

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR  
ET DE LA RECHERCHE SCIENTIFIQUE  
UNIVERSITE MENTOURI CONSTANTINE  
FACULTE DES SCIENCES DE L'INGENIEUR  
DEPARTEMENT D'INFORMATIQUE**

**No d'Ordre :**

**No de Série :**

**Thèse De Doctorat en Sciences en Informatique  
Spécialité : Informatique**

**Intitulé :**

# **LE RAISONNEMENT QUALITATIF DANS LES SYSTEMES MULTI AGENTS**

**Présentée par: GUERRAM TAHAR**

**Thèse soutenue le ..... Devant le jury composé de :**

**Président: Boufaida Mahmoud, Professeur (Université de Constantine)**

**Rapporteur: Ramdane Maamri Maître de Conférences (Université de Constantine)**

**Examineurs : Zarour Naceredine, Professeur (Université de Constantine)**

**Kazar Okba, Maître de Conférences (Université de Biskra)**

**Mokhati Farid Maître de Conférences, (Université d'Oum El Bouaghi)**

**Directeur de Thèse : Ramdane Maamri Maître de Conférences (Univ. de Constantine)**

*Thèse Préparée au sein du Laboratoire LIRE – U.M.C*

## RESUME

Le raisonnement est une activité mentale que les êtres humains exercent d'une manière continue afin de résoudre les problèmes qu'ils rencontrent dans leur vie quotidienne, ce raisonnement est souvent effectué (par les êtres humains) en l'absence d'informations quantitatives. Ce type de raisonnement s'appelle le raisonnement qualitatif. Kenneth de Forbus, le père de la Théorie des Processus Qualitatifs [Forbus 1984], le définit comme suit : « Le raisonnement qualitatif est la branche de L'IA qui crée de représentations des aspects du monde tels que l'espace, le temps et la quantité, permettant de raisonner avec très peu d'informations ».

Le raisonnement qualitatif est né avec les travaux de Hayes [Hayes 1979] sur la physique « naïve » tentant à modéliser les connaissances de sens commun chez l'être humain afin de les utiliser pour raisonner sur les objets physiques. Depuis, des efforts conjugués par des chercheurs en intelligence artificielle, en mathématiques en psychologie cognitive ont permis de développer cette branche, qui est actuellement en plein essor et évolution.

Actuellement des simulateurs qualitatifs [Kuipers 1986] [Bredeweg et al. 2009] à vocation recherche existent et permettent de modéliser et simuler les systèmes écologiques et biologiques dans le but de prédire leurs comportements typiques.

Dans le paradigme agent, un ensemble d'agents en interaction permet de modéliser et de simuler des systèmes tels les colonies de fourmis et la propagation de l'épidémie afin de prédire leurs comportements [Drougol 1993] [Carley et al. 2006]..

L'objectif de ce travail consiste à étudier les différentes approches de raisonnement qualitatif afin d'explorer toutes les possibilités conduisant à conférer à l'agent l'aptitude de simuler le raisonnement humain en raisonnant qualitativement en l'absence d'informations complètes.

L'acquisition de cette faculté cognitive par l'agent permettra d'enrichir ses capacités de raisonnement et confère au paradigme agent une meilleure appréhension des systèmes complexes.

Nous avons montré dans le cadre de cette thèse, que la relation technologie agent – technologie raisonnement qualitatif est mutuellement bénéfique à travers deux contributions :

Ma première contribution intitulée “ARDGIAL : an Argumentation Based Dialogue in Multi agent Systems” ( publié dans International Review on Computers and Software, vol. 4 No 3, May 2009, pp 392-401, ISSN : 1828-6003) se veut un exemple d'utilisation du raisonnement qualitatif dans les systèmes multi agents. Ce travail propose un protocole de dialogue de demande d'action basé argumentation entre deux agents cognitifs qui intègre un processus de persuasion.

Ma deuxième contribution intitulée “A Multi agent Based Architecture to Qualitative Temporal Constraint Reasoning “ (Proceedings of the European Computing Conference 2010 (ECC’10), Bucharest, Romania, pp : 249-259, ISBN : 1790-5117; ISSN : 978-960-474-178-) est un exemple d’utilisation de l’approche multi agents dans le raisonnement qualitatif. L’objectif de ce travail consiste à proposer une approche multi agents pour le raisonnement qualitatif temporel.

En outre, deux autres contributions ayant trait à la modélisation et l’inférence qualitatives ont fait l’objet de deux articles acceptés qui paraîtront dans les actes de la Conférence Internationale Arabe des Technologies de l’Information 2010 (ACIT 2010) qui se tiendra en Libye du 14-16 Décembre 2010.

---

**Mots Clés :** Agent, système multi agents, raisonnement qualitatif.

## SUMMARY

Reasoning is a mental activity that human beings exert in a continuous way in order to solve problems which they encounter in their everyday life, this reasoning is often carried out (by the human beings) in the absence of quantitative information. This type of reasoning is called Qualitative Reasoning. Kenneth de Forbus, the father of Qualitative Process Theory [Forbus 1984] defines it as follows:

“Qualitative Reasoning is the area of A.I which creates representations aspects of the world such as space, time and quantity, which supports reasoning with very little information”

Qualitative reasoning was born with Hayes’s work [Hayes 1979] on "naive" physics trying to model human being’s common sense knowledge so using it to reason on physical objects. Since, efforts combined by researchers in artificial intelligence, in mathematics in cognitive psychology made it possible to develop this area, which is currently in full rise and evolution.

Currently qualitative simulators [Kuipers 1986] [Bredeweg et al. 2009] in research area exist and can model and simulate ecological and biological systems aiming to predicting their typical behaviors.

In agent paradigm, an interacting set of agents make it possible to model ant colonies and virus propagation in order to predict their behaviors [Drougol 1993] [Carley, et al 2006].

The objective of this Thesis consists of studying qualitative reasoning approaches in order to explore all the possibilities aiming to confer to the agent the aptitude to simulate human being’s reasoning by reasoning qualitatively in absence of complete information.

The acquisition of this cognitive faculty by agents will make it possible to enrich their reasoning capacities and confers to the paradigm agent a better apprehension of complex systems.

We showed within the work of this thesis, that the relation agent technology – qualitative reasoning technology is mutually beneficial through two contributions:

My first contribution entitled “ARDGIAL” : an Argumentation Based Dialogue in Multi agent Systems” (published in International Review on Computers and Software, vol. 4 No 3, May 2009, pp 392-401, ISSN : 1828-6003) wants to be an example of use of qualitative reasoning in multi agent systems. This work proposes an argumentation based request for action dialogue between two cognitive agents which integrates a persuasion process.

My second contribution entitled “A Multi agent Based Architecture to Qualitative Temporal Constraint Reasoning “ (In the Proceedings of the European Computing Conference

2010 (ECC'10), Bucharest, Romania, pp : 249-259, ISBN : 1790-5117; ISSN : 978-960-474-178-6. ) is an example of use of multi agent approach in qualitative reasoning. This work proposes a multi agent based approach to qualitative temporal reasoning.

Moreover, two other papers relating to qualitative modeling and inference were accepted and will appear in the proceedings of the Arab International Conference on Information Technologies 2010 (ACIT 2010) which will be held in Libya on December 14-16th, 2010.

---

**Key words:** Agent, multi agent system, qualitative reasoning

## ملخص

التفكير هو نشاط ذهني يقوم به الإنسان بشكل غير منقطع و عفوي من أجل حل مشاكله التي تعترضه خلال حياته اليومية. هذا النشاط الذهني الذي يمارسه الإنسان غالبا في غياب معلومات رقمية دقيقة يطلق عليه التفكير الكيفي . كينيت دو فرييس صاحب نظرية السيرورات الكيفية يعرفه كمايلي : [Forbus 1984] :

التفكير الكيفي هي الشعبة من الذكاء الاصطناعي التي نهتم بأنشاء تمثيلات لمظاهر الكون كالزمان و المكان و الكميات و الني تسمح بالتفكير باستعمال معلومات قليلة و غير كاملة .

نشأ التفكير الكيفي مع الأبحاث العلمية لهائيس في ميدان الفيزياء "البسيطة" [Hayes 1984] من أجل نمذجة المعارف القاعدية عند الإنسان لأستخدامها في التفكير عن الأشياء الفيزيائية. منذ ذلك الوقت أثمرت الأبحاث العلمية في ميادين الرياضيات و الذكاء الاصطناعي و علم النفس المعرفي الى تطوره و ازدهاره.

يوجد في الوقت الحاضر محاكيات برمجية كيفية موجهة للبحث العلمي [Kuipers 1986] [Bredeweg et al. 2009] و تسمح بمحاكاة و نمذجة الأنظمة البيئية و البيولوجية. لغرض معرفة سلوكياتها. [Carley, et al 2006] [Drougol 1993].

الغرض المرجو من خلال هذه الرسالة هو دراسة مختلف طرق التفكير الكيفي كخطوة أولى تحضر لخطوة ثانية من شأنها أن تستكشف إمكانيات تخول للعميل محاكاة أساليب التفكير الكيفي عند الكائن الإنساني و ذلك بمنحه امكانية التفكير في غياب المعلومات الكاملة . إن اكتساب هذه الكفاءة من طرف العميل يسمح له بإثراء قدراته الذهنية كما تسمح لتكنولوجيا العميل بالتكفل أحسن بدراسة الأنظمة المعقدة.

لقد بينا في هذه الرسالة بأن العلاقة الموجودة بين تكنولوجيا العميل و تكنولوجيا التفكير الكيفي هي علاقة نفع متبادل من خلال إسهامين:

الإسهام الأول مقالة علمية منشورة في مجلة مختصة ( أنظر لتفاصيلها أسفل) تبين استخدام المحاجة كأسلوب للتفكير الكيفي من طرف نظام متعدد العملاء. هذه المقالة تقترح بروتوكول حوار لطلب القيام بانجاز مهمة بين عميلين.

“ARDGIAL : an Argumentation Based Dialogue in Multi agent Systems”

International Review on Computers and Software, vol. 4 No 3, May 2009, pp 392-401, ISSN: 1828-6003.

الإسهام الثاني مقالة علمية منشورة في كتاب الندوة الأوروبية للحساب الألي لسنة 2010 وتبين ايجابية استخدام نظام تعدد العملاء في التفكير الكيفي الزمني :

“A Multi agent Based Architecture to Qualitative Temporal Constraint Reasoning “

Proceedings of the European Computing Conference 2010 (ECC'10), Bucharest, Romania, pp : 249-259, ISBN : 1790-5117; ISSN : 978-960-474-178-6.

بالإضافة الى ذلك فقد تم تقديم مقالتين للندوة الدولية العربية لتكنولوجيا المعلومات 2010 و التي ستجرى فعاليتها بالجمهورية الليبية 14-16 ديسمبر القادم وقد حضيتا بالقبول و ينتظر نشرهما لاحقا.

الكلمات المفاتيح: العميل ، نظام متعدد العملاء، التفكير الكيفي.

## **REMERCIEMENTS**

Je tiens à remercier tout particulièrement les personnes ayant permis l'aboutissement de mes travaux de thèse :

Tout d'abord, je tiens à remercier le Prof. Boufaïda Mahmoud pour avoir accepté de présider ma soutenance.

Je remercie vivement Prof. Zarour Naceredine pour avoir accepté de juger mes travaux.

Je remercie vivement Dr. Kazar Okba pour avoir accepté de juger mes travaux.

Je remercie vivement Dr. Mokhati Farid pour avoir accepté de juger mes travaux.

Je remercie également et vivement Dr Ramdane Mamri et Pr Sahnoun Zaidi et je leur exprime ici ma profonde gratitude pour leur confiance en moi, leur aide considérable et leurs orientations enrichissantes.

## **DEDICACES**

A la mémoire de mon père, qu'Allah l'accueille dans son vaste paradis.

A ma chère maman

A ma chère épouse

Et à mes chers enfants Walid, Salah Eddine et Aya.

## SOMMAIRE

<b>Introduction Générale.....</b>	<b>15</b>
- Contexte et problématique .....	15
- Travail demandé .....	15
- Contributions .....	15
- Organisation de la thèse .....	16
<b>Chapitre 1 : Le raisonnement qualitatif.....</b>	<b>18</b>
1.1 Introduction .....	18
1.2 Les différentes approches de raisonnement qualitatif .....	19
1.2.1 Les approches fondées sur la physique qualitative .....	19
1.2.1.1 L'approche centrée processus .....	19
1.2.1.2 L'approche centrée composant .....	20
1.2.1.3 L'approche centrée contrainte .....	21
1.2.1.4 Comparaison des approches fondées sur la physique qualitative.....	22
1.2.2 Les approches logiques .....	23
1.2.2.1 Logique de défaut .....	23
1.2.2.2 Logique modale .....	23
1.2.2.3 Logique auto-épistémique .....	24
1.2.2.4 L'argumentation .....	24
- Introduction .....	24
- L'argumentation, une approche pour le raisonnement réfutable .....	25
- Système de Simari & Loui .....	25
- Le système de Dung .....	27
- Le système de de Hage et Veheij .....	30
- Le système de Bondarenko et al.....	33
- Le système d'Amgoud & Cayrol .....	35
- Comparaison des systèmes logiques d'argumentation.....	36
1.2.3 Les approches algébriques .....	37
1.2.3.1 Le raisonnement qualitatif temporel .....	37
- L'algèbre des intervalles .....	37
- L'algèbre des points .....	37
1.2.3.2 Le raisonnement qualitatif spatial .....	38
- Méreologie .....	39
- Topologie .....	39
- Mére topologie .....	40
1.2.3.3 Le raisonnement qualitatif causal.....	42
- Définition d'une carte causale .....	42
- Historique des cartes causales .....	42
- Applications des cartes cognitives .....	43
- Inférence d'une carte causale .....	43
1.3 Avantages et les limites du raisonnement qualitatif .....	44
1.4 Les domaines d'applications du raisonnement qualitatif .....	44
1.5 Les tendances actuelles de recherche en raisonnement qualitatif .....	44
1.6 Conclusion .....	45

<b>Chapitre 2 : Les systèmes complexes .....</b>	<b>47</b>
2.1 Introduction .....	47
2.2 Présentation et définition des systèmes complexes .....	47
2.3 Modélisation des systèmes complexes .....	49
2.3.1 Approche basée sur les automates cellulaires .....	49
2.3.2 Approche basée agents .....	50
2.3.3 Approche basée sur le raisonnement qualitatif .....	53
2.4 Conclusion .....	57
<b>Chapitre 3 : Les systèmes multi agents .....</b>	<b>58</b>
3.1 Introduction .....	58
3.2 Les Agents .....	59
3.2.1 Structure conceptuelle des agents .....	60
3.2.1.1 Les agents avec états .....	60
3.2.1.2 Les agents avec buts .....	60
3.2.1.3 Agents avec utilité .....	61
3.2.2 Les agents réactifs .....	62
3.2.3 Les agents cognitifs ou délibératifs .....	63
3.2.4 Les agents hybrides .....	64
3.3 Les systèmes multi agents .....	66
3.3.1 Définitions des Systèmes Multi-Agents .....	66
3.3.2 Les avantages des SMA .....	67
3.3.3 Types d'environnement .....	68
3.4. L'interaction dans les systèmes multi agents .....	69
3.4.1 Définitions .....	69
3.4.2 Situations d'interactions .....	69
3.4.3 Coordination d'actions .....	71
3.4.4 La négociation .....	72
3.4.4.1 Définitions .....	72
3.4.4.2 Types de négociation et organisation des Systèmes multi agents .....	74
3.4.4.3 Différentes méthodes de résolution de conflits .....	76
- Les méthodes centralisées .....	76
- Les méthodes décentralisées .....	79
- La méthode heuristique .....	81
- La négociation basée argumentation A.N.A .....	81
3.5. Méthodologies de conception des systèmes multi-agents .....	81
3.6 Les plateformes S.M.A .....	83
3.7 Conclusion .....	90
<b>Chapitre 4 : Le raisonnement qualitatif et les systèmes multi agents .....</b>	<b>91</b>
4.1 Introduction .....	91
4.2 Apports du raisonnement qualitatif au paradigme agent .....	92
4.2.1 L'argumentation dans le raisonnement des agents autonomes .....	92
4.2.1.1 L'argumentation pour la révision des croyances .....	92
4.2.2 L'argumentation pour la planification et la délibération .....	92
4.2.3 L'argumentation pour la communication entre agents .....	93
4.2.3.1 jeux de dialogues .....	94
4.2.3.2 plans d'argumentation .....	94
4.2.3.3 L'argumentation pour la négociation entre agents .....	94

4.3 Apports du paradigme agent au raisonnement qualitatif .....	98
4.4 Conclusion .....	98
<b>Chapitre 5 : Contributions .....</b>	<b>101</b>
5.1 Introduction .....	101
5.2. Première Contribution : Un protocole de dialogue entre agents .....	...
basé argumentation .....	102
5.2.1 Problématique .....	102
5.2.2 Travaux similaires .....	102
5.2.3 Description du protocole proposé ARGDIAL .....	103
5.2.4 Implémentation .....	104
5.2.5 Conclusion et travaux futurs.....	105
5.3 Deuxième contribution : Une architecture multi agents pour le raisonnement.....	107
qualitatif temporel.....	107
5.3.1 Problématique .....	108
5.3.2 Travaux similaires .....	108
5.3.3 L'approche proposée .....	109
5.3.4 Conclusion et futurs travaux .....	112
5.4 Troisième contribution : Un algorithme pour le raisonnement qualitatif causal ...	113
5.4.1 Problématique .....	114
5.4.2 Principe de l'algorithme.....	114
5.4.3 Description détaillée de l'algorithme.....	114
5.4.4 Implémentation .....	114
5.4.5 Conclusion et futurs travaux .....	118
<b>Conclusion Générale et Perspectives.....</b>	<b>119</b>
<b>Références bibliographiques.....</b>	<b>123</b>
<b>Annexe : Quelques détails d'implémentation.....</b>	<b>136</b>
1. Présentation de l'outil d'édition et de manipulation des cartes cognitives.....	136
2. Spécification en langage Maude du protocole de dialogue inter agents .....	140

**LEXIQUE :**

QSIM .....	Qualitative Simulation.
QPT.....	Qualitative Process Theory
QPE.....	Qualitative Process Engine
RCTQ .....	Réseau de Contraintes Temporelles Qualitatives
RCSQ.....	Réseau de Contraintes Spatiales Qualitatives
QAC.....	Qualitative Arc Consistency
QPC.....	Qualitative Path Consistency
RCC8 .....	Région Connections Calculus – 8 -
ARGDIAL .....	ARGumentation based DIALogue

**LISTE DES FIGURES :**

Figure 1.1 : Description des systèmes dans l'approche centrée contrainte .....	21
Figure 1.2 : Les Treize relations d'Allen.....	38
Figure 1.3. Un R.C.Q.T basé intervalles.....	38
Figure 1.4 : un R.C.T.Q basé points.....	38
Figure 1.5 : Le R.C.C-8 .....	41
Figure 1.6 Réseau de contraintes spatiales qualitatives .....	41
Figure 2.1 La simulation qualitative par GARP3 .....	56
Figure 3.1 : Différence entre objets et agents.....	59
Figure 3.2: Architecture B.D.I .....	64
Figure 3.3 : Cycle Perception / Délibération/ Action d'un agent cognitif.....	64
Figure 3.4 : Architecture InteRRap.....	66
Figure 3.5 : Représentation d'un agent en interaction avec son environnement Et les autres agents.....	67
Figure 3.6: Les différentes catégories de négociation et leurs structures.....	74
Figure 3.7 : Les différents types de négociation .....	75
Figure 3.8: Graphe de transition des états de la négociation entre un manager et un contractant dans une extension du CNP.....	78
Figure 3.9 L'espace de négociation.....	80
Figure 3.10: Facette AEIO au sein d'un agent.....	84
Figure 3.11: Structure générale de Swarm .....	88
Figure 5.1 : Machine d'états finis du protocole ARGDIAL.....	104
Figure 5.2 : Architecture conceptuelle interne d'un agent argumentatif .....	105
Figure : 5.3 : une partie du code MAUDE développé.....	106
Figure 5.4 : une configuration initiale.....	106
Figure 5.5 : une configuration initiale.....	106
Figure 5.6: Résultats des réécritures limitées des configurations initiales.....	107
Figure 5.7 : Architecture du système multi – agents .....	110
Figure 5.8 : Données en entrée structurées.....	111
Figure 5.9 : La matrice d'adjacence.....	111
Figure 5.10 : L'algorithme Arc Consistency de Mackworth.....	112
Figure 5.11 : Algorithme QPC.....	112
Figure 5.12 : L'algorithme Consistency .....	113
Figure 5.13 : Description détaillée de l'algorithme .....	115
Figure 5.14 : Carte cognitive « infection virale » .....	116
Figure 5.15 : Un scenario d'exécution de l'algorithme proposé.....	117
Figure 5.16 : Détail du scenario d'exécution .....	117
Figure 5.17 : Détail du vecteur d'état E4.....	117
Figure a.1 : Capture d'écran pour la catégorie « Accueil ».....	136
Figure a.2 : Capture d'écran pour la catégorie « Création ».....	136
Figure a.3 : L'architecture de l'outil logiciel développé.....	137
Figure a.4 : Capture d'écran pour la catégorie « Optimisation ».....	138
Figure a.5 : Capture d'écran pour la catégorie « Intégration ».....	138
Figure a.6 : Capture d'écran pour la catégorie « Analyse et Inférence ».....	139
Figure a.7 : Capture d'écran pour la fenêtre « Scénario Possibles ».....	139
Figure a.8 : Capture d'écran pour la fenêtre « Confluences ».....	140

**LISTE DES TABLEAUX**

Tableau 1.1 : Comparaison des approches qualitatives fondées sur la physique Qualitative.....	22
Tableau 1.2 : Comparaison des systèmes logiques d'argumentation.....	36
Tableau 2.1 Exemples de systèmes complexes.....	48
Tableau 3.1 Situations d'interactions entre agents .....	71
Tableau 3.2 : Les différents environnements de programmation.....	85
Tableau 3.3 : Liste de plates-formes disponibles et utilisables.....	87
Tableau 5.1 : Locutions communicatives utilisées dans ARGDIAL.....	104

## INTRODUCTION GENERALE

### - Contexte et problématique

Dans le cadre des recherches en intelligence artificielle, le raisonnement est un thème très étudié qui joue un rôle très important dans la mise en œuvre des outils et méthodes servant à mieux simuler l'intelligence humaine. Dans sa vie quotidienne, l'homme utilise ses facultés cognitives de base connues sous le nom de « connaissances de sens commun » pour accomplir ses tâches, pour expliquer les phénomènes physiques et pour se protéger contre les méfaits et les dangers de la nature. Toutes ces activités sont réalisées par l'être humain sans avoir besoin d'informations numériques détaillées. La conduite par une personne d'un véhicule dans une cité à trafic routier important est un exemple type de cette faculté cognitive par laquelle est doté l'être humain. Par ailleurs, dans la mise en œuvre d'applications informatiques issues du monde réel, souvent de type continues, dynamiques et complexes, on ne dispose pas d'informations quantitatives précises décrivant ces systèmes. L'ensemble des techniques informatiques permettant de simuler et d'expliquer le comportement des systèmes, physiques ou autres, sans le recours à des connaissances détaillées au sujet de ces systèmes sont désignées par l'appellation « raisonnement qualitatif ».

Les systèmes multi agents sont utilisés pour modéliser les systèmes complexes par le biais de la distribution des données et du raisonnement. Dans ce paradigme multi agents, l'entité de base qui est l'agent est dotée de mécanismes de raisonnement tant sur le plan individuel (délibération, prise de décision) que sur le plan collectif (négociation, coordination). Il en résulte que si on voudrait élargir le champ d'application des systèmes multi agents, il faudrait reconsidérer ces mécanismes de raisonnement intra-agents et inter-agents afin de mieux simuler le raisonnement qualitatif chez l'être humain.

### - Travail demandé

Dans le cadre de cette thèse, notre travail consiste en premier lieu à l'étude de différentes approches de raisonnement qualitatif qui existent actuellement dans la littérature. Le deuxième volet de notre travail consiste à utiliser les résultats de cette étude pour proposer un modèle d'agent et/ ou d'une architecture de systèmes multi agents supportant ce type de raisonnement.

### - Contributions

Après étude de l'état de l'art relatif au raisonnement qualitatif, nous avons pu distinguer trois classes différentes d'approches de raisonnement qualitatif. L'approche basée sur la physique qualitative, l'approche basée sur la logique non monotone et l'approche basée sur les algèbres qualitatives. Dans le cadre de la deuxième classe d'approches, nous avons proposé un protocole de dialogue entre agents cognitifs basé sur l'argumentation qui est un style de

raisonnement non monotone. Ce protocole de dialogue, nous l'avons spécifié et validé par le langage de programmation formel basé sur la logique de réécriture -Maude -.

Dans le cadre de la troisième classe d'approches, nous avons proposé une architecture multi agents pour le raisonnement qualitatif sur des contraintes temporelles décrites dans le langage d'Allen. Ces dernières sont représentées par un graphe temporel, appelé réseau de contraintes temporelles. La résolution de ce problème de satisfaction de contraintes, qui est de complexité NP HARD, est réalisée d'une manière collaborative par un groupe d'agents de l'architecture proposée dans le but d'obtenir les différents scénarios possibles. Nous avons montré dans le cadre de cette contribution que contrairement aux autres approches visant à réduire l'expressivité du langage d'Allen, l'approche multi agents que nous avons proposée permet d'atténuer la complexité de ce problème de résolution de contraintes qualitatives temporelles sans pour autant diminuer l'expressivité du dit langage.

La troisième contribution s'inscrit également dans le cadre de la troisième classe d'approches, plus précisément le raisonnement causal. Ici, nous avons proposé un algorithme d'inférence pour le raisonnement qualitatif causal sur les cartes cognitives. Pour valider notre proposition, nous avons choisi un domaine d'application qui est l'infection virale. Avec l'aide d'experts virologues et des cadres de la santé publique de la wilaya d'Oum El Bouaghi, nous avons pu modéliser ce processus biologique. Pour la simulation de l'infection virale et le test de notre algorithme d'inférence, nous avons développé un outil pour la construction et la manipulation des cartes cognitives.

## **- Organisation de la Thèse**

Cette thèse est organisée de la manière suivante. Le premier chapitre intitulé raisonnement qualitatif permet de faire une étude détaillée des approches de raisonnement qualitatives qui existent dans la littérature.

Le deuxième chapitre donnera un aperçu général sur les systèmes complexes. Nous donnons dans ce chapitre les caractéristiques de ce type de systèmes ainsi que leurs méthodes de conception et de mise en œuvre.

Le troisième chapitre sera consacré à l'étude des agents et des systèmes multi agents où nous mettrons essentiellement l'accent sur les différentes architectures d'agents et leurs mécanismes de raisonnement. Autres éléments d'informations telles les méthodologies de conceptions des systèmes multi agents et les plate - formes pour la mise en œuvre de ces systèmes y seront aussi présentées.

Le quatrième chapitre sera consacré à l'étude des travaux de recherche ayant trait au raisonnement qualitatif et les systèmes multi agents. L'objectif de ce chapitre est d'étudier l'apport de chacune des deux technologies par rapport à l'autre.

Le cinquième chapitre, sera dévolu à la présentation de mes contributions. La première contribution consiste à un dialogue inter - agents basé sur l'argumentation. La deuxième contribution est la modélisation d'un problème de contraintes qualitatives temporelles par le biais d'un système multi agents. Ma troisième et dernière contribution est la proposition d'un algorithme d'inférence qualitative causale sur les cartes cognitives.

Finalement une conclusion générale me permettra de conclure ce travail de thèse et d'en donner des perspectives.

# CHAPITRE I : LE RAISONNEMENT QUALITATIF

## 1.1 Introduction

Le raisonnement qualitatif est une branche de l'IA visant à prédire et à expliquer le comportement d'un système en l'absence d'informations numériques complètes caractérisant ce système. Les premiers travaux dans le domaine remontent aux années soixantes avec les travaux en économie de H.A Simon [Simon 60] qui a montré la possibilité d'inférer des conclusions pertinentes à partir de connaissances purement qualitatives. Le premier produit à base de connaissances utilisant le raisonnement qualitatif s'appelle Netwon [de Kleer 77] et a été dédié à la mécanique physique. Les travaux de P. J Hayes [Hayes 1979][Hayes 1985] sur l'identification et l'axiomatisation des connaissances humaines de sens commun ayant trait aux objets physiques, connus aussi sous l'appellation de « la physique naïve », a tenté de proposer des modélisations « naïves » du monde physique mais ayant un caractère prédictif. Les travaux de Hayes ont eu un impact considérable sur l'avancée des recherches en raisonnement qualitatif car les concepts qu'il a avancés tels « l'espace d'une quantité » et « l'histoire d'une variable » ont été repris en détail par K.D Forbus dans sa théorie des processus qualitatifs (Q.P.T) [Forbus 1984] basée sur le concept que tout changement n'est que le résultat de l'interaction d'un nombre fini de processus. Peu après, Benjamin Kuipers proposa son approche appelée la simulation qualitative (QSIM) [Kuipers 1986] qui modélise un système physique par un ensemble d'équations différentielles qualitatives et un ensemble de contraintes qualitatives. Par la donnée d'un état initial, la résolution de ces équations différentielles permet d'obtenir les comportements du système modélisé qui vérifient les contraintes imposés. Les recherches sur le domaine se sont par la suite multipliées visant à affiner les approches existantes [Haton et al 1991], parmi ces travaux on peut citer [Kuipers 87][Dormoy 87][Bredeweg et Wielinga 88a][Bredeweg et Wielinga 88b][Kuipers94].

Une autre classe d'approches basées sur la logique ont été proposées telle la logique des défauts [Reiter 80], logique modale non monotone [Mc Dermot and Doyale 1980], logique auto-épistémique [Moore 85] et les modèles logiques d'argumentation [Simari & Loui 1992] [Dung 1993] [Hage & Verheij 1995] [Bondarenko et al 1997] [Amgoud and Cayrol 2002][Amgoud and Prade 2005].

Une troisième classe d'approches qualitatives basées sur les algèbres qualitatives qui sont le raisonnement qualitatif temporel [Allen 1983][Vilain and Kautz 1986], le raisonnement qualitatif spatial et le raisonnement qualitatif causal [Axelrod 1976] [Kosko 1986] [Florentin 2003].

Actuellement le raisonnement qualitatif a une communauté de chercheurs très large à travers le monde (à titre d'exemple : Le groupe d'excellence européen de recherche en raisonnement qualitatif et basé modèles MONET [MONET] et le groupe de recherche en raisonnement qualitatif QRG à l'université de Texas aux états unis [QRG]) travaillant sur des

projets trouvant leurs applications dans des secteurs divers tels le secteur industriel, le secteur socio-économique, le secteur écologique. et le secteur biologique.

Nous allons consacrer ce chapitre à l'étude détaillée et la comparaison de toutes ces approches. A la fin, Nous clôturons ce chapitre par une conclusion.

## **1.2 Les différentes approches de raisonnement qualitatif**

### **1.2.1 Les approches fondées sur la physique qualitative**

De point de vue historique, les premières approches de raisonnement qualitatif ont eu comme domaine d'application les systèmes réels physiques continus, c'est pourquoi ces approches sont souvent connues sous le nom d'approches de la physique qualitative. La présentation détaillée de ces approches est donnée ci-dessous.

#### **1.2.1.1 L'approche centrée processus**

L'approche Q.P.T (Qualitative Process Theory) [Forbus 1984] conçoit que tout changement n'est qu'une interaction d'un nombre fini de processus. Elle permet de mieux modéliser le raisonnement de sens commun du fait qu'elle sous-entend l'existence de causalité entre ces processus ce qui permet d'expliquer le comportement des systèmes réels. Tout processus au sens de cette théorie possède les paramètres suivants :

- a) Individuals : Ce sont les objets participant à ce processus.
- b) Preconditions : les préconditions qui doivent être satisfaites pour le déclenchement du processus
- c) Quantity conditions : les variables sur lesquelles opère le processus.
- d) Relations : désignent les relations entre les variables du processus.
- e) Influences : désignent les influences sur les variables (les conséquences) : elles sont de deux types qui sont les influences directes et les proportionnalités qualitatives.

