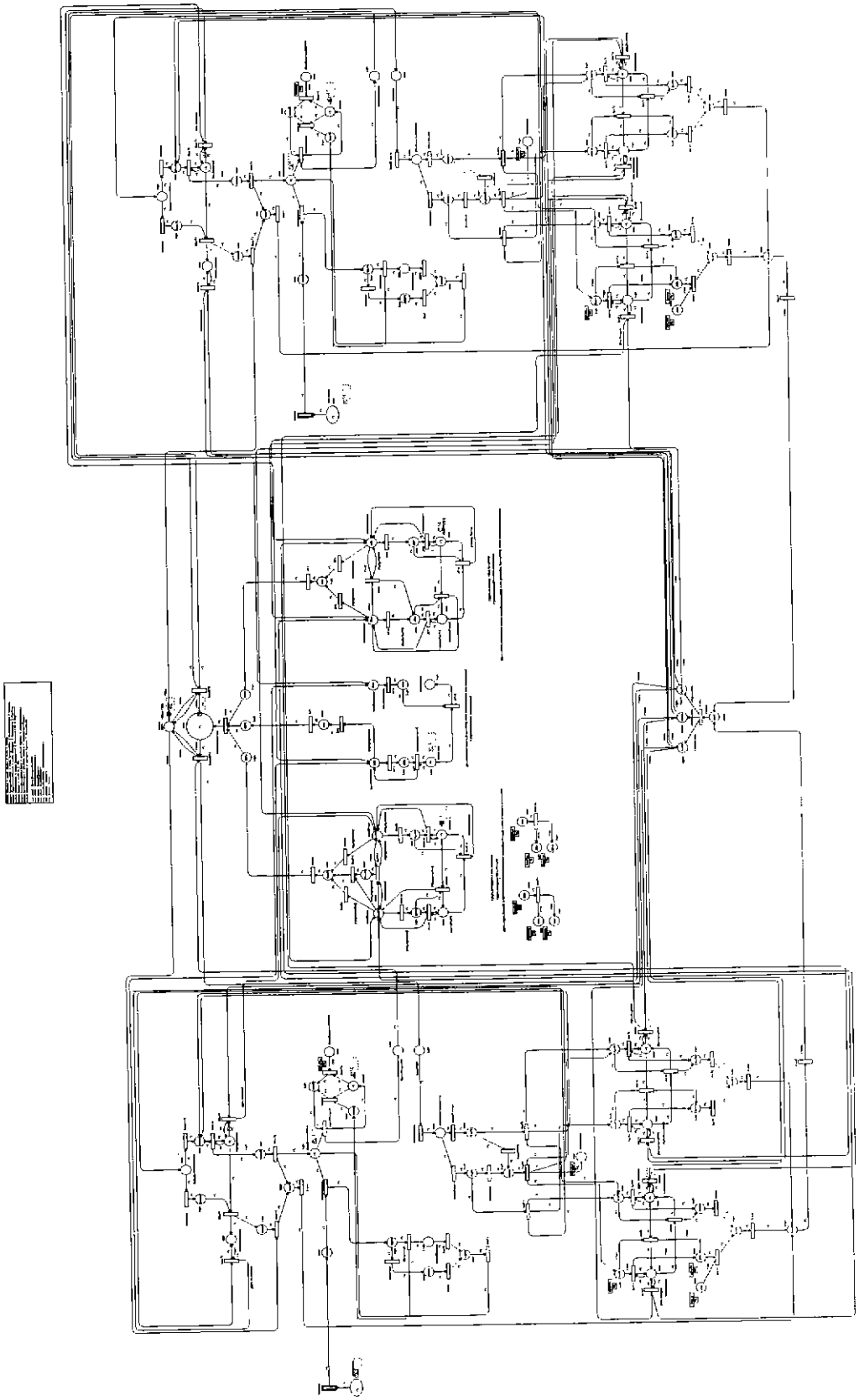


FIG. B.20 – *Mise en œuvre du modèle*

Annexe C

Représentation logique

C.1 Logique propositionnelle

La logique propositionnelle constitue la base de toutes les approches logiques développées en particulier en I.A. et est une référence pour le **traitement symbolique de l'information**.

C.1.1 Syntaxe

Un *ensemble dénombrable* P de *propositions* p et les cinq connecteurs logiques $\{ \neg, \wedge, \vee, \supset, \equiv \}$ forment le *vocabulaire* de la logique propositionnelle et le *calcul des propositions* fournit le cadre formel pour assembler ces «mots» en phrases syntaxiquement correctes : les *formules bien formées du langage.*

Définition C.1 *Le* **langage** $\mathcal{L}$ *est l'ensemble des* formules bien formées *qui sont obtenues sur la base du vocabulaire défini ci-dessus de la manière suivante :*

- *toute* proposition p *de l'ensemble* P *(dénombrable) des propositions du vocabulaire est une* formule bien formée,
- *si* ϕ *et* ψ *sont des* **formules bien formées** *du langage* $\mathcal{L}$, *alors* $\phi \vee \psi$, $\phi \wedge \psi$, $\neg\phi$, $\neg\psi$, $\phi \supset \psi$, $\phi \equiv \psi$ *sont des formules bien formées.*

Un sous-ensemble quelconque de formules bien formées du langage $\mathcal{L}$ constitue une **théorie logique** dans ce langage.

C.1.2 Sémantique

Une sémantique attribuée à ces formules se base sur le principe suivant : une *proposition est soit vraie, soit fausse* (du moins en logique propositionnelle). Le domaine sémantique est constitué des deux **valeurs de vérité**

X	¬X
V	**F**
F	**V**

TAB. C.1 – *Table de vérité pour la négation*

{**Vrai**, **Faux**} et interpréter une formule bien formée du langage revient à lui attribuer l'une des deux valeurs de vérités précédentes. Ceci se fait à partir des valeurs de vérités des propositions que contient la formule et de la façon dont ils sont reliés entre eux par les connecteurs logiques.

Un *littéral* est soit une proposition p, soit sa négation $\neg p$.

La méthode des tables de vérités permet d'obtenir simplement toutes les valeurs de vérité possible de n'importe quelle formule à partir des valeurs de vérités des littéraux qui la composent. Ainsi, la table C.1 donne la sémantique de la négation et la table C.2 décrit la sémantique attachée à chacun des connecteurs dyadiques du vocabulaire.

X	Y	X∧Y	X∨Y	X⊃Y	X≡Y	¬ X ∨ Y	¬ X ∨ ¬ Y	¬ X ∧ ¬ Y
V	**V**	**V**	**V**	**V**	**V**	**V**	**F**	**F**
V	**F**	**F**	**V**	**F**	**F**	**F**	**V**	**F**
F	**V**	**F**	**V**	**V**	**F**	**V**	**V**	**F**
F	**F**	**F**	**F**	**V**	**V**	**V**	**V**	**V**

TAB. C.2 – *Table de vérité pour les connecteurs dyadiques*

Par exemple, on peut vérifier les lois de De Morgan dans la table C.2 :

$$\neg(X \wedge Y) \Leftrightarrow (\neg X \vee \neg Y) \tag{C.1}$$
$$\neg(X \vee Y) \Leftrightarrow (\neg X \wedge \neg Y) \tag{C.2}$$

ou les équivalences logiques suivantes :

$$(X \supset Y) \Leftrightarrow (\neg X \vee Y) \tag{C.3}$$
$$(X \equiv Y) \Leftrightarrow (\neg X \vee Y) \wedge (\neg Y \vee X) \tag{C.4}$$

Remarque C.1 (algèbre de Boole de formules logiques)

Les équivalences logiques ci-dessus permettent de voir que l'ensemble de formules bien formées d'un langage $\mathcal{L}$ est muni d'une structure d'algèbre de Boole ($(L, \vee, \wedge, \neg)$, algèbre de Lindenbaum-Tarski (Carvallo, 1962)) par les opérateurs dyadiques $\vee$ et $\wedge$ et l'opérateur involutif $\neg$.

De même, dans la table C.3, on voit que les trois connecteurs $\vee$, $\wedge$ et $\equiv$ peuvent être redéfinis à partir de la négation $\neg$ et de l'implication $\supset$.

X	Y	X∧Y	¬X⊃Y	X∨Y	¬(X⊃¬Y)	X≡Y	(X⊃Y)∧(Y⊃X)
V	**V**	**V**	**V**	**V**	**V**	**V**	**V**
V	**F**	**F**	**F**	**V**	**V**	**F**	**F**
F	**V**	**F**	**F**	**V**	**V**	**F**	**F**
F	**F**	**F**	**F**	**F**	**F**	**V**	**V**

TAB. C.3 – *Redéfinitions possibles de* $\wedge$, $\vee$ *et* $\equiv$.

Définition C.2 *Une* **interprétation** *est alors une fonction i qui a toute proposition p associe une valeur de vérité* **V** *ou* **F** *: son extension I par la méthode des tables de vérité associe également une valeur de vérité* **V** *ou* **F** *à toute formule bien formée du langage.*

Étant donné une formule f, toute interprétation I qui donne la valeur de vérité **V** *à f est appelée un* **modèle** *de la formule f (on peut noter* $[\![f]\!]$ *l'ensemble des modèles de f).* [1]

C.1.3 Formules valides, inconsistantes

Définition C.3 *Une formule est alors* **consistante** *si et seulement si elle admet au moins un modèle, c'est-à-dire s'il existe une interprétation telle qu'elle reçoive la valeur de vérité* **V** *: par exemple,* $(p \vee q)$ *et* $(p \wedge q)$ *sont consistante.*

Définition C.4 *Une formule f est :*

- **inconsistante** *si et seulement si elle n'admet aucun modèle* $[\![f]\!] = \emptyset$ *:* $(p \wedge \neg p)$ *est inconsistante.*
- **valide** *si et seulement si elle prend la valeur de vérité* **V** *dans toutes les interprétations possibles : ainsi* $(p \vee \neg p)$, $(p \supset p)$ *et* $(\neg\neg p \equiv p)$. *On note* $\models f$.
- **contingente** *lorsqu'elle n'est ni valide, ni inconsistante, comme* $(p \vee q)$ *et* $(p \wedge q)$.

On trouvera dans (Thayse and al., 1990) une description des algorithmes classiques que l'on peut mettre en œuvre pour décider si une formule propositionnelle est valide (resp. consistante) à savoir l'algorithme de réduction, l'algorithme de Quine, la stratégie de Davis et Putnam, et les méthodes de preuve fondées sur le *principe de résolution.* La première étape du travail à

1. Cette définition d'un *modèle* ne doit pas être confondue avec la notion du même nom en Automatique ou en Mathématiques, qui est une description du système étudié : on parlera en particulier dans la suite de *modèle d'évolution* en faisant référence à la définition communément admise en Automatique et non à celle utilisée en Logique.

accomplir passe souvent par la mise sous **forme conjonctive normale** de la formule f :

Définition C.5 *Une* **clause** *est une disjonction ($\vee$) finie de littéraux ($p_1 \vee \ldots \vee p_n$).*
Une **clause de Horn** *est une clause contenant au plus un littéral positif ($\neg p_1 \vee \ldots \vee \neg p_n \vee q$).*
Une **forme conjonctive normale** *est une conjonction ($\wedge$) finie de clauses :* $(p \vee \neg q) \wedge (p \vee r \vee s) \wedge \ldots \wedge (t \vee v \vee \ldots \vee w)$.

Toutefois, il faut noter que le problème de décider si une formule du calcul des propositions est valide (resp. consistante) est de complexité *NP-complet* et que la méthode de résolution ne s'avère efficace que dans les cas particuliers où f se met sous forme d'une conjonction de *clauses de Horn* (cf. définition C.5 ci-dessus).

Propriété C.1 *Si on considère un ensemble Φ de formules, un* **modèle** *de Φ est une interprétation du langage telle que toutes les formules de Φ soient vraies. L'ensemble des modèles de Φ est donc l'intersection des ensembles de modèles de chaque formule de Φ :*

$$[\![\Phi]\!] = \cap_{f \in \Phi} [\![f]\!]$$

Définition C.6 *Une théorie Φ est dite* complète *si dans les modèles de Φ toute formule du langage est soit valide, soit inconsistante.*

Une théorie Φ est dite catégorique *si elle admet un modèle unique (à un isomorphisme près entre différents domaines d'interprétation).*[2]

Propriété C.2 *Si Φ admet plusieurs modèles, chaque modèle ω est une interprétation rendant vraie toutes les formules de Φ d'une part et attribuant une valeur de vérité aux autres propositions et formules du langage d'autre part.*

Pour chaque modèle ω, on obtient une théorie catégorique décrivant ω *en ajoutant à Φ toutes les formules n'apparaissant pas dans Φ et rendues vraies dans le modèle ω* et *la négation de toutes les formules n'apparaissant pas dans Φ et rendues fausses dans ω.*

Définition C.7 *Une théorie Φ est* **inconsistante** *si elle n'admet aucun modèle :*

$$[\![\Phi]\!] = \emptyset.$$

2. Il est vivement souhaitable que toute théorie complète soit catégorique et inversement.

C.1.4 Théorie logique

On dira que φ est une **conséquence logique** de Φ ($\Phi \models \varphi$) si et seulement si tout modèle de Φ est un modèle pour φ. Partant d'un ensemble de formules Φ, on peut ainsi former sa **fermeture logique** $CL(\Phi)$ contenant toutes les conséquences logiques de Φ ($\Phi \in CL(\Phi)$). $CL(\Phi)$ et Φ ont évidemment les mêmes modèles et donc la même sémantique : $[\![CL(\Phi)]\!] = [\![\Phi]\!]$.

On appellera **théorie logique** un ensemble de formules du langage et *théorie logique fermée* une théorie contenant sa propre fermeture logique ($CL(\Phi) = \Phi$).

Ainsi lorsque chaque formule d'une théorie $\mathcal{A}$ s'interprète comme une propriété de l'environnement, la théorie $\mathcal{A}$ constitue une description de l'état de cet environnement. Malheureusement la logique des propositions n'offre qu'un pouvoir d'expression limité : en particulier, elle ne permet pas de raisonner sur des lois générales s'appliquant à un ensemble d'individus pris dans un ensemble : les propositions manipulées sont des formes atomiques et dans une formule ne présente aucune propriété générique telle que l'on pourrait échanger une proposition p_1 pour une autre p_2 et déduire sans autre calcul les mêmes propriétés vraies pour p_1 le sont pour p_2.

En d'autres termes, il manque à la logique propositionnelle la possibilité de manipuler des variables et des expressions dépendant de la valeur prise par ces variables.

C.2 Logique des prédicats du premier ordre

C.2.1 Vocabulaire

Le vocabulaire du calcul des prédicats du premier ordre reprend les connecteurs dyadiques du calcul des prédicats { $\neg$, $\wedge$, $\vee$, $\supset$, $\equiv$ }. S'y ajoutent en particulier les symboles suivants :

- des constantes (ou constantes individuelles),
- des variables,
- des prédicats, dont très couramment le prédicat d'égalité $=$,
- des fonctions (ou constantes fonctionnelles),
- le quantificateur universel $\forall$ et existentiel $\exists$.

Les constantes sont des symboles directement mis en correspondance avec les objets que l'on décrit ou sur lesquels on veut raisonner. Les variables

($\{x, y, z, ...\}$) peuvent être instanciées dans un ensemble de constantes possibles ($\{a, b, c, ...\}$). Les fonctions ($f(x, z), g(x, y, z), \ldots$) prennent comme arguments des variables ou des constantes pour retourner des valeurs prises parmi l'ensemble de constantes possibles. Les «mots» constitués des variables, des constantes et des fonctions appliquées à ces variables ou constantes forment les **termes** du langage : $t_1 := x$, $t_2 := g(x, y, b)$, $t_3 := a, \ldots$.

Les **prédicats** ($P(t_1, ..., t_n), Q(t_1, t_2), \ldots$) ont pour arguments les termes du langage. Le langage prédicatif comporte de même toutes les *formules* bien formées à partir des *formules atomiques* ($\phi := P(t_1, ..., t_n)$, $\psi := Q(t_1, t_2)$, $\varphi := (t1 = t_2), \ldots$), des connecteurs dyadiques de la logique propositionnelle $\{ \neg, \wedge, \vee, \supset, \equiv \}$ et des quantificateurs $\forall$ et $\exists$: si ϕ et ψ sont des formules bien formées du langage $\mathcal{L}$, alors $\phi \vee \psi$, $\phi \wedge \psi$, $\neg\phi$, $\neg\psi$, $\phi \supset \psi$, $\phi \equiv \psi$, $\forall x\, \phi$, $\exists x\, \psi, \ldots$ sont des formules bien formées.

Définition C.8 (formule fermée)

Une formule fermée *est une formule où toutes les variables sont liées. Lorsqu'une variable x apparaît dans une formule bien formée φ du langage, on dit que x est* quantifiée *si x apparaît sous une occurrence de la forme $\forall x$ ou $\exists x$. La variable x est* liée *s'il y a dans φ une occurrence quantifiée de x, elle est* libre *si elle n'est ni liée ni quantifiée.*

Le langage prédicatif contient le langage propositionnel car une proposition est un prédicat sans arguments.