Le raisonnement en Q.P.T se fait en quatre (04) phases :

- a) Détermination de tous les objets participants vérifiant les préconditions
- b) Détermination de toutes les instances de processus qui sont mutuellement consistants
- c) Détermination des effets directs et indirects sur les objets participants, c'est-à-dire les influences.
- d) Prédiction (analyser l'effet de ces influences sur le comportement du système) sur la base des influences déterminées dans la phase (c)

Q.P.E, l'abréviation de Qualitative Process Engine, [Forbus, 1990] est l'implémentation de la théorie de processus qualitative (Q.P.T) . Q.P.E trouve d'abord, toutes les entités actives et processus puis détermine les effets de ces derniers sur les entités ou les objets participants par

la résolution des influences directes. Le programme propage alors les effets de ces influences aux proportionnalités qualitatives qui, sont alors analysées. Une fois que ceci est accompli, QPE essaye d'identifier comment les changements qui se sont produits affectent le comportement du système. C'est-à-dire, identifier quelles quantités pourraient atteindre des valeurs de borne limite et ce qui arrivera aux entités et aux processus. Ce processus est réitéré jusqu'à ce qu'aucun autre changement ne puisse être trouvé.

- Caractéristiques de l'approche centrée processus :

- 1) prise en compte de la causalité
- 2) Modélisation compositionnelle
- 3) Se prête bien pour modéliser les systèmes dynamiques car selon cette méthode, les processus et les entités du système ne sont pas figés dès le début mais peuvent naître ou disparaître avec le temps.

### **1.2.1.2 L'approche centrée composant**

L'approche centrée composant [de Kler and Brown 1984] permet le développement d'une version qualitative de la physique conventionnelle : la physique qualitative. Dans cette approche, le comportement du système étudié est obtenu à partir de la description de la structure de ce système, exprimé par une collection de composants entre lesquels sont définies des relations ou des connections. La structure physique, selon cette approche est faite à partir de trois constituants qui sont les matériaux ( fluide, électricité ,gaz..), composant (vanne, transistor , résistance..) et conduits ( tuyau, fil, ..) . Les composants sont les parties actives des systèmes qui peuvent changer la forme des matériaux et les conduits sont des éléments passifs qui transportent les matériaux d'un composant à un autre. La structure du système peut être représentée par un graphe dont les nœuds représentent les composants et les arcs représentent les conduits. Pour déduire les comportements globales du système à partir des comportements de ses composants, on y introduit la notion de calcul qualitatif (addition et multiplication qualitatives) en utilisant des équations différentielles qualitatives, appelées *confluences*, définies sur des variables qualitatives. L'ensemble des états possibles d'un système, obtenus par « environement » [Hayes 1979], représenté graphiquement par un diagramme d'états, est un sous ensemble du produit cartésien des états de ses composants. La modélisation et l'étude du comportement d'un système physique, selon cette approche, passe par les étapes suivantes :

- 1) décomposer le système en ses constituants
- 2) donner le diagramme de la structure de ce système

- 3) Définir le modèle qualitatif du système : les confluences et les contraintes sur les variables du système
- 4) déduction de tous les états possibles système par « envisionment » (arbre des états possibles)
- 5) Interprétation

- Caractéristiques de l'approche centrée composant :

- 1) prise en compte de la causalité
- 2) modélisation compositionnelle
- 3) modélisation des systèmes dont la topologie est fixe et donc ne se prête pas pour modéliser les systèmes dynamiques.

### 1.2.1.3 L'approche centrée contrainte [Kuipers 1984][Kuipers 1986]

Dans cette approche, le système étudié est décrit par le biais d'un ensemble de paramètres physiques (variables et fonctions) et par un ensemble de relations entre ces paramètres (voir Fig. 1).

Un modèle d'équations différentielles qualitatives (QDE) est une description symbolique exprimant des connaissances incomplètes d'un système continu, et est donc une abstraction d'un nombre infini d'équations différentielles ordinaires. La simulation qualitative (QSIM), qui est l'implémentation de cette approche, permet de prédire l'ensemble de comportements possibles vérifiant les contraintes Q.D.E. On appelle un état qualitatif, une itération de la simulation qualitative permettant de créer un changement dans, au moins un des paramètres du système. Une succession achevée d'états qualitatifs constitue un scénario ou un comportement possible du système.

```

/* Définition des paramètres */
x = <min max> . définit la valeur qualitative du paramètre .x
QS(F,t) = <Qval,Qdir> . définit l'état qualitative du paramètre F à l'instant t.
    Qval la quantité de changement et Qdir le sens de changement
.....
/* Définition des relations */
/* les contraintes arithmétiques */
ADD (f,g,h) ..... f(t)+g(t)=h(t)
MULT (f,g,h) ..... f(t)*g(t)=h(t)
.....
/* contraintes fonctionnelles */
M+(f,g) : signifie f (t)= H(g(t)) avec H'(t) > 0 ,pour tout t
M- (f,g) : signifie f (t)= H(g(t)) avec H'(t) < 0 ,pour tout t
...../* contraintes dérivée */
DERIV (f,g) ..... signifie que : f'(t) =g(t)

```

Figure 1.1 : Description des systèmes dans l'approche centrée contrainte

Les travaux récents visant à améliorer les performances de QSIM ont été dominés par la recherche de solutions aux problèmes de la complexité permettant de générer des prévisions tout en respectant l'échelle de données en entrée. Une variété de méthodes existe maintenant pour appliquer une analyse plus approfondie, par exemple l'augmentation de la « résolution » de la description du système en intégrant des données quantitatives supplémentaires soigneusement choisies [Kuipers 1988].

Après la définition du modèle qualitatif du système, Le raisonnement sur ce dernier revient à résoudre un problème à contraintes qualitatives selon un cycle composé des étapes suivantes :

- propagation des contraintes
- prédiction
- interprétation
  
- Caractéristiques de l'approche centrée contrainte :
  - Expression explicite du temps dans le modèle qualitatif du système.
  - Utilisation de « l'envisonnement » pour déduire tous les comportements du système.
  - Possibilité d'avoir des états cycliques et des états divergents.
  - Intégration de données quantitatives pour réduire la complexité combinatoire.

#### 1.2.1.4 Comparaison des approches fondées sur la physique qualitative

En guise de comparaison de ces approches nous donnons le tableau ci-après. Les critères que nous avons retenus pour cette comparaison sont distinctifs et permettent de mettre en valeur les caractéristiques de chaque approche.

	Approche centrée processus	Approche centrée composant	Approche Centrée Contrainte
<b>Concepts</b>	Processus, vues individuelles	Structure, composant, connexion entre les composants, confluences, « envoisinent »	La simulation qualitative.
<b>Principe</b>	le processus, est l'origine de tout changement.	Le comportement du système et l'ensemble des comportements de ses composants	résolution de contraintes qualitatives par la simulation
<b>Objets manipulés</b>	Processus, individus, objet, vue individuelle, espaces de quantités	Composants, conduits et matériaux	Paramètres, relations et équations diff. qualitatifs
<b>Etat qualitatif et comportement qualitatif</b>	Valeurs qualitatives des variables	Valeurs qualitatives des variables	Valeurs qualitatives des variables
<b>?</b>			

Limites /Avantages de l'approche	Modélise les systèmes dynamiques	Ne peut modéliser les systèmes dynamiques	Modélise les systèmes dynamiques
Applications et implémentations	GARP3 [Bredeweg et al. 2009]	ENVISION [de Kleer and Brown 1984]	QSIM [Kuipers 1986]

Tableau 1.1 : Comparaison des approches qualitatives fondées sur la physique qualitative

## 1.2.2 Les approches logiques

### 1.2.2.1 Logique de défaut [Reiter 1980]

Est une logique non monotone qui permet de raisonner en l'absence de connaissances complètes sur le domaine de discours. Dans cette logique proposée par Reiter, on définit un ensemble de règles d'inférences appelées défauts (D) qui permet de raisonner sur une base de connaissances (A), exprimée dans un langage de prédicats de 1er ordre.

Une théorie de défaut est donc définie par le couple (A, D) où chaque défaut est de la forme  $a : b / c$  ; qui se lit « si a est vraie et si b est cohérente avec ce qui est connu alors inférer c ». a, b et c sont des formules bien formées et nommées respectivement, la condition, la justification et la conséquence du défaut.

Une logique de défaut affecte à une théorie de défaut une collection d'extensions qui sont un ensemble de croyances (faits). On distingue plusieurs types d'extension : sans conflit, minimale et maximale, minimale sans conflit et maximale sans conflit

### 1.2.2.2 Logique modale [McDermott & Doyle 1980]

McDermott et Doyle étaient les premiers ayant introduit la logique modale pour modéliser le raisonnement non monotone. Ils ont introduit la modalité de consistance  $M$  de telle sorte que la formule modale  $M\phi$  signifie «  $\phi$  est consistante avec les choses déjà connues ». Leur contribution réside dans le fait d'avoir introduit une règle d'inférence non monotone formulée comme suit : « on ne peut inférer  $\neg \phi$  »  $\vdash M\phi$ . Le schéma d'axiomes comprend, en plus de cette règle, les schémas d'axiomes de la logique de prédicats ainsi qu'un ensemble d'axiomes modaux. Si le système comporte le schéma d'axiome de distribution  $K$  et celui de la connaissance  $T$ , il sera appelé le système non monotone  $KT$ . Si ce dernier comporte le schéma d'axiomes 4 ou 5, il sera appelé non monotone  $S4$  ou non monotone  $S5$ .

Dans cette logique, la règle de défaut : « les oiseaux généralement volent » peut être exprimée par :  $\text{oiseaux}(x) \wedge M \text{vole}(x) \supset \text{vole}(x)$ .

A partir d'un ensemble de prémisses (théorie initiale), on peut inférer l'ensemble de toutes les formules mutuellement consistantes qui constitue les théorèmes de la théorie.

### 1.2.2.3 Logique auto – épistémique [Moore 1985]

Est un cadre logique basé sur la logique modale permettant de formaliser le raisonnement non monotone portant sur les croyances et les connaissances. Elle permet de formaliser des énoncés de la forme « si je ne crois pas que  $\varphi$  est vérifiée alors  $\psi$  est vérifiée ». L'opérateur modal utilisé est l'opérateur de nécessité  $L$  qui est le dual de l'opérateur de possibilité  $M$ . La signification de la formule modale  $L\varphi$  est « il est cru que  $\varphi$  est vérifiée ». Notons qu'il y'a une relation entre les deux opérateurs modaux  $M$  et  $L$  de la sorte:  $M\varphi \equiv \neg L\neg\varphi$ . C'est pourquoi que la logique de Moore est vue comme une reconstruction de la logique de McDermott. La règle de défaut « les oiseaux généralement volent » est formalisée par :

$$\text{Oiseaux}(x) \wedge \neg L\neg \text{vole}(x) \supset \text{vole}(x).$$

Dans la logique de croyance les opérateurs modaux  $L$  et  $M$  signifient respectivement : « est cru », « l'inverse n'est pas cru ». Dans la logique de connaissance les deux opérateurs signifient respectivement « est connu », « l'inverse n'est pas connu »

Des implémentations basées sur la logique auto épistémique ont été réalisées. A titre d'exemple, l'étude et l'implémentation du problème de déduction d'une extension stable de croyances à partir d'un ensemble de croyances de départ [Antonio 97]

### 1.2.2.5 L'argumentation

#### - Introduction

L'argumentation est une pratique cognitive quotidienne de l'être humain qui consiste à soutenir ou à contester une opinion, une thèse sur un thème ou un sujet, mais aussi à agir sur le destinataire ou l'auditoire en cherchant à le persuader ou de le dissuader.

L'argumentation est une approche du raisonnement non monotone ou révisable. C'est un modèle pour raisonner sur des informations incomplètes, imprécises ou éventuellement incohérentes [Aubry 2005]. Ce modèle est fondé sur la construction d'arguments et de contre arguments en faveur ou non d'une thèse donnée, de la comparaison de ces arguments puis ensuite en choisir les plus acceptables. Une autre définition de l'argumentation a été donnée par [van Eemeren et al. 1996] comme suit :

“Argumentation is a verbal and social activity of reason aimed at increasing (or decreasing) the acceptability of a controversial standpoint for the listener or reader, by putting

forward a constellation of propositions intended to justify (or refute) the standpoint before a rational judge “

Plusieurs logiques formelles d'argumentation ont vu le jour en tant qu'un style de formalisation du raisonnement non monotone et qui sortent du cadre de la logique classique. Dans le raisonnement non monotone, une conclusion peut ne pas demeurer valable avec l'arrivée de nouvelles informations, c'est pourquoi le raisonnement non monotone est souvent appelé raisonnement révisable.

Une règle réfutable se base sur une ou plusieurs raisons réfutables pour aboutir à une conclusion. Lorsque nous enchaînons plusieurs raisons réfutables pour arriver à une conclusion, nous dirons que nous avons procédé à une argumentation et non à une preuve.

#### - **L'argumentation, une approche pour le raisonnement réfutable**

C'est grâce à Hart [Hart 1951] que fut introduit le concept de la réfutation (defeasibility, en Anglais) dans le raisonnement. En 1958 Toulmin [Toulmin 1958] proposa un modèle conceptuel pour le raisonnement réfutable basé argumentation qui décompose l'argument en ses parties et de reconnaître le rôle de chacune de ces parties qui sont : Data (D), Warrant(W), Backing (B), C (Claim), Qualifier (Q) et la Restriction (R). Cette structure se lit de la façon : « Etant données D et d'après le principe ou la (es) règle (es) W supporté (es) par les raisons B, l'assertion C est obtenue avec le quantificateur modal Q , mais avec la restriction R ». En 1974, Pollock [Pollock 1974] a développé une théorie pour le raisonnement révisable. Doyle [Doyle 1979] a développé le système TMS de maintien de vérité où les raisons sont réfutables. En 1980, Reiter [Reiter 1980] a développé la logique de défaut. Dès 1980, les philosophes et les épistémologistes ont commencé à s'intéresser à cet axe de recherche. Jhon Pollock et Donald Nute ont joué un grand rôle dans le rapprochement de l'épistémologie et l'IA [Carlos et al 2000].

Plusieurs modèles logiques d'argumentation existent. Nous allons présenter ci-après les plus importants de ces modèles.

#### - **Système de Simari & Loui** [Simari & Loui 1992]

La base de connaissances est représentée par un couple  $(K, \Delta)$  ;  $K$  est la connaissance monotone , et  $\Delta$  est l'ensemble des règles non monotones exprimées par :  $p \dashv\vdash q$  et se lit par «  $p$  est la cause de  $q$  ».

Un sous ensemble  $T$  des instances des éléments de  $\Delta$  est un argument pour la proposition  $h$  si et seulement si :

$$1. K \cup T \vdash h \quad (T \text{ dérive } h)$$

2.  $K \cup T \rightarrow \perp$  (*T est consistant avec K*)  
 3.  $\exists T' \subset T$  tel que.  $K \cup T' \vdash h$  (*T est minimal*)  
 $\langle T, h \rangle$  définit une structure d'argument. Les arguments contiennent les sous arguments. La structure d'argument  $\langle S, j \rangle$  est un sous argument de la structure d'argument  $\langle T, h \rangle$  si  $S \subseteq T$ .

Ce système prévoit le conflit entre arguments. Trois cas de conflit sont considérés :  
 Le désaccord, l'opposition et la défaite. Ces cas de conflit sont définis formellement ci-après :

- Deux arguments  $\langle A_1, h_1 \rangle$  et  $\langle A_2, h_2 \rangle$  sont en désaccord si et seulement si :

$$K \cup \{ h_1, h_2 \} \vdash \perp$$

- Deux arguments  $\langle A_1, h_1 \rangle$  et  $\langle A_2, h_2 \rangle$  sont en opposition au niveau du littéral  $h$  si et seulement si :

$\exists$  Sous -argument  $\langle A, h \rangle$  de  $\langle A_2, h_2 \rangle$  tel que  $\langle A_1, h_1 \rangle$  et  $\langle A, h \rangle$  soit en désaccord.

-  $\langle A_1, h_1 \rangle$  défait , dans le sens fort ou faible,  $\langle A_2, h_2 \rangle$  si et seulement si  $\exists$  un sous argument  $\langle A, h \rangle$  de  $\langle A_2, h_2 \rangle$  tel que  $\langle A_1, h_1 \rangle$  oppose  $\langle A, h \rangle$  au littéral  $h$  et  $\langle A_1, h_1 \rangle$  est strictement plus spécifique que  $\langle A, h \rangle$  ou  $\langle A_1, h_1 \rangle$  est non relié par spécificité à  $\langle A, h \rangle$ .

La défaite entraîne l'opposition et l'opposition entraîne le conflit entre arguments.

### Exemple 1.1 : [Carlos 00]

Soit la connaissance suivante :

Les voitures fiables sont souvent recommandables.

Les voitures fiables dont la production s'est arrêtée sont souvent non recommandables.

Les voitures européennes dont la pièce détachée est disponible sont souvent fiables.

Les voitures ayant des problèmes de température sont souvent non fiables.

Les voitures avec moteur en arrière ont des problèmes de température.

Les voitures ayant moteur en arrière ont souvent une attraction arrière.

Les voitures qui consomment moins d'énergie et qui ont une attraction arrière sont souvent fiables.

Les voitures italiennes sont des voitures européennes.

Et supposons que nous savions que  $f_6$  est une voiture italienne dont la pièce est disponible, a un moteur en arrière, consomme moins d'énergie et dont la chaîne de production s'est arrêtée.

Ces connaissances peuvent être exprimées dans le système de Simari et Loui par :

L'ensemble  $K_{cp} = \{ \text{moteur\_arriere}(f6), \text{italienne}(f6), \text{moins-conso}(f6), \text{prod\_arret}(f6), \text{piece-dispo}(f6) \}$ , qui représente les connaissances contingentes, un ensemble  $K_{GN} = \{ \text{italienne}(x) \rightarrow \text{européenne}(x) \}$  qui représente la connaissance nécessaire, et l'ensemble  $\Delta$  des règles défaisables définies par :

$\Delta = \{ \text{fiable}(x) >-- \text{recommandable}(x), \text{fiable}(x) \wedge \text{prod\_arret}(x) >-- \neg \text{recommandable}(x), \text{européenne}(x) \wedge \text{piece\_dispo}(x) >-- \text{fiable}(x), \text{moteur\_arriere}(x) >-- \text{systeme\_arriere}(x), \text{moins\_conso}(x) \wedge \text{systeme\_arriere}(x) >-- \text{fiable}(x), \text{moteur\_arriere}(x) >-- \text{prob\_temperature}(x), \text{prob\_temperature}(x) >-- \neg \text{fiable}(x) \}$

A partir de ces connaissances, on peut tirer les arguments  $A1, A2, A3, A4$  suivants :

$A1 = \{ \text{européenne}(f6) \wedge \text{piece\_dispo}(f6) >-- \text{fiable}(f6), \text{fiable}(f6) >-- \text{recommandable}(f6) \}$   
pour  $\text{recommandable}(f6)$  ;

$A2 = \{ \text{moteur\_arriere}(f6) >-- \text{prob\_temper}(f6), \text{prob\_temperature}(f6) >-- \neg \text{fiable}(f6) \}$   
Pour  $\neg \text{fiable}(f6)$  ;

$A3 = \{ \text{européenne}(f6) \wedge \text{piece\_dispo}(f6) >-- \text{fiable}(f6), \text{fiable}(f6) \wedge \text{prod\_arret}(f6) >-- \neg \text{recommandable}(x) \}$  pour  $\neg \text{recommandable}(f6)$  ;

$A4 = \{ \text{moteur\_arriere}(f6) >-- \text{systeme\_arriere}(f6), \text{moins\_conso}(f6) \wedge \text{systeme\_arriere}(f6) >-- \text{fiable}(f6) \}$  pour  $\text{fiable}(f6)$  .

- **Le système de Dung** [Dung 1993]

Dung a développé une théorie de l'argumentation dont la notion centrale est l'acceptabilité des arguments. Il a montré que la plupart des approches du raisonnement non monotone en Intelligence Artificielle et en programmation logique sont des formes particulières de sa théorie d'argumentation.

Une programmation logique étendue (avec négation explicite) est composée de causes de la forme :

$$L_0 \leftarrow L_1, L_2, \dots, L_m, \neg L_{m+1}, \dots, \neg L_{m+n}$$

$L_i$  : sont des littéraux positifs ou négatifs de la forme  $A$  ou  $\neg A$  et sont appelés des Littéraux objectifs

$\neg L_i$  : sont appelés des littéraux subjectifs considérés en tant qu'hypothèses sur lesquelles se base la preuve car l'acceptabilité de la preuve est basée sur l'acceptabilité des hypothèses.

Les arguments sont donc vus comme un ensemble d'hypothèses construites à partir d'un ensemble de littéraux subjectifs de base.

Un argument  $A$  supporte un littéral objectif  $L$  s'il existe une preuve pour  $L$  basée sur des hypothèses contenues dans  $A$ . Un argument se défait s'il supporte  $L$  et  $\neg L$ .

Dung distingue deux manières pour attaquer un argument :

- Un argument  $A$  attaque un argument  $A'$  par Reduction & Absurdum ( R-AA attack) si :

$A \cup A'$  se met en échec.

- Un argument  $A$  attaque un argument  $A'$  par ground attack ( G-attack) s'il existe une hypothèse  $\neg L$  dans  $A'$  telle que  $L$  soit supporté par  $A$ .

Etant donnée une programmation logique étendue  $P$ , la sémantique de  $P$  est définie par la sémantique du cadre formel suivant :

$$AF(P) = (AR p, attacks, g-attacks)$$

Tel que :  $attacks = R-AA attacks \cup G-attacks$ .

Un ensemble d'arguments  $S$  où il n'existe aucun argument  $A$  qui attaque un autre argument  $A' \in S$  est appelé ensemble sans conflit.

Un argument  $A$  est acceptable vis-à-vis d'un ensemble d'arguments  $S$  si pour n'importe quel argument  $B$  attaquant  $A$ ,  $B$  est G-attaqué par un certain argument  $A' \in S$ .

Une extension préférée d'une structure argumentative (argumentation framework) est l'ensemble maximal d'arguments acceptables.

Un ensemble d'arguments sans conflit  $S$  est une extension stable si  $S$  G-attack tout argument n'appartenant pas à  $S$ .

Chaque extension stable est une extension préférée mais la réciproque n'est pas vraie.

Un ensemble  $S$  est l'ensemble réponse d'une programmation logique étendue  $P$  si et seulement s'il existe une extension stable  $E$  de  $AF(P)$  tel que :

$$S = \{L / L \text{ est supporté par un argument de } E\}$$

La fonction caractéristique d'une structure argumentative :

$Faf(S) = \{A / A \text{ est acceptable vis-à-vis de } S\}$ .  $Faf$  est monotone par rapport à l'inclusion.

Un ensemble d'arguments sans conflit est acceptable si et seulement si  $S \subseteq Faf(S)$ .

Un ensemble d'arguments acceptables est appelé une extension complète si et seulement si chaque argument acceptable vis-à-vis de  $S$  appartient à  $S$ . Dung a montré que chaque extension préférée est une extension complète minimale. Un peu plus tard, [Dung 95] a proposé une structure argumentative plus abstraite que celle d'origine et qui constitue les fondements de base des approches connues en raisonnement non-monotone.

**Exemple 1.2 :** [Dung 1993]

Dung illustre sa théorie en proposant le débat suivant entre deux personnes :

I : mon gouvernement ne peut pas négocier avec votre gouvernement car votre gouvernement ne reconnaît même pas notre gouvernement

K : votre gouvernement ne reconnaît non plus notre gouvernement

I : mais votre gouvernement est terroriste

Notons par  $i_1$  et  $i_2$  le premier et le second argument de la personne I et  $k$  l'argument de la personne K. Le débat ci-dessus peut être représenté par la structure suivante :

$AF = (AR, attacks)$  comme suit :

$AR = \{i_1, i_2, k\}$  ;  $attacks = \{(i_1, k), (k, i_1), (i_2, k)\}$  ; L'extension préférée  $E = \{i_1, i_2\}$

**Exemple 1.3 :** [Carlos 00]

Considérons le programme de la logique étendue suivant :

$innocent(x) \leftarrow not\ guilty(x)$

$guilty(x) \leftarrow confess\_guilt(x)$

$\neg innocent(x) \leftarrow confess\_guilt(x), not\ lies(x)$

$confess\_guilt(Jhon)$

$lies(Jhon)$

$confess\_guilt(Peter)$ .

Soient les ensembles des suppositions (hypothèses) suivantes :

$A_0 = \emptyset$ ,  $A_1 = \{not\_guilty(Jhon)\}$ ,  $A_2 = \{not\ lies(Peter)\}$ ,  $A_3 = \{not\ guilty(Peter)\}$

Alors,  $A_0$  supporte guilty (Jhon), l'argument  $A_1$  supporte innocent (Jhon), l'argument  $A_2$  supporte  $\neg$  innocent(Peter) et l'argument  $A_3$  supporte innocent(Peter).

Remarquons que :  $A_2$  RAA- attaque  $A_3$  car l'argument  $A_2 \cup A_3$  s'auto-défait du moment qu'il supporte innocent (Peter) et  $\neg$  innocent(Peter) en même temps. Aussi, l'argument  $A_0$  G-attaque  $A_1$ .

- **Le système de Hage & Verheij** [Hage & Verheij 1995]

Hage & Verheij [Hage & Verheij 1995], ont présenté un modèle d'argumentation basé sur la raison. Une raison peut consister en plusieurs sous raisons. Les arguments sont de la forme :

*(sous-raison<sub>1</sub>, sous-raison<sub>2</sub>, sous-raison<sub>3</sub>,..., sous-raison<sub>k</sub>) Donc (Conclusion)*

Les arguments peuvent être combinés par subordination ou par coordination définies formellement par :

-  $(A, B)$  sont liés par une relation de subordination si et seulement si la conclusion  $h$  de l'un des deux arguments est la sous raison de l'autre.

-  $(A, B)$  sont liés par une relation de coordination si et seulement si  $A$  et  $B$  supportent la même conclusion  $h$ .

**Exemple 1.4** : [Verheij 1996]

Soit l'argument :

Le soleil brille, donc, il fait beau.

L'argument suivant :

Un collègue, tout mouillé, dit qu'il pleut

Donc, il pleut probablement.

Donc, il est prudent de mettre son imperméable.

Est le résultat de subordination de l'argument un collègue, tout mouillé, dit qu'il pleut donc, il pleut probablement et de l'argument il pleut probablement. Donc, il est prudent de mettre son imperméable.

La coordination des deux arguments Le soleil brille, donc, il fait beau et l'argument Le soleil brille, donc, il fait beau donne l'argument résultat : le soleil brille ; le ciel est bleu. Donc il fait beau.

Verheij distingue entre défaite par « undercutting » et défaite par « rebutting », mais il a discuté d'autres types de défaites qui sont :

Défaite par affaiblissement séquentiel ( sequential weakning) : les arguments ne peuvent pas être enchaînés d'une manière indéfinie.

Défaite par renforcement parallèle ( parallel strengthening) : le cas dans lequel la coordination de deux arguments défait un des deux arguments.

Défaite collective (collective defeat) : le cas ou un groupe d'argument est défait, alors que chaque argument appartenant au groupe n'est pas individuellement défait.

Défaite indéterminée (undeterministic defeat) : apparaît quand il y'a plusieurs choix d'arguments que l'on peut créer et le choix est résolu d'une manière indéterministe. Il est nécessaire de prendre un des arguments comme défait afin de supporter les autres.

Un argument peut avoir deux états : défait ou non défait. Un argument est défait s'il ne justifie pas sa conclusion et il est non défait s'il la justifie.

**Exemple 1.5** : [Verheij 1996]

Cet exemple illustre les types de défaites (d'arguments) prévues par le modèle de Verheij

- Undercutting defeat

Soient les deux arguments :

The object looks red . So, the object is red.

The object is illuminated by a red light.

Le fait « the object is illuminated by red light » undercuts l'argument que « the object is red »

- Rebutting defeat

Soient les deux arguments :

Jhon likes French fries. So he orders French fries

Jhon is on a low calorie diet. So he doesn't order French fries

Ces deux arguments supportent deux conclusions opposites. Chacun des deux arguments défait l'autre par rebutions.

- Sequential weakening

Soit l' argument suivant:

This body of grains of sand is a heap.

So, this body of grains of sand minus 1 grain is a heap.

So, this body of grains of sand minus 2 grains is a heap.

So, this body of grains of sand minus 3 grains is a heap

.....

So, this body of grains of sand minus  $n$  grains is a heap

Chaque pas de l'argument est correct mais ceci ne peut être correct indéfiniment puisque à la fin il n'y aura aucune graine de sable qui va rester. Le problème ici réside dans la difficulté de dire à quel point l'argument devient défait.

- Parallel strengthening

-

Soient les arguments suivants :

Jhon is a miner first offender. So, Jhon should not be punished. (1)

Jhon has robbed Alex. So, Jhon should be punished. (2)

Jhon has robbed Alex; Jhon injured Alex in a fight. So, Jhon should be punished. (3)

- Collective and deterministic defeat

Soient les arguments:

Jhon is qualified. So, Jhon is hired. (1)

Alex is qualified. So, Alex is hired (2)

Mary is qualified. So Mary is hired (3)

Si l'employeur ne peut pas recruter plus de deux personnes alors deux solutions sont envisageables : soit on défait les trois arguments et on introduit d'autres critères de sélection, soit on défait un argument au choix d'une manière indéterministe.

- **Le processus d'argumentation**

La construction de nouveaux arguments à partir des arguments pris en compte dans une étape donnée nous conduit à une nouvelle étape. Débutant par un ensemble donné de prémisses, les arguments sont construits graduellement par six (06) manières élémentaires, qui sont les suivantes :

a) introduire une nouvelle déclaration ou assertion.

b) Ajouter une conclusion d'un argument déjà existant afin de supporter une nouvelle conclusion (un pas en avant)

c) Ajouter un pas en arrière, c'est à dire que la prémisse d'un argument existant est remplacée par une ou plusieurs nouvelles prémisses.

d) Ajouter un grand pas : c'est le cas où la conclusion d'un argument existant est supportée par une raison supplémentaire.

e) Subordination de deux arguments

f) Coordination de deux arguments

- **Le système de Bondarenko et al.** [Bondarenko et al. 1997]

Bondarenko et al. ont développé un système d'argumentation abstrait basé sur les hypothèses (Abstract assumption based Framework) pour le raisonnement par défaut. C'est une combinaison de la programmation logique et la notion d'extension comme présentée dans le théoricien de Poole. [Carlos 2000]. Dans Ce système, l'hypothèse ou la supposition peut être attaquée si le contraire pourrait être prouvé.

Formellement, ce système est donné par le triplet :  $(T, Ab, \Gamma)$  tels que :

$T$  : une théorie, un ensemble de croyances  $\subseteq \xi$  ;  $\xi$ . étant un langage formel.

$Ab$  : ensemble d'hypothèses non vide  $\subseteq \xi$

$\Gamma$  : est un mapping défini de  $Ab \subseteq \xi \rightarrow \xi$ .

La fermeture déductive  $Th(T \cup \Delta)$  avec  $\Delta \subseteq Ab$ , est souvent appelée une extension dans la littérature de la logique non monotone. La question est de savoir sous quelles conditions une extension pourrait être acceptée.

Ce système a véhiculé plusieurs concepts tels le concept de l'extension et ses différents types, le concept de l'admissibilité des arguments ainsi que la préférence entre arguments. Nous donnons dans ce qui suit un résumé de ces concepts.

- **L'extension naïve**

Etant donnée une structure argumentative :  $(T, Ab, \Gamma)$  comme définie ci-dessus. Un ensemble d'hypothèses  $\Delta \subseteq Ab$  est sans conflit si et seulement si :

$\forall \alpha \in Ab, (T \cup \Delta)$  ne produit pas  $\alpha$  et  $\bar{\alpha}$ .

Un ensemble d'hypothèses  $\Delta \subseteq Ab$  est maximal et sans conflit si et seulement si  $\Delta$  est sans conflit et il n'existe pas  $\Delta'$  sans conflit tel que  $\Delta \subset \Delta'$

Une extension est dite la plus simple des extensions ou « naïve » si elle est maximale et sans conflit.

- **L'extension stable**

Un ensemble d'hypothèses  $\Delta$  est dit stable si et seulement si :

- a)  $\Delta$  est fermé
- b)  $\Delta$  ne s'attaque pas (consistante)
- c)  $\Delta$  attaque toute hypothèse (supposition)  $\alpha \notin \Delta$ .

- **Relation entre l'extension maximale sans conflit et l'extension stable.**

- Toute extension stable est une extension maximale sans conflit.

- La structure  $(T, Ab, \Gamma)$  est appelée normale si et seulement si tout ensemble d'hypothèses maximal et sans conflit est stable.