À titre d'exemple, voici une théorie logique simple dans le formalisme de la logique des prédicats du premier ordre ($dans(x, y)$ se traduit par «*x est dans y*» et $sur(x, y)$ se traduit par «*x est sur y*») :

Exemple C.1

$$\mathcal{A} = \begin{cases} \exists y, & dans(Fred, y) \wedge (\neg(y = vélo)) \\ \exists y, & dans(vélo, y) \wedge (\neg(y = Fred) \wedge \neg(y = bureau_114)) \\ \forall x, y, & dans(x, y) \supset \neg(dans(y, x)) \\ \forall x, y, & dans(x, y) \supset \neg(x = y) \\ \forall x, y, & sur(x, y) \supset \neg(x = y) \\ \forall x, y, z, & (sur(x, y) \wedge dans(y, z)) \supset dans(x, z) \\ \forall x, y, z, & (sur(x, y) \wedge dans(x, z)) \supset dans(y, z) \\ \forall x, y, z, & (dans(x, y) \wedge \neg(y = z)) \supset \neg(dans(x, z)) \\ \forall x, y, & sur(x, y) \supset ((x = Fred) \wedge (\neg(dans(Fred, bureau_114))) \\ & \wedge (y = vélo)) \end{cases}$$

C.2.2 Interprétation

De même que dans le cas de la logique propositionnelle, on s'intéresse à la sémantique d'une formule, ou d'un ensemble de formules en recherchant

ses modèles. Une **interprétation** I pour une théorie logique, consiste ainsi à associer une valeur de vérité à chaque formule bien formée du langage. L'extension du vocabulaire entre le calcul des propositions et le calcul des prédicats entraîne cependant une certaine complication.

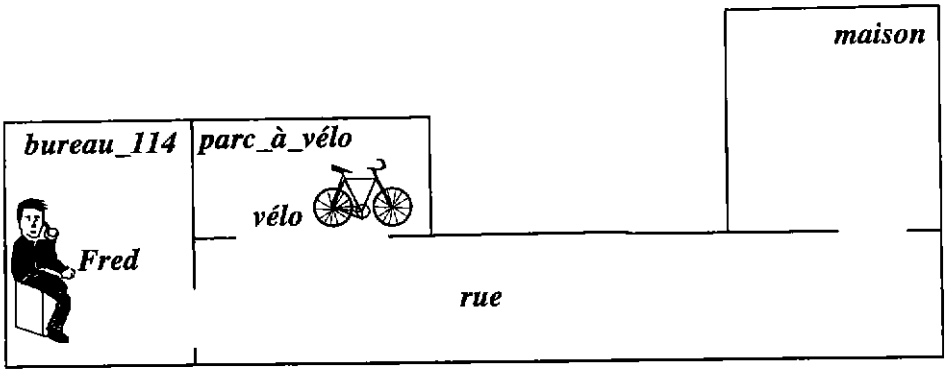


FIG. C.1 – *Un état possible de l'environnement de Fred*

L'interprétation du langage demande la définition d'un **domaine d'interprétation** D constitué d'un ensemble non vide de symboles faisant référence à des entités que l'on pourra appeler *objets de l'environnement.* Ainsi, l'interprétation des constantes dans le calcul des prédicats fournit le lien premier et trivial entre chaque symbole de constante a et un objet de l'environnement représenté dans D : ainsi seules les constantes *Fred*, *vélo*, *bureau*_114 apparaissent dans $\mathcal{A}$, pour retrouver le monde de la figure C.1, il faut rajouter *parc_à_vélo*, *rue* et *maison* mais on pourrait rajouter aussi *ailleurs* (cf. figure C.2), etc. . L'interprétation d'une variable x se fait alors en faisant correspondre x avec un des éléments de D. L'interprétation d'une fonction f de n variables se fait en faisant correspondre à tout n-uplet de D^n un élément de D.

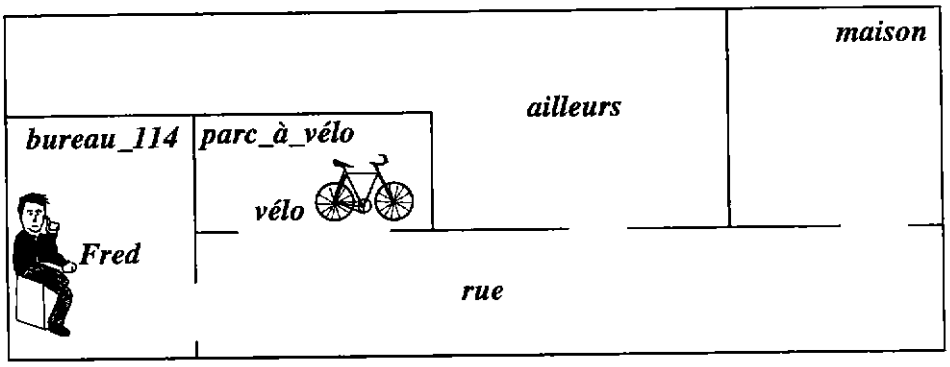


FIG. C.2 – *Un état possible de l'autre environnement de Fred*

On voit aisément que le choix de D conditionne l'ensemble des termes que l'on pourra générer dans le langage et donc l'interprétation correspondante. Ainsi, la théorie $\mathcal{A}$ peut décrire plusieurs environnements différents de Fred, pour lesquels il y a plusieurs états possibles à chaque fois. En ce sens, il existe une **infinité d'interprétations possibles** des formules du calcul des prédicats si l'on ne restreint pas le *domaine d'interprétation* et les termes du langage.

Dans notre exemple, on peut choisir (cf. figure C.1) :

$$D = \{Fred,\ vélo,\ parc_à_vélo,\ rue,\ bureau_114,\ maison\}$$

On obtient alors les différentes positions possibles du vélo :

$$dans(vélo, parc_à_vélo) \ \vee \ dans(vélo, rue) \ \vee \ dans(vélo, maison)$$

et de Fred :

$$dans(Fred, bureau_114) \ \vee \ dans(Fred, parc_à_vélo) \ \vee \ dans(Fred, rue) \ \vee \ dans(Fred, maison)$$

Si D devient (cf. figure C.2):

$$D = \{Fred,\ vélo,\ parc_à_vélo,\ rue,\ bureau_114,\ maison,\ ailleurs\}$$

on obtient pour le vélo :

$$dans(vélo, parc_à_vélo) \ \vee \ dans(vélo, rue) \ \vee \ dans(vélo, maison) \ \vee \ dans(vélo, ailleurs)$$

et pour Fred :

$$dans(Fred, bureau_114) \ \vee \ dans(Fred, parc_à_vélo) \ \vee \ dans(Fred, rue) \ \vee \ dans(Fred, maison) \ \vee \ dans(Fred, ailleurs)$$

On voit en particulier l'influence du domaine d'interprétation choisi lorsque la théorie comprend des *négations* de formules comportant des variables : pour supprimer ces négations de la théorie, on est contraint à l'énumération des valeurs possibles des variables. Cela revient à recenser tous les termes possibles du langage. Il est à noter que si la théorie logique étudiée comprend une fonction $f(x)$ de la variable x (ce n'est pas le cas de $\mathcal{A}$) et que $D \neq \emptyset$, alors l'ensemble des *termes* du langage comprend au moins le sous-ensemble des termes de la forme : (si $a \in D$) $f(a)$, $f(f(a))$, $f(f(f(a)))$, $f(f(f(f(a))))$, $\ldots$, $f^{(n)}(a)$, $\ldots$, qui est infini (dénombrable).

Pour un prédicat donné P à n arguments, si l'on connaît l'ensemble de tous les termes possibles du langage et que celui-ci est fini (de cardinal N), on peut former toutes les formules atomiques de la formes $P(t_1, ..., t_n)$, mais ceci peut

devenir très long (N^n possibilités). Mieux vaudrait pouvoir se restreindre à un sous-ensemble de ces formules atomiques : par exemple celles ne comportant que des constantes en arguments.

De même, si l'on répète l'opération d'énumération des formules atomiques possibles à partir de chaque prédicat du langage, on forme l'ensemble des formules atomiques du langage. Mais si on cherche alors à attribuer une valeur de vérité à chacune de ces formules atomiques, on doit d'abord parcourir pour chaque variable le domaine d'interprétation D dans lequel elle s'interprète. Ainsi il apparaît encore une fois qu'on devrait pouvoir ne s'intéresser qu'aux formules atomiques n'ayant plus que des constantes comme arguments et plus aucune variable : de plus il semble judicieux de ne prendre en compte que le sous-ensemble minimal des constantes de D intervenant réellement dans la théorie en question.

Cela nous amène tout naturellement à parler des **interprétations de Herbrand**.

C.2.3 Interprétations de Herbrand

Soit une formule f dont on souhaite tester l'inconsistance. On peut réécrire f sous **forme clausale**[3], c'est-à-dire en une expression équivalente F de la forme :

$$F \ : = \ \forall x_1 \forall x_2 ... \forall x_n \, (M_f)$$

où (M_f) est une *forme conjonctive normale* (cf. définition C.5 en page 190).