### - Sémantique de l'admissibilité

La différence entre une extension stable et une extension admissible, c'est que cette dernière attaque toute supposition qui l'attaque, formellement :

Un ensemble fermé de suppositions  $\Delta \subseteq Ab$  est admissible si et seulement si :

- a)  $\Delta$  n'attaque pas soi même
- b) Pour tout ensemble fermé de suppositions  $\Delta' \subseteq Ab$ , si  $\Delta'$  attaque  $\Delta$ , alors  $\Delta$  attaque  $\Delta'$

### - Sémantique préférentielle

Une extension est acceptable si elle est maximale admissible, dans le sens qu'aucun sous ensemble de l'extension n'est aussi admissible.

On montre que pour une structure argumentative  $(T, Ab, \Gamma)$  normale, les notions d'extension maximale sans conflit, d'extension stable et d'extension préférée sont équivalentes.

### - Sémantique de l'extension complète

Une extension admissible de telle sorte qu'elle contienne toutes les suppositions qu'elle défend est appelée une extension complète. Cette sémantique est intermédiaire entre l'admissibilité et la sémantique préférentielle. La sémantique de l'extension complète considère qu'une extension est acceptable si elle est admissible et si elle contient toutes les suppositions qu'elle défend.

Un ensemble de suppositions  $\Delta$  défend une supposition  $\alpha$  si et seulement si pour tout ensemble fermé  $\Delta'$ , si  $\Delta' \subset \Delta$  attaque  $\alpha$ , alors  $\Delta$  attaque  $\Delta' - \Delta$ .

**Exemple 1.6 :** [Bondarenko et al. 1997]

La structure argumentative basée hypothèses qui correspond au programme logique :

Soit le programme logique  $T = \{p \leftarrow \text{not } q, q \leftarrow \text{not } p, r \leftarrow p, r \leftarrow q\}$ .

La structure argumentative basée hypothèses qui correspond au programme  $T$  comprend deux ensembles d'hypothèses stables  $\{\text{not } p\}$  et  $\{\text{not } q\}$  qui sont aussi complètes.

**Exemple 1.7 :** [Bondarenko et al. 1997]

Soit le programme logique de défaut :

$T = \{ \neg \text{guilty} \rightarrow \text{innocent} \}$

$T = \{ p \leftarrow \text{not } q, q \leftarrow \text{not } r, r \leftarrow \text{not } p \}$  comprend:

- Un ensemble préféré d'hypothèses réduit à  $\emptyset$  qui n'est pas maximale sans conflit.
- Deux ensembles d'hypothèses maximale sans conflit sont :  $\{ \text{not } p \}$ ,  $\{ \text{not } q \}$ ,  $\{ \text{not } r \}$  qui ne sont pas admissibles.

**Exemple 1.8 :** [Bondarenko et al 1997]

$\left\{ D = \frac{M \neg \text{guilty}}{\neg \text{guilty}} \right\}$  ; (Mp : veut dire que la proposition « p » est consistante)

Dans le système d'argumentation basée hypothèses correspondant à ce programme, un seul ensemble maximal sans conflit est défini :  $\{ M \neg \text{guilty} \}$

- **Le système d'Amgoud & Cayrol** [Amgoud & Cayrol 2002]

Dans ce système, ses auteurs ont utilisé le système d'argumentation de Dung [Dung 95] pour proposer un cadre d'argumentation plus général ayant comme principal concept l'acceptabilité des arguments et basé non seulement sur la relation de défaite, comme c'est le cas dans le système de Dung, mais également sur une relation de préférence, qui est une relation d'ordre entre les arguments. C'est pourquoi ; ce système est connu sous le nom du système d'argumentation basé sur les préférences.

Formellement, ce système est défini par le triplet  $\langle A, R, \text{Pref} \rangle$  où  $A$  est l'ensemble des arguments,  $R$  est une relation de défaite entre les arguments et  $\text{Pref}$  est une relation de préférence entre ces arguments. Ce système permet de définir trois classes d'arguments qui sont l'ensemble des arguments acceptable, noté par  $\text{AccR, Pref}$ , l'ensemble des arguments rejetés, noté par  $\text{RejR, Pref}$  et l'ensemble des arguments non rejetés et non acceptés, noté par  $\text{AbR Pref}$ .

Dans ce système, on dit qu'un argument  $A$  défait un autre argument  $B$  s'il le défait par la réfutation (rebutting) ou par le dégageant (undercutting) et que  $B$  n'est pas préféré à  $A$ . Ce système propose un processus d'argumentation où deux interlocuteurs, le participant et son adversaire, échangent des arguments et de contre-arguments au sujet d'un premier argument  $A_0$  proposé par le participant. Le processus d'argumentation est vu comme un jeu de dialogue entre deux joueurs et il est représenté par un arbre appelé arbre de dialogue. La stratégie du joueur participant est de défendre l'acceptabilité de l'argument  $A_0$ , tandis que le joueur adversaire essaye de le défaire. Le gagnant du jeu est celui qui présente le dernier argument.

Ce système a été généralisé par la suite pour exprimer les concepts de confiance et de récompense durant le processus d'argumentation [Amgoud and Prade 2005]

### Exemple 1.9

Soit  $\langle A, R, \text{Pref} \rangle$  un système d'argumentation basé sur les préférences  $= \{ a, b, c, d, e \}$ ,  
 $R = \{ (c, d), (d, c), (a, e), (b, d) \}$  and  $b, c \gg \text{pref } d$  ;  $a \gg \text{pref } e$  , alors nous avons :  
 $\text{Acc}R, \text{Pref} = \{ a, b, c \}$ ,  $\text{Rej}R, \text{Pref} = \{ d, e \}$  and  $\text{Ab}R, \text{pref} = \emptyset$

### - Comparaison des systèmes logiques d'argumentation

En guise de comparaison entre les cadres logiques d'argumentation présentés ci-dessus, nous donnons le tableau comparatif ci-dessous. Les critères de comparaison que nous avons jugés pertinents sont : la manière dont est spécifié l'argument (nature de l'argument), si le

	Modèle l'argumentation	Représentation de l'argument	Relations de conflit entre arguments	Acceptabilité des arguments	Processus d'argumentation
Système de Simari et and Loui	Système basé raisons	Sous ensemble de règles non monotones	Désaccord défaite, opposition	Logique	Non
Système de Dung	Système basé prétentions ou hypothèses	Ensemble d'hypothèses Supportant une conclusion	"RAA attack", "G- attack"	arguments sans conflit	Non
Système de Hage & Verheij	Système basé sur les raisons	Ensemble de raisons supportant une conclusion	Défaite par "undercutting" ou par "rebutting "	un argument est accepté s'il justifie sa conclusion	Oui
Système de Bondarenko et al.	Système basé prétentions hypothèses	Ensemble d'hypothèses Supportant une conclusion	« Tackle »	acceptabilité d'un ensemble fermé d'hypothèses	Non
Système de Amgoud and Cayrol	Système basé préférences entre arguments	$\langle H, h \rangle$ H: support; h : conclusion	Défaite	Basé sur une relation de défaite et sur une relation de préférence	Oui

Tableau 1..2 : Comparaison des systèmes logiques d'argumentation

conflit entre arguments est prévu par le système, si le système traite l'acceptabilité des arguments, et enfin si le système propose ou non une démarche pour la génération des arguments.

### **1.2.3 Les approches algébriques**

Cette classe d'approches permet de représenter et de raisonner qualitativement sur les relations de type temporel, spatial ou causal. Ces approches qualitatives sont de type algébrique où une structure d'algèbre est définie sur un ensemble fini de relations (temporelles, spatiales ou causales), appelé ensemble de base, muni d'opérations algébriques telle l'addition, la multiplication, la conversion et l'intersection. Nous allons maintenant présenter ci-dessous ces différentes approches.

#### **1.2.3.1 Le raisonnement qualitatif temporel**

Le temps comme l'espace constitue une composante fondamentale dans le domaine de L'I.A car l'information temporelle est particulièrement présente dans la description de la dynamique des objets et événements et des interactions qui font notre monde. A titre d'exemples, nous pouvons citer deux applications majeures du raisonnement temporel qui sont la planification et l'ordonnancement des tâches. Lorsque l'information temporelle numérique disponible fait défaut ou elle est incomplète, on tente d'appliquer un raisonnement qualitatif temporel au problème donné dans le but d'en trouver des solutions qualitatives acceptables. Il existe deux cadres formels pour le raisonnement qualitatif temporel : L'algèbre des intervalles [Allen 1983] et l'algèbre des points [Vilain et Kautz 1986]. Nous allons présenter ci-dessous ces deux formalismes.

##### **- L'algèbre des intervalles**

Ce framework permet de raisonner sur les relations temporelles qui existent entre les variables du problème étudié où chaque variable est assignée un intervalle de temps. Sur ces intervalles de temps, Allen définit treize (13) relations temporelles de base qui sont : { meets(m), met\_by(mi), overlaps(o), overlapped\_by(oi), starts(s), started\_by(si), during(d), contains(di), finishes(f), finished\_by(fi), before(b), after(a), equal(e)}. (Voir Figure 1.2).

##### **- L'algèbre des points**

Ce framework a été proposé par ses auteurs Vilain et Kautz dans le but de fournir une alternative au formalisme de l'algèbre des intervalles qui présente des problèmes de complexité lors des éliminations des inconsistances (consistance d'arc et consistance de chemin) ou lors de la recherche des scénarios (solutions). L'algèbre des points utilise trois

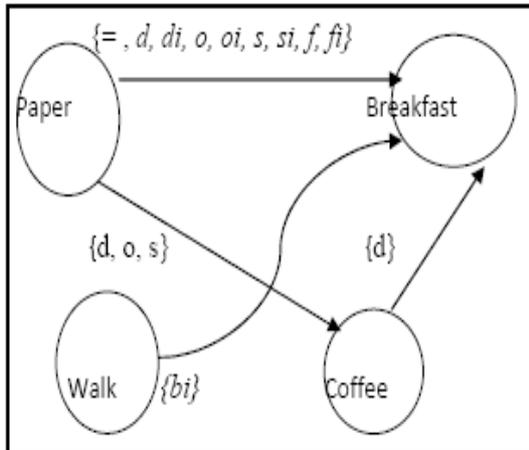


Figure 1.3. Un R.C.Q.T basé intervalles

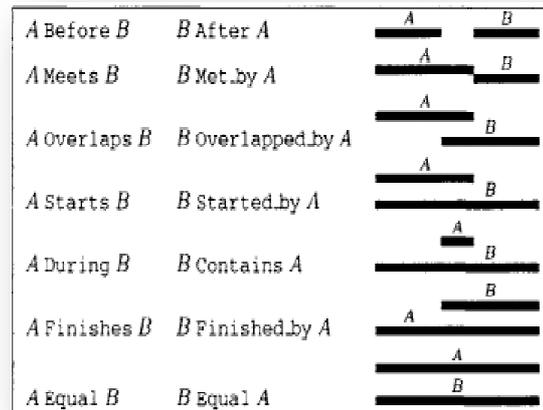


Figure 1.2 : les treize (13) relations d'Allen

(03) relations temporelles de base qui sont :{ <, >, = } d'où on tire l'ensemble de relations possibles :{ <, >, =, {>, =}, {<, =}, {<, >, ? } , voir la figure 1.4 dû à R. Dechter [Dechter 2003].

La représentation de ces connaissances temporelles qualitatives est faite par le biais d'un graphe temporel appelé réseaux de contraintes temporelles qualitatives, voir l'exemple de la Figure 1.3 et Figure 1.4 tirés de [Dechter 2003]. Le raisonnement qualitatif temporel va se résumer donc à la résolution d'un problème de satisfaction de contraintes.(P.S.C).

### 1.2.3.2 Le raisonnement qualitatif spatial

L'espace comme le temps, constitue une catégorie fondamentale dans le domaine de la représentation des connaissances car l'information spatiale est omniprésente dans les activités et des interactions des objets dans notre monde. Cependant, contrairement à la représentation et au raisonnement temporel, la représentation et le raisonnement spatial ont mis longtemps à devenir une branche de l'I.A à part entière car l'information spatiale est multidimensionnelle et de natures diverses, par exemple informations sur la position, sur l'orientation, et sur la forme de cet objet [Bouzig 2006].

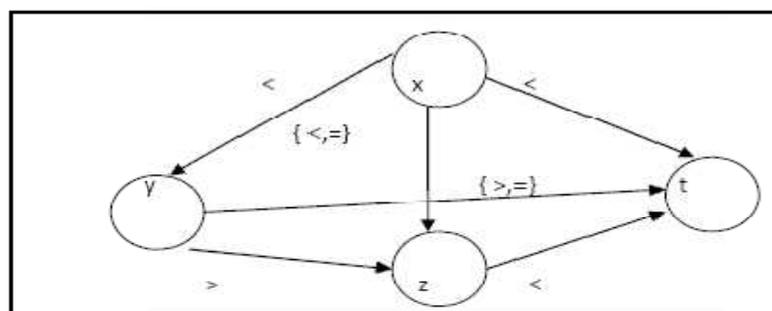


Figure 1.4 : un R.C.T.Q basé points

Les disciplines de l'IA ayant contribué au développement du raisonnement sur les informations spatiales sont la robotique, la vision artificielle et les systèmes d'informations géographiques qui utilisent des modèles quantitatifs classiques tels la topologie et la géométrie euclidienne. Ce n'est que vers les débuts des années 90 que des travaux concernant spécifiquement le raisonnement spatial que se sont diversifiés. Ces travaux concernent aussi bien les méthodes quantitatives que les méthodes qualitatives.

Le problème auquel se confrontaient les recherches en raisonnement spatial qualitatif est la granularité de l'entité spatiale. Faut-il considérer le point, la ligne ou la région en tant qu'entité spatiale? De même, quelle dimension doit-elle avoir cette entité? Une seule dimension, deux ou trois dimensions.

Des travaux de recherche ont abouti à construire des cadres formels pour la représentation et le raisonnement sur l'information spatiale selon que cette information décrit la position, l'orientation,, ou la topologie des ces entités. Nous allons décrire ci-dessous ces cadres formels d'une manière succincte.

### - Méréologie

La méréologie est une théorie générale qui sert à modéliser des propriétés abstraites sur des entités de même niveau ontologique. La relation de méréologie  $P$  vérifie les axiomes suivants pour toutes entités  $x, y, z$ .

- a)  $P(x, x)$  -----  $p(1)$
- b)  $P(x, y)$  et  $P(y, x) \rightarrow x = y$  -----  $p(2)$
- c)  $p(x, y)$  et  $P(y, z) \rightarrow P(x, z)$  -----  $p(3)$

Autres notions définies sur cette base :

- partie propre de :

$PP(x, y) =_{df} P(x, y)$  et  $\text{not } P(y, x)$ .

- recouvrement

$O(x, y) =_{df} \exists z (P(z, x) \wedge P(z, y))$

- Méréologie étendue

-  $\text{not } P(x, y) \rightarrow \exists z (P(z, x) \wedge \neg O(z, y))$  .....  $p(4)$

### - Topologie

Etudie la relation de contact et de connexion entre les entités de même ordre ontologique. Contrairement à la théorie de méréologie, elle manipule des entités faites d'un seul morceau.

La théorie topologique de base vérifie les axiomes suivants :

C est la relation topologique x, y et z des entités spatiales

- $C(x, x)$  ----- (C1)
- $C(x, y) \rightarrow C(y, x)$  ----- (C2)
- $P(x, y) \rightarrow \forall z (C(z, x) \rightarrow C(z, y))$  ----- (C3)

Théorie alternative : une solution alternative est de n'avoir que C comme primitive :, remplacer (c3) par c3').

Quelque soit z,  $((C(z, x) \leftrightarrow C(z, y)) \rightarrow x = y)$  ----- (C3')

#### - **Méré topologie**

Combinaison entre les axiomes de la théorie de méréologie et ceux de la topologie. Par exemple la méréotopologie MT est définie par les axiomes : P1-P3 auxquels on ajoute les axiomes C1-C3.

Ces théories ont servi en tant que bases mathématiques solides pour la définition d'un formalisme de représentation et de raisonnement sur les relations spatiales entre régions appelé Region Connection Calculus (RCC) [Cohen et al 1996]. Par exemple RCC8 décrit d'une manière abstraite les relations topologiques binaires entre les régions d'un espace topologique. Ces relations sont au nombre de huit (08) (Voir Fig. 1.5) qui sont : disconnected (DC), externally connected (EC), equal (EQ), partially overlapping (PO), tangential proper part (TPP), tangential proper part inverse (TPPi), non-tangential proper part (NTPP), non-tangential proper part inverse (NTPPi).

A partir de ces relations de base, des combinaisons de relations peuvent être construites par exemple la relation partie propre (PP) est l'union des relations TPP et NTPP. [Renz 02]

Une table de composition de ces relations 8\*8 est utilisée pour raisonner sur les relations topologiques, comme le montre l'exemple suivant que je préfère le garder dans sa langue d'origine.

#### **Exemple 1.10:** [Renz 2002]

“Two houses are connected via a road. Each house is located on an own property. The first house possibly touches the boundary of the property. The second one surely does not. What can we infer about the relation of the second property to the road? ”

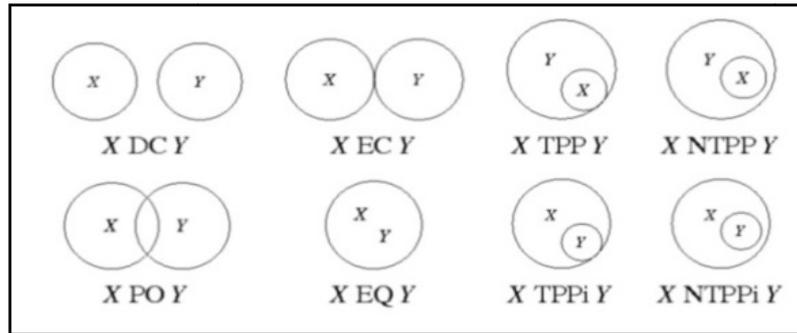


Figure 1.5 : Le R.C.C-8

Traduction du texte de l'exemple dans le formalisme RCC-8 :

House1 *DC* house2  
 House1 {*TPP, NTPP*} property1  
 House1 {*DC, EC*} property2  
 House1 *EC* Road  
 House2 {*DC, EC*} Property1  
 House2 *NTPP* Property2  
 House2 *EC* road  
 Property1 {*DC, EC*} Property2  
 Road {*DC, EC, TPP, TPPi, PO, EQ, NTPP, NTPPi*} Property1  
 Road {*DC, EC, TPP, TPPi, PO, EQ, NTPP, NTPPi*} Property2

Ce texte peut être représenté par un graphe spatial appelé réseau de contraintes spatiales qualitatives (R.C.S.Q) de la figure 1.6.

En utilisant la table de composition 8\*8 et l'algorithme de consistance de chemin, on peut inférer ce qui suit : Road { *PO, EC* } property1 ; Road { *PO, TPP* } Property2.

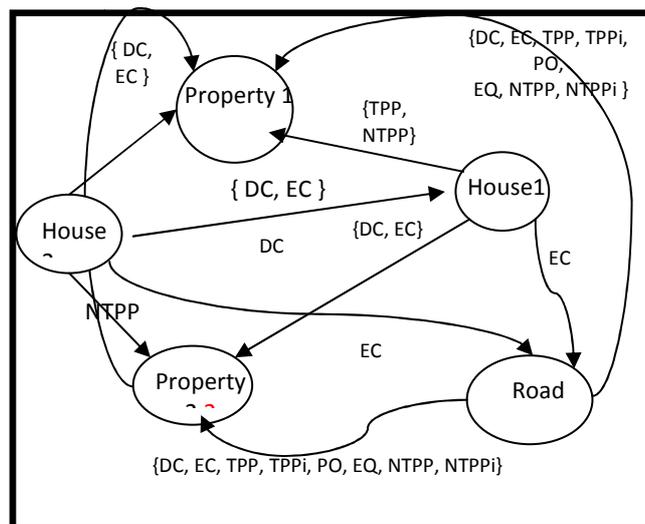


Figure 1.6 Réseau de contraintes spatiales qualitatives

### 1.2.3.3 Le raisonnement qualitatif causal

Le raisonnement causal est une faculté cognitive de sens commun permettant de trouver les liens de causalités et leur poids entre les entités-causes et les entités-effets d'un système donné. L'objectif de ce type de raisonnement est de se donner le moyen pour expliquer et prédire les comportements du système étudié dans le but de pouvoir prendre les meilleures décisions. Pour simuler ce type de raisonnement, une algèbre causale est utilisée conjointement avec un modèle graphique appelé la carte cognitive ou tout simplement la carte causale.

#### - Définition d'une carte causale.

Une carte causale, est un outil de représentation et de raisonnement sur les connaissances causales, représentée par un graphe orienté, étiqueté et cyclique dont les nœuds représentent des causes ou des effets et dont les arcs représentent des relations causales entre ces nœuds, telles augmente, diminue, favorise, et défavorise. Formellement une carte cognitive est définie par le triplet  $\langle S, A, V \rangle$  /  $X$  : l'ensemble des concepts,  $A : S \times S$  : l'ensemble des arcs et  $V$  : l'ensemble des valeurs causales ( étiquettes ) appartenant à l'ensemble de base  $\{ +, -, 0 \}$ . La sémantique de ces valeurs causales est la suivante :

- + : relation de causalité ou d'influence positive entre deux concepts,
- : relation de causalité ou d'influence négative entre deux concepts,
- 0 : aucune influence entre les deux concepts.

Une carte cognitive présentant des cycles est dite rétroactive, c'est-à-dire avec feedback. Cette propriété est d'une grande importance car elle nous permet de déceler et d'explicitier les comportements causaux de base « cachés » dans la carte.

#### - Historique des cartes causales

En 1948, Edward Tolman [1948] a introduit le concept « Carte Cognitive » dans son papier intitulé « Cognitive maps in mice and men ». Dans son article, il a décrit des expériences faites avec des rats, lesquels sont formés pour suivre un sentier complexe afin d'arriver à une boîte remplie de nourriture. Dans ce contexte l'expression « carte cognitive » est utilisée pour définir la représentation mentale qu'un individu fait de l'environnement spatial, dans lequel il se trouve.

En 1976, Robert Axelrod a décrit les cartes cognitives comme des graphes binaires, reflétant un modèle mental en vue de prédire, comprendre et améliorer les décisions des gens dans le domaine politico-économique. Peu après, en 1986, Kosko [Kosko 1986] a étendu les graphes d'Axelrod au « flou », pour donner naissance aux cartes cognitives floues dont les relations sont graduées dans l'ensemble  $[-1, 1]$ . En 1988, Styblinsky et Meyer ont utilisé les cartes cognitives floues pour l'analyse des circuits électriques. En 1994, Dickerson et Kosko ont proposé l'utilisation des cartes cognitives floues afin d'obtenir une modélisation globale

d'un monde virtuel. En 1999, Styblios et Groupos ont introduit l'utilisation des cartes cognitives floues comme modèle de supervision dans des systèmes complexes.

Enfin, en 2003, W.B.Vasantha Kandasmy et Florentin Smarandache [Vasantha & Florentin 2003] on introduit dans leur livre « Fuzzy cognitive maps and Neutrosopic cognitive maps », la notion des cartes cognitives neutrosophiques, que est une généralisation des cartes cognitives floues où ils ont introduit la relation d'indéterminisme « ? » à l'ensemble de relations causales de base.

#### - Applications des cartes cognitives

Les applications des cartes cognitives sont abondantes et touchent presque tous les domaines. Les premières de ses applications étaient dans le domaine politique où on cherchait à trouver les meilleures stratégies pour résoudre des situations de conflits politiques entre nations. En 1988, Styblinsky et Meyer ont utilisé les cartes cognitives floues pour l'analyse des circuits électriques. En 1994, Dickerson et Kosko ont proposé l'utilisation des cartes cognitives floues afin d'obtenir une modélisation globale d'un monde virtuel. En 1999, Styblios et Groupos ont introduit l'utilisation des cartes cognitives floues comme modèle de supervision dans les systèmes complexes.

#### - Inférence d'une carte cognitive

L'inférence dans une carte causale permet de répondre aux questions « qu'arrivera-t-il si ». qui consiste à examiner l'influence d'un ou plusieurs nœuds par rapport aux autres nœuds de la carte en stimulant la carte d'adjacence ( représentation matricielle de la carte) par le vecteur approprié via une multiplication itérative jusqu'à ce qu'on obtienne un vecteur déjà obtenu dans les itérations précédentes. (Voir l'exemple ci-dessous).

#### Exemple 1.11 :

Soit une carte cognitive (CC) ayant quatre (04) concepts A, B, C et D .représentés par sa matrice d'adjacence ADJ(CC) comme suit :

$$ADJ(CC) =$$

	A	B	C	D
A	Ze	Nz	Ze	Ne
B	Ze	Ze	Ze	Po
C	Ze	Po	Ze	Ze
D	Ze	Ze	Po	Ze

Ze : relation nulle, Nz : relation non nulle, Po : relation positive ; Ne : relation négative.

Stimulons cette matrice par le vecteur  $V_0 = (1, 0,0,0)$  pour déduire l'influence du nœud A sur le reste des nœuds de la carte , comme suit :

$(1, 0, 0, 0) * \text{ADJ}(\text{CC}) = (0, \text{nz}, 0, \text{ne}) = \text{V1}$	1 <sup>ère</sup> itération
$\text{V1} * \text{ADJ}(\text{CC}) = (0, 0, \text{ne}, \text{nz}) = \text{V2}$	2 <sup>ème</sup> itération
$\text{V2} * \text{ADJ}(\text{CC}) = (0, \text{ne}, \text{nz}, 0) = \text{V3}$	3 <sup>ème</sup> itération
$\text{V3} * \text{ADJ}(\text{CC}) = (0, \text{nz}, 0, \text{ne}) = \text{V4} = \text{V1}$	Arrêt.

Interprétation : on dit que le concept A a une influence négative sur le Concept D et une influence non nulle sur le concept B.

### 1.3 Avantages et limites du raisonnement qualitatif

La technologie du raisonnement qualitative offre l'avantage de permettre la modélisation et la prédiction des différents comportements typiques d'un système même en absence d'informations complètes. Ces prédictions sont imprécises mais elles sont correctes. L'ensemble des comportements ou des scénarios fourni par un raisonnement qualitatif est concis est facilement explorable. Ces prédictions peuvent être raffinées par la suite avec l'arrivée de nouvelles informations [Iwazaki 1997].

Par contre, la construction d'un modèle qualitatif n'est pas du tout une chose simple et acquiert une attention particulière à donner aux paramètres qualitatifs entrant dans la description du modèle qualitatif. Un défi auquel est confronté le raisonnement qualitatif est l'amélioration de l'efficacité des techniques y inhérentes. [Iwazaki 1997].

### 1.4 Les domaines d'application du raisonnement qualitatif

Le raisonnement qualitatif a été appliqué dans plusieurs domaines et secteurs, A titre d'exemples, en industrie pour des fins de diagnostic et de conception des systèmes, dans le domaine médical pour faire le diagnostic médical, en écologie pour modéliser étudier le comportement des écosystèmes et la dynamique des populations et en biologie pour étudier le comportement des processus biologiques

### 1.5 Les tendances actuelles de recherche en raisonnement qualitatif

Les axes de recherche actuels en raisonnement qualitatif couvrent les points suivants :

- Recherche d'autres modèles de représentation qualitative des connaissances de sens commun.
- Les méthodes permettant de faire le mapping entre les connaissances quantitatives et les connaissances qualitatives
- Nouvelles approches pour la résolution qualitative des problèmes.

## 1.6 Conclusion

Dans ce premier chapitre, j'ai présenté l'état de l'art des approches de raisonnement qualitatif. Ces approches peuvent être regroupées en trois classes différentes : les approches fondées sur la physique qualitative, les approches fondées sur la logique non classique et les algébriques.

La première classe de ces approches concerne celles fondées sur la physique qualitative et renferme trois approches qui sont l'approche centrée sur les processus, l'approche centrée sur les composants et l'approche centrée sur les contraintes. L'approche centrée sur les processus se base sur une idée intuitive stipulant que n'importe quel variation ou changement dans l'état d'un système n'est que le résultat d'un ou plusieurs processus en interaction. L'approche centrée sur les composants prend en considération la structure du système étudié et les comportements de ses composants et leurs liaisons pour déduire le comportement global du système. Pour obtenir tous les états possibles du système étudié, ces deux approches utilisent le principe de la technique « envisionment » qui génère une structure d'arbre appelée le diagramme d'états. L'approche centrée sur les contraintes quant à elle, modélise le système en un ensemble de contraintes qualitatives définies par des équations différentielles qualitatives sur des paramètres qualitatifs. Pour obtenir les différents comportements du système, une simulation qualitative (QSIM) est appliquée à ces équations après la donnée d'un état initial. Ces approches ont été par la suite comparées (Voir Tableau 1.1) en se servant d'un ensemble de critères permettant de mettre en évidence les caractéristiques de chaque approche.

Les approches qualitatives faisant partie de la deuxième classe d'approche sont basées sur la logique non classique et qui prennent en charge le non complétude des connaissances disponibles. Ces logiques sont non monotones, c'est-à-dire que les connaissances acquises au sujet d'un domaine de discours donné peuvent faire l'objet d'une mise à jour au vu de l'arrivée de nouvelles informations. Parmi ces logiques, on cite la logique de défaut [Reiter 80], la logique modale non monotone [Mc Dermott and Doyle 1980] , logique auto épistémique [Moore 1985]. Le raisonnement par l'argumentation est une approche de raisonnement non monotone utilisée par les êtres humains dans la vie quotidienne et qui permet à deux personnes de raisonner au sujet d'une thèse donnée en utilisant leurs informations conflictuelles, et ce par l'échange d'arguments et de contre arguments. Les modèles logiques de l'argumentation ont été présenté ci dessus [Simari & Loui 1992] [Dung 1993] [Verheij 1996] [Bondarenko et al 1997] [Amgoud & Cayrol 2002] et comparés (Voir Tableau 1.2) et le résultat de comparaison favorise ce dernier modèle car outre le processus d'argumentation proposé, les arguments de ce modèle sont doublement liés par une relation de défaite et une relation de préférences ce qui confère à ce système une meilleure force d'expression.

La dernière classe d'approches qualitatives englobe trois approches basées sur des algèbres qualitatives dont elles tirent le nom d'approches algébriques. Ces dernières sont appelées respectivement le raisonnement qualitatif temporel [Allen 1983][Vilain & Kautz

1986] le raisonnement qualitatif spatial [Cohen et al. 1996] et le raisonnement qualitatif causal [Axelrod 1976][Kosko 1986][Florentin 2003]. Les algèbres qualitatives sont définies par un ensemble de relations de base et un ensemble d'opérations sur ces relations. Selon le cas, ces relations peuvent être de type temporel, spatial ou causal. Nous avons montré que dans le cas d'un raisonnement qualitatif spatial ou temporel, on est finalement ramené à résoudre un problème de satisfaction de contraintes de telle sorte que les solutions qualitatives obtenues représentent les différents scénarios ou comportements du système étudié. Les cartes cognitives ou causales quant à elles, représentent le modèle de représentation et de raisonnement sur les Informations causales. Ces dernières sont représentées graphiquement par un ensemble de concepts liés par des relations causales de type « augmente », « diminue », « sans effet » et « indéterminée » symbolisées respectivement par  $\{+, -, 0, ?\}$ . Grâce à une algèbre causale, nous pouvons raisonner sur ces informations pour déduire les comportements typiques du système étudié...

## CHAPITRE 2 : LES SYSTEMES COMPLEXES

### 2.1 Introduction

La disponibilité des ordinateurs peu coûteux ayant une grande capacité de calcul ont suscité de l'intérêt pour la modélisation et la mise en œuvre des systèmes complexes. Le corps humain, l'économie mondiale, les marchés financiers ; l'écosystème, le réseau mondiale internet sont des exemples de systèmes complexes. Ces systèmes présentent généralement des propriétés et des comportements émergents résultants des interactions locales entre leurs entités constituantes. Ces comportements émergents ne sont pas prévisibles car les relations entre les entités de ces systèmes sont non linéaires et que la structure de telles interactions change dynamiquement grâce à l'auto-organisation.

La simulation de ces systèmes complexes par un modèle mathématique ou autre est une technique visant à comprendre ces systèmes. L'intérêt de ces simulations est de pouvoir considérer aussi bien des paramètres quantitatifs que qualitatifs [Ferber 2006].