Une **interprétation de Herbrand** de F est une interprétation dans le domaine particulier D_f^H appelé *domaine ou univers de Herbrand.* Celui-ci est défini de manière à restreindre au maximum le domaine d'interprétation permettant de décider si la forme clausale F est inconsistante.

Le *domaine de Herbrand* D_f^H de la forme clausale F est ainsi l'ensemble minimal composé des éléments suivants :

- pour chaque constante a apparaissant dans F, on associe la constante de Herbrand[4] a dans D_f^H ;

3. Une expression logique de la forme $S \ : = \ Q_1 x_1 \, Q_2 x_2 ... Q_n x_n \ (M)$, où les Q_i sont des quantificateurs ($\forall$ ou $\exists$) et M une formule du langage sans quantificateur, est appelée *forme prénexe* (M est la matrice et $Q_1 x_1 \, Q_2 x_2 ... Q_n x_n$ est le préfixe) - si tous les Q_i sont des quantificateurs universels ($S \ : = \ \forall x_1 \forall x_2 ... \forall x_n \ M$), c'est une *forme de Skolem*, et si M est sous forme conjonctive normale (cf. définition C.5 en page 190), alors S est une **forme clausale**. On trouvera dans (Thayse and al., 1990) un exposé plus étoffé quoique limité aux logiques du premier ordre sans prédicat égalité = (pour plus de détail, il faut se reporter à (Chang and Lee, 1973) et (Manna, 1974)).

4. On peut choisir un symbole différent pour la constante de Herbrand et la constante apparaissant dans f, mais on a toujours la possibilité d'identifier les symboles par la suite.

- s'il n'y a aucune constante dans F, on choisit une constante de Herbrand par défaut $c \in D_f^H$;
- pour chaque fonction g de n variables apparaissant dans F, on rajoute dans D_f^H l'ensemble des termes s'écrivant à partir du symbole g et des n-uplet $\{h_1, h_2, \ldots h_n\}$ d'éléments de D_f^H sous la forme $g(h_1, h_2, \ldots h_n)$.

Proposition C.1 *En présence d'une fonction dans F, le domaine de Herbrand est toujours infini car au moins dénombrable. Dans la pratique, on se limitera exclusivement à des* **domaines d'interprétations au plus dénombrables mais le plus souvent finis.**

Dans l'exemple de la théorie $\mathcal{A}$,

$$F := \forall x \forall y \left\{ (\neg(sur(x,y)) \vee (x = Fred)) \wedge \ (\neg(sur(x,y)) \vee (y = vélo)) \wedge (\neg(sur(x,y)) \vee dans(Fred, bureau_114)) \right\}$$

est la forme clausale (cf. note 3 page 195) pour la formule f : $f := \left\{ \forall x, y, \quad sur(x,y) \supset ((x = Fred) \wedge (\neg(dans(Fred, bureau_114))) \wedge (y = vélo)) \right\}$ et il n'y a pas de fonctions :

$$D_f^H := \{ Fred, bureau_114, vélo \}$$

Les éléments de D_f^H sont des symboles à fonction purement syntaxique, sans signification concrète. Cela revient à instancier toutes les variables des formules atomiques pour se ramener à un vocabulaire uniquement composé des *formes fondamentales* qui sont les formes prédicatives du type : $P(t_1, t_2, \ldots t_n)$ dont les prédicats P interviennent dans F (ou f, c'est pareil) et les termes $(t_1, t_2, \ldots t_n)$ appartiennent à D_f^H. Dans notre exemple, il y a trois prédicats à deux places et trois éléments dans D_f^H, donc $3 \times 3^2 = 27$ formes fondamentales.

Les interprétations de Herbrand de F sont alors exactement déterminées par les valeurs de vérité associées à ces formes fondamentales. Ainsi, les interprétations de Herbrand peuvent être assimilées à des interprétations du calcul des propositions avec le vocabulaire des formes fondamentales associées à F. Un **modèle de Herbrand** de F est une interprétation de Herbrand qui rend F **Vrai**. L'intérêt des interprétations de Herbrand vient du résultat suivant :

Théorème C.1 *Une forme clausale F est inconsistante* si et seulement si *elle est fausse pour toutes les interprétations de Herbrand.*

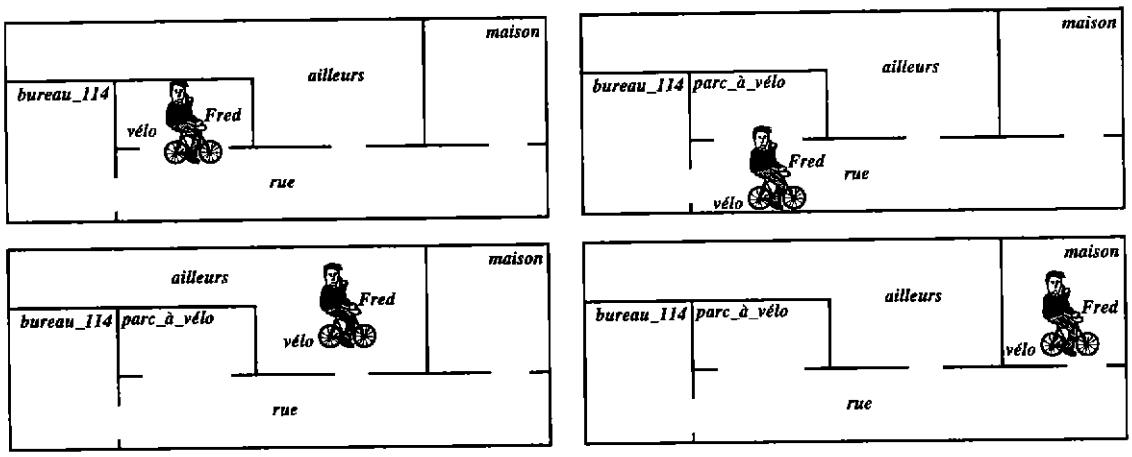


FIG. C.3 – *Modèles où Fred est sur le vélo*

La condition est évidemment nécessaire, et pour prouver qu'elle est suffisante, on montre que pour tout modèle de F avec un domaine d'interprétation D, on peut construire un modèle de F, dit *modèle de Herbrand* avec le domaine de Herbrand D_f^H. On trouvera la démonstration de ceci dans (Thayse and al., 1990).

C.3 Modèles de Herbrand et mondes possibles

C.3.1 Modèles de Herbrand

La formule de notre exemple n'est pas inconsistante puisqu'elle admet au moins le modèle de Herbrand qui correspond aux figures C.1 et C.2 :

Vrai	**Faux**
$dans(Fred, bureau_114)$	$dans(vélo, bureau_114)$
	$sur(Fred, vélo)$

et celui correspondant aux cas de la figure C.3.

Vrai	**Faux**
$sur(Fred, vélo)$	$dans(Fred, bureau_114)$
	$dans(vélo, bureau_114)$

Il y a d'autres modèles de Herbrand de f (cf. figure C.4). En particulier, nous n'énumérerons pas les modèles possibles de f qui ne sont pas compatibles avec la théorie $\mathcal{A}$ de l'exemple C.1 (et pas représentables par une figure), mais il y a notamment tous les modèles où $\neg(sur(Fred, vélo))$ est **Vraie** et où les

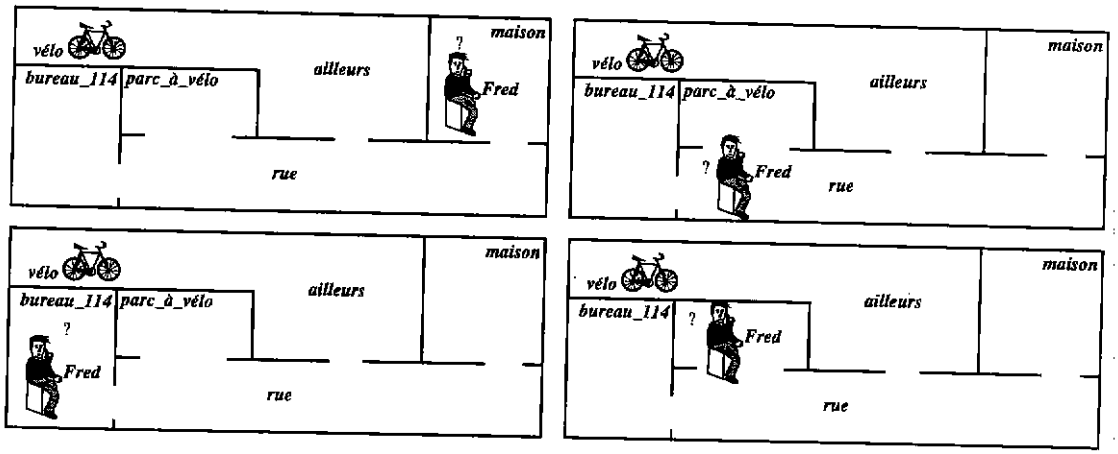


FIG. C.4 – *Fred cherche son vélo*

formes fondamentales suivantes sont également interprétées à **Vraies**.