Ce chapitre est consacré à l'étude des systèmes complexes. Nous commençons par la présentation de cette discipline dans la section 2.2 où nous donnons deux définitions complémentaires d'un système complexe. Dans la section 2, 3 nous présenterons l'état de l'art des approches de modélisation des systèmes complexes existantes. La section 2.4 relate les travaux menés jusqu'à présent dans le domaine. Enfin nous clôturons ce chapitre par une conclusion.

### 2.2 Présentation et définition des systèmes complexes

Les systèmes sont généralement constitués d'un grand nombre d'entités en interaction. De tels systèmes sont qualifiés de complexes lorsque, si malgré que les interactions entre ces entités sont bien comprises, il en résulte un comportement global émergent qui n'était pas directement prédictible à partir des règles d'interaction entre ces entités. En d'autres termes, lorsque l'on veut modéliser ce système, on conçoit un certain nombre de règles d'évolution, puis l'on simule le système en itérant ces règles jusqu'à obtenir un résultat structuré. Cette limitation de ne pouvoir prédire ce comportement global émergent n'est pas due à la longueur des calculs, ni d'un comportement aléatoire, ou de la sensibilité aux conditions initiales mais de l'impossibilité de modéliser le système par des équations solvables et prédictives. Un système complexe est caractérisé par le nombre de paramètres très important entrant dans sa définition et le degré d'influence de chacun sur le comportement global de ce système. Une méthode généralement utilisée pour prévoir ce comportement global émergent est la simulation qui prend en compte tous ces paramètres. Le tableau suivant (Tableau 2.1) montre quelques exemples des systèmes complexes.

	<b>Entités</b>	<b>Interactions</b>	<b>Comportement émergent</b>
S.C1	gènes	activation et inhibitions	Points fixes
S.C2	neurones	impulsions électriques	Pensée
S.C3	fourmis	phéromones	Fourmilière
S.C4	investisseurs	transactions	bulle financière

Tableau 2.1 Exemples de Systèmes Complexes

### **- Définition d'un système complexe**

Il n'existe pas une définition précise largement acceptée par la communauté informatique. Nous allons donner deux définitions, l'une se base sur la structure ou la constitution du système et l'autre se base sur son comportement. Prédire le comportement d'un système à partir de sa structure est d'ailleurs un des problèmes encore ouvert en intelligence artificielle.

### **- Définition basée sur la Constitution du système**

Nous considérons qu'un système est complexe s'il a les caractéristiques suivantes :

- Il est composé d'un grand nombre, fini, d'entités. Typiquement entre 50 et un million.
- Les interactions sont locales, de même que la plupart des informations, il y a peu d'organisation centrale.
- Le graphe d'interaction est non trivial et les interactions sont non linéaires
- Il y a des boucles de rétroaction (en anglais feedback) : l'état d'une entité a une influence sur l'état d'autres entités, qui en retour ont une influence sur la première

Les entités peuvent être elles aussi des systèmes complexes. On parle alors de systèmes complexes récurrents.

- Un système complexe a généralement toutes ses entités interchangeable et définies à partir de quelques paramètres.

Dans un système complexe, l'environnement est extérieur et a une influence à travers les frontières.

Les systèmes complexes s'auto organisent, c'est-à-dire que le nombre d'entités et les relations entre ces entités peuvent changer dynamiquement pour faire face à une situation donnée.

### **Définition basée sur le comportement du système**

On peut aussi définir les systèmes complexes d'après leurs comportements. Un système est complexe s'il présente la plupart des comportements suivants.

– Il y a auto-organisation et émergence de propriétés ou de structures cohérentes, apparition de motifs. Le système a donc une mémoire de son évolution, son état influence les états futurs.

– Le système présente une robustesse locale et fragilité à moyenne échelle : puisqu'il y a de nombreux liens (éventuellement créés ou remaniés par le système lui même), si un élément est affecté par un événement extérieur, ses voisins le seront aussi. Il s'ensuit que le système est souvent plus robuste à une petite perturbation locale qu'il ne le serait sans les liens. Mais c'est grâce à cette même interdépendance des constituants que modifier globalement le système (et donc potentiellement le contrôler) peut être fait grâce à une perturbation moins grande que dans le système sans liens. Bien cibler cette perturbation est cependant très difficile. Les virus (issus d'une longue sélection naturelle) sont un bon exemple : avec une dizaine de gènes, un virus est capable de modifier profondément (jusqu'à la mort...) un organisme de plusieurs dizaines de milliers de gènes, et ce même avec une faible concentration au départ.

– On observe une brisure de symétrie : la connaissance d'une partie du système ne permet pas d'affirmer que le reste du système est en moyenne dans le même état.

– Plusieurs comportements possibles sont en compétition, certains sont simples, d'autres chaotiques (désordonnés). C'est souvent lorsque le système est à la frontière entre les deux et alterne ces deux types de comportement qu'il est le plus intéressant.

## **2.3 Modélisation des systèmes complexes**

Modéliser un système signifie en faire une représentation dans l'objectif de prédire ses différents comportements. Les modèles permettent d'apprendre des choses sur le monde et d'explorer des hypothèses et d'étudier des questions sans expérimenter directement sur le monde réel.

### **2.3.1 Approche basée sur les automates cellulaires**

Les automates cellulaires sont utilisés comme modèle de système complexe. C'est un modèle simple souple et formalisé. Informellement, un automate cellulaire consiste en une grille de cellules, chaque cellule pouvant être dans un nombre fini d'états (souvent représentés par des couleurs). À chaque pas de temps, les cellules calculent leur nouvel état en fonction des états des cellules voisines. Toutes les cellules utilisent la même fonction et sont mises à jour en même temps. Les automates cellulaires ont été tout d'abord, et sont encore aujourd'hui, très étudiés comme modèle de calcul (voir par exemple [Kari 2005] qui passe en revue cette approche des automates cellulaires). La question fondamentale de ce domaine est « que peut-on calculer avec un automate cellulaire ? ». Plusieurs problèmes ont été ainsi posés et résolus : Firing Squad (Synchronisation d'une ligne de cellules à partir des interactions locales), calcul des nombres premiers, construction de diverses formes géométriques, simulation d'autres modèles de calcul... De façon complémentaire, des résultats d'indécidabilité ont été montrés, ainsi par exemple il ne peut pas exister d'algorithme pour décider dans tous les cas d'une

propriété aussi simple que la nilpotence (i.e. si le calcul global mène toujours à une configuration où toutes les cellules sont dans le même état).

Dans une approche de type système dynamique, on a cherché à classifier les automates cellulaires, d'abord par une simple observation visuelle [Wolfram 1984] puis de façon formelle. Une revue de l'approche des automates cellulaires en tant que systèmes dynamiques est disponible dans [Blanchard, Kurka, et Maass 1997].

Les automates cellulaires sont étudiés de façon plus appliquée comme paradigme de modélisation pour réaliser des simulations en physique et plus particulièrement dans le cadre de la dynamique des fluides. Des extensions des automates cellulaires pour incorporer non plus des états discrets mais des états continus ont été utilisées dans le cadre de la dynamique des populations en écologie, en biologie pour modéliser les processus biologiques. L'une des applications la plus attractive et celle du jeu de la vie dans le domaine de la vie artificielle [Langton 1988]. Comme règle générale, nous pouvons dire que les automates cellulaires sont applicables dans les domaines où les entités sont caractérisées par des règles locales et où entre lesquelles (les entités) existent des interactions de voisinage.

### **2.3.2 Approche basée agents**

Dans cette approche le système complexe est modélisé par un ensemble d'agents autonomes en interaction et de granularité faible. Pour la simulation multi agents deux aspects importants sont à considérer : la discrétisation du temps et l'ordre d'activation des agents.

Deux types d'activation seront à considérer, l'une synchrone et l'autre asynchrone. Une activation synchrone des agents permet d'approcher réellement la nature du système modélisé sur une machine séquentielle. Le principe de l'approche synchrone est alors de garantir pour un pas de temps donné, que tous les agents perçoivent le même état du monde et décident ensuite de l'action à effectuer. Les actions de tous les agents sont mises en œuvre à la fin du pas de temps, lorsque tous les agents ont été activés. Ainsi, dans l'approche synchrone, un pas de temps correspond à un instant de temps. C'est d'ailleurs la modélisation la plus naturelle du temps au regard de ce que l'on entend par discrétisation temporelle. Il n'y a donc pas de problème d'ordonnement et les agents peuvent être activés dans un ordre quelconque. Cependant, parce que les actions sont mises en œuvre à la fin, il peut être nécessaire d'avoir des politiques de gestion de conflit [Michel 2004]. Suivant le système modélisé et les interactions possibles entre les agents et l'environnement, les conflits entre des actions effectuées par des agents différents sur le même objet du monde peuvent être très difficiles à gérer [Bonneaud 2009]

Dans l'approche asynchrone, les agents perçoivent, décident et agissent chacun à leur tour. Les conflits entre deux actions de deux agents différents sont alors impossibles, car les

agents modifient le monde au moment de leur activation et les agents, activés pas au même temps, perçoivent le monde ainsi modifié. L'approche asynchrone correspond plus à l'idée que le système est fortement distribué et non organisé et synchroniser les agents impliquent déjà un contrôle centralisé sur le monde. Cependant, le pas de temps n'est plus strictement instantané et l'approche asynchrone pose alors le problème de l'ordonnancement des agents [Bonneaud 2009]. Classiquement, les agents sont ordonnancés de manière pseudo-aléatoire, ce qui permet de minimiser les biais que peut entraîner un ordre spécifique d'activation.

Les travaux dans le domaine de la simulation des systèmes complexes sont actuellement en pleine expansion et les applications de simulation développées jusqu'à présent touchent des domaines multiples et variés. Dans ce qui suit, nous allons présenter trois simulateurs appliqués respectivement au domaine médical, en écologique et en biologie.

### **- BIOWAR (propagation des maladies infectieuses)**

Une équipe de chercheurs à l'université de Carnegie Mellon a développé un système de simulation appelé BioWar [Carley, et al 2006] pour examiner l'impact des événements menaçants la vie des populations au niveau des villes. Ce modèle a été utilisé pour étudier la propagation des maladies contagieuses telle l'Anthrax, la Mallpox, et de la grippe et pour comprendre l'efficacité relative de différentes stratégies d'intervention, sous l'hypothèse d'une épidémie de la grippe à Norfolk- Virginie [Lee, et al 2008]. Le scénario de simulation s'est déroulé en Virginie et a mis en jeu 1.6 million d'agents représentant approximativement la population de cette région métropolitaine. L'environnement virtuel comprend des maisons, des écoles, des endroits pour les loisirs et des infrastructures de santé telles les pharmacies, les bureaux du docteur et des salles de secours. Les agents peuvent se localiser dans l'une de ces structures mais peuvent aussi changer librement d'endroits. Chaque étape de la simulation est de période de quatre heures et la simulation s'est étalée sur une année et tournait sur un super ordinateur Cray XD1. L. Chaque agent a été doté d'un ensemble de caractéristiques socio démographiques pour refléter la réalité socio démographique de la population de la région. De même, les caractéristiques de contagion de la grippe ont été prises en considération. Deux cents (200) agents "ont été infectés" et répartis aléatoirement sur la ville. Chaque agent peut changer d'emplacement toutes les quatre heures, c'est-à-dire à chaque étape de la simulation, et choisit d'interagir avec les autres agents ou non. Ils peuvent aussi s'engager dans d'autres activités telles que acheter de médicaments, faire un diagnostic médical chez un « agent Docteur », et peuvent être dans les états suivants : être infecté, être guéri, mort ou être saint. Les agents infectés se transmettent la contagion par interaction. La simulation tient compte également de l'infection aéroportée, portée par les eaux, ou portée par les aliments. Des alertes médicales et/ou des décisions de fermeture de locaux publics (écoles, cafés, restaurants.) peuvent être annoncés à tout moment, permettant aux agents de changer leurs comportements. L'étude des résultats de cette simulation ont permis d'arrêter de stratégies de lutte contre la propagation de l'épidémie.

**- SIMDELTA**

Le simulateur SIMDELTA est le résultat de travail de plusieurs spécialistes (halieutes, écologistes, pédologues, biologistes, anthropologues, etc.) ayant conjugué leurs efforts pendant plusieurs années pour étudier le système de pêche du Delta central du Niger au Mali. Il s'agissait de pouvoir modéliser des informations tant quantitatives (concernant l'évolution des crues du Niger par exemple) que qualitatives (telles que les techniques de pêches). Ce simulateur, réalisé par F. Bousquet et C. Cambier [Bousquet et al. 1993], permet de simuler en même temps la dynamique de population des poissons, en tenant compte des nombreux facteurs biologiques et topologiques qui peuvent affecter son évolution, et la prise de décision des pêcheurs. Les agents sont ici les bancs de poissons et les pêcheurs. Trois types d'agents: composent le système de simulation: les biotopes qui représentent des portions d'environnement, les poissons qui ont un comportement réactif et les pêcheurs qui se comportent comme des agents cognitifs. L'environnement est constitué de biotopes dont les connexions peuvent être modifiées dynamiquement lorsque le niveau de l'eau change. Pour chaque biotope, une fonction de ressource indique la quantité de nourriture disponible pour une population de poissons en fonction du temps. Les bancs de poissons sont des agents qui représentent des ensembles de poissons et dont les paramètres (taille et nombre d'œufs, processus de migration, régime, etc.) caractérisent leur stratégie d'adaptation. Chaque pêcheur est représenté sous la forme d'un agent cognitif et son comportement est décrit par un système à base de connaissances composé d'une base de données qui contient ses croyances et sa mémoire de pêcheur et d'un système de règles qui représente sa stratégie cognitive pour exploiter les biotopes.

Les simulations ont été effectuées pour essayer de comprendre et de prédire la dynamique de la population des poissons sous l'effet d'une activité variable de pêche pouvant être régulées par des stratégies de rationalité économique ou par des stratégies faisant intervenir des règles d'accès à l'espace. Les résultats de simulation fondées sur ces dernières stratégies sont les plus meilleurs. [Bousquet et al. 1993] [Cambier 1994].

**- SIMPOP**

SIMPOP [Bura et al. 1993] est un simulateur basé agents modélisant la dynamique de l'évolution d'un système de villes et, plus particulièrement, la genèse, le développement et la concentration des fonctions urbaines à différents niveaux durant une longue période de temps. L'environnement est représenté par un ensemble de "places" de tailles et de formes variées essentiellement des carrées et des hexagones). Elles sont caractérisées par leur nature (plaines, montagnes, mer, marais), leurs ressources naturelles (agriculture, pêche, minerais) ainsi que par des éléments divers tels que la présence d'une voie de communication (rivière, route, etc.). Les ressources correspondent au potentiel qu'une population peut exploiter, la productivité dépendant de facteurs tels que les possibilités techniques ou l'activité d'un peuplement voisin. Chaque place comporte un agent "peuplement" que l'on appelle communément une ville (en

fait ces “villes” peuvent aller d’un simple hameau à une grande métropole). Les villes sont caractérisées par la taille de leur population, leur richesse économique et les fonctions qu’elles possèdent (agriculture, économie, industrie, administration de ses habitants, ces derniers étant représentés par des fonctions économiques correspondant aux principaux groupes sociaux. Par exemple, les hameaux voient la majeure partie de leur population associée à la fonction agricole. Les villes sont donc les principaux agents du système et, du fait qu’elles ne bougent pas, les interactions s’effectuent essentiellement par des transferts de biens, de monnaies, de services et d’habitants qui s’expriment sous la forme de mécanismes d’offres et de demandes entre villes. De ce fait, certaines villes tendent à grossir et de nombreux phénomènes locaux viennent renforcer les différences apparaissant entre elles de manière à former ce que l’on nomme une “hiérarchie” de villes, portant sur leur taille et leur richesse.

### - MANTA

MANTA [Drogoul 1993] est un système multi agents de simulation des comportements d’une colonie de fourmis. Les agents qui composent le système sont de trois groupes : les travailleurs, les reines et les males. Ces agents sont réactifs et communiquent par envoi de signaux. Les agents fourmis sont disposés à exécuter plusieurs tâches compétitives de nature primitives. Chaque agent ne reçoit que les stimuli correspondants à ses tâches. Une tâche donnée d’un agent quelconque de la colonie est activée si la force du signal est au dessus du seuil d’activation de la même tâche. Si deux tâches compétitives d’un même agent sont requises en même temps alors la plus prioritaire ou celle dont le stimulus est le plus fort qui va être activée et exécutée. Les simulations par MANTA ont montré et ont confirmé qu’une société d’agents basée sur la répartition de tâches élémentaires induisant des comportements élémentaires présente des qualités d’adaptation à l’environnement par le biais de la prise de décision collective et distribuée. La synergie de ces comportements individuels permet à cette société de survivre sans avoir besoin à un organe de contrôle centralisé. Pour avoir de plus amples informations concernant ce simulateur, consultez [Drogoul 1993].

### 2.3.3 Approche basée sur le raisonnement qualitatif

Cette approche utilise comme technique de base le raisonnement qualitatif pour modéliser et simuler qualitativement les systèmes complexes. Les modèles qualitatifs visent à capturer le comportement fondamental d’un système dans un modèle traitable par un ordinateur, tout en supprimant une grande partie du détail. Ces modèles sont définis par les expressions qualitatives, abstrait, imprécis, vague, et inachevé. Des méthodes telles que l’abstraction et l’approximation sont souvent employées pour établir des modèles basés sur des aspects plutôt qualitatifs que quantitatifs (numériques)

Dans beaucoup de problèmes de technologie, les ingénieurs ne disposent pas de toute l’information détaillée concernant le problème à résoudre. Ceci rend impossible la construction d’un modèle quantitatif détaillé. Un des avantages principaux du raisonnement qualitatif est

qu'il est possible d'employer une approche qui fournit une solution approximative à un problème donné quand toute l'information précise détaillée nécessaire est indisponible.

Deux simulateurs qualitatifs existent actuellement : QSIM (Qualitative Simulation) [Kuipers 1986] basée sur l'approche centrée sur les contraintes et GARP3 [Bredeweg et al. 2009] basée sur l'approche centrée sur les processus [Forbus 1984]. A titre d'exemple, nous allons présenter dans ce qui suit le simulateur GARP3.

### - GARP 3

GARP3 [Bredeweg et al. 2009] est un environnement de raisonnement qualitatif indépendant du domaine étudié et utilise les concepts définis dans la théorie des processus qualitatifs, Q.P.T (Qualitative Process Theory) [Forbus 1984]. La description du modèle qualitatif sous GARP3 peut être introduite manuellement en utilisant un éditeur de textes ou par le biais d'un outil de modélisation graphique. Pour ce faire, des ingrédients du modèle doivent être définis. Ces ingrédients sont divisés en quatre (04) catégories.

- **Fragments du modèle (Agrégats)**

Sont des modèles partiels composés de plusieurs ingrédients. Les fragments modèles ont la forme d'une règle. Ceci signifie que des ingrédients modèles sont incorporés dans les conditions ou les conséquences des règles. En outre, des fils de fragments modèles peuvent être faites, ce qui enrichi le fragment de modèle parent avec de nouveaux ingrédients. D'une manière générale, Les fragments modèles décrivent une partie de la structure et du comportement d'un système. L'état actuel du système, dit scénario contient les ingrédients de conséquence. Il y a trois types différents de fragments modèles :

- **Fragments statiques**, utilisés pour décrire les parties de la structure du système. Tous les ingrédients peuvent se produire excepté des agents et des influences.

- **Fragments processus** : utilisé pour décrire les processus intervenant dans le système, Contient au minimum une influence, mais pas d'agents..

- **Fragments Agents** : utilisés pour décrire les influences des entités externes sur le système appelées « agents ». Ils peuvent contenir une ou plusieurs influences.

- **Scénarios** : décrivent l'état actuel du système, et peuvent être composés de tous les ingrédients qui peuvent être utilisés comme conditions dans les fragments modèles. Des scénarii sont employés comme entrée pour le simulateur qualitatif et sont considérés comme état de début du système à partir duquel sont obtenus les comportements du système sous forme d'un graphe d'états.

- **Les ingrédients décrivant la structure du modèle.**

La deuxième catégorie des ingrédients des modèles les ingrédients ayant trait à la structure du système.

- **Entité** : Objets physiques ou abstraits qui jouent un rôle dans le système. Ces entités sont organisées dans une hiérarchie permettant ainsi aux concepts les plus généraux d'être utilisés dans des fragments du modèle.
- **Les configurations** sont employées pour modéliser les relations entre les entités.
- **Les attributs** sont des propriétés des entités qui ne varient pas pendant la simulation. Pour cette raison elles n'ont pas un espace de quantité associé
- **Les agents** sont un type particulier d'entités. Elles sont employées pour modéliser des entités externes ayant leurs influences sur le système.

- **Hypothèses (Assumptions) sur le modèle**

Un ingrédient modèle qui ne décrit ni les aspects liés à la structure du système ni à son comportement sont dites les hypothèses ou les contraintes car elles contraignent le comportement du modèle.

- **Les ingrédients relatifs au comportement du modèle**

La dernière catégorie des ingrédients du modèle sont les dépendances comportementales. Ces relations sont employées pour indiquer des informations sur des quantités, des dérivés et les valeurs des quantités d'espace Les proportionnalités et les influences désignées souvent sous le nom des relations causales.

- **Les quantités** : sont les objets variables des entités. Par exemple, le volume, la taille et la pression sont des quantités d'un liquide contenu. Chaque quantité a deux espaces de quantités associés un espace de quantité à définir et un autre espace de quantité par défaut {décroit, zéro, croît} pour la dérivée.
- **Les espaces de quantité** : indiquent un intervalle de valeurs qualitatives qu'une quantité ou une dérivée peut avoir. Les valeurs qualitatives dans un espace de quantité forment un ordre total. Chaque valeur qualitative est un point ou un intervalle, Un exemple d'un espace de quantité pour la taille d'un liquide contenu dans un récipient est {zéro, positif, pleins}.
- **Les inégalités** ( $<, =, >$ ) peuvent être employées dans trois manières différentes. Premièrement, elles sont employées pour indiquer des différences relatives dans les valeurs des quantités (relation entre quantités) ou des dérivés (relation entre dérivés). Deuxièmement, des

inégalités peuvent être indiquées entre les points d'espaces de quantité appartenant à différentes quantités, pour indiquer, par exemple, qu'une valeur qualitative est plus grande que l'autre.

- **Les valeurs :** sont des abréviations pour des relations d'égalité entre une quantité et une valeur qualitative dans son espace de quantité.
- **Les influences,** sont des relations entre deux quantités. On distingue les influences négatives noté par I- et les influences positives notées par I+. Deux quantités liées par la relation I+ par A I+ B alors B augmente si A est positif. Tandis que si A I- B, alors B diminue si A est négative. Les influences sont aussi connues sous le nom de confluences directes.
- **Les proportionnalités,** sont des relations également orientée entre deux quantités, mais ne causent pas le changement aux quantités directement comme font les influences. Au lieu de cela, les proportionnalités propagent les effets d'une variation d'une quantité à une autre liées par cette relation. Par exemple A P+ B fait que B augmente si A augmente. Tandis que la relation A P- B fait que B diminue si A diminue.
- **Plus/Min :** ces relations sont employées pour calculer la somme ou la différence de deux quantités ou de deux dérivés. La relation de plus/min peut avoir une relation (d'inégalité) elle-même. Habituellement une relation d'inégalité est indiquée entre la relation de plus/min et une quantité pour indiquer que la quantité est (in)égal au résultat du calcul.
- **La simulation qualitative**

Après avoir décrit les ingrédients du modèle qualitatif, la simulation de ce dernier est effectué (Voir Figure 2.1) par l'introduction de tous les fragments du modèle conjointement avec les conditions initiales représentées par scénario. Le résultat de la simulation qualitative est un graphe d'états.

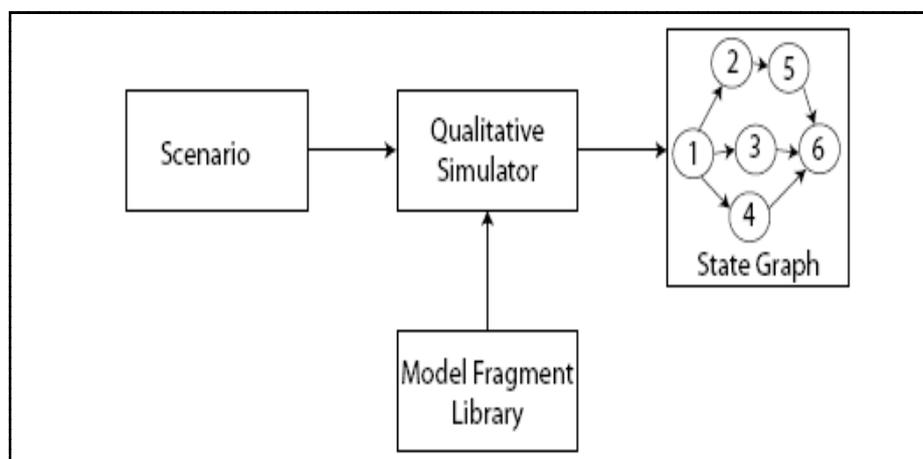


Figure 2.1 La simulation qualitative par GARP3.

## 2.4 Conclusion

Les systèmes complexes est une branche de l'I.A reconnue en tant qu'une science en 1987 par Stefen Wolfram qui a crée la première revue spécialisée en la matière intitulée « Complex Systems ». Un système complexe est fait d'un nombre important d'entités ayant des comportements primitifs où chaque entité interagit avec les entités voisines selon des règles simples. Les systèmes complexes, depuis la cellule jusqu'à l'écosphère, résultent de processus d'évolution et d'adaptation. Ils présentent des propriétés d'auto organisation et d'émergence:

L'auto organisation est une forme d'adaptation permettant au système complexe de modifier la structure et les poids de son réseau de liaisons faisant émerger des formes organisées au niveau macroscopique. Cette propriété d'émergence ne peut être prévue par les méthodes mathématiques classiques. Pour comprendre et élucider les comportements de tels systèmes, la technique de modélisation et de simulation du modèle est appliquée mettant en jeu de nouvelles approches de l'I.A.

L'approche multi agents pour les systèmes complexes offre un grand avantage pour modéliser et simuler ces systèmes car le paradigme agent dispose de tous les concepts (objet est une entité, comportement réactifs, interaction, adaptation, auto organisation) nécessaires pour prendre en charge cette classe de systèmes. Plusieurs simulateurs basés agents ont été développés par les équipes de recherche [Bousquet et al. 1993] [Bura et al. 1993] [Drogoul 1993][Carley, et al 2006].

L'approche basée sur le raisonnement qualitatif offre l'avantage de pouvoir modéliser parfaitement les systèmes complexes car les informations quantitatives caractérisant ces systèmes sont souvent incomplètes et imprécises. Deux plates formes de simulation qualitatives ont été développées par la communauté des chercheurs en raisonnement qualitatif qui sont QSIM [Kuipers 1986] et GARP3 [Bredeweg et al. 2009]. Ces deux simulateurs ont été utilisés pour l'étude de systèmes complexes dans des domaines divers, nous citons à titre d'exemples [Heidtke, et al. 1998][Jong, et al 01][Bredeweg & Salles 2005].

Pour conclure ce chapitre, nous pouvons dire que l'approche basée agents et l'approche basée sur le raisonnement qualitatif offrent toutes les deux des avantages pour modéliser et simuler les systèmes complexes et l'utilisation combinée de ces deux approches permettra de mieux appréhender cette classe de systèmes et aura certainement son impact positif sur l'évolution de la recherche dans ce domaine.

## CHAPITRE 3 : LES SYSTEMES MULTI - AGENTS

### 3.1 Introduction

Initialement, le domaine de l'Intelligence Artificielle (I.A) cherchait surtout à décrire et à résoudre des problèmes complexes identifiés par des experts. Dans ce domaine, il est possible de construire des programmes informatiques, capables d'exécuter un nombre important de tâches en centralisant « l'intelligence » au sein d'un système unique [Erceau et Ferber, 1991].

Il est cependant difficile d'entrer dans une même « base », les connaissances, les compétences d'individus totalement différents qui communiquent entre eux.

L'apport de l'Intelligence Artificielle Distribuée (I.A.D) permet de « distribuer l'intelligence » entre plusieurs agents. Comme le souligne Bond et Gasser [Bond et Gasser, 1988]. L'intelligence Artificielle Distribuée présente les avantages suivants :

- L'IAD est bien adaptée à la distribution de problèmes spatiaux, logiques...
- L'héritage de modules permet aux différentes parties du système de développer de façon indépendante un système continu et extensible,
- Les processus distribués entre différents ordinateurs augmentent la vitesse de calcul et de raisonnement,
- Le contrôle du processus local peut être isolé ou séparé du système,
- Dans certain cas, les systèmes confèrent aux agents individuels des ressources limitées pour résoudre les problèmes et la coopération et la coordination sont essentielles à la résolution de ces problèmes.

Une des branches issue de ce domaine, les Systèmes Multi-Agents (S.M.A), permet d'introduire dans un système, un ensemble d'individus (ou agents) dotés de connaissances, d'intentions et de capacité d'évolution différente. Ces agents sont capables d'interagir entre eux.

« Dans le cadre général des sciences de la complexité, qui propose de reposer les questions scientifiques par l'étude des interactions entre entités élémentaires et de leurs organisations, les chercheurs en S.M.A vont participer à cette dynamique dans un double mouvement. D'une part, les SMA vont constituer une méthode pour catalyser la reformulation de certaines questions en sciences sociales et en sciences de la nature. D'autre part, les chercheurs informaticiens vont puiser dans cette interaction les éléments pour mieux préciser les questions et proposer de nouveaux concepts dans leur discipline. » [Bousquet et Le Page, 2001]

D'un point de vue informatique, l'approche multi-agents peut être considérée comme une évolution du paradigme orienté-objet. Du point de vue conceptuel, un objet est simplement une structure de données à laquelle sont associées des fonctions. Les agents sont des entités autonomes, ce qui signifie que leur comportement ne dépend pas d'une pression extérieure, contrairement aux objets. (Figure 3.1)

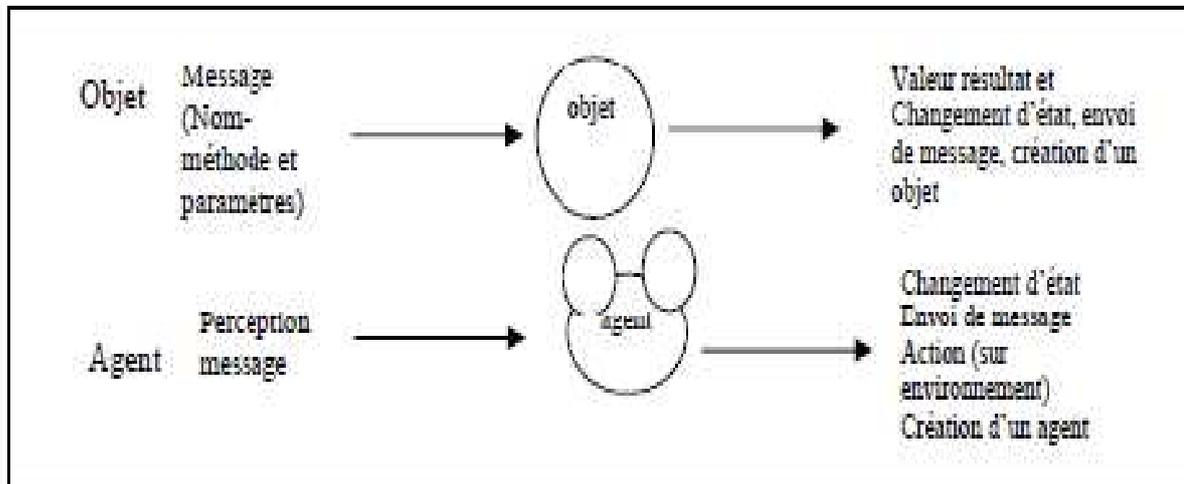


Figure 3.1 : Différences entre objets et agents [Briot, 2002]

D'un point de vue transdisciplinaire, le concept d'agent a fait l'objet d'études depuis plusieurs années dans des disciplines telles que : l'écologie, l'éthologie, la psychologie, la sociologie et l'économie.

### 3.2 Les Agents

Il existe un grand nombre de typologies d'agents [Ferber, 1995] [Chaib-Draa, 1996] [Wooldridge et Jennings, 1994] mais celle la plus répandue est fondée sur le processus de prise de décision de l'agent. Elle distingue trois types d'agents : réactifs, cognitifs ou délibératifs et hybrides.

Pour un certains nombre d'auteurs [Maes, 1994] [O'Hare et Jennings, 1994] [Weiss, 1999], un agent est défini comme étant une entité logicielle d'un système informatique qui possède les caractéristiques suivantes :

- **Autonomie** : les agents contrôlent leurs actions et leurs états internes. Le système dans son ensemble est capable de réagir sans l'intervention d'un humain ou d'un autre agent. Il n'y a pas de définition unique du terme agent, par contre, il y a un consensus général pour considérer l'autonomie comme notion centrale de l'agent.
- **Flexibilité** : le système doit être un système dans lequel (i) les agents perçoivent l'environnement et peuvent répondre dans le temps requis aux changements que celui-ci peut entraîner sur les agents (ii) les agents prennent en considération leur comportement général pour permettre de prendre les initiatives appropriées aux changements de l'environnement (iii)

les agents interagissent avec les autres agents afin d'accomplir leurs buts. Pour cela les agents ont donc les caractéristiques suivantes :

- **Réactivité** : ils perçoivent leur environnement et réagissent aux changements qui s'y produisent dans le temps requis ;
- **Proactivité** : ils exhibent un comportement proactif et opportuniste pour ne pas agir uniquement par réaction à leur environnement mais prendre des initiatives selon leurs buts individuels ;
- **Sociabilité** : ils sont capables d'interagir les uns avec les autres quand la situation l'exige afin d'accomplir leurs tâches ou d'aider les autres agents à accomplir leurs buts.

Jacques Ferber propose la définition suivante :

«Une entité physique ou abstraite qui est capable d'agir sur elle-même et son environnement, qui dispose d'une représentation partielle de son environnement, et qui, dans un Système Multi-Agent, peut communiquer avec d'autres agents, et dont le comportement est la conséquence de ses observations, de sa connaissance et des interactions avec les autres agents ». [Ferber, 1995]

Pour Ferber [Ferber, 1995], la distinction cognitif/réactif « définit un axe pratique d'évaluation de la capacité des agents à accomplir individuellement des tâches complexes et à planifier leurs actions. »

### 3.2.1 Structure conceptuelle des agents

#### 3.2.1.1 Les agents avec états

Les agents avec états, comme leur nom l'indique, maintiennent un état interne qui mémorise la séquence des perceptions de l'agent et, dans certains cas, les actions effectuées par l'agent. L'agent a besoin de maintenir cet état interne pour faire la distinction entre des états de l'environnement qui lui semblent identiques selon la perception qu'il en a, mais qui sont néanmoins différents. Soit  $S = \{s_1, s_2, \dots\}$  l'ensemble des états internes de l'agent. Dans ce cas, les fonctions de l'agent se modifient de la manière suivante :

- **action** :  $S \rightarrow A$  est maintenant définie sur l'ensemble des états internes de l'agent, ce qui revient à dire que l'agent décide de l'action à effectuer en fonction de son état interne ;
- **suiv** :  $S \times P \rightarrow S$  est une nouvelle fonction qui modélise le changement de l'état interne de l'agent en fonction de ses perceptions et de l'état interne courant ;
- **inter** :  $S \times P \rightarrow I$  et **inter** :  $S \times I \rightarrow A$  sont modifiées pour montrer que la décision sur les interactions avec d'autres agents dépend aussi de l'état interne ;
- **voir** :  $E \rightarrow P$  et **env** :  $E \times A \rightarrow P(E)$  ne sont pas modifiées.

#### 3.2.1.2 Les agents avec buts

Connaître l'état de l'environnement et la séquence des perceptions mémorisées dans l'état interne n'est pas toujours suffisant pour décider quelle est la meilleure action à effectuer à un moment donné. Pour fonctionner d'une manière efficace, l'agent doit essayer d'obtenir le maximum de performance et il doit donc choisir ses actions en conséquence.

Il existe plusieurs façons de définir la mesure de performance. Une première solution est d'indiquer à l'agent ce qu'il doit faire en toute circonstance pour avoir du succès. C'est le cas des agents réactifs, où le concepteur définit les règles condition-action dans chaque cas de perception sur l'environnement. Toutefois, puisque les agents intelligents sont autonomes et proactifs, il serait intéressant de dire à l'agent que ce qu'il faut faire, sans avoir à lui dire exactement comment le faire dans chacune des situations. Dans ce cas, il est possible de définir la mesure de performance de l'agent en fixant des états-buts ou états désirables :

**but** :  $E \rightarrow \{0, 1\}$  est une fonction qui a la valeur 1 pour les états-but et 0 pour les autres.

Si les états-buts sont connus d'avance, on peut définir l'ensemble **B** des buts de l'agent,  $B \subseteq E$ .

Par exemple, un robot qui ramasse des objets sur une surface pour les transporter à la base sait que l'état-but est l'état "parvenir à la base avec le maximum d'objets". Si les états-buts ne sont pas connus mais que l'on connaît uniquement certaines propriétés désirables de ces états, alors la fonction **but** fait l'évaluation de chaque état pour voir si l'état possède ou non ces propriétés.

Par exemple, si le robot ne connaît pas d'avance la position de la base mais qu'il peut la percevoir une fois arrivé près d'elle, le robot doit tester dans chaque position s'il est près de la base ou pas.

Une troisième solution pour définir la mesure de performance de l'agent est décrite dans ce qui suit.

### 3.2.1.3 Agents avec utilité

La mesure de performance d'un agent peut être définie d'une manière plus fine en associant à chaque état de l'ensemble  $E$  une valeur réelle, l'utilité, qui mesure la désirabilité de cet état pour l'agent. On peut voir les états-buts comme un cas particulier des états avec utilité, puisque cela correspond au cas où l'utilité ne peut prendre que les valeurs 1 et 0. Pour associer une utilité aux états, on introduit une nouvelle fonction :

Utilité :  $E \rightarrow R$ , où  $R$  est l'ensemble des nombres réels.

La fonction utilité est mieux adaptée que la fonction but dans deux situations. Premièrement, si l'agent a des buts contradictoires, par exemple l'agent « X » veut aller au cinéma et étudier pour son prochain examen ( 02 buts) , la fonction utilité indique le but à choisir. Deuxièmement, si l'agent agit dans un environnement non déterministe, il n'est pas toujours certain de pouvoir atteindre ses buts. Dans ce cas, l'utilité des buts offre un moyen de rapporter la probabilité de succès à l'importance des buts. En général, dans un environnement

non déterministe, l'agent va souvent avoir besoin de la fonction utilité pour prendre des décisions sur les diverses actions à effectuer tout en sachant que les résultats de ces actions ne sont pas certains.

### 3.2.2 Les agents réactifs

Ces agents dits réactifs ont hérité de la vie artificielle. Ils ont un cycle de type Perception / Action [Wooldridge, 1999] Un agent de type réactif est constitué d'un ensemble de comportements permettant d'accomplir une tâche donnée. Chaque comportement est une machine à états finis qui établit une relation entre une entrée sensorielle et une action en sortie.

L'ensemble des comportements est représenté sous la forme d'une hiérarchie dans laquelle les couches de niveaux inférieurs représentent des comportements moins abstraits, et les couches de niveaux supérieurs des comportements plus abstraits [Chaib -Draa et al. 2001].

Cette approche repose sur le concept d'émergence. Elle avance par ailleurs, l'idée qu'il n'est pas nécessaire que chaque agent soit personnellement « intelligent » pour parvenir à un comportement intelligent de l'ensemble [Wooldridge, 1999].

Un exemple d'agents réactifs est celui du système MANTA (Modeling an ANThill Activity) dû à Drogoul [Drogoul, 1993] pour simuler une communauté de fourmis. « Dans ce système, l'architecture d'un agent comporte les opérateurs de perception, de sélection et d'activation qui manipulent un ensemble de tâches. » [Boissier, 2001].

Ce système a mis en évidence que l'organisation du travail dans une colonie de fourmis peut résulter d'un ensemble d'interactions et de contrôles locaux, sans qu'il soit nécessaire de faire intervenir une quelconque entité centrale de régulation. Les performances adaptatives de la société se présentent comme issue de la confrontation des comportements élémentaires de chacun de ses membres [Ferber, 1995].

Les hypothèses principales pour le développement d'agents réactifs sont :

- La manipulation syntaxique prévaut à la représentation symbolique pour modéliser la prise de décision,
- Les idées d'intelligence et de comportement rationnel sont liées à l'environnement de l'agent et non aux agents eux-mêmes,
- Les agents n'ont pas de représentation de leur environnement et des autres agents et se faisant, ils sont incapables de prévoir et d'anticiper les actions à accomplir.
- Individuellement, ils sont très faibles, mais leur force vient de leur capacité à constituer des colonies capables de s'adapter à leur environnement. Ils peuvent faire face à des tâches

complexes et peuvent rivaliser en termes de performance avec des agents plus sophistiqués [Ferber, 1995].

Cette conception fait toutefois l'objet de critiques que Chaib-Draa et al. [Chaib-Draa et al., 2001] formulent ainsi:

- Si les agents ne possèdent pas de modèle de leur environnement, ils doivent posséder suffisamment d'informations locales leur permettant de choisir une action acceptable ;
- Les agents basent leurs décisions sur des informations locales, il est donc difficile de savoir comment ils peuvent tenir compte d'informations non locales ;
- Les agents réactifs offrent peu de mécanismes d'explications et ils ont des difficultés avec les tâches qui nécessitent une connaissance du monde qui ne peut être obtenue que par la mémoire ou le raisonnement et non par la perception.

### **3.2.3 Les agents cognitifs ou délibératifs**

D'une manière générale, ce type d'agents appelés cognitifs ou délibératifs selon les communautés suit un cycle Perception /Délibération/Action comme le montre la Figure 3.3. On dit aussi que ces agents sont intentionnels car ils possèdent des buts et des plans explicites leurs permettant d'accomplir leurs buts. Pomerol [Pomerol, 1997] définit un but comme le résultat de ce qu'un décideur souhaiterait obtenir. Le résultat de la planification est défini alors comme suit ; étant donné un but et un état courant, il s'agit de trouver une séquence d'actions qui permettent d'aller de l'état courant au but. Cette séquence d'actions est appelée plan.

Du fait de la vision partielle de ces agents sur le monde, il est nécessaire de concevoir des liens entre la « réalité » du monde et les bases de connaissances de chacun. Ainsi, l'agent cognitif peut mettre à jour ses connaissances, en fonction, d'une part de l'interaction avec les autres agents et d'autre part à l'environnement. Ceci suppose des communications inter-agents. Il est donc nécessaire de développer des théories dans le domaine de la communication mais aussi des protocoles de coopération entre les agents.

Contrairement aux agents réactifs, ce type d'agents a la capacité de raisonner sur des représentations du monde, de mémoriser des situations, de les analyser, de prévoir des réactions possibles à toute action, d'en tirer des conduites pour les événements futurs et donc de planifier son propre comportement [Ferber, 1995].

Une approche emblématique est celle initiée dans les années 80 [Bratman et al., 1988] qui propose une architecture basée sur trois attitudes : Belief, Desire, Intention ( Figure 3.2). Elle fait depuis l'objet de nombreux travaux [Kumar et Shapiro, 1994] [Rao et Georgeff, 1995] [Busetta et Ramamohanarao, 1998] [d'Inverno et al., 1998]. Le concept Belief correspond à la représentation de l'état interne de l'agent. Il est actualisé à chaque instant en fonction de l'état

de l'environnement. Desire, correspond aux objectifs fixés par l'état interne de l'agent. Intention, correspond aux buts en cours d'achèvement.

### 3.2.4 Les agents hybrides

Certains auteurs [Chaib-Draa, 1996] [Chaib-Draa et al., 2001] [Fischer et al., 1995a] [Georgeff et al., 1999] ont été amenés à proposer des architectures d'agents hybrides pour améliorer en particulier les temps de la décision et temps de l'action. L'agent hybride est alors conçu comme alliant comportement réactif et comportement cognitif

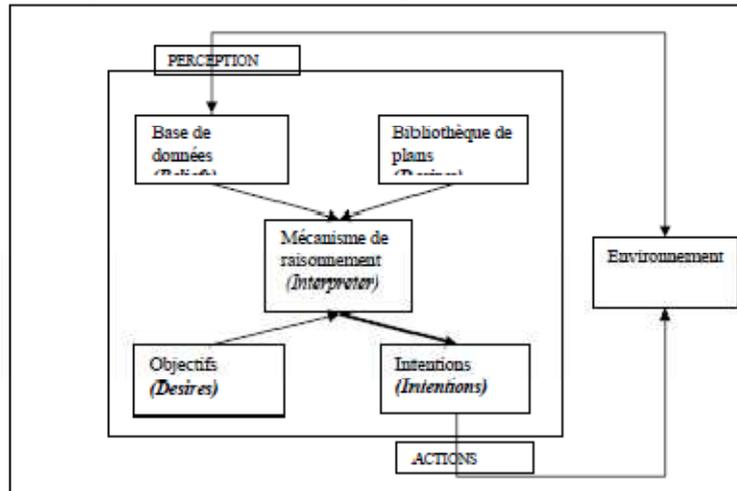


Figure 3.2 : Architecture BDI [Rao et Georgeff, 1995] [Bratman et al. 1988] [d'Inverno et al., 1998]

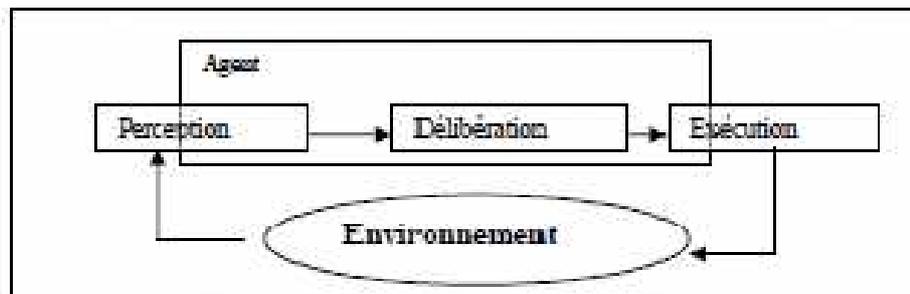


Figure 3.3 : Cycle Perception / Délibération/ Action d'un agent cognitif [Wooldridge, 1999]

Dans ce type d'architecture, les agents sont conçus comme étant composés de niveaux hiérarchiques qui interagissent entre eux. Chaque niveau gère un aspect du comportement de l'agent [O'Hare et Jennings, 1994] :

- Au plus bas niveau de l'architecture, on retrouve habituellement une couche purement réactive qui prend ses décisions en se basant sur des données brutes en provenance de l'environnement.

- La couche intermédiaire fait abstraction des données brutes et travaille plutôt avec une vision qui se situe au niveau des connaissances de l'environnement.

- La couche supérieure se charge des aspects sociaux de l'environnement, c'est-à-dire du raisonnement tenant compte des autres agents [Chaib-Draa, 1996] [Chaib-Draa et al., 2001].

L'architecture InteRRaP (INTEgration of Reactive Behavior And Rational Planning) [Fischer et al., 1995a] qui s'inspire des concepts BDI est un exemple de ce type d'agent. Elle a été utilisée dans des applications telles que le transport routier [Fischer et al., 1995a][Fischer et al., 1995b] et la robotique [d'Inverno et al., 1998].

Un agent InteRRap est constitué (Figure 3.4) (i) d'une base de connaissances, (ii) d'une interface avec le monde extérieur qui lui permet de percevoir et d'agir, (iii) d'une unité de contrôle hiérarchique organisée en trois couches interagissant les unes avec les autres. Ces trois niveaux correspondent à trois niveaux d'abstraction de la base de connaissances de l'agent : le modèle du monde, le modèle mental et le modèle social.

Chaque couche possède des fonctions qui permettent de gérer la reconnaissance d'une situation, activer un but, planifier et ordonnancer [Boissier, 2001] :

- La couche C (comportementale) réagit à l'évolution de l'environnement (condition-action). La reconnaissance des situations et l'activation des buts correspondent aux croyances sur l'état de l'environnement.

- La couche PL (planification locale) sélectionne des plans pouvant faire appel à des sous-plans ou capacités de la couche C. La reconnaissance des situations et l'activation des buts correspondent ici aux croyances que l'agent a sur lui-même

- La couche PC (planification coopérative) élargit les capacités de développement de planification à un univers multi-agent qui manipule des plans communs aux autres agents. À ce niveau l'agent utilise des protocoles et de stratégies d'interaction. La reconnaissance des situations et l'activation des buts correspondent aux croyances que l'agent possède sur les autres agents du système.

Cette architecture utilise des plans hiérarchiques dans une librairie de plans. Les plans sont formés à partir d'opérations primitives (procédures), de joncteurs (composition séquentielle), de structures de contrôle (si-alors-sinon, tant-que-faire). La notion de plan est rendue opérationnelle en y rajoutant des informations temporelles pour former des plans ordonnés.

Lorsque la couche de planification locale ne peut pas résoudre la situation, elle délègue à la couche de planification coopérative. Cette dernière couche permet aux agents de coordonner leurs actions en négociant et en exécutant des plans joints. Le contrôle de cette couche est activé par des requêtes de la couche de planification locale lorsqu'un conflit apparaît qui ne peut être résolu localement ou lorsqu'une tâche est explicitement coopérative.

L'architecture InteRRap présente différents avantages :

- La base de connaissances de l'agent est gérée selon trois niveaux hiérarchiques ce qui permet à l'agent d'avoir différents comportements,
- le même agent peut être utilisé dans des domaines d'applications différents,
- ce système exploite le parallélisme, ce qui permet par exemple à un agent de mener simultanément un plan local et un processus coopératif à un niveau social.

### 3.3 Les systèmes multi agents

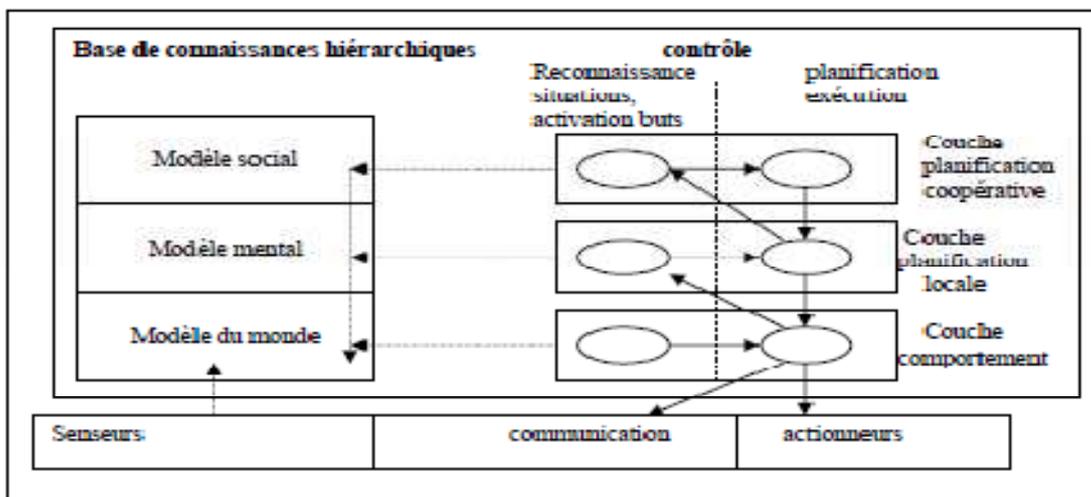


Figure 3.4 : Architecture InteRRap [Fischer et al., 1995a]

#### 3.3.1 Définitions des Systèmes Multi-Agents

Parmi les différentes définitions des systèmes multi-agents nous retiendrons celle de Ferber [Ferber, 1995] qui le définit comme un système composé des éléments suivants (Figure 3.5):

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

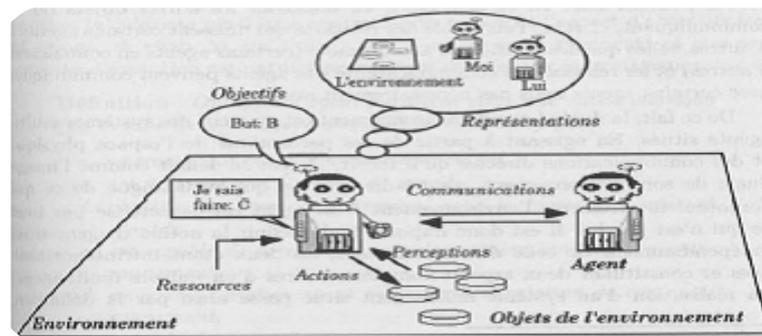


Figure 3.5 : Représentation d'un agent en interaction avec son environnement  
Et les autres agents [Ferber, 1995]

### 3.3.2 Les avantages des SMA

L'utilisation des Systèmes Multi-Agents (SMA) présente une série d'avantages [Brandolese et al., 2000] :

#### - Un Système dynamique

Les SMA héritent des bénéfices de l'IA au niveau du traitement symbolique des connaissances. En revanche, contrairement aux approches traditionnelles de l'Intelligence Artificielle qui simulent, dans une certaine mesure, les capacités du comportement humain, les SMA permettent de modéliser un ensemble d'agents qui interagissent. Les agents sont structurés afin d'exercer une influence sur chacun pour faire évoluer le système dans son ensemble. On rencontre de nombreuses interactions entre agents telles que la coordination (organiser la résolution d'un problème de telle sorte que les interactions nuisibles soient évitées, ou que les interactions bénéfiques soient exploitées), la négociation (parvenir à un accord acceptable pour toutes les parties concernées), la coopération (travailler ensemble à la résolution d'un but commun) [Chaib-Draa et al. 2001]. Cette approche est particulièrement bien adaptée à la simulation des systèmes complexes dont le fonctionnement global émerge des actions des individus. Les SMA permettent de faire vivre virtuellement des agents autonomes sur ordinateur et d'y effectuer des expériences difficiles, voire impossible à mener dans la réalité, d'où la qualification de laboratoires virtuels [Treuil et Mullon, 1997].

#### - Un nombre important d'agents.

Un grand nombre d'agents est au cœur du problème dans ce type de modélisation contrairement à la théorie des jeux où rarement plus de trois acteurs sont représentés [Moss, 2001].

- Souplesse de l'outil informatique qui permet de modifier le comportement des agents, ajouter ou supprimer des actions possibles, étendre les informations disponibles à l'ensemble des agents à la différence des modèles traditionnellement utilisés en science économique. Le modèle Multi-Agent est rendu opérationnel grâce à une implémentation informatique qui n'impose alors aucune exigence analytique spécifique mais l'emploi des langages

informatiques évolués (Langage Orienté Objet (LOO)) qui permet de développer le programme de façon modulaire. La programmation des processus au niveau local dans différents modules et l'utilisation d'entités individualisées apportent alors une grande flexibilité. Les modifications ne nécessitent pas de large restructuration du programme.

- Une résolution distribuée de problèmes. Il est possible de décomposer un problème en sous-parties et de résoudre chacune de façon indépendante pour aboutir à une solution stable.

Cette solution n'est pas forcément optimale au sens de la rationalité complète mais elle peut être « satisfaisante » au sens de Simon [Simon, 1969].

- Lorsqu'un de ses éléments tombe en panne, La réponse d'un système multi agent ne se dégrade pas dans l'ensemble.

### 3.3.3 Types d'environnements

Un agent est situé dans un environnement. Pour modéliser la structure de l'agent il faut avoir un modèle de l'environnement. L'environnement peut être vu comme étant dans un état parmi un ensemble d'états  $E = \{e_1, \dots, e_k, \dots\}$ . L'environnement peut changer son état soit d'une manière spontanée soit comme résultat des actions de l'agent.

L'évolution de l'environnement se modélise différemment selon les caractéristiques que l'on prend en compte, et les simplifications que l'on s'autorise. Les principales distinctions à faire sur les types d'environnements sont [Ferber, 1995] [Russel et Norvig, 1995]:

- Accessible vs inaccessible : un environnement accessible est un environnement dans lequel l'agent peut obtenir une information complète sur l'état de cet environnement. Le plus accessible des environnements est celui construit pour que les agents puissent y agir.

- Déterministe vs non-déterministe : un environnement déterministe est un environnement dans lequel les actions ont un effet. Il n'y a pas d'incertitude sur l'état qui résultera de l'action en cours.

- Episodique vs non-épisodique : dans un environnement épisodique, la performance d'un agent est dépendante du nombre d'épisodes discrets, et sans aucun lien avec la performance d'un agent dans différents scénarios, par exemple, un courrier sortant du système.

- Statique vs dynamique : un environnement dynamique est un environnement dans lequel aucun autre processus n'agit sur lui.

- Discret vs continu : un environnement est discret s'il y a un nombre fini d'actions possibles. L'échiquier d'un jeu de dame est l'exemple d'un environnement discret. Par contre un chauffeur de taxi évolue dans un environnement continu.

Les caractéristiques de l'environnement influencent la façon dont on conçoit un agent car il faut tenir compte de l'évolution de l'environnement, de la capacité de l'agent de saisir cette

évolution et de sa capacité à décider en conséquence. Par exemple, si on a plusieurs agents qui agissent dans un même environnement, chaque agent va percevoir l'environnement comme dynamique et non déterministe, car l'état de l'environnement changera en raison des actions des autres agents, et une même action exécutée dans un certain état aura des résultats différents en fonction des actions de ces autres agents.

L'environnement peut être utilisé implicitement par les agents pour interagir. Dans [Le Strugeon, 1995], le système multi-agents gère une équipe de robots devant explorer une planète. Dans cette application « le SMA perçoit et agit sur son environnement » mais l'environnement lui ne peut agir sur cette application : « l'environnement est conçu comme une entité globale statique, ce qui signifie qu'il n'évolue pas lui-même mais en fonction des actions effectuées sur lui ». Dans cet exemple, l'interaction est implicite puisqu'un agent peut constater une modification de l'environnement sans en connaître nécessairement la cause (un autre agent, évolution normale de l'environnement...). L'interaction entre les agents par perception de l'environnement est souvent utilisée pour représenter les modèles biologiques comme les fourmis [Drogoul, 1993] [Nakamura et Kurumatani 1995] ou les araignées [Bourjot et al., 1999]. Pour les fourmis, la communication est explicite dans le sens où chaque agent décide de déposer un message dans l'environnement et c'est la proximité d'un agent avec le message qui transforme ce dernier en récepteur [Drogoul, 1993] [Nakamura et Kurumatani 1995].

### **3.4. L'interaction dans les systèmes multi agents**

#### **3.4.1 Définitions**

Une interaction est une mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques. Les interactions sont un élément indispensable à la constitution d'organisations sociales [Morin, 1977]. C'est grâce à la coopération que les agents peuvent accomplir des actions et dégager plus que la somme de leurs actions.

L'interaction est donc le composant essentiel de toute organisation. C'est à la fois la source et le produit de la permanence de cette organisation. Un agent sans interaction avec d'autres agents n'est plus qu'un corps isolé. Plusieurs facteurs influencent l'interaction des agents comme les relations de dépendance, la confiance et la capacité de persuasion [Esmahi, 2001].

#### **3.4.2 Situations d'interactions**

Les différentes formes d'interactions sont la collaboration et la coordination d'actions. La première consiste à répartir le travail entre plusieurs agents. La seconde s'intéresse à la manière dont les actions des agents sont organisées dans le temps et dans l'espace pour accomplir les buts. D'une manière générale, les interactions entre individus jouent un rôle essentiel dans le développement d'un être vivant, que ce soit au niveau réactif ou cognitif. [Ferber, 1995] définit une situation d'interaction comme « un ensemble de comportements

résultant du regroupement d'agents qui doivent agir pour satisfaire leurs objectifs en tenant compte des contraintes provenant des ressources plus ou moins limitées dont ils disposent et de leurs compétences individuelles ».

Dans un SMA, les principales situations d'interaction peuvent être classées par rapport à trois critères : les objectifs (buts) ou intentions des agents, les relations que les agents entretiennent envers les ressources qu'ils possèdent ainsi que les moyens (ou compétences) dont ils disposent pour parvenir à leurs fins.

1) Des agents seront dans une situation de collaboration ou d'indifférence si leurs buts sont compatibles. Si leurs buts ne sont pas compatibles, ils seront dans une situation d'antagonisme.

2) Les ressources dont disposent les agents sont une composante très importante dans une situation d'interaction. Une ressource est un élément environnemental ou matériel utile à la réalisation d'une action. Par conséquent, tout agent requiert des ressources pour accomplir ses tâches. La quantité de ressources limitée est souvent une source de conflit entre agents.

3) Si un agent a besoin de ressource pour accomplir une tâche, il a également besoin d'une ou de plusieurs compétences. S'il dispose de toutes les compétences requises pour accomplir sa tâche, il peut l'exécuter seul. Au contraire, il se peut que l'agent ne possède pas toutes les compétences nécessaires, auquel cas, il est contraint de travailler en collaboration avec un autre agent ayant les compétences manquantes. Il peut ainsi sous-traiter une tâche à un autre agent.

Les interactions sont donc dans ce cas très bénéfiques puisque les actions de chacun des agents contribuent à satisfaire un objectif commun. Le système résultant de ces interactions dispose alors de propriétés nouvelles qui s'expriment parfois comme une fonctionnalité émergente.

La coopération est une forme d'interaction. Elle consiste à établir qui fait quoi, avec *quels moyens*, de quelle manière et avec qui. Cela sous-entend qu'il faut trouver des solutions aux différents sous-problèmes que constitue la collaboration par répartition de tâches, la coordination d'actions et la résolution de conflits. La coopération se résume donc par la formule :

Coopération = collaboration + coordination d'actions + résolution de conflits

Ces concepts sont expliqués plus en détails dans les paragraphes suivants. La résolution de conflit supposant une négociation pour lever ce conflit. Plusieurs agents coopèrent ou sont dans une situation de coopération si l'une des conditions suivantes est vérifiée :

- L'ajout d'un nouvel agent permet d'accroître les performances du groupe
- L'action des agents sert à éviter ou à résoudre des conflits potentiels ou actuels.

Les situations d'interactions sont classées dans le tableau suivant :

Buts	Ressources	Compétences	Type de situation	Catégorie
<b>compatibles</b>	suffisantes	suffisantes	indépendances	Indifférence
<b>compatibles</b>	suffisantes	insuffisantes	collaboration simple	coopération
<b>compatibles</b>	insuffisantes	suffisantes	encombrement	
<b>compatibles</b>	insuffisantes	insuffisantes	collaboration coordonnée	
<b>incompatibles</b>	suffisantes	suffisantes	compétition individuelle pure	antagonisme
<b>incompatibles</b>	suffisantes	insuffisantes	compétition collective pure	
<b>incompatibles</b>	insuffisantes	suffisantes	conflits individuels pour des ressources	

Tableau 3.1 : Situations d'Interaction entre Agents [Ferber 1995]

### 3.4.3 Coordination d'actions

Des agents qui travaillent dans un SMA accomplissent un certain nombre de tâches. Il existe également des tâches supplémentaires liées à la mise en relation des autres tâches pour améliorer globalement le déroulement. Il s'agit d'une articulation des actions individuelles de chacun des agents de manière à ce que l'ensemble du système se comporte de façon cohérente et performante.

Ces tâches supplémentaires sont appelées des tâches de coordination. [Malone, 1988] décrit la coordination d'actions comme « l'ensemble des activités supplémentaires qu'il est nécessaire d'accomplir dans un environnement multi-agents et qu'un seul agent poursuivant les mêmes buts n'accomplirait pas ».

La coordination des actions est donc l'une des principales méthodes pour assurer la coopération entre agents autonomes. Il y a plusieurs raisons pour utiliser une coordination : dans un SMA, les ressources sont limitées et il faut bien souvent éliminer les actions inutiles qui consomment des ressources. On recherche donc une réduction des coûts. En outre, les agents ont besoin d'informations et de résultats que seuls d'autres agents peuvent fournir.

On note plusieurs formes de coordination d'action :

- La coordination par synchronisation : c'est la forme la plus basique pour laquelle les actions sont décrites précisément au niveau de leur enchaînement. Très utilisée en automatisme, cette forme de coordination se formalise souvent par des réseaux de Petri.
- La coordination par planification : elle s'exécute en deux phases. La première consiste à créer un ensemble de plans d'actions qui décrivent précisément les actions à effectuer pour atteindre un but. La seconde est l'exécution de l'un de ces plans. Du fait de la dynamique des SMA, l'environnement dans lequel évoluent les agents change en

permanence et de ce fait les agents sont contraints de revoir leur planification en temps réel, ce qui est très difficile à implémenter. Amal El Fallah a travaillé depuis quelques années sur cet épineux sujet [El Fallah & Haddad, 1996].