$$\left|\begin{array}{l} dans(Fred, vélo) \\ dans(Fred, Fred) \\ dans(vélo, vélo) \\ dans(vélo, Fred) \\ dans(vélo, bureau_114) \\ dans(bureau_114, bureau_114) \\ dans(bureau_114, Fred) \\ dans(bureau_114, vélo) \end{array}\right|$$

On voit par exemple que dans les interprétations de Herbrand de la formule F de notre exemple, la position du *vélo* n'est pas contrainte et on peut ainsi exhiber l'interprétation de Herbrand suivante, qui est un modèle de Herbrand de F et n'en est pas un pour $\mathcal{A}$:

Vrai	**Faux**
$sur(Fred, vélo)$	$dans(Fred, bureau_114)$
$dans(Fred, Fred)$	
$dans(vélo, bureau_114)$	
$dans(bureau_114, vélo)$	

C.3.2 Inconsistance, déduction

Consistance, inconsistance

Grâce aux interprétations de Herbrand, on voit que pour tester la consistance d'un ensemble de formules Φ, on peut se restreindre à l'ensemble des instances fondamentales de Φ, c'est-à-dire aux formules obtenues en instanciant les variables dans le domaine de Herbrand.

On peut même se ramener aux instances fondamentales des clauses intervenant dans les formules de Φ, une fois celles-ci mises sous forme conjonctive normale : la consistance (resp. l'inconsistance) de cet ensemble est équivalente à la consistance (resp. l'inconsistance) de l'ensemble de formules de départ Φ.

Si le domaine de Herbrand est **fini**, on peut appliquer les algorithmes du calcul des propositions (Davis et Putnam, méthode de résolution) avec les mêmes réserves dues à la complexité algorithmique du problème. Dans le cas contraire, cela reste possible en théorie mais n'a pas grand intérêt du point de vue de l'efficacité. Dans ce cas, on fait appel à l'opération d'*unification* ((Thayse and al., 1990), (Manna, 1974)) pour appliquer la méthode de résolution à des *clauses de Horn*[5] prédicatives non instanciées (puisqu'elles peuvent avoir une infinité d'instanciations possibles). Le principe même de la programmation logique s'appuie sur cette méthode.

Approches axiomatiques et déduction

Il existe également des *approches axiomatiques* de la preuve en logique : elles font appel d'une part à un schéma d'*axiomes* (Kleene, 1952), c'est-à-dire des formules du langage postulées vraie, et d'autre part à la mise en œuvre de *règles d'inférences*, c'est-à-dire des mécanismes de déduction de formules vraies à partir d'autres formules vraies. On utilise le symbole de déduction $\vdash$ de la façon suivante (f une formule et φ un ensemble de formules) :

$\vdash f$	$\longleftrightarrow$	*f peut être déduit à partir du schéma d'axiomes*
$\varphi \vdash f$	$\longleftrightarrow$	*f peut être déduit à partir du schéma d'axiomes augmenté des formules de φ considérée comme vraies.*

La règle du **Modus Ponens** s'énonce alors de la façon suivante :

$$Si\ \{\vdash P\}\ et\ \{\vdash (P \vdash Q)\}\ alors\ \{\vdash Q\}$$

Un système axiomatique est *adéquat* si $\forall f$:

$$\vdash f \longrightarrow \models f$$

Un système axiomatique est *complet* si $\forall f$:

$$\models f \longrightarrow \vdash f$$

La déduction $\vdash$ et la conséquence logique $\models$ sont *équivalentes* si et seulement si le système est *adéquat* et *complet*.

Dans un système axiomatique donné un ensemble de formules φ est inconsistant si et seulement si $\varphi \vdash Faux$.

5. Voir définition C.5 en page 190

C.3.3 Décidabilité

Définition C.9 *Une théorie Φ est* **décidable** *s'il existe un algorithme permettant de décider en temps* **fini** *si une formule $f \in \Phi$ est valide, inconsistante ou contingente.*[6]

Du fait de l'existence d'une infinité d'interprétations possibles pour les formules du calcul des prédicats, il n'existe pas d'algorithme général pour **décider** si une formule du calcul des prédicats est valide, contingente ou inconsistante (voir (Thayse and al., 1990)).

De fait, la logique des prédicats n'est pas réellement adaptée pour la description d'un état unique de l'environnement, mais plutôt pour la description de tous les états possibles, ce qui se traduit par l'**explosion combinatoire** de l'espace de recherche et du temps de calcul à laquelle on doit parfois faire face lorsqu'on manipule des théories logiques, même lorsqu'il existe un algorithme en temps fini (voir définition C.9).

Il y a pourtant un point pour lequel le fait de travailler pour la représentation d'un environnement donné est un avantage indéniable : le domaine d'interprétation est connu a priori. Il est alors souhaitable de construire la théorie logique Φ de manière adaptée au monde décrit et en y faisant apparaître tous les objets de l'environnement décrit sous forme de constantes.

6. Voir § C.1.3

Annexe D

Réseaux de Petri colorés

Un réseau de Petri coloré (Jensen, 1994) est un tuple *CPN=(P, T, A, N, Col, C, E, I)* tel que :

- P est un ensemble fini de places,
- T est un ensemble fini de transitions,
- A est un ensemble fini d'arcs ;
- N est une fonction qui associe à chaque arc une transition et une place :

$$N \ : \ A \rightarrow P \times T \cup T \times P$$

- Col est un ensemble fini de couleurs et C une fonction qui associe à chaque place une couleur :

$$C \ : \ P \rightarrow Col$$

  C spécifie le type de jeton pouvant se trouver dans une place :

- E est une fonction qui associe à chaque arc une combinaison linéaire de couleurs à ajouter ou à retirer d'une place : elle définit la *capacité* de l'arc ;
- I est le marquage initial du réseau.

Un réseau de Petri interprété et coloré est un tuple *ICPN=(CPN, Guards, Sync, States, Assess)* tel que :

- CPN est un réseau de Petri coloré dont l'ensemble Col contient la couleur Logic (cette couleur correspond au jeton classique utilisé habituellement dans les réseaux de Petri interprétés) ;

- *Guards* est un ensemble de fonctions booléennes ;
- *Sync* est une fonction qui associe à chaque élément de T un élément de *Guards* ;
- *States* est un ensemble de variables booléennes,
- *Assess* est une fonction qui associe à chaque place une variable booléenne :

$$Assess \ : \ P \rightarrow States$$

On appellera les variables de *States* des variables d'états car elles caractérisent les différents états du modèle. Enfin, une variable d'état associée à une place p_i prend la valeur *vrai* quand p_i est marquée par un jeton de type Logic (noté $\bullet$) et la valeur *faux* dans le cas contraire.

Une transition t_j est dite *franchissable* si et seulement si, d'une part le nombre de jetons colorés dans chaque place d'entrée de t_j est supérieur ou égal à la capacité des arcs reliant ces places à t_j, et d'autre part la fonction associée à t_j prend la valeur *vrai*. Lorsque la transition est *tirée*, on retire de chaque place d'entrée de cette transition le nombre et le type (c'est-à-dire la couleur) de jetons définis par la capacité de l'arc qui la relie à la transition, et on ajoute dans chaque place de sortie de cette transition le nombre et le type de jetons définis par les arcs reliant la transition à chacune de ces places.

Prenons un exemple : la transition de la figure D.1 est franchissable car il y a un jeton $< yes >$ dans la place d'entrée et la fonction associée à la transition a pour valeur *vrai*; on retire alors un jeton $< yes >$ de cette place et on ajoute un jeton Logic dans la place de sortie : la variable d'état X prend alors la valeur *vrai*.

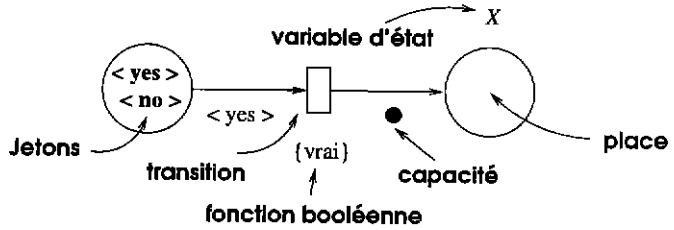


FIG. D.1 – *Réseau de Petri interprété*

Bibliographie

(1993). *Design/CPN Tutorial for X-Windows.* Meta Software Corporation, Cambridge, USA.