- La coordination réactive : c'est en quelque sorte le contraire de la planification. Des agents réactifs gèrent leur boucle de perception-action en temps réel. En fonction d'un stimulus de l'environnement, l'agent déclenche spontanément une action réflexe. Il n'y a donc pas du tout de planification. Bien souvent cette approche n'assure pas une optimisation du système sur le plan de l'efficacité mais assure une robustesse du système et une grande adaptabilité. Ce type de coordination est celui qu'utilisent les bancs de poissons, les oiseaux migrateurs ou les meutes de loups. Des simulateurs comme ICHTYUS [Mesle, 1994] ont montré la validité de cette approche.
- La coordination par réglementation : cela consiste à édicter des règles de comportement qui visent à éliminer les conflits potentiels entre agents. On pourrait appeler ces règles des heuristiques. C'est une extension d'une coordination réactive avec des règles d'un niveau plus haut. Le meilleur exemple est le code de la route.

Lorsqu'un conflit apparaît entre deux ou plusieurs agents, la résolution de ces conflits et la recherche du consensus entre ces agents est réalisée grâce à la négociation.

### **3.4.4 La négociation**

La négociation est abordée dans plusieurs domaines scientifiques : l'économie, les sciences des organisations, les sciences de l'aide à la décision et l'Intelligence Artificielle Distribuée.

#### **3.4.4.1 Définitions**

Smith [Smith, 1988] définit la négociation : «By Negotiation, we mean a discussion in which the interested parties exchange information and come to an agreement. For our purposes negotiation has three important components (a) there is a two-way exchange of information, (b) each party to the negotiation evaluates the information from its own perspective, (c) final agreement is achieved by mutual selection».

Dans cette définition, la négociation implique deux éléments essentiels, d'une part la communication (a) et d'autre part la prise de décision (b) et (c). La négociation est abordée par différentes disciplines : l'économie, les sciences des organisations, les sciences de l'aide à la décision, l'Intelligence Artificielle Distribuée (IAD) [Lander et Lesser, 1993][Sycara, 1988] que dans celui de la coordination entre agents [Davis et Smith, 1980] [Sycara, 1992].

Dans les SMA, la négociation donne lieu à deux approches différentes [Osborne et Rubinstein, 1994] [Lepperhoff, 2002] :

1. La première est une approche analytique et normative fondée sur la théorie des jeux. Zlotkin et Rosenschein [Zlotkin et Rosenschein, 1993] ont été les premiers à appliquer les outils de la théorie des jeux dans le domaine des SMA. Ils proposent une architecture en couche permettant de modéliser différents niveaux d'interaction. Leurs protocoles permettent de spécifier des types de négociations que les agents peuvent réaliser, aussi bien que les séquences de propositions et de contre propositions qui sont permises. Dans leurs approches, on retrouve des considérations sur les stratégies de prise en compte globale du temps et des contacts multiples entre agents [Chaib-Draa et Gageut, 2002]. Dans les travaux de Zlotkin et Rosenschein [Zlotkin et Rosenschein, 1991] ou Kraus et al. [Kraus et al., 1998], chaque agent possède sa propre fonction d'utilité et son choix est celui qui conduit à l'équilibre de Nash [Nash, 1953] c'est-à-dire un état dont aucun agent n'a intérêt à s'écarter.

2. La seconde, d'inspiration plus sociologique se rapproche plus du comportement des acteurs dans le processus même de la négociation [Epstein et Axtell, 1996]. Dans ce cadre, l'essentiel des travaux menés débouche sur l'élaboration de protocoles de négociation. Le plus connu est le Contract Net Protocol (CNP) [Davis et Smith, 1980] développé pour la coordination. Axelrod [Axelrod, 1997] et Parunak et al, [Parunak et al., 1998] soulignent que la simulation est la seule approche viable pour étudier des populations d'agents de grande taille et qui interagissent selon des règles non linéaires. [Müller, 1996] distingue trois éléments fondamentaux dans la négociation (Figure 3.6) :

a. La négociation qui s'appuie sur des langages comportant des primitives de communication pour la négociation, leurs sémantiques, et leurs usages dans les protocoles. Les primitives concernent les envois et réceptions de messages et intègrent des actes illocutoires. La structure de l'objet de la négociation fait appel à un langage de description de l'objet. Le protocole spécifie les séquences possibles d'actions et les conditions suivant lesquelles une requête peut être effectuée.

b. Les processus de négociation qui peuvent être, soit des procédures pré-établies, soit des comportements. Le processus concerne la proposition de solutions, l'analyse et la révision des solutions préférables.

c. La décision individuelle qui peut être prise en fonction de critères d'utilité, de stratégie, de systèmes de préférences, d'une fonction de comparaison et de mise en correspondance (matching). L'observation des différentes interactions humaines et l'analyse théorique ont permis de mettre en évidence dans les processus de prise de décision d'un acteur, une dimension individuelle et collective et la mobilisation conjointe de l'une et de l'autre en relation avec l'apprentissage [Zeng et Sycara, 1997].



Figure 3.6: Les différentes catégories de négociation et leurs structures.  
D'après Müller [Müller, 1996]

### 3.4.4.2 Types de négociation et organisations des systèmes multi agents

Deux types de négociation : Intégrative et Distributive

La négociation en IAD trouve place dans des situations de conflits entre agents qui peuvent porter sur les buts, les moyens ou les décisions [Vercouter et al. 1998]. Ceci peut se produire en situation réelle, entre des acteurs antagonistes, en compétition ou en coopération. Simos [Simos, 1990] distinguent un gradient entre des situations d'affrontement et d'entente (Figure 3.7). A chaque pôle Simos [Simos, 1990] associe un type de négociation (au sens IAD), la négociation distributive correspondant à une situation d'affrontement et la négociation intégrative correspondant à une situation d'entente. Aux limites de la négociation se trouvent :

- L'affrontement : seul le rapport de force joue ; il y a recherche de la soumission totale de l'adversaire.

- L'entente : les interlocuteurs oublient leurs intérêts propres et se retrouvent en situation de résolution de problème, c'est-à-dire vont chercher une solution optimale (du point de vue mathématique).

Certains auteurs [Moraitis, 1994] [Baeijs et Demazeau, 1996] [Simos, 1990] distinguent différentes modalités de négociation en fonction de leurs structures organisationnelles :

- Organisation à membre unique : un seul agent va effectuer toutes les tâches et actions qu'il souhaite.

- Hiérarchie simple : Le modèle dispose d'un agent qui dirige l'ensemble du système et répartit le problème en sous-problèmes entre les différents agents (appelés esclaves).

- Hiérarchie multi-niveaux : L'organisation est la même que précédemment mais un agent

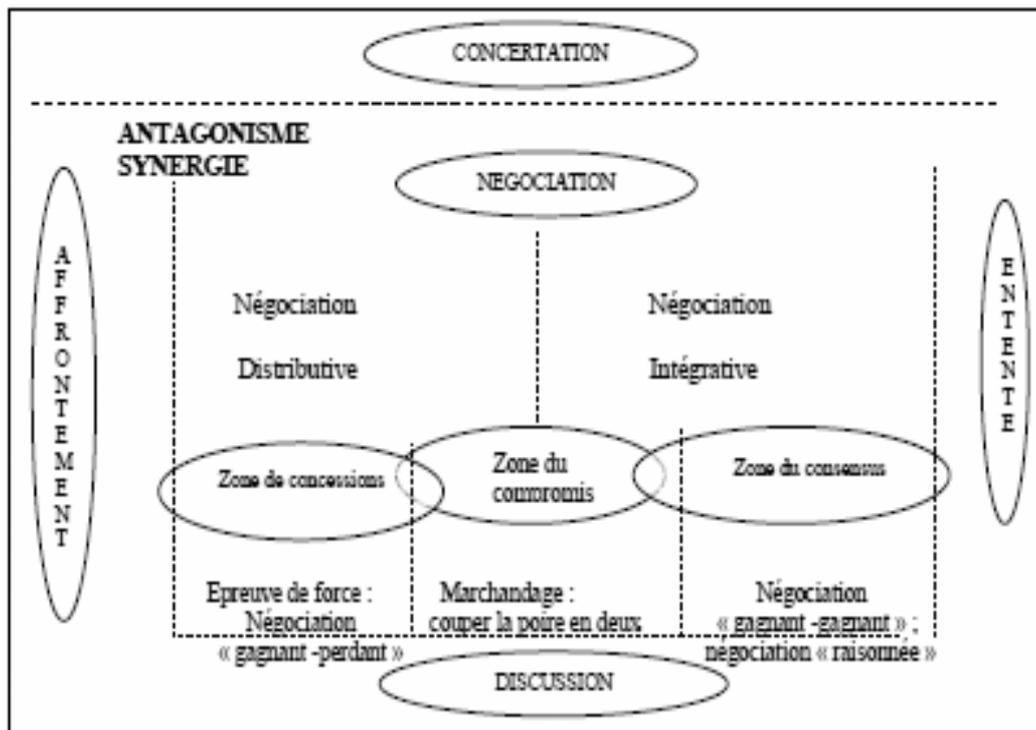


Figure 3.7: Les différents types de négociation [Simos, 1990]

peut être le maître d'un ensemble d'agents et l'esclave d'un autre agent.

- Organisation décentralisée : Les groupes sont autonomes et communiquent sans relation d'autorité. Chaque groupe est maître de ses décisions et peut négocier avec tous les autres groupes sans rapport de force.

- Organisation de marché : C'est le modèle du « Contract Net Protocol » défini par Davis et Smith en 1980 [Davis et Smith, 1980].

Le concepteur d'un SMA choisira le modèle organisationnel en fonction des spécificités de son problème. S'il y a un grand nombre d'agents, une organisation hiérarchique est souvent préférable car elle limite le nombre des communications ; si les agents n'ont pas d'ascendant les uns sur les autres, on optera plutôt pour une organisation de marché,...Lorsque l'organisation n'est pas connue a priori, une organisation dynamique des agents est à envisager, appelée souvent auto-organisation [Le Strugeon, 1995].

On peut distinguer différents types de conflits [Béron et al., 1996] :

- Le conflit de buts entre deux agents tel que la réalisation d'un but par l'un empêche l'autre d'atteindre le sien, même partiellement [Sycara, 1988].

- Les conflits de résultats qui surviennent quand plusieurs agents donnent des résultats différents à un tiers pour une même information demandée par celui-ci. Ces conflits dépendent fortement de l'organisation du système et du protocole de communication utilisé.

- Les conflits de ressources interviennent lorsque les agents utilisent des ressources communes dont le nombre n'est pas suffisant pour permettre de réaliser leurs buts entièrement ou partiellement [Zlotkin et Rosenschein, 1991] [Zlotkin et Rosenschein, 1993].

On peut également distinguer un gradient dans l'ampleur du conflit, il peut être spécifique à un agent ou faire intervenir tout ou partie des agents du système.

Pour résoudre ces situations de conflits, différentes approches sont proposées. Elles peuvent être gérées de façon centralisée ou décentralisée.

La négociation peut s'effectuer (i) directement d'individu à individu [Lander et Lesser, 1993], (ii) être réglée de façon centralisée, en faisant appel à un médiateur ou à un arbitre [Sycara, 1992], (iii) ou à un dispositif hiérarchique [Durfee et Moss, 1997] [Davis et Smith, 1980].

Par ailleurs Kraus et al. [Kraus et al., 1998] proposent différents critères pour qualifier ces méthodes :

- Distribution ou non : les règles d'interaction sont centralisées ou non.
- Symétrie ou équité pour la satisfaction de chaque agent.
- Rapidité : c'est à dire la part prise par la négociation dans la durée de simulation.
- Efficacité : renvoie au nombre d'agents satisfaits par les résultats de la négociation.
- Simplicité des stratégies mises en oeuvre dans la négociation.
- Stabilité. Cette caractéristique correspond au respect d'une seule stratégie par agent tout au long de la négociation.

### **3.4.4.3 Différentes méthodes de résolution de conflits**

De manière générale, dans les méthodes suivantes, la résolution de conflits se fonde sur la recherche d'un compromis acceptable par l'ensemble des antagonistes. Il s'agit ici de résolution plutôt démocratique. Ces compromis peuvent conduire à infléchir différents points de l'état mental d'un protagoniste et les relations entre agents [Aknine, 2002]. Par exemple :

- Accepter les conditions d'un autre agent peut entraîner qu'un agent soit dans l'obligation de modifier son propre état interne (croyances, préférences....).
- Accepter les conditions d'un agent peut entraîner la modification des relations avec un tiers (ou plusieurs) et générer avec ceux-ci des situations de conflits.

Dans ces approches, même un agent non coopératif peut être amené à adhérer à un consensus partagé par l'ensemble de la communauté. Il peut avoir notamment intérêt à ce que les conflits ne se généralisent pas. Il peut également subir la pression des autres agents. Pour convaincre un agent « égoïste » plusieurs méthodes peuvent être employées. Par exemple, menacer de le gêner dans sa propre activité, promettre de l'aider en retour, Donner une récompense « monétaire » comme en théorie des jeux, en économie [Kraus, 2001]... De son côté, l'« altruiste » peut être amené à différer la satisfaction de ces propres objectifs en participant à la satisfaction des objectifs d'autres agents.

Ces méthodes font généralement appel à la théorie multi-attribut de l'utilité initiée par Von Neumann et Morgenstern [Neumann et Morgenstern, 1944]. Elles reposent sur une augmentation des connaissances dont dispose chaque agent sur les autres agents (préférences, argumentation, blocage, ...) en cours du processus. Dans ces méthodes, une négociation prend fin dès qu'un consensus est trouvé, quand on aboutit à une impasse (aucune nouvelle proposition n'est effectuée) ou, dans nombre de ces méthodes, quand le temps de la négociation excède une limite fixée.

Jennings cité par Beer et al. [Beer et al., 1998] identifie trois composantes qui font l'objet de la recherche en négociation : (i) les protocoles de négociation, (ii) les objets de la négociation, (iii) le raisonnement des agents.

- Les protocoles de négociation définissent l'ensemble des règles qui gouvernent l'interaction. Le protocole concerne (i) le type de participant possible (les négociateurs et les candidats possibles), (ii) les états de la négociation (par exemple les propositions acceptables, la fin de la négociation), (iii) les événements qui entraînent une transition d'état, (iv) les actions valides que les négociateurs peuvent engager pour chaque état particulier. Kreifelts et Von Martial [Kreifelts et Von Martial, 1991] considèrent important qu'un protocole de négociation (dans des situations de résolution de problèmes distribués) s'attache (i) à prendre en compte la conservation d'un certain niveau de communication entre les différents agents tout en respectant l'autonomie, (ii) à favoriser l'échange de plans d'actions entre des agents, (iii) à synchroniser les actions, (iv) à laisser ouverte différentes voies possibles pour atteindre un compromis.

- Les objets de la négociation sont la gamme de résultats vers laquelle doit tendre la négociation. Ce résultat peut être simple comme un prix, ou un résultat en relation avec un prix, un délai de livraison, ... on trouve aussi ici les opérations possibles sur les objets.

- Le raisonnement des agents fournit le dispositif de prise de décision dont chaque participant dispose pour atteindre ses objectifs.

Les méthodes de résolutions de conflits supposent donc d'identifier clairement l'objet du conflit, les agents impliqués et mettre en place une conversation basée sur des actes de langage strictement spécifiés [Koning et al., 1998].

#### - **Les méthodes centralisées**

- Une méthode hiérarchique : Le Contract Net Protocol (CNP)

Ce protocole est largement utilisé. Il a été développé par Davis et Smith [Davis et Smith, 1980] comme modèle de négociation pour la résolution de problèmes distribués. Il est fondé sur la notion d'appels d'offres sur les marchés publics. Les relations entre les clients et les fournisseurs se créent grâce à un mécanisme d'appels d'offres émis par un manager. La décision finale est adoptée par celui-ci à partir des réponses envoyées par les contractants.

Ce protocole se décompose en quatre phases :

1. Décomposition du problème : Appel d'offres pour la réalisation d'une tâche.
2. Distribution des « sous problèmes » : le message est envoyé par le manager à tous les agents qu'il estime capable de réaliser la tâche ou à tous les agents du système. A partir de la description de la tâche, les agents contractants construisent une proposition qu'ils envoient au manager.
3. Solution des « sous- problèmes » : le manager reçoit et évalue les propositions et attribue la tâche au meilleur contractant.
4. Réponse : le contractant auquel la tâche est confiée envoie un message au manager en lui confirmant son intention de l'accomplir.

Aknine et al. [Aknine et al., 2000] ont proposé une extension du CNP (Figure 3.8) dans lequel un même contractant peut travailler pour différentes tâches et un manager en organiser plusieurs. Sa structure hiérarchique favorise une rapidité du processus. Elle limite les désengagements des contractants [Aknine et al., 2000]. Son usage ne se conçoit que dans une situation où les objectifs sont partagés par différents agents. Krishna et Ramesh [Krishna et Ramesh, 1998] soulignent que ce protocole n'est utilisable que dans le cas où les agents ont un but commun et ne l'est plus dès lors que les agents ont des objectifs conflictuels.

- La négociation via un médiateur : PERSUADER

La négociation est employée par Sycara pour résoudre des conflits de buts [Sycara, 1988]. Ce système a été implémenté dans PERSUADER [Sycara, 1992] pour aider à résoudre des conflits de travail entre les syndicats d'une entreprise et la direction. Dans ce modèle, trois types d'agents interviennent : un agent médiateur et deux agents opposants correspondant respectivement à la direction et aux syndicats.

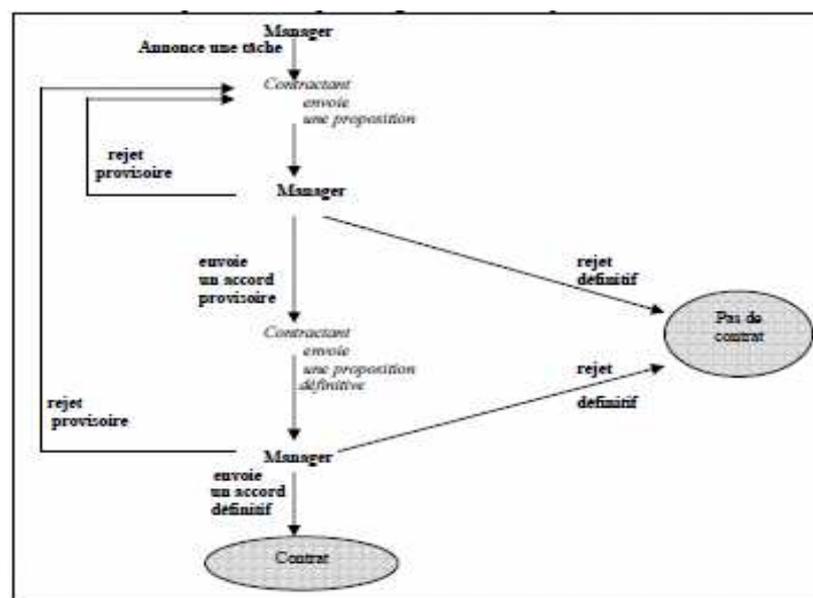


Figure 3.8: Graphe de transition des états de la négociation entre un manager et un contractant dans une extension du CNP. [Aknine et al., 2000].

Chaque agent opposant possède sa propre fonction d'utilité. Chacun cherche à optimiser. Toutefois, en cours de négociation ils seront amenés à réduire leurs exigences en termes d'utilité pour faire aboutir la négociation. En ce sens, on peut dire qu'il s'agit d'agents coopératifs. L'agent de coordination (ou agent médiateur) aide à trouver un compromis entre les deux agents décrits précédemment. Cet agent de coordination (i) propose un compromis initial aux deux opposants, (ii) corrige cette proposition en cas de refus, (iii) reçoit toutes les justifications de chaque agent associées à une réponse. Ainsi, le médiateur acquiert progressivement une vision de plus en plus complète du système. Il s'arrête dès qu'un premier compromis est obtenu.

On retrouve le même type d'approche mais pour des agents non-coopératifs chez Kreifelts et Von Martial [Kreifelts et Von Martial, 1991].

#### - **Les méthodes décentralisées**

Les protocoles de négociation que nous allons décrire succinctement reposent sur une représentation commune du processus de négociation [Jennings et al. 2000], [Jennings et al. 2001].

Généralement ces approches s'inscrivent dans le courant BDI (Belief, Desire, Intention) [Rao et Georgeff, 1995] (définie ultérieurement) ou des courants proches.

A chaque étape de la négociation, on distingue (Figure 3.9) :

- Un objet qui est défini par différents critères. Cet objet peut être une croyance, un but, un plan.
- Un espace de négociation dont les dimensions sont les critères de l'objet. Dans la Figure 24 les dimensions sont au nombre de deux.
- Des agents négociateurs qui sont situés dans l'espace de négociation par :
  - Un sous-espace qui correspond à un ensemble en extension dans lequel chaque point correspond à une proposition,
  - Des préférences qui sont l'expression des poids relatifs des différents critères de l'objet,
  - Une fonction d'utilité qui met en œuvre un critère unique de synthèse quantifiable, construit à partir des critères et des préférences. Cette fonction d'utilité est générale à l'ensemble de la négociation.
  - Chaque point du sous-ensemble est quantifié par son utilité.
  - Au départ de la négociation on crée un objet initial et un agent initiateur de la négociation.

Les agents négociateurs sont proactifs car ils peuvent dans tous les cas faire évoluer leur sous-espace en fonction d'un pronostic porté sur l'évolution de la négociation. A tout moment

de la négociation un agent peut faire entrer un tiers dans le processus pour l'aider à accomplir l'objectif poursuivi.

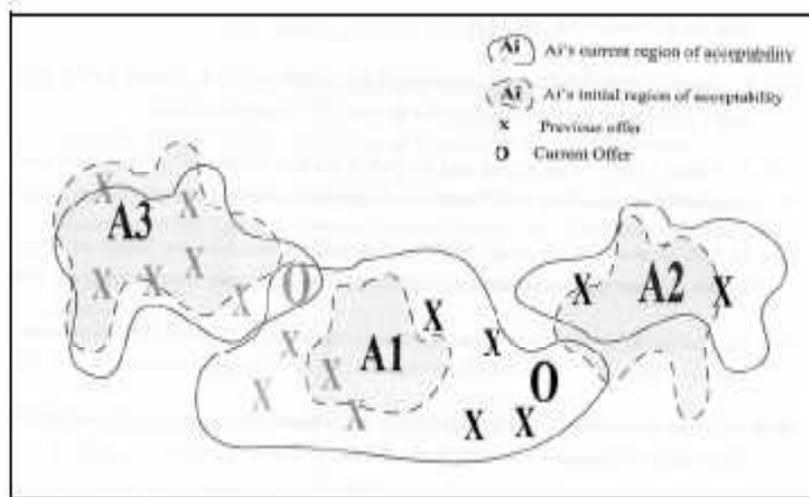


Figure 3.9 L'espace de négociation [Jennings et al. 2001]

On présentera trois méthodes : Les deux premières (recherche négociée, recherche basée sur des méthodes heuristiques) vont conduire à une évolution de l'espace de négociation par formulation de propositions ou contre propositions. Dans ces deux premiers cas, l'évolution de l'espace des négociations est l'image de l'état interne de chaque agent décidé par lui-même (Belief, préférence,...). La troisième (basée sur l'argumentation) conduit également à une modification de l'espace mais peut également conduire à l'évolution des états internes des agents sous l'influence de l'argumentation invoquée par l'un d'entre eux.

- La recherche négociée : TEAM

Dans cette méthode, les agents ne sont pas fondamentalement hostiles. La recherche d'un compromis se fait de façon itérative par élimination progressive des solutions inacceptables par l'ensemble des agents [Lander et Lesser, 1993].

Au départ, un agent fait une proposition, les autres l'évaluent en fonction de leurs propres critères. Les agents qui refusent cette proposition opèrent un retour en arrière compte tenu de leurs désaccords. Ceci conduit éventuellement certains agents à émettre une contre proposition.

Celle-ci est examinée par les autres agents. Ce cycle est répété jusqu'à trouver une proposition acceptée par l'ensemble des agents. Au cours de ce processus incrémental les agents sont amenés à réduire leurs exigences pour favoriser une convergence vers un consensus [Zlotkin et Rosenschein, 1991].

Au plan de l'implémentation, cette méthode met en œuvre un mécanisme de résolution par satisfaction de contraintes [Sycara, 1988].

Cette méthode s'applique à un système distribué et est mise en œuvre dans TEAM [Lander et Lesser, 1993] qui propose des agents hétérogènes et réutilisables.

### - **La Méthode heuristique**

Dans cette méthode Jennings et al. [Jennings et al., 2000] ont développé un ensemble d'algorithmes de négociation fondé sur la théorie multi-attribut de l'utilité dans lesquels un agent a (i) une stratégie de négociation, c'est à dire comment conduire la négociation lors d'une confrontation donnée (ii) des tactiques comme moyen d'accomplir ces stratégies. Ces stratégies et tactiques peuvent évoluer au cours de la négociation.

Les auteurs distinguent trois grandes attitudes en négociation [Faratin et al., 2000][Faratin et al., 2001] [Sycara, 1988] : faire des concessions, échanger des objets de négociation, proposer de nouveaux objets de négociation dans la confrontation en cours. Ces auteurs proposent [Faratin et al., 2000][ Faratin et al., 2001][Sycara, 1988] des algorithmes et des heuristiques. Ceux-ci mobilisent le mécanisme de satisfaction de contraintes. Ces méthodes sont particulièrement utilisées dans des approches distribuées avec des agents homogènes. Matos et al. [Matos et al., 1998] ont utilisé ce type d'approche dans le domaine financier. Il s'agissait de désigner l'agent le plus approprié pour effectuer une tâche tout en respectant les conditions prédéfinies.

### - **La négociation basée sur l'argumentation : ANA**

Cette méthode vise à repousser certaines limites attachées aux deux modèles précédents [Jennings et al. 2001] : (i) Les propositions correspondent en général à un seul point de l'espace de négociation (ii) La seule réaction à une proposition est d'émettre une contre-proposition (un autre point de l'espace), d'accepter la proposition ou la refuser. (iii) Il est difficile de changer les objets de la négociation au cours de celle-ci (ce qui correspondrait à changer l'espace de négociation en ajoutant de nouvelles dimensions).

Dans cette méthode [Jennings et al, 2001] [Kraus et al. 1998], les agents sont de type BDI. Ils peuvent être coopératifs ou ne pas l'être. L'argumentation est un processus itératif qui émerge des échanges entre agents pour amener l'autre et lui faire changer ses intentions. Un échange s'opère en mobilisant des illocutoires qui peuvent être de type promesse, menace, demande...et des modèles de logiques qui peuvent être des logiques modales, floues,...Une formalisation peut être trouvée dans Moraïtis [Moraïtis, 1994].

Plus de détails sur ce type d'interaction entre agents seront donnés dans le chapitre 4.

## **3.5. Méthodologies de conception des systèmes multi-agents**

Le paradigme mutli-agents connaît ces dernières années une nouvelle étape dans son évolution. Des travaux récents montrent le besoin de normaliser le processus d'analyse et de modélisation multi-agent. Pour preuve, de nombreux travaux proposent de faire le point sur l'activité du domaine. Un état de l'art peut être trouvé dans [Mandiau et al., 2002]. Une première approche consiste à définir un état de l'existant en proposant des critères permettant

soit une typologie [Huget et al. 1998] [Ferber, 1995] [Wooldridge et al, 2000] [Padgham et Winikoff, 2002], soit une comparaison des plateformes existantes [Asa, 1999]. Une seconde approche consiste à définir une méthodologie dans la conception d'un système multi agents [Collinot et al. 1996][Demazeau, 1995].

#### - CASSIOPEE

Dans Collinot [Collinot et al., 1996], les auteurs présentent la méthode Cassiopée comme « une façon d'appréhender un type de résolution de problèmes qui suppose la mise en œuvre de comportements collectifs par un ensemble d'agents logiciels ». Afin d'aider le concepteur, cette méthode propose une analyse de l'approche d'un problème par le paradigme mutli-agents. Les auteurs modélisent ce cadre méthodologique suivant trois étapes :

- Définition des Agents : le concepteur répartit les connaissances et les compétences nécessaires aux systèmes entre les agents.

- Définition des Interactions : le concepteur définit le mode de communication entre les agents, la nature de leurs échanges.

- Définition de l'Organisation : le concepteur définit les liens entre les agents qu'il a conçus, comment travailleront-ils en commun afin d'atteindre l'objectif global que le système dans son ensemble doit atteindre.

#### - GAIA

Gaia est une méthodologie permettant la modélisation de l'agent et de son système [Wooldridge et al., 2000]. Elle suppose que les actions des agents sont concurrentes toutes vis-à-vis à d'un but global à atteindre. Les relations entre les agents d'un système construit à l'aide de Gaia sont statiques (elles ne varient pas au cours de la vie du système). Comme la plupart des méthodologies de modélisation, Gaia découpe la construction du système en trois phases : analyse des besoins conception architecturale du système et conception détaillée du même système mais elle ne prend en compte que les deux dernières phases.

Cette méthodologie peut être : (i) abstraite (qui n'a aucune représentation directe dans le système final). Les entités abstraites sont le modèle de rôles et le modèle d'interactions. (ii) Concrète (qui se retrouve dans le système construit). Les entités concrètes quant à elles sont le modèle d'agents, le modèle de services et le modèle d'acointances.

#### - PROMETHEUS

Prometheus [Padgham et Winikoff, 2002] est une méthodologie complète de spécification, conception et implémentation de systèmes d'agent intelligents. Elle est adaptée à la conception des systèmes fermés contenant des agents contrôlés et fiables, par contre, ne l'est pas pour la conception de systèmes ouverts.

A la spécification des besoins, les actions, perceptions et fonctionnalités du système sont définies. (i) les actions et les perceptions définissent l'interface entre les agents et leur environnement. (ii) les fonctionnalités décrivent en un sens plus large ce que devrait faire le système. (iii) Des scénarii de cas d'utilisation sont créés pour fournir une vue plus globale de l'interconnexion entre actions, perceptions, et fonctionnalités.

La phase de conception se décompose en deux sous phases :

- La première appelée conception architecturale consiste à définir les agents du système et leur fonctionnalités. Au cours de cette sous-phase, on définit également les événements auxquels les agents réagissent, les messages qu'ils peuvent recevoir ou émettre. Les protocoles d'interaction sont donc spécifiés sur la base des diagrammes d'interaction. Enfin, les données partagées sont identifiées à cette étape.

- La deuxième sous-phase est la conception détaillée. Elle se préoccupe de la structure interne des agents et de la façon dont ils exécutent leur tâche.

- VOYELLES

La méthodologie voyelles (AEIO) [Demazeau, 1995] (Figure 3.10) se décompose en quatre parties :

La facette A permet de représenter l'ensemble des fonctionnalités du raisonnement interne (planification par exemple) de l'agent.

La facette E permet de définir l'ensemble des capacités de perception et d'action d'un agent sur son environnement.

La facette I permet de définir l'ensemble des interactions avec les autres agents (protocoles de communication par exemple).

La facette O est liée aux capacités de structuration et de gestion des relations des agents entre eux.

### **3.6 Les plateformes S.M.A**

Comme nous l'avons indiqué, les SMA constituent un instrument conceptuel pour modéliser des domaines complexes [Müller, 1998] [Mandiau et al, 2002] [Müller, 2002]. Cependant le processus de développement reste complexe [Wooldridge et Jennings, 1999] tant au niveau conception de la distribution et de la communication entre agents, qu'à celui des architectures et de l'implémentation [Boissier et al. 1999].

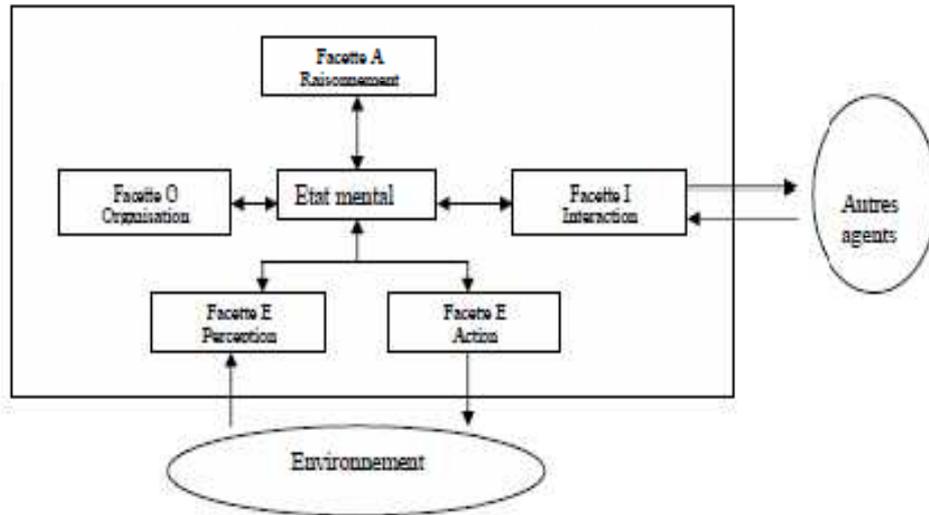


Figure 3.10 : Facette AEIO au sein d'un agent [Boissier, 2001]

De nombreuses propositions sont restées conceptuelles sans aboutir à des réalisations pratiques permettant de les valider. Certains modèles sont devenus des SMA opérationnels mais généralement les systèmes qui en résultent sont restés très dépendants de leurs applications d'origine [Wooldridge et Jennings, 1999]. Ce facteur limite beaucoup la diffusion des SMA. Chaib-Draa et Gageut [Chaib-Draa et Gageut, 2002] suggèrent que des outils formels permettraient :

- L'analyse et le raisonnement des SMA,
- La description des SMA
- La spécification et la vérification des propriétés des SMA
- L'élaboration d'une base commune sur laquelle peuvent être construites des théories plus profondes sur l'action sociale, l'interaction et la coopération.

Ces mêmes auteurs [Chaib-Draa et Gageut, 2002] pensent que la logique présente des avantages certains :

- Une utilisation de la logique permet aux concepteurs de SMA de spécifier, de vérifier et de raisonner sur ces systèmes.
- Un langage défini qui identifie des classes d'objets syntaxiquement acceptables.
- Une sémantique bien définie qui attribue à chaque objet une signification formelle.
- Une théorie de la preuve et de la réfutabilité bien définie qui permet d'examiner de nouvelles inférences.

Les outils génériques relèvent de deux soucis principaux, d'une part la réalisation de classes de base traitant d'agents, d'interactions, d'organisations, ..., d'autre part des outils plus intégrés prenant en charge la coordination entre les différents éléments du système [Guessoum et Ocello, 2001]. Actuellement, on observe une prolifération de plates-formes (Tableau 2)

cherchant à répondre à l'ensemble de ces problèmes. Ces plateformes sont conçues [Guessoum et Ocelllo, 2001]:

- Par rapport à un modèle d'agent particulier et orienté vers une communication entre systèmes distribués (par exemple : ZEUS, MadKit, JACK),
- Par rapport à un domaine d'application (CORMAS),
- Comme agents mobiles (exemple : AGLETS),
- Pour être orientées vers la construction de modèles de simulations (SWARM, CORMAS).

Actuellement, on peut regrouper ces environnements de développement en cinq catégories [Guessoum et Ocelllo, 2001] (Tableau 3.2) :

1. Les outils pour la simulation qui permettent de fournir un ensemble d'outils et de bibliothèques pour faciliter le développement de simulations multi-agents.
2. Les outils pour l'implémentation d'architectures d'agents.
3. Les outils pour la conception fondés sur un modèle componentiel.
4. Les outils pour la conception et l'implémentation offrant un ensemble d'utilitaires pour définir un groupe d'agents,
- 5 Les outils pour la conception, l'implémentation et la validation.

Certains critères d'analyses peuvent être à la base du choix d'une plateforme :

- L'information fournie : le degré de couverture qu'une plateforme produit à cette étape, principalement au niveau de la documentation et des outils disponibles.
- L'application : la plateforme est-elle utilisable par rapport à notre problématique ?
- La complexité : la compétence nécessaire pour le développeur et la qualité de travail que cette tâche impose.
- La réutilisation : la quantité de travail réutilisable pour de prochains travaux.
- Le langage de programmation utilisé : le plus souvent les langages de programmation diffèrent d'une plateforme à l'autre et cela pose le problème de compatibilité si l'on souhaite utiliser différentes plateformes.

Catégorie	Environnements
Outils pour la simulation	Swarm, Cormas, geomas, Mice
Outils pour l'implémentation	Actalk, DECAF, JACK, MAGES IV, Magique, Mocab, OSAKA, Pandora II
Outils pour la conception	Aladdin, JAF, Maleva
Outils pour la conception et l'implémentation	AgentBuilder, DIMA, Mask, Mercure, MACE
Outils pour la conception, l'implémentation et la validation	Desire, dMars, MAST, Zeus

Tableau 3.2 : Les différents environnements de programmation.  
D'après [Guessoum et Ocelllo,2001]

On peut aussi envisager d'autres critères d'analyses de plateforme SMA tels que :

- La disponibilité : Est-ce une version stable ? Le code source est-il disponible ? Coût de l'outil ?
- Le support : il y a t il un futur développement de cette plate-forme ? Cette plate-forme est elle largement utilisée ?

On notera que ces plateformes présentent de grandes variabilités au niveau des langages d'implémentation, des formalismes de représentation des connaissances et des théories d'agents sous-jacentes. Elles sont de plus souvent très marquées par un type d'application visée.

CORMAS [Bousquet et Le Page, 2001]

Si on considère l'agent comme une extension de l'objet, la plateforme CORMAS (Common pool Ressources and Multi-Agents Systems) est axée vers la construction de modèles de simulation mais plus particulièrement utilisée dans le domaine de la gestion de ressources renouvelables. Cette plateforme gratuite a été développée par le CIRAD (Centre de coopération Internationale en Recherche Agronomique pour le Développement) et facilement téléchargeable à l'adresse : <http://cormas.cirad.fr/>

Cette plateforme de simulation est basée sur l'environnement de programmation VisualWorks ; qui permet de développer des applications en Smalltalk. L'environnement de programmation Cormas permet le développement de modèles de simulation des modes de coordination entre des individus et des groupes qui exploitent des ressources en commun.

Ce système est structuré en trois modules :

- Un premier module qui permet de définir les différentes entités du système à modéliser, c'est ce qu'on appelle les agents informatiques et leurs interactions. Celles-ci s'expriment par des procédures de communication directes telles que l'envoi de messages ou le contrôle de la dynamique globale.

- Un second module qui permet le contrôle de la dynamique globale. Le modélisateur doit définir la méthode qui lui permet de définir son pas de temps. Chaque entité peut avoir sa propre méthode d'évaluation du temps. Lors de la simulation, il est possible de définir le nombre de pas de temps voulu pour l'ensemble.

- Le troisième module qui permet de définir les différents points de vue.

Un des aspects attrayant de cette plateforme est la visualisation des processus dynamiques (propagation de feu de forêt par exemple) sur un support spatial correspondant à une grille, chaque cellule étant définie comme un automate cellulaire.

Produit	Compagnie	Langage	Description
AgentBuilder	Reticular Systems, Inc.	Java	Integrated Agent and Agency Development Environment
AgentTalk	NTT	LISP	Multi-agent Coordination Protocols
AgentX	International Knowledge Systems	Java	Agent Development Environment
Aglets	IBM Japan	Java	Mobile Agents
Concordia	Mitsubishi	Java	Mobile Agents
DirectIA SDK	MASA Group	C++	Adaptive Agents
Gossip	Trillian	Java	Mobile Agents
Grasshopper	IKV++	Java	Mobile Agents
IGEN	CHI Systems	C/C++	Cognitive Agent Toolkit
Intelligent Agent Factory	Bits & Pixels	Java	Agent Development Tool
Intelligent Agent Library	Bits & Pixels	Java	Agent Library
JACK Intelligent Agents	Agent Oriented Software Pty. Ltd.	JACK Agent Language	Agent development Environment
JAM	Intelligent Reasoning Systems	Java	Agent Architecture
Jumping Beans	Ad Astra Engineering, Inc	Java	Mobile Components
LiveAgent	Alcatel	Java	Internet Agent Construction
MadKit	Madkit Development Group	Java, Scheme, Jess	Multiagent Development tool
Microsoft Agent	Microsoft Corporation	Active X	Interface creatures
NetStepper	JW's Software Gems		Agent Development Environment
Network Query Language	NQL Solutions		Programming Language
Pathwalker	Fujitsu	Java	Agent-oriented programming library
Social Interaction Framework	DFKI (German research Institute for AI)	Java	Multi-agent System toolkit
Sodasdot	MIT Artificial Intelligence Lab		Software Agent User-Environment and Construction System
SOMA (Secure and Open Mobile Agent)	Universite de Bologne	Java	Agent Programming Environment
Swarm	Swarm development Group	Objective C, Java	MultiAgent Simulation
TeamBots	The Robotics Institute Carnegie Mellon University	Java	Multiagent Mobile Robotics
TuCSoN	Université de Bologne		Model for the coordination of internet agents
UMPRS	Intelligent Reasoning Systems	C++	Agent Architecture
Via : Versatile Intelligent Agents	Kinetoscope	Java	Agent Builder Blocks
Voyager	Object Space	Java	Agent-Enhanced ORE
Zeus	British Telecommunications Labs	Java	Agent Building Environment

Tableau 3.3 : Liste de plates-formes disponibles et utilisables (www.multiagent.com)

## SWARM [Terna, 1998]

Le simulateur Swarm (Figure 3.11) est un ensemble de bibliothèques portables que l'on peut utiliser dans des environnements variés. Ce simulateur est gratuit et facilement téléchargeable sur : <http://www.swarm.org>

Swarm est un ensemble de bibliothèques facilitant l'implémentation des modèles basés agents.

L'inspiration de Swarm provient du domaine de l'intelligence artificielle (IA). Deux stratégies caractérisent ces types d'approches :

- Evaluation empirique des dynamiques

La combinaison des entités autonomes dans un environnement distribué est typiquement un processus récursif qui ne peut être résolue de façon analytique. Or, dans de nombreux systèmes la seule façon de connaître quelle sera la dynamique globale d'un système est de le simuler sur un grand nombre de pas de temps et ceci un grand nombre de fois.

- La deuxième idée est synthétique

La vie artificielle semble extrapoler des connaissances biologiques au lieu de suggérer de nouvelles expériences. Swarm fournit des éléments permettant de faire des extrapolations pour réaliser la simulation.

Ainsi, Swarm est une machine virtuelle qui permet :

- De décrire des comportements d'agents, un par un, contexte par contexte, tout en gardant la notion de temps et de concurrence dans le modèle.
- De composer et de décomposer des hiérarchies d'agents.

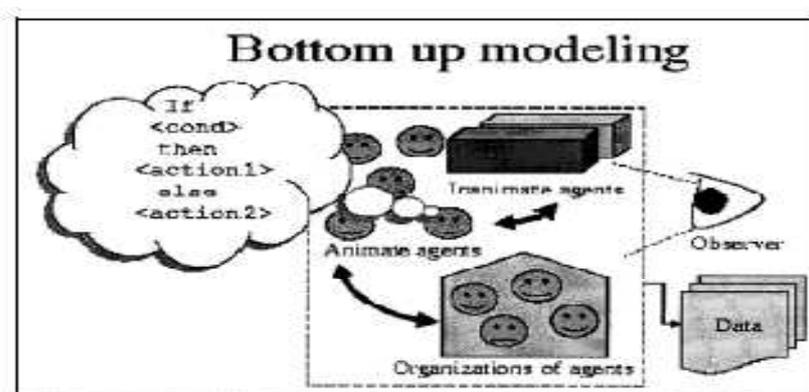


Figure 3.11: Structure générale de Swarm [SWARM, 2001]

L'environnement de développement est celui qu'utilise le modélisateur Swarm pour écrire son programme. Cela peut être de la simple ligne d'éditeur à un environnement programmable de type Emacs, ou un environnement de développement java comme JBuilder.

L'environnement Swarm contient des méthodes et objets pour la simulation. Pour l'essentiel tous les éléments accessibles en Objective C sont stockés dans cet environnement.

Une des caractéristiques principales de Swarm est l'absence d'une classe agent à la base de l'outil. Ceci permet à l'utilisateur de créer ses propres agents avec la sémantique qu'il souhaite.

Dans Swarm, l'agent est vu comme une extension du processus. L'originalité de ce système provient de la notion d'activité associée à un swarm ou essaim qui permet un contrôle du déroulement des actions effectuées par les agents. Un swarm est le composant de base du système.

Dans une application de ce type, l'utilisateur spécifie le comportement d'un ensemble d'agents sous la forme d'un ensemble d'actions déclenchées selon un calendrier. Il est possible de créer plusieurs swarm liés entre eux par une hiérarchie.

JACK [Busetta et Ramamohanarao, 1998].

JACK est une plateforme qui permet le développement de l'environnement des agents BDI. Cette plateforme est décrite comme étant un environnement pour construire, exécuter et intégrer des systèmes multi-agents commerciaux, écrits en Java et utilisant une approche orientée composants. Cette plateforme payante a une utilisation limitée dans le temps. Une version d'évaluation est téléchargeable sur : <http://www.agent-software.com>

JACK inclut tous les éléments de développement dans un environnement Java et des extensions pour implémenter le comportement de l'agent. Les agents intelligents de JACK sont construits sur le modèle d'agent BDI. Cette plateforme s'intéresse principalement à l'étape de développement.

Les agents sont des éléments autonomes qui ont des objectifs explicites [Busetta et Ramamohanarao, 1998]. Chaque agent dispose de :

- Un ensemble de données sur l'environnement,
- Un ensemble d'évènements auxquels il pourra répondre,
- Un ensemble d'objectifs (goals) qu'il peut souhaiter atteindre,
- Un ensemble de plans qui décrivent la façon d'atteindre ses différents objectifs.

Les outils logiciels sont le JDE (Jack Development Environment), un environnement de programmation classique, le compilateur du Jack Agent Language (JAL), qui traduit les

programmes écrits en JAL en JAVA pur, et la librairie de classes permettant l'exécution des agents, appelée Jack Agent Kernel.

Le langage agent JACK est constitué de cinq grandes classes :

- Agent. L'agent construit est utilisé pour définir le comportement de l'agent logiciel. Cela introduit, le type de messages et les événements qui correspondent, et les plans à utiliser pour accomplir ses objectifs.
- Capacité. Permet de définir les éléments réutilisables par l'agent. Ceci peut être des plans, des données....
- Base de données. Il s'agit d'une base de données relationnelle.
- Evènements. Décrivent une occurrence qu'un agent doit prendre en compte pour ses actions
- Plan. Les plans des agents sont analogues aux fonctions.

### 3.7 Conclusion

Nous avons pu couvrir à travers ce chapitre tous les aspects qui concernent La technologie agent. Ce dernier en tant qu'entité de base et concept clé de ce paradigme a eu un espace important dans cette étude. Nous avons vu que cette entité a une structure interne, un environnement, un type qui pourrait être réactif, cognitif ou hybride et a un cycle de vie qui définit son comportement individuel.

Un agent a aussi un comportement social lui permettant d'interagir avec les autres agents pour faire justement le tout, qui est le système multi agents car un ensemble d'agents sans interaction n'est qu'un ensemble d'entités isolées.

L'interaction dans un système multi agent est primordiale car elle permet aux agents d'atteindre leurs buts et d'accomplir leurs tâches qui lui ont été assignées par leurs concepteurs et ce, par le biais de la coordination, la collaboration et la négociation. C'est pour cette raison que cet aspect social de l'agent a été présenté avec suffisamment de détails.

Le processus de développement d'un système multi agents a été aussi débattu via la présentation des méthodologies de conception de ces systèmes et les plateformes permettant de les mettre en œuvre. Il faut noter qu'un besoin insistant de normalisation de ces méthodologies et plateformes est consenti par la communauté multi agents.

Après avoir présenté en détail le raisonnement qualitatif dans le premier chapitre et les systèmes multi agents dans le présent chapitre, le chapitre suivant discutera de la relation entre les deux technologies.

## CHAPITRE 4 : LE RAISONNEMENT QUALITATIF ET LES SYSTEMES MULTI AGENTS

### 4.1 Introduction

Après avoir présenté, d'une manière plus ou moins détaillée, les systèmes complexes et les systèmes multi agents, objet des chapitres 2 et chapitre 3 respectivement, ce présent chapitre va être dévolu aux travaux de recherche menés jusqu'à présent et touchant conjointement les systèmes multi agent et le raisonnement qualitatif.

Le raisonnement qualitatif est la faculté cognitive permettant d'inférer des connaissances portant sur les comportements typiques ou essentiels d'un système en l'absence d'informations quantitatives complètes. Nous avons aussi mentionné dans le premier chapitre que ce type de raisonnement, pratiqué par les être humains fait appel à des connaissances de base, appelées connaissances de sens commun.

Un agent est une entité logicielle capable de percevoir son environnement, de raisonner sur ces perceptions, de planifier ses actions, et d'agir sur son environnement afin d'atteindre ses objectifs. Un système multi agents est une collection d'entités logicielles qui interagissent dans le cadre d'une coopération ou d'une coordination d'actions afin d'atteindre un objectif commun. Cette interaction est réalisée selon un protocole d'interaction exprimé dans un langage de communication inter - agents commun. Le paradigme agent ou la programmation orienté agents en utilisant une méthodologie de développement multi agents et d'une plateforme de développement de systèmes multi agents. L'approche multi agents est pour le moment un moyen pour la modélisation et la mise en œuvre des systèmes complexes. Un vrai défi pour le paradigme agent dans le futur proche, est la possibilité de modéliser n'importe quel système complexe, quelque soit le degré de sa complexité. La prise en compte du raisonnement qualitatif dans le comportement individuel et collectif de l'agent permettra de doter le paradigme agent d'une capacité supplémentaire qui lui permettrait de relever ce défi.

Le but attendu de ce présent chapitre est de pouvoir cerner les points de concours pouvant exister entre les deux domaines ainsi que de dévoiler l'apport de chacune des deux technologies par rapport à l'autre.

Le reste de ce chapitre sera organisé de la manière suivante. Le premier point sera consacré à l'étude des travaux dont l'objectif est de montrer l'apport du raisonnement qualitatif au paradigme agent. Le deuxième point discutera de la relation inverse, c'est-à-dire l'apport du paradigme agent au raisonnement qualitatif. Finalement, une conclusion nous permettra de clôturer ce présent chapitre.

## **4.2 Apports du raisonnement qualitatif au paradigme agent**

Le raisonnement qualitatif par l'argumentation est un domaine de recherche riche et multidisciplinaire mariant la philosophie, la science de la communication, la linguistique et la psychologie.

Dans un environnement dynamique, l'argumentation, est un moyen pour l'agent afin de mettre à jour ses croyances après la perception de nouvelles informations issues de son environnement ainsi que de résoudre les conflits qui pourraient exister entre ses croyances et les croyances des autres agents à travers la communication et la négociation.

Plusieurs travaux récents ont proposé des modèles d'argumentation que les agents logiciels peuvent utiliser pour effectuer les activités de négociation pour arriver à un consensus dans des situations de non coopération et quand les agents disposent de connaissances incomplètes au sujet d'eux-mêmes et au sujet de leur environnement. Les plus récents des travaux ont tenté de développer des protocoles de négociation basée argumentation et des modèles d'agents capables d'utiliser les techniques d'argumentation. Dans ce qui suit, nous allons faire le tour d'horizon sur les travaux réalisés dans ce domaine.

### **4.2.1 L'argumentation dans le raisonnement des agents autonomes**

#### **4.2.1.1 L'argumentation pour la révision des croyances**

Un agent a besoin de réviser ses croyances après la perception de nouvelles informations de son environnement. Un défi majeur du raisonnement non monotone est de définir des méthodes efficaces pour la révision des croyances. Le paradigme A.G.M [Gardenfors et al 1988] définit les postulats de rationalité à vérifier par tout processus de révision de croyances. Les méthodes de révision de croyances connues sont T.M.S [Doyle 1979], la logique de défaut [Reiter 1980] et la logique de circonscription [McCarthy 1980]. L'argumentation est une méthode alternative pour automatiser le raisonnement non monotone où des arguments et des contre - arguments au sujet de certaines conclusions, sont construits et comparés. La non-monotonie réside dans le fait que de nouvelles prémisses peuvent générer de nouveaux arguments qui vont supporter de nouvelles croyances.

Plusieurs cadres formels basés argumentation pour le raisonnement non monotone ont été proposés durant les deux dernières décades, tels [Fox & Ambler 1992] [Loui 1987], [Pollock 1987] [Prakken and Sartor 2001][Simari and Loui 1992], mais leur incorporation dans le raisonnement d'un agent autonome reste une tâche non encore achevée. Pour ce faire, une représentation adéquate de l'environnement ainsi que l'intégration des perceptions de l'agent dans le mécanisme de révision des croyances du même agent sont nécessaires.

#### **4.2.2 L'argumentation pour la planification et la délibération**

En plus du maintien des informations concernant son environnement, un agent doit se fixer un but (parmi plusieurs) et doit aussi arrêter un plan (parmi plusieurs) pour atteindre ce but.

Des travaux récents ont utilisé l'argumentation comme moyen pour choisir parmi plusieurs désirs en conflit [Amgoud 2003] et parmi plusieurs buts en conflit [Amgoud & Kaci 2004]. Une autre approche [Kakas & Moraitis 2003] utilisant des arguments et des préférences entre ces arguments a été proposée pour générer des buts sur la base d'un contexte dynamique.

Aussi, plusieurs travaux récents dans la planification basée argumentation ont été proposés. Par exemple dans [Hulstijn et Van der Torre 2004][Rueda et al. 2002], des approches basées argumentation ont été proposées pour générer des plans qui sont relativement simples par rapport aux plans que génèrent les algorithmes de planification [Georgeff 1987].

### **4.2.3 L'argumentation pour la communication entre agents**

Afin d'achever leurs buts, les agents ont besoin de communiquer. L'argumentation permet d'étudier et de formaliser les différents aspects de communication entre agents. Les théoriciens de l'argumentation Douglas Walton et Erik Krabbe ont défini une typologie de dialogues [Walton and Kabbe 1995] en se basant sur les croyances des participants et sur le résultat attendu par le dialogue, nous en donnons ci-dessous une description informelle.

- Recherche d'information : un participant au dialogue cherche à obtenir des réponses à des questions venant d'un autre participant. le premier participant croit que le deuxième est en mesure d'y répondre.
- Persuasion : deux ou plusieurs participants ont des croyances conflictuelles et un des participants veut changer les croyances d'un autre participant.
- Enquête : un nombre de participant collaborent pour répondre ensemble à une question ouverte. Aucun des participants ne connaît la réponse.
- Délibération : un nombre de participants cherchent à décider sur le cours d'une action
- Négociation : un nombre de participants ayant des intérêts conflictuels coopèrent pour arriver à un consensus.

Pour formaliser ces types de dialogues, deux principaux concepts issus de la théorie de l'argumentation qui sont les jeux de dialogues (dialogue games) et les plans d'argumentation (argumentation schemes) ont été adoptés par la communauté M.A.S. Nous allons décrire ci-dessous chacun des deux concepts.

#### 4.2.3.1 Jeux de dialogues

Un jeu de dialogue (dialogue game) est un ensemble d'interactions entre deux ou plusieurs joueurs où chaque joueur effectue un mouvement par le biais d'une locution exprimée dans un langage commun et conformément à des règles prédéfinies. Récemment, les jeux de dialogues ont été utilisés pour spécifier les protocoles [Maudet and Chaib-draa 2003]. Un protocole d'un jeu de dialogue est défini par un ensemble de locutions et un ensemble de règles : on en distingue les règles de commencement (commencement rules), les règles de terminaison (termination rules), les règles de combinaison (combination rules) et les règles qui définissent l'effet de chaque locution. (commitment rules). Les protocoles de jeux dialogue précédemment cités ont été spécifiés formellement [Hulstij, 2001][McBurney et al. 2003] [Pollock 1987] [Hitchcock et al. 2001].

#### 4.2.3.2 Plans d'argumentation

Les plans d'argumentation (Argumentation Schemes, en Anglais) permettent de capturer les formes stéréotypiques de raisonnement, déductive ou non déductives qu'on trouve dans les discours de tous les jours. Un plan d'argumentation est associé un ensemble de questions permettant de générer des arguments relatifs à un sujet donné. Les plans d'argumentation permettent de réduire le coût de génération et d'évaluation des arguments. Le travail de Chris Reed et Douglas Walton intitulé : « Towards a formal and implemented model of argumentation schemes in agent communication » présente une implémentation d'un cadre formel pour la spécification des plans d'argumentation en utilisant le langage AML (XML-based argument markup language). Ce travail ouvre des horizons pour l'utilisation pratique des plans d'argumentation dans les systèmes multi agents.

#### 4.2.3.3 L'argumentation pour la négociation entre agents

Parmi les premiers travaux dans le domaine, nous pouvons citer le système de Sycara [Sycara 1990], un système intelligent d'aide à la résolution des conflits à travers la négociation/médiation. L'argumentation persuasive a été adoptée comme mécanisme pour la résolution de problèmes par un groupe d'agents qui ne sont pas complètement coopératifs. La construction des arguments est effectuée par l'intégration de raisonnement basé cas, recherche dans un graphe et l'approximation des fonctions d'utilité des agents.

Sillince dans [Sillince 1994] propose un cadre formel pour la résolution des conflits basé sur l'argumentation. Dans son framework, les arguments sont construits par les agents qui peuvent adopter des croyances inconsistantes jusqu'à ce que d'autres agents les attaquent au moyen d'arguments plus forts.

Kraus et son équipe de recherche [Kraus et al 1998] proposent un modèle logique d'argumentation faisant usage de six (6) types d'arguments ayant de la force persuasive dans la

négociation chez les êtres humains [Karlins and Abelson 1970] qui permettent aux agents de négocier d'une manière persuasive. Ces arguments sont : menace (threat), promesse d'une future récompense ( promise of future reward), appel d'une récompense passée ( appeal to past reward), appel à une contradiction ( appeal to " precedents as counterexamples"), appel à une pratique prévalente ( appeal to prevailing practice) et appel à un intérêt personnel ( appeal to self interest). Dans ce modèle logique, les états mentaux des agents c'est-à-dire leurs intentions, désirs, croyances, buts et préférences sont représentés et l'argumentation est vue comme un processus itératif qui tente à les modifier par le mécanisme de persuasion. Les agents s'envoient des messages de la forme request (f, [A]), reject (f, [A]), accept (f, [A]) où f est une formule qui représente un état mental et A un type d'argument qui peut être un des six types d'arguments précédemment cités. Kraus et al. ont implémenté un modèle d'agent appelé ANA (Automated Negotiation Agent) qui agit dans une simulation multi agent où l'utilisateur peut définir les états mentaux des agents et les règles d'inférence servant à la génération des arguments. Une fois créés, les agents tentent d'accomplir leurs désirs, en utilisant les types d'arguments qui conviennent. Cette approche permet à l'utilisateur de tester les différents types d'argument et d'analyser leur impact sur les capacités de négociation des agents.

Parsons et son équipe de recherche [Parsons et al. 1998], inspirés par le travail de [Kraus et al 95] proposent un cadre générique d'argumentation qui permet de simuler le raisonnement interne d'un agent et de supporter la négociation entre agents. Ces derniers ont les capacités de négociation minimales qui sont : proposer (propose) et répondre (respond). Une réponse peut jouer le rôle d'une critique, c'est-à-dire une contre-proposition. Pour justifier sa position et persuader les autres agents à changer leurs positions, un agent a besoin d'arguments tels utilisés dans [Sycara 1990] et [Kraus et al 1998] comme la menace, la récompense et les appels. Parsons et al. ont proposé un processus d'argumentation qui commence par l'échange de propositions (solutions à un problème), de critiques (remarque, commentaire ou contre propositions) , d'explications ( peuvent être des arguments ou des méta - informations qui permettent d'aider un autre agent à trouver une autre suggestion plus acceptable . Ces méta-informations sont uniquement autorisées en l'absence d'une proposition ou une critique. Le processus d'argumentation se termine si une proposition ou une contre proposition est acceptée par toutes les parties ou si la négociation est avortée sans que les agents ne parviennent à un consensus. Parsons et al. proposent un système d'argumentation qui permet deux types de défaite entre arguments : défaite par « undercutting » et défaite par « rebutting » . Ils ont défini aussi cinq classes d'acceptabilité permettant d'évaluer les arguments. Un argument dans le système de Parson et al. est une succession d'inférences prouvant qu'une conclusion pourrait être vraie.

Amgoud et ses collègues [Amgoud et al 2000a] présentent un modèle de dialogue basé argumentation qui se veut plus général que celui proposé par [Parson et al 1998]. Ce modèle est inspiré par le travail de Dung [Dung 1995] mais augmenté par la notion de préférence entre les arguments [Amgoud and Cayrol 2000].

Jung et ses collègues [Jung et al 2001] ont étudié la négociation basée argumentation dans les systèmes de taille importante (large scale systems). Son travail se base sur le questionnement suivant : Si la négociation basée argumentation a donné des résultats dans les systèmes de petite échelle (small scale systems), est-ce que ceci est valable dans les systèmes à grande échelle ? La réponse est évidemment non car la vitesse de convergence de la résolution des conflits lors de la négociation n'est pas la même. Ils ont proposé un modèle d'argumentation pour ces types de systèmes basée sur la satisfaction distribuée des contraintes. Ce modèle se base sur l'idée suivante : la communication des contraintes locales d'un agent à d'autres agents permet à ces derniers de les utiliser pour accélérer le processus de résolution des conflits. Jung et al. ont implémenté et expérimenté leur modèle de négociation et ont montré que la convergence de la résolution des conflits dépend de la stratégie appliquée par les agents dans la propagation des contraintes.

[Schroeder 2000] réclame que la visualisation du processus d'argumentation est une approche prometteuse permettant à l'utilisateur de comprendre d'une manière intuitive le raisonnement de plusieurs agents en interaction sans avoir besoin de connaissances en logique. Ceci a fait l'objet du projet ULTIMATE RATIO [Schroeder 2000] dont le principe consiste à construire dynamiquement un arbre qui garde trace du processus d'argumentation entre agents appelé arbre de preuves.

Les travaux récents dans le domaine tentent à étudier l'effet des considérations sociales, telles la confiance, la crédibilité et la réputation d'un agent sur le processus d'argumentation. Par exemple dans [Schillo et al 99], un agent évalue la réputation d'un autre agent sur la base des observations directes et les témoignages des autres agents. L'importance de l'intégration d'un mécanisme de gestion explicite de la réputation des agents permet de lever l'indécision dans le cas de conflit entre deux arguments émanant de deux agents en interaction.

Après avoir montré l'apport très important de l'argumentation pour l'agent et le système multi agents, Continuons par présenter deux autres travaux qui attestent eux aussi de l'avantage du raisonnement qualitatif dans les systèmes multi agents.

Dans son travail intitulé « The Role of Qualitative Reasoning in Multi Agent Systems » [ Marik et al 1997] , l'auteur présente une architecture multi-agents dont le comportement global est observé par un agent spécifique de l'architecture appelé l'agent observateur dont le rôle est d'observer d'une manière continue le fonctionnement d'un système multi-agent dans le but d'améliorer l'efficacité et la fiabilité du SMA par l'assurance d'une répartition appropriée des tâches sur les différents agents permettant la réduction des besoins de communication inter - agents. Ce méta- agent aura à assurer les fonctions suivantes : détection des cas de pannes ou de dysfonctionnement chez un agent, redistribuer la charge des agents par la réallocation des tâches et clonage des agents s'il est nécessaire. Pour cela, l'agent observateur maintient et contrôle les trois bases d'informations suivantes :

- a) base d'informations permanentes et 'semi' permanentes au sujet des agents du système. (Exemple d'une 'information permanent : adresse fixe sur le réseau pour un agent situé. Exemple d'une information non permanente : message envoyés à d'autre agents)
- b) base d'informations au sujet des états courants des agents.
- c) base des tâches que les agents des systèmes peuvent assurées.

Ainsi l'agent observateur peut à tout moment connaître toutes les tâches en exécution par les agents ainsi que la liste des agents participants. Pour accomplir sa tâche d'observateur et de supervision du comportement global du système, l'agent observateur utilise le raisonnement qualitatif pour prédire les séquences possibles de transitions d'états du SMA et donc avoir la possibilité d'éviter au système de tomber dans des états non souhaitables ( par exemple état où la charge des agents est relativement non homogène ou utilisation abusive des ressources). Concrètement parlant, l'agent observateur aura à utiliser un modèle qualitatif décrivant et mettant en relation les paramètres qui entrent dans la description du comportement du système multi agents sur lequel un raisonnement qualitatif va être appliqué pour tirer les comportements possibles. Il est à noter que l'agent observateur ne contredit pas l'esprit du paradigme agent qui prône la distribution du contrôle et des données car son travail principal consiste à observer le comportement du système et le corriger le cas échéant. Malgré que le type de systèmes multi agents utilisé dans cette technique est fait d'un petit ensemble d'agents ayant des rôles fixes et une organisation fixe, les résultats obtenus sont encourageants [Marik et al 97] et montrent que le modèle de supervision qualitative menée par l'agent observateur a eu un rôle important dans l'amélioration du comportement du système.