Adler, M. R., Davis, A. B., Weihmayer, R., and Worrest, R. W. (1989). Conflict-resolution strategies for nonhierarchical distributed agents. In Gasser, L. and Huhns, M. N., editors, *Distributed Artificial Intelligence*, volume 2, pages 139–161. Pitman Publishing.

Agha, G. (1986). *Actors: a model of concurrent computation for distributed systems.* MIT Press.

Agre, P. E. and Chapman, D. (1987). Pengi: An implementation of a theory of activity. In *AAAI-87*, pages 268–272.

Alquier, N. (1998). *Suivi de situation et de détection des conflits sur un aéroport.* Mémoire de DEA de l'ENSEEIHT.

Azarm, K. and Schmidt, G. (1997). A decentralized approach for the conflict-free motion of multiple mobile robots. *Advanced Robotics*, 11(4):323–340.

Barr, A. and Feigenbaum, E. A. (1982). *The Handbook of Artificial Intelligence*, volume I, III. Pitman.

Barrouil, C., Castel, C., Fabiani, P., Mampey, R., Secchi, P., and Tessier, C. (1998). A perception strategy for a surveillance system. In *ECAI'98.* Brighton, UK.

Berge-Cherfaoui, V. and Vachon, B. (1991). A multi-agent approach of the multi-sensor fusion. In *Fifth International Conference on Advanced Robotics, Robots in Unstructured Environments*, pages 1264–1269. Pisa, Italy.

Bond, A. H. and Gasser, L. (1988). An analysis of problems and research in DAI. In Bond, A. H. and Gasser, L., editors, *Readings in Distributed Artificial Intelligence*, pages 3–35. Morgan Kaufmann Publishers.

Bouron, T. (1992). *Structures de communication et d'organisation pour la coopération dans un univers multi-agent.* Mémoire de thèse de l'Université Paris VI, Paris, France.

Boutilier, C., Friedman, N., and Halpern, J. Y. (1998). Belief revision with unreliable observations. In *AAAI'98*.

Bouzouane, A. (1995). Modélisation multi-agents pour la gestion stratégique d'une délégation d'assurances. *Revue des systèmes de décision*, 4(1):43–61.

Bratman, M. E. (1992). Shared cooperative activity. *The Philosophical Review*, (101):327–341.

Bressolle, M.-C., Pavard, B., Decortis, F., and Salembier, P. (1996). Traitement cognitif et organisationnel des micro-incidents dans le domaine du contrôle aérien : analyse des boucles de régulation formelles et informelles. In Terssac, G. D. and Friedberg, E., editors, *Coopération et conception*. Octarès.

Brooks, R. A. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, RA-2(1):139–159.

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

Cammarata, S., McArthur, D., and Steeb, R. (1983). Strategies of cooperation in distributed problem solving. In *IJCAI'83*, pages 767–770.

Carvallo, M. (1962). *Monographie des treillis et algèbre de Boole*. Gauthier-Villars.

Castel, C., Chaudron, L., and Tessier, C. (1996). 1st order C-Cubes for the interpretation of Petri nets: an application to dynamic scene understanding. In *TAI'96 - 8th Int. Conf. on Tools with AI*, pages 366–373, Toulouse, France.

Chang, C. and Lee, R. (1973). *Symbolic Logic and Mechanical Theorem Proving*. Academic Press, New York.

Chapman, D. (1987). Planning for conjunctive goals. *Artificial Intelligence*, (32):333–377.

Chaudron, L. (1995a). A model of symbolic uncertainty for knowledge fusion in the situation assessment process. In *NATO Symposium on coping with uncertainty in Defence Decision making (unclassified)*, volume 2, pages 15.1–10, The Hague, the Netherlands.

Chaudron, L. (1995b). *Ordres et treillis*. Rapport final ONERA-Cert/DERI, n. 1/360003.

Chaudron, L., Cossart, C., Maille, N., and Tessier, C. (1997a). A purely symbolic model for dynamic scene interpretation. *International Journal on Artificial Intelligence Tools*, 6(4).

Chaudron, L., Fiorino, H., Maille, N., and Tessier, C. (1997b). From Lakatos to lattices: a semantic control of cooperative dialogues. In *IJCAI'97, Workshop on Collaboration, Cooperation and Control in Dialogue Systems*. Nagoya, Japan.

Chaudron, L., Fiorino, H., and Tessier, C. (1997c). Towards a pragmatic theory of multiagent cooperation. In *15th Iasted Int. Conf. on Applied Informatics*, pages 310–313, Innsbruck, Austria.

Chaudron, L., Fiorino, H., and Tessier, C. (1998). Agents humains et artificiels : vers une coopération fructueuse. In *Entretiens Science et Défense*, Paris, France.

Chaudron, L., Mangeret, M., and Poirel, P. (1995). Fustation et Ecosit : une approche de systèmes multiagents frustes pour des traitements complexes. In *3e Journées Francophones sur l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents*, Chambéry, France.

Chaudron, L. and Tessier, C. (1995). SuprA-cooperation: when difference and disagreement are constructive. In *International Workshop on the design of Cooperative Systems (COOP'95)*. Antibes-Juan-les-Pins, France.

Cholvy, L. (1993). Proving theorems in a multi-source environment. In *IJCAI'93*, volume 1, pages 66–71, Chambéry, France.

Cholvy, L. (1998). A general framework for reasoning about contradictory information and some of its applications. In *ECAI'98, Workshop on conflicts among agents*, Brighton, UK.

Cohen, P. R. and Levesque, H. J. (1990). Intention is choice with commitment. *Artificial Intelligence*, (42):213–261.

Conry, S. E., Kuwabara, K., Lesser, V. R., and Meyer, R. A. (1991). Multistage negotiation for distributed constraint satisfaction. *IEEE Transactions on SMC*, 21(6):1462–1477.

Corkill, D. D., Lesser, V. R., and Hudlicka, E. (1982). Unifying data-directed and goal-directed control: an example and experiments. In *AAAI'82*, pages 143 – 147. Los Altos, CA.

Dalal, M. (1989). Investigation into a theory of knowledge base revision: preliminary report. In *IJCAI'89*, pages 475–479.

Dario, P., Ribechini, F., Genovese, V., and Sandini, G. (1991). Instinctive behariors and personalities in societies of cellular robots. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pages 1927–1932. Sacramento, CA.

de Saint-Cyr, F. D. (1996). *Gestion de l'évolutif et de l'incertain en logiques pondérées.* Mémoire de thèse de l'Université Paul Sabatier, Toulouse, France.

Dhalluin, J. F., Gabillard, R., and Koursi, M. E. (1987). Application des réseaux de Petri à la commande - contrôle de processus en sécurité. *APII*, (21):531–551.

Djeffal, S. and Chaib-Draa, B. (1998). Stratégie de changement des croyances basées sur la crédibilité des agents-sources dans un système multi-agent. *Revue d'Intelligence Artificielle*, 12(1):103–123.

Drogoul, A. and Ferber, J. (1992). Multi-agent simulation as a tool for modeling societies: Application to social differentiation in ant colonies. In *4th European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW'92)*. S. Martino al Cimino, Italy.

Durfee, E. H. and Lesser, V. R. (1987). Using partial global plans to coordinate distributed problem solvers. In *IJCAI'87*, pages 875–883. Milan, Italy.

Durfee, E. H. and Lesser, V. R. (1989). Negotiation task decomposition and allocation using partial global planning. In Gasser, L. and Huhns, M. N., editors, *Distributed Artificial Intelligence*, volume 2, pages 229–243. Pitman Publishing.

Durfee, E. H. and Lesser, V. R. (1991). Partial global planning: a coordination framework for distributed hypothesis formation. *IEEE Transactions on System, Man and Cybernetics*, 21(5):1167–1183.

Durfee, E. H., Lesser, V. R., and Corkill, D. D. (1987). Coherent cooperation among communicating problem solvers. *IEEE Transactions on System, Man and Cybernetics*, C-36(11):1275–1291.

Durfee, E. H., Lesser, V. R., and Corkill, D. D. (1989). Trends in cooperative distributed problem solving. *IEEE Transactions on Knowledge and Data Engineering*, 1(1):63–83.

Durfee, E. H. and Montgomery, T. A. (1991). Coordination as distributed search in a hierarchical behavior space. *IEEE Transactions on System, Man and Cybernetics*, 21(6):1363–1378.

Engelmore, R. and Morgan, T. (1988). *Blackboard systems.* Addisson-Wesley.

Ephrati, E. and Rosenschein, J. S. (1994). Divide and conquer in multi-agent planning. In *AAAI'94*, pages 375–380. Seattle, WA, USA.