Le troisième travail intitulé « Application of Qualitative Reasoning to Robotic Soccer » [Fraser et al 04], ses auteurs ont doté les robots joueurs de football par le raisonnement qualitatif. L'équipe de robots joueurs est un système multi agents où chaque agent est doté d'une architecture hybride ayant deux couches, une couche inférieure réactive et l'autre supérieure proactive. A partir de l'image du terrain prise de haut par un capteur, les informations quantitatives relatives aux emplacements des joueurs et leurs relations par rapport à eux-mêmes et par rapport au ballon par rapport aux différentes zones du terrain et la position du ballon dans le terrain sont traduites à des informations qualitatives décrivant des relations spatiales topologiques et d'orientation entre les robots et les objets du terrain (Ballon, différentes régions du terrain) par le langage de prédicats. Les informations de l'environnement perçues par les robots au moyen de leurs senseurs sont de type quantitatif et servent à interpréter ces prédicats pour déduire l'état du monde observé et déduire le plan d'actions à entreprendre par chaque robot. Ce système est un exemple d'utilisation du raisonnement qualitatif dans un monde réel. La représentation instantanée du monde physique par un modèle qualitatif offre l'avantage de réduire considérablement l'espace de recherche de solutions possibles et offre au système une meilleure robustesse en prenant en charge les variations et le bruit générés par les capteurs.

### 4.3 Apports du paradigme agent au raisonnement qualitatif

Le premier travail de recherche qui montre l'avantage de l'application du paradigme agent au raisonnement qualitatif est celui de Christian Krey du Centre Allemand de Recherche en Intelligence Artificielle (DFKI) [Krey 2001]. Cet article propose un système multi agents pour le raisonnement qualitatif spatial baptisé « SPACE » qui est un guide touristique mobile. Les tâches principales du système de raisonnement qualitatif spatial sont l'identification des objets, localisation des objets, description des dimensions des objets et traduction graphique des localisations des objets. Chacune de ces tâches est assurée par un agent du système et les requêtes utilisateurs sont gérées par l'agent d'ordonnancement. SPACE utilise un serveur de base de données géographique (SIG) et une base de données GPS de la ville allemande Heidelberg où le système a été testé. Les résultats obtenus après deux années de développement et d'amélioration de ce système montrent que la modélisation de ce système de raisonnement qualitatif spatial par un S.M.A a doté le processus de raisonnement qualitatif spatial de flexibilité et d'extensibilité.

Le deuxième travail de recherche [Pyka & Grebel 2003] est dû à Andreas Pyka de l'université de Augsburg en Allemagne et de Thomas Grebel de CNRS France. Il s'agit d'étudier le changement qualitatif (qualitative change) de l'activité économique. Pour cela, les auteurs considèrent le processus d'entrepreneuriat qui est un facteur très important dans l'évolution économique et le modélisent par un système multi agents. Le processus d'entrepreneuriat est un processus dynamique et complexe et met en jeu des considérations sociales, économiques, politiques, psychologiques et même culturelles. Les auteurs déterminent d'abord les variables de ce processus qui sont : les acteurs (personnes physiques : entrepreneurs et personnes morales : entreprises) dont le rôle et la prise de décision de différentes natures telle la décision de production, de ventes, de partenariat, de recrutement, et de licenciement, etc. La qualité de ces décisions est en relation avec les moyens matériels et les qualités morales de l'entrepreneur tels le degré de la motivation ou l'ambition, l'éducation et le savoir faire et les moyens financiers. Les décisions d'entrepreneuriat ne sont pas des faits isolés mais influent et sont influencés par le contexte et les interactions sociales. La décision de création d'une entreprise et donc soit adoptée et intégrée au réseau d'opérateurs socio-économique ou simplement rejetée. L'évolution de ce réseau est aléatoire et sujet aux variables aléatoires endogènes (socio économiques) et exogènes (politiques). Les mesures qualitatives opérées sur ce réseau d'opérateurs socio économique permet d'estimer et de prédire le changement qualitatif dans l'activité économique.

La simulation de ce processus par un système multi agents et les résultats ayant été obtenus montrent l'avantage et l'apport du paradigme multi agent dans la modélisation de ce système complexe mettant en interaction plusieurs disciplines.

### 4.4 Conclusion

Dans ce chapitre, nous avons fait un survol des travaux de recherche mariant les systèmes multi agents et le raisonnement qualitatif pour mettre en évidence les relations entre

les deux technologies et afin de mesurer l'apport de l'une des technologies par rapport à l'autre. Les travaux que nous avons présentés sont typiques et significatifs.

En ce qui concerne l'apport des approches de raisonnement qualitatif à la technologie des systèmes multi agents, nous avons commencé par présenter les travaux menés dans l'approche de raisonnement qualitatif par l'argumentation où nous avons montré que cette approche, en tant que mode de raisonnement qui tolère le conflit et l'incertitude des connaissances, permet aux agents logiciels de réviser leurs croyances [Fox & Ambler 1992] [Loui 1987], [Pollock 1987], [Prakken and Sartor 2001], [Simari and Loui 1992], de communiquer [Maudet and Chaib-draa 2003] et de négocier [Sycara 1990] [Sillince 1994] [Kraus et al 1998] [Parsons et al 1998] [Amgoud et al 2000a] [Amgoud and Cayrol 2000] [Jung et al 2001], et de planifier leurs tâches et délibérer leurs plans [Kakas & Moratis 2003]. La communication par l'argumentation à l'aide des jeux de dialogues et des plans d'argumentation constitue un axe de recherche actuellement très actif.

D'autre part, le travail de [Marik et al 1997] montre que le raisonnement qualitative peut être utilisé pour améliorer le comportement d'un système multi agents par l'équilibrage de la charge de travail sur l'ensemble des agents ainsi que par la réduction du nombre de communications inter agents sur le réseau.

Un autre travail [Frazer et al 2004] où le raisonnement qualitatif est appliqué dans le domaine de la robotique ou un ensemble de robots joueurs se disputent une rencontre de football. L'ensemble est vu en tant qu'un système multi agents où chaque agent est de type hybride. Le monde observé est capté, d'une manière continue, à l'aide d'un senseur puis l'image obtenue est représentée qualitativement sous formes de prédicats décrivant des relations topologiques, d'orientation et de distance entre les robots. Les perceptions quantitatives de ces derniers permettent d'instancier les prédicats représentant le monde observé et donc de déclencher les actions dont les conditions sont vérifiées. La planification des ces actions par la couche proactive de l'agent permet d'arrêter le plan adéquat à exécuter.. Cette application nous renseigne du pouvoir d'abstraction des représentations qualitatives car une variable qualitative correspond à une infinité de valeurs quantitatives, un état qualitatif correspondant à un nombre très important d'états quantitatifs et à un scénario qualitatif correspondant à une infinité de solutions quantitative. Ce pouvoir d'abstraction confère aux solutions obtenues une robustesse et une efficacité.

Dans le deuxième volet de ce chapitre, ont été présentés deux travaux de recherche qui concernent l'utilisation de la technologie agent pour le raisonnement qualitatif. Le premier travail [Krey 2001] consiste en l'implémentation d'un guide touristique « virtuel » utilisant les techniques de raisonnement qualitatif spatial par un système multi agents. Ce cas d'utilisation montre que le paradigme agent a doté le raisonnement qualitatif spatial de la souplesse et de l'extensibilité.

Le deuxième travail [Pyka & Grebel 2003] , dans le cadre du même volet, présente la modélisation du processus d'entrepreneuriat par un système multi agents afin de décrire et de prédire le changement qualitatif dans l'activité économique d'un pays. Ce processus met en jeu plusieurs disciplines qui sont les sciences économiques et sociales, les sciences politiques et les sciences psychologiques. Les agents des systèmes sont les acteurs représentés par les entrepreneurs, les groupes et les firmes qui composent ensemble « le réseau d'entrepreneuriat ». L'étude de la dynamique de ce réseau et sa simulation par un système multi agents a prouvé une fois encore la force de ce paradigme à pouvoir modéliser les systèmes complexes.

## CHAPITRE 5 : CONTRIBUTIONS

### 5.1 Introduction

Ce dernier chapitre est dévolu à la présentation des contributions que j'ai pu réalisées dans le cadre de cette thèse. La première contribution s'inscrit dans l'axe de recherche visant à doter l'agent d'une capacité d'interaction lui permettant de mener des « conversations » ou des dialogues en présence d'informations incomplètes et conflictuelles. Cet axe de recherche s'appelle raisonnement par l'argumentation. L'argumentation, comme nous l'avons mentionné dans le premier chapitre est un style de raisonnement qualitatif non monotone c'est à dire que les agents sont amenées à réviser leur croyances en cours de l'interaction sur la base de nouvelles informations argumentées issues des parties participantes au dialogue. Pour pouvoir s'engager dans un processus d'argumentation, les agents concernés doivent être dotés d'un langage, d'un protocole de communication et d'une architecture interne adéquate. Cette première contribution montre l'utilité d'utilisation du raisonnement qualitatif basée argumentation dans le comportement social d'un système multi agents.

La deuxième contribution quant à elle, s'inscrit dans l'axe de recherche intitulé raisonnement qualitatif temporel qui est une des techniques du raisonnement qualitatif permettant le raisonnement en l'absence d'informations temporelles quantitatives exactes et complètes. Comme nous l'avons vu au premier chapitre, cette technique de raisonnement est algébrique car elle est basée sur une algèbre qualitative. Raisonner qualitativement sur des informations temporelles revient à résoudre un problème de satisfaction de contraintes temporelles qualitatives. Ces connaissances qualitatives temporelles sont représentées par un modèle qualitatif appelé réseau de contraintes temporelles sur lequel sont effectuées les tâches de raisonnement qualitatif, qui sont la consistance d'arcs, la consistance de chemins et la recherche de scénarios possibles. C'est-à-dire les solutions qualitatives au problème en question. L'objectif visé par cette contribution est de montrer l'apport du paradigme multi agents au raisonnement qualitatif.

La troisième contribution consiste à la proposition d'un algorithme pour le raisonnement qualitatif causal dans les cartes cognitives permettant de lier les concepts-effets et leurs influences désirées à des concepts-causes avec leurs influences. Cet algorithme permet de compléter l'inférence classique dans les cartes cognitives basée sur la réponse à la question « Qu'arrivera -t-il si ? ».

Ce chapitre est structuré comme suit. La deuxième, la troisième et la quatrième section seront dévolues à la présentation de mes contributions. Ces trois sections sont organisées de la manière suivante : Après avoir posé la problématique et décrit les travaux similaires, je donne la description de la solution préconisée et je termine par donner une conclusion et des perspectives. Finalement, ce chapitre sera clôturé par une conclusion.

## **5.2. Première contribution : Un protocole de dialogue entre agents basé argumentation**

### **5.2.1 Problématique**

L'utilisation de l'argumentation comme style de raisonnement non monotone pour prendre en charge les aspects de raisonnement dans les systèmes multi agents lorsque ces derniers disposent d'informations incomplètes, au sujet de leur environnement et au sujet des autres agents, a connu beaucoup de travaux de recherche dans les domaines de la communication inter agents [Maudet and Chaib-draa 2003], [Hulstij, 2001], [McBurney et al. 2003], [Pollock 1987], [Hitchcock et al. 2001] dans la planification et la délibération [Kakas & Moraitis 2003][Hulstijn et Van der Torre 2004] [Rueda et al. 2002] et dans la négociation entre les agents [[Sycara 1990] Kraus et al 95] [Amgoud et al.1998] [Kraus et al 1998] ] [Parsons and Sierra 98] [Sadri et al 2001] [Dignum et al. 2001] [Hitchcock et al. 2001] [Moulin et al 2002] [Maudet and Chaib Draa, 2003] [Rahwan et al 2003] [Ramchurn et al 2003] [Amgoud and Kaci 2004] ] [Rahwan et al 2004]. [ Chaib Draa et al. 2003] . La grande partie de ces travaux a été surtout dédiée pour traiter l'interaction basée sur l'argumentation dans le système multi agents, notamment la négociation car l'interaction des agents ayant des informations incomplètes au sujet d'eux-mêmes et au sujet de l'environnement donne très souvent naissance à des situations de conflits.

D'autres travaux de recherche ayant comme objectif de créer des systèmes d'argumentation ont eu comme résultats plusieurs modèles logiques d'argumentation [Simari & Loui 1992] [Dung 1993][Verheij 1996][ Bondarenko et al 1997][Amgoud & Cayrol 2002][Amgoud & Prade 2005]. La plupart de ces modèles d'argumentation n'ont pas encore fait l'objet d'une intégration dans le mécanisme de raisonnement de l'agent. [Rahwan et al 2004]]

Dans un système multi agents, le protocole de demande d'exécution d'une action ou d'une ressource menant à son exécution revêt d'une grande importance car il permet d'incarner la coopération entre les agents du système leur permettant d'achever leurs tâches.

L'objectif visé est de proposer un dialogue de demande d'action (Request for action dialogue, en Anglais) entre agents utilisant le modèle logique d'argumentation de Amgoud et Cayrol [Amgoud et Cayrol 2002] et qui intègre un processus de persuasion entre agents, à l'encontre des travaux qui existent déjà. Ce protocole a été spécifié et validé formellement par le langage Maude [Meseguer 1992].

### **5.2.2 Travaux similaires**

Dans le premier travail [Sadri et al 2001], les auteurs ont proposé un protocole de demande d'une ressource entre deux agents « égoïstes ». Pour réaliser leurs buts, chaque agent utilise ses croyances afin d'élaborer un plan qui met en jeu un certain nombre de ressources.

Chaque plan est assigné une fonction de coût mesurant le nombre de ressources manquantes. Les agents utilisent dans leur dialogue, entre autres, des locutions issues de la théorie de l'argumentation qui sont : la locution « promettre », visant à offrir une ressource, la locution « défier », visant à demander une justification et la locution « justifier » répondant à la locution de défi. Un agent qui n'a pas besoin d'une ressource doit le céder au premier agent demandeur ou il peut l'offrir en échange d'une autre ressource. Quand un agent ne peut ni céder ni échanger une ressource, il a la possibilité de proposer un autre plan à l'agent expéditeur de la requête utilisant des ressources différentes. Le plan proposé est accepté si la valeur de la fonction de coût lui est associée est inférieure à celle du plan d'origine.

Ce dialogue de demande d'une ressource met en jeu un ensemble de locutions argumentatives mais lorsque les agents entre en situation de conflit et que l'un des agents propose un autre plan à l'autre agent, alors l'acceptation ou le rejet de cette proposition n'est basée que sur une fonction de coût.

Le deuxième travail [Chaib Draa et al 2003] inspiré par la théorie d'argumentation de Walton et Krabbe [Walton and Krabbe 1995] et la théorie des engagements sociaux pour reformuler un protocole de demande d'exécution d'une action de Winograd and Flores [Winograd and Flores 1987] basé sur les engagements. Le dialogue proposé se compose d'un ensemble de dialogues de base de la théorie de l'argumentation qui sont : request (gr), Offer (go), information (gi), ask (ga) et persuasion (gp). pour ce faire, quatre opérateurs sont définis sur ces dialogues de base qui sont l'opérateur de séquencement(;) de choix(), et de pré séquencement ( $\rightarrow$ ). L'expression du dialogue composé est de la forme :

$$(gr;(go)^*) \rightarrow ((gi|ga) < (gp)^*).$$

Introduire les jeux de dialogue issus de la théorie d'argumentation, en tant que modèles de dialogue entre humains, pour spécifier les dialogues inter agents est une approche très prometteuse. Cependant tous les dialogues de base composant le dialogue en question ont été spécifiés, excepté le dialogue de base « persuasion » .

Dans le premier travail, la résolution de conflit entre agents n'est pas basée sur un processus de persuasion mais basé simplement sur une fonction de coût. D'autre part, le deuxième travail envisage un processus de persuasion entre les agents comme composant important du dialogue, mais malheureusement il n'y a pas été spécifié d'une façon détaillée.

### 5.2.3 Description du protocole proposé ARGDIAL

Le dialogue inter agents proposé est basé argumentation et inclut un processus de persuasion (voir Figure 5.1). Le système multi agents est composé de deux agents cognitifs et argumentatifs ayant la même structure. Les agents communiquent entre eux en utilisant les locutions indiquées dans le tableau 5.1.). Les agents utilisent le modèle d'argumentation de [Amgoud et cayrol 2002] pour générer, évaluer et sélectionner les arguments.

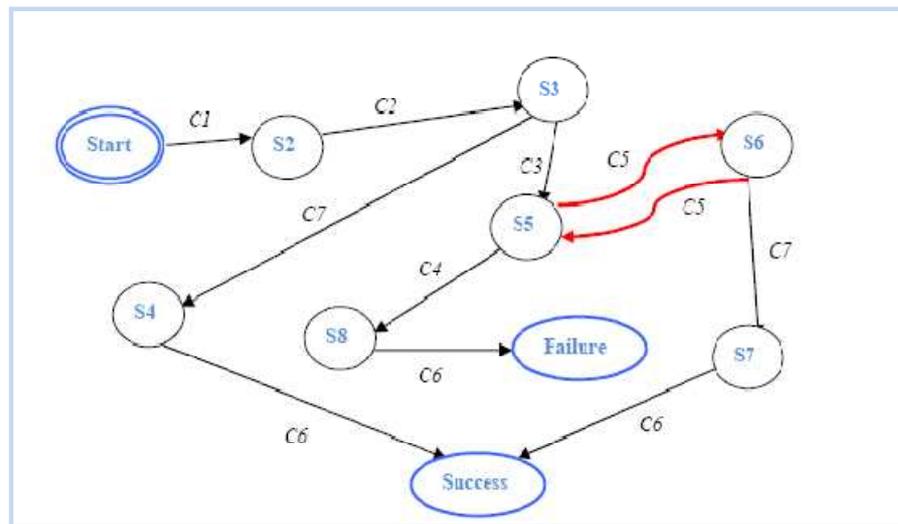


Figure 5.1 : Machine d'états finis du protocole ARGDIAL

Ab br.	Locution syntax	Locution meaning
<i>C1</i>	<i>InformOpen (A, B)</i>	The initiator agent <i>A</i> informs the participant agent <i>B</i> that the dialogue is opened and invites it to formulate a request.
<i>C2</i>	<i>Request (B, A, Tsk)</i>	The participant agent <i>B</i> requires from the initiator agent <i>A</i> to carry out the task <i>Tsk</i> .
<i>C3</i>	<i>Criticise (A, B, Arg)</i>	The agent <i>A</i> criticise agent <i>B</i> about the execution of the task <i>Tsk</i> and presents to him the argument <i>Arg</i> .
<i>C4</i>	<i>Retract (A, B)</i>	agent <i>A</i> sends to the agent <i>B</i> a notification of cancellation of its request
<i>C5</i>	<i>Argue (A, B, Arg)</i>	The agent <i>A/B</i> presents to <i>B /A</i> the argument <i>Arg</i> .
<i>C6</i>	<i>InformClose (A, B)</i>	The initiator agent <i>A</i> informs the participant agent <i>B</i> that the dialogue is closed.
<i>C7</i>	<i>Agree (A, B)</i>	Agent <i>A</i> sends to the agent <i>B</i> a notification of acceptance of its request

Tableau 5.1 : Locutions communicatives utilisés dans ARGDIAL

Les composants de la structure interne de chaque agent et les relations entre eux sont donnés par la figure 5.2

### 5.2.4 Implémentation

Ce protocole d'interaction ARGIAL est spécifié et validé par le langage formel MAUDE [Meseguer 1992]. Figure 5.3 montre une partie du code développé. Elle visualise les règles de réécriture qui décrivent le protocole de dialogue dans son état intermédiaire S5, l'initiateur reçoit un argument envoyé par le participant et l'agent initiateur accepte d'exécuter la tâche de l'agent participant car il n'a pas un argument plus fort à présenter

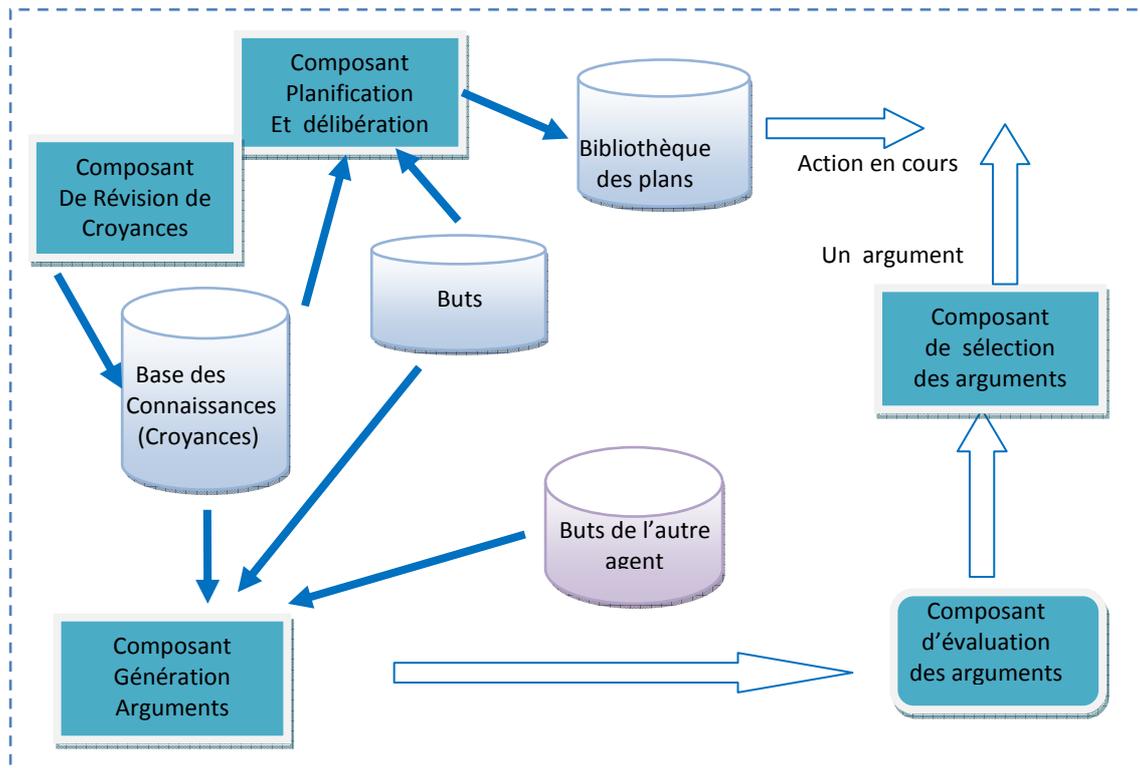


Figure 5.2 : Architecture conceptuelle interne d'un agent argumentatif participant au dialogue ARGDIAL.

et le protocole de dialogue termine avec l'état succès. . De plus, figure 5.3 montre la réécriture limitée (après 3 et 4 étapes de réécriture) de la configuration de la figure 5.4 et de la figure 5.5 respectivement. Le résultat de la réécriture de telle des ces configurations initiales est illustré par la figure 5.6. Le premier résultat illustre que l'initiateur accepte d'accomplir la tâche et le protocole de dialogue se trouve dans son état intermédiaire S6. Tandis que, le deuxième résultat montre le protocole de dialogue dans son état intermédiaire S5 et un message « retract » est émis pour annuler la requête.

### 5.2.5 Conclusion et travaux futurs

Ma première contribution consiste en la proposition d'un protocole de dialogue de demande d'exécution d'une action qui utilise un modèle d'argumentation formel et qui intègre un processus de persuasion contrairement aux travaux similaires existants. L'intégration de la persuasion dans le dialogue proposé assure la modification de l'état mental de l'agent ne pouvant pas présenter un argument plus fort (convaincre) l'autre agent, qui est d'ailleurs l'esprit même d'une « conversation argumentative ». En plus, le dialogue proposé a été spécifié et validé par le langage formel MAUDE [Meseguer 1992].

```

Java - DP.maude - Eclipse SDK
File Edit Navigate Search Project Run Maude Window Help
Maude Console
Ready.
Advisory: redefining module COAL-LIST.
Advisory: redefining module KNOWLEDGE-BASE.
Advisory: redefining module ARGUMENT.
Advisory: redefining module ARGUMENT-INFORMATIONS.
Advisory: redefining module ARGUMENT-LIST.
Maude> Advisory: redefining module DIALOG-PROTOCOL.
rewrite [4] in DIALOG-PROTOCOL : ||< P : Agent | GList : {G2 : G4 : G6},ArgList
: (Arg(Sup2, Concl(Arg2, 14)) :: Arg(Sup4, Concl(Arg4, 11)) :: Arg(Sup6,
Concl(Arg6, 12))) > < D : Dialog | Initiator : I,Participant : P,State : |
Start ; Initial| >| < I : Agent | GList : {G1 : G3 : G5},ArgList : {Arg|
Sup1, Concl(Arg1, 15) :: Arg(Sup3, Concl(Arg3, 13)) :: Arg(Sup5, Concl|
Arg5, 5)|| >| InformOpen(I, P) .
rewrites: 101 in 111753ms cpu |15ms real |0 rewrites/second
result Configuration: < I : Agent | GList : {G1 : G3 : G5},ArgList : {Arg|Sup3,
Concl(Arg3, 13)| :: Arg(Sup5, Concl(Arg5, 5))} > < P : Agent | GList : {G2
: G4 : G6},ArgList : (Arg(Sup4, Concl(Arg4, 11)) :: Arg(Sup6, Concl(Arg6,
12))) > < D : Dialog | Initiator : I,Participant : P,State : |S6 ;
Intermediate| > Agree(I, P, Tsk)
rewrite [3] in DIALOG-PROTOCOL : ||< P : Agent | GList : {G2 : G4 : G6},ArgList
: (Arg(Sup2, Concl(Arg2, 14)) :: Arg(Sup4, Concl(Arg4, 11)) :: Arg(Sup6,
Concl(Arg6, 12))) > < D : Dialog | Initiator : I,Participant : P,State : |
Start ; Initial| >| < I : Agent | GList : {G1 : G3 : G5},ArgList : {Arg|
Sup1, Concl(Arg1, 15) :: Arg(Sup3, Concl(Arg3, 13)) :: Arg(Sup5, Concl|
Arg5, 5)|| >| InformOpen(I, P) .
rewrites: 98 in 111739ms cpu |1ms real |0 rewrites/second
result Configuration: < I : Agent | GList : {G1 : G3 : G5},ArgList : {Arg|Sup3,
Concl(Arg3, 13)| :: Arg(Sup5, Concl(Arg5, 5))} > < P : Agent | GList : {G2
: G4 : G6},ArgList : (Arg(Sup2, Concl(Arg2, 14)) :: Arg(Sup4, Concl(Arg4,
11)) :: Arg(Sup6, Concl(Arg6, 12))) > < D : Dialog | Initiator : I,
Participant : P,State : |S5 ; Intermediate| > Retract(I, P, Tsk)
Maude>

```

Figure : 5.3 : Résultats des réécritures limitées des configurations initiales de La figure 5.4 et de la figure 5.5

```

InformOpen(I, P)
< I : Agent | GList : (G1 : (G3 : G5)), ArgList : (Arg(Sup1, Concl(Arg1, 15)) :: (Arg(Sup3, Concl(Arg3,
13)) :: Arg(Sup5, Concl(Arg5, 5)))) >
< P : Agent | GList : (G2 : (G4 : G6)), ArgList : (Arg(Sup2, Concl(Arg2, 14)) :: (Arg(Sup4, Concl(Arg4,
11)) :: Arg(Sup6, Concl(Arg6, 12)))) >
< D : Dialog | Initiator : I, Participant : P, State : (Start ; Initial) >

```

Figure 5.4 : une configuration initiale

```

InformOpen(I, P)
< I : Agent | GList : (G1 : (G3 : G5)), ArgList : (Arg(Sup1, Concl(Arg1, 15)) :: (Arg(Sup3, Concl(Arg3,
13)) :: Arg(Sup5, Concl(Arg5, 5)))) >
< P : Agent | GList : (G2 : (G4 : G6)), ArgList : (Arg(Sup2, Concl(Arg2, 14)) :: (Arg(Sup4, Concl(Arg4,
11)) :: Arg(Sup6, Concl(Arg6, 12)))) >
< D : Dialog | Initiator : I, Participant : P, State : (Start ; Initial) >

```

Figure 5.5 : une configuration initiale

```

oal [C512] : Argue(I, P, Ar)
  < I : Agent | GList : GLI, ArgList : ArLI >
  < P : Agent | GList : GLP, ArgList : ArLP >
  < D : Dialog | Initiator : I, Participant : P, State : (S5 ; Intermediate) >
  =>
  < I : Agent | GList : GLI, ArgList : ArLI >
  < P : Agent | GList : GLP, ArgList : ArLP >
  < D : Dialog | Initiator : I, Participant : P, State : (S6 ; Intermediate) >
  Agree(I, P, Tsk)
  if ArgumentStoreStrong(Ar, ArLI) == NoArgument..

al [C72] : Agree(I, P, Tsk)
  < D : Dialog | Initiator : I, Participant : P, State : (S6 ; Intermediate) >
  -> < D : Dialog | Initiator : I, Participant : P, State : (S7 ; Intermediate) >
  InformClose(I, P) ..

al [C63] : InformClose(I, P)
  < D : Dialog | Initiator : I, Participant : P, State : (S7 ; Intermediate) >
  => < D : Dialog | Initiator : I, Participant : P, State : (Success : Final) > ..

endal

rew [4] InformOpen(I, P)
  < I : Agent | GList : (G1 : |G3 : G5|), ArgList : (Arg(Sup1, Concl(Arg1, 1)) :: |Arg(Sup3, Concl(Arg3, 13)) :: |Arg(S
  < P : Agent | GList : (G2 : |G4 : G6|), ArgList : (Arg(Sup2, Concl(Arg2, 19)) :: |Arg(Sup4, Concl(Arg4, 11)) :: |Arg(S
  < D : Dialog | Initiator : I, Participant : P, State : (Start : Initial) > ..

rew [3] InformOpen(I, P)
  < I : Agent | GList : (G1 : |G3 : G5|), ArgList : (Arg(Sup1, Concl(Arg1, 15)) :: |Arg(Sup3, Concl(Arg3, 13)) :: |Arg(S
  < P : Agent | GList : (G2 : |G4 : G6|), ArgList : (Arg(Sup2, Concl(Arg2, 14)) :: |Arg(Sup4, Concl(Arg4, 11)) :: |Arg(S
  < D : Dialog | Initiator : I, Participant : P, State : (Start : Initial) > ..

```

Figure 5.6: Une partie du code MAUDE développé

Par ailleurs, notre contribution a quelques limites et insuffisances, à compléter dans les travaux futurs, qui sont :

- Tenir compte de l'aspect temps dans le dialogue. Par exemple la durée de temps tolérée avant qu'un agent ne reçoive une réponse à sa demande et aussi le délai pour qu'un agent justifie sa demande.
- Pour des raisons de performance, la liste des arguments candidats doit être maintenue et mise à jour automatiquement après une révision de croyances de l'agent.
- Etendre notre protocole de dialogue pour une utilisation dans les systèmes multi agent ouverts où les agents sont malveillants, et peuvent « tricher » et « mentir » pour achever leurs buts.
- Vérification formelle des spécifications produites du protocole de dialogue en utilisant le « model checker » du langage Maude. [Meseguer 1992].

**5.3 Deuxième contribution :** Une architecture multi agents pour le raisonnement qualitatif temporel.

### 5.3.1 Problématique

L'information temporelle est particulièrement présente dans la description de la dynamique des objets et évènements du monde qui nous entoure. La planification et l'ordonnancement des tâches sont deux applications de ce type de raisonnement. Lorsque l'information temporelle numérique disponible fait défaut ou elle est incomplète, on tente d'appliquer un raisonnement qualitatif temporel au problème donné dans le but d'en trouver des solutions qualitatives acceptables. Il existe deux cadres formels pour le raisonnement temporel qualitatif. L'algèbre des intervalles [Allen 1983] et l'algèbre des points [Vilain et Kautz 1986].

Raisonnement