Erceau, J., Chaudron, L., Ferber, J., and Bouron, T. (1994). *Systèmes coopératifs : de la modélisation à la conception*, chapter Systèmes Personne(s)-Machine(s) : patrimoines cognitifs distribués et mondes multi-agents, coopération et prises de décision collectives. Éditions Octares.

Erman, L. D., Hayes-Roth, F., Lesser, V. R., and Reddy, D. R. (1980). The Hearsay-II speech-understanding system: Integrating knowledge to resolve uncertainty. *Computing Surveys*, 12(2):115–152.

et C. Tessier, L. C. H. F. N. M. (1998). De Lakatos à l'algèbre des cubes : un contrôle sémantique du dialogue coopératif. In *11e Congrès de Reconnaissance des Formes et Intelligence Artificielle*. Clermont-Ferrand, France.

Fabiani, P. (1996). *Représentation dynamique de l'incertain et stratégie de perception pour un système autonome en environnement évolutif*. Mémoire de thèse de l'École Nationale Supérieure de l'Aéronautique et de l'Espace, Toulouse, France.

Fagin, R. and Halpern, J. Y. (1985). Belief, awareness, and limited reasoning: preliminary report. In *IJCAI'85*, pages 491–501.

Favier, M. (1995). Le travail asynchrone et distribué: quelles technologies collaboratives choisir? *Revue des systèmes de décision*, 4(1):63–77.

Ferber, J. (1995). *Les Systèmes multi-agents*. InterEdition.

Ferber, J. and Jacopin, E. (1990). The framework of eco-problem solving. In *Second European Workshop on Modelling Autonomous Agents in a multi-Agent World (MAAMAW'90)*, pages 181–193. Saint Quentin en Yvelines, France.

Fiorino, H. (1998). Principles for cooperative conjecture elaboration in a multiagent context. In *ECAI'98*. Brighton, UK.

Fiorino, H. and Maille, N. (1998). Conflict solving through common conjecture elaboration. In *ECAI'98, Workshop on Conflicts among agents, Brighton, UK*.

Fiorino, H. and Tessier, C. (1994). A functional and behavioural models working together to diagnose failures more accurately. In *14th International Avignon Conference*, pages 301–310. Paris, France.

Fiorino, H. and Tessier, C. (1998a). Agent cooperation: a Petri net based model. In *ICMAS'98*. Paris, France.

Fiorino, H. and Tessier, C. (1998b). Modèle de coopération pour l'élaboration de conjectures communes. In *4e Rencontres des Jeunes Chercheurs en Intelligence Artificielle*, Toulouse, France. Traduction française de ICMAS'98.

Fiorino, H. and Tessier, C. (1998c). Modelling cooperation as a common conjecture elaboration. In *COOP'98, 3rd International conference on the design of cooperative systems*, Cannes, France.

Florin, G. (1996). Structures de contrôle pour les applications coopératives. In *Contrôle Réparti dans les Applications Coopératives (CRAC'96)*. Paris, France.

Gaillet, G. (1996). *Détermination d'éléments pertinents pour la reconnaissance de certains objets dans le cadre du projet* PERCEPTION. Mémoire de DESS de l'Université Paul Sabatier.

Galliers, J. R. (1989). The positive role of conflict in cooperative multi-agent systems. In *First European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW'89)*. Cambridge, England.

Garcia, F. (1993). *Révision des croyances et révision du raisonnement pour la planification*. Mémoire de thèse de l'École Nationale Supérieure de l'Aéronautique et de l'Espace, Toulouse, France.

Gärdenfors, P. (1988). *Knowledge in Flux*. Bradford Books. MIT Press.

Gauvin, D. (1995). Un environnement de programmation orienté agent. In *3e Journées Francophones sur l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents*. St-Baldoph, France.

Genesereth, M. R., Ginsberg, M. L., and Rosenschein, J. S. (1986). Cooperation without communication. In *AAAI'86*, pages 51–57. Philadelphia, PA.

Georgeff, M. (1983). Communication and interaction in multi-agent planning. In *AAAI'83*, pages 125–129.

Ghedira, K. (1993). *Masc: une approche multi-agent des problèmes de satisfaction de contraintes*. Mémoire de thèse de l'École Nationale Supérieure de l'Aéronautique et de l'Espace.

Ghose, A. K. and Goebel, R. (1998). Belief states as default theories: studies in non-prioritized belief change. In *ECAI'98*, pages 8–12.

Grégoire, E. (1988). Discussion on autoepistemic logic. In Smets, P., Mamdani, E. H., Dubois, D., and Prade, H., editors, *Non-Standard Logics for Automated Reasoning*, pages 132–136. Academic Press.

Grosz, B. and Kraus, S. (1993). Collaborative plans for group activities. In *IJCAI'93*, pages 367–373. Chambéry, France.

Gutknecht, O. and Ferber, J. (1998). Un modèle d'analyse, de construction et d'exécution de systèmes multi-agents. In *Journées Francophones sur l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents.*

Halpern, J. Y. and Moses, Y. (1990). Knowledge and common knowledge in a distributed environment. *Journal of the Association for Computing Machinery*, 37(3):549–587.

Hewitt, C. (1977). Viewing control structures as patterns of passing messages. *Artificial Intelligence*, 8(3):323–364.

Hewitt, C. (1986). Offices are open systems. *ACM Transactions on Office Information Systems*, 4(3):271–287.

Jennings, N. R. (1992). Towards a cooperation knowledge level for collaborative problem solving. In *ECAI'92*, pages 224–228.

Jennings, N. R. (1994). *Cooperation in industrial multi-agent systems.* Number 43 in Computer Science. World Scientific.

Jensen, K. (1994). An introduction to the theoretical aspects of coloured Petri nets. In de Bakker, J., de Roever, W., and Rozenberg, G., editors, *A Decade of Concurrency*, volume 803, pages 230–272. Springer-Verlag.

Katsuno, H. and Mendelzon, A. O. (1991a). On the difference between updating a knowledge base and revising it. In *International Conference on Principles of Knowledge Representation and Reasonig*, pages 387–394.

Katsuno, H. and Mendelzon, A. O. (1991b). Propositional knowledge base revision and minimal change. *Artificial Intelligence*, (52):263–294.

Khedro, T. and Genesereth, M. (1994). Modeling multiagent cooperation as distributed constraint satisfaction problem solving. In *ECAI'94*, pages 249–253.

Kleene, S. (1952). *Introduction to Mathematics.* North-Holland, Amsterdam.

Korf, R. E. (1987). Planning as seach: a quantitative approach. *Artificial Intelligence*, (33):65–88.

Kornfeld, W. A. and Hewitt, C. E. (1981). The scientific community metaphor. *IEEE Transactions on System, Man and Cybernetics*, 11(1):24–33.

Lakatos, I. (1984). *Preuves et Réfutations.* Hermann.

Lander, S. E. and Lesser, V. R. (1993). Understanding the role of negotiation in distributed search among heterogenous agents. In *IJCAI'93*, pages 438–444. Chambéry, France.

Lemaitre, M. and Verfaillie, G. (1996). *Systèmes complexes distribués coopératifs.* Rapport technique ONERA, n.3/3552.00/DERI.

Lesser, V. R. and Corkill, D. D. (1981). Functionally accurate, cooperative distributed systems. *IEEE Transactions on Systems, Man and Cyberbetics*, 11(1):81–96.

Levesque, H. J., Cohen, P. R., and Nunes, J. (1990). On acting together. In *AAAI'90*, pages 94–99. Boston, MA.

Luo, R. C. and Kay, M. G. (1989). Multisensor integration and fusion in intelligent systems. *IEEE Transactions on Systems, Man and Cybernetics*, 19(5):901–931.

Maes, P. (1994). Agents that reduce work and information overload. *Communication of the ACM*, 37(7):30–40. Special Issue on Intelligent Agents.

Maille, N. (1997). *Du treillis des termes au treillis des cubes.* Rapport technique ONERA-Cert/DTIM, n. 3/3600.82.

Manna, Z. (1974). *Mathematical Theory of Computation.* MacGraw-Hill, New York.

Maudet, N. and Evrard, F. (1998). Using implicature to avoid conflicts in dialogue. In *ECAI'98, Workshop on conflicts among agents*, Brighton, UK.

Mazur, B. (1997). Conjecture. *Synthese, an International Journal for Epistemology, Methdology and Philosophy of Science*, 11(2):197–210.

Ouzrout, Y., Kabachi, N., and Vincent, L. (1996). Application des systèmes multi-agents à la simulation d'organisation productives. In *Actes de la Journée PRC-GDR IA SMA*, pages 81–92. Toulouse, France.

Perception (1996). Project int. and final reports. Technical Report 1-2/7995.02-3575.00, ONERA-Cert, BP 4025, 31055 Toulouse Cedex 04, France. In French.

Popper, K. (1973). *La logique de la découverte scientifique.* Payot.

Prigogine, I. and Nicolis, G. (1977). *Self-organization in Nonequilibrium Systems.* Wiley-Interscience.

Robert, F. (1996). *Coopération multi-robots par insertion incrémentale de plans.* Mémoire de thèse de l'Institut National Polytechnique de Toulouse, France.

Rosenschein, J. S. (1993). Consenting agents: Negotiation mechanisms for multi-agent systems. In *IJCAI'93*, pages 792–799. Chambéry, France.

Rosenschein, J. S. and Zlotkin, G. (1994). *Rules of encounter: Designing convention for automated negotiation among computers.* Artificial Intelligence. MIT Press.

Rosenschein, S. J. and Kaelbling, L. P. (1995). A situated view of representation and control. *Artificial Intelligence*, (73):149–173.

Sallantin, J., Quinqueton, J., Barboux, C., and Aubert, J. P. (1991). Théorie semi-empirique: éléments de formalisation. *Revue d'Intelligence Artificielle*, 5(1):69–92.

Sathi, A., Morton, T. E., and Roth, S. F. (1986). Callisto: An intelligent project management system. *AI Magazine*, 7(5):34–52.

Seguin, C. (1992). *De l'action à l'intention: vers une caractérisation formelle des agents.* Mémoire de thèse de l'Université Paul Sabatier, Toulouse, France.

Shoham, Y. (1993). Agent-oriented programming. *Artificial Intelligence*, (60):51–92.

Simon, J. C. (1984). *La reconnaissance des formes par algorithmes.* Études et Recherches en Informatique. Masson.

Singh, M. P. (1994). *Multiagent systems: A Theorical Framework for Intentions, Know-How and Communications.* Number 799 in Lecture notes in Artificial Intelligence. Springer-Verlag, Heidelberg, Germany.

Smith, R. G. and Davis, R. (1981). Frameworks for cooperation in distributed problem solving. *IEEE Transactions on Systems, Man and Cyberbetics*, 11(1):61–69.

Steels, L. (1990). *Decentralized AI*, chapter Cooperation between distributed agents through self-organisation. Elsevier Science Publishers.

Stirling, W. C. (1993). Coordinated intelligent control via epistemic utility theory. *IEEE Control Systems*, pages 21–29.

Sycara, K. P. (1989). Multiagent compromise via negotiation. In Gasser, L. and Huhns, M. N., editors, *Distributed Artificial Intelligence*, volume 2, pages 119–137. Pitman Publishing.

Tessier, C. (1997). Reconnaissance de scènes dynamiques à partir de données issues de capteurs : le projet Perception. In *NATO Research and Technology organization meeting on "Multi-sensor and data fusion for telecommunications, remote sensing and radars"*. Lisbonne, Portugal.

Tessier, C. and Chaudron, L. (1996). Constructive difference and disagreement: A suprA-cooperation among agents. *CSCW: The Journal of Collaborative Computing*, (5):323–336.

Thayse, A. and al. (1990). *Approche logique de l'intelligence artificielle*, volume 1 of *Series Informatique*. Dunod.

Theraulaz, G. and Bonabeau, E. (1996). Rôle de l'auto-organisation et de la stigmergie dans l'intelligence collective des insectes bâtisseurs. In *Atelier de cognition partagée PRESCOT : le contrôle dans les systèmes coopératifs*.

van Eijk, R. M. and de Boer, F. S. (1998). Information-passing and belief revision in multi-agent systems. In *5th International Workshop on Agent theories, Architectures and Languages (ATAL'98)*, pages 75–89. Cité des Sciences - La Villette, Paris, France.

Verjus, J. P. (1991). Introduction aux systèmes répartis. In Balter, R., Banâtre, J. P., and Krakowiak, S., editors, *Construction des systèmes d'exploitation répartis*, chapter 1. INRIA.

Volpe, R., Balaram, J., Ohm, T., and Ivlev, R. (1997). Rocky 7: a next generation Mars rover prototype. *Advanced Robotics*, 11(4):341–358.

Waltz, E. L. and Buede, D. M. (1986). Data fusion and decision support for command and control. *IEEE Transactions on Systems, Man and Cybernetics*, 16(6):865–879.

Werner, E. (1989). Cooperating agents: A unified theory of communication and social structure. In Gasser, L. and Huhns, M. N., editors, *Distributed Artificial Intelligence*, volume 2, pages 3–36. Pitman Publishing.

Wittig, T., Jennings, N. R., and Mamdani, E. H. (1994). Archon: Framework for intelligent cooperation. *Intelligent Systems Engineering*, 3(3):168–179.

Wooldridge, M. (1994). Coherent social action. In *ECAI'94*, pages 279–283.

Wooldridge, M. and Jennings, N. R. (1994). Towards a theory of cooperative problem solving. In *6th European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW'94)*. Odense, Denmark.

Wooldridge, M. and Jennings, N. R. (1995). Intelligent agents: theory and practice. *The Knowledge Engineering Review*, 10(2):115–152.

Zeghal, K. and Ferber, J. (1993). A reactive approach for distributed air traffic control. In *Proceedings of Avignon'93, 13th International Conference*, pages 381–390. Avignon, France.

Zhou, M. C. and Dicesare, F. (1989). Adaptative design of Petri net controllers for error recovery in automated manufacturing systems. *IEEE Transactions on System, Man and Cybernetics*, 19(5).

Zlotkin, G. and Rosenschein, J. S. (1993). A domain theory for task oriented negotiation. In *IJCAI'93*, pages 416–422. Chambéry, France.

Résumé

Cette thèse se situe dans le champ de l'intelligence artificielle distribuée et des systèmes multiagents, domaine en plein essor, tant sur le plan théorique que sur celui des applications. Le thème général de cette recherche est l'élaboration, par un ensemble de systèmes autonomes coopérants (par exemple des gnoptères, des engins sous-marins sans pilote, des robots) – désignés par le terme générique d'«agents» – d'une représentation commune à partir des conjectures qu'ils ont élaborées. Ce thème est central dans de nombreuses activités pratiques telles que la surveillance multicapteur, la robotique d'intervention, la gestion d'équipes d'engins non habités.

Ces agents sont hétérogènes et ne perçoivent pas nécessairement les mêmes informations. De plus, à cause de leur perception locale, leurs connaissances sont incertaines, mutuellement contradictoires, voire erronées : nous les qualifions de *croyances*. L'élaboration d'un point de vue commun ne consiste donc pas simplement à juxtaposer des points de vue locaux : les agents doivent résoudre des conflits portant sur leurs croyances afin d'aboutir à ce résultat commun.

Dans ce contexte, nous développons notre contribution selon les points suivants : nous proposons un protocole générique de coopération pour des agents devant *élaborer une conjecture commune*. Cette étude constitue un travail de synthèse et de formalisation *a posteriori* de méthodes de coopération effectives ; nous formalisons la notion de *conflits*, puis nous introduisons la notion de *croyance commune* comme une conjecture caractérisant le point de vue commun d'un groupe d'agents et permettant une prise de décision collective et *cohérente*.

Mots-clés : Intelligence Artificielle Distribuée, systèmes multiagents, coopération de systèmes autonomes, croyance commune, conflits, reconnaissance de situation.

Abstract

This thesis is within the field of distributed artificial intelligence and multiagent systems, which is expanding both from the theoretical and the applicative points of view. The main topic of this research is the elaboration of a common representation, which is achieved by a set of cooperative autonomous systems or "agents" (e. g. gnopters, autonomous underwater vehicles, robots) that gather their locally inferred conjectures. This is a topic of great concern for applications such as multisensor surveillance systems, robotics, unhabited vehicle teams.

The agents that are considered are heterogeneous and do not necessarily get the same information. Furthermore, because of their local perception, their knowledge is uncertain, mutually contradictory and even erroneous: it is called a *belief*. Consequently, the elaboration of a commun point of view is not the mere juxtaposition of the local points of view: the agents have to cope with their conflicting beliefs in order to set out a common result.

Our contribution to these issues is the following: a generic cooperation protocol for agents that have to elaborate a common conjecture is proposed. This protocol aims to synthesize and formalize effective and practical cooperation methods; the *conflict* concept is formalized and the notion of *common belief* intended as a conjecture characterizing the point of view of a group of agents, and allowing a collective and *coherent* decision process to be built, is then introduced.

Keywords : Distributed Artificial Intelligence, multiagent systems, cooperating autonomous systems, common belief, conflicts, situation recognition.

ÉCOLE NATIONALE SUPÉRIEURE DE L'AÉRONAUTIQUE ET DE L'ESPACE

10, avenue Édouard Belin - B.P. 4032 - 31055 TOULOUSE CEDEX 4

Téléphone : 05 62 17 80 80 - Télécopie : 05 62 17 83 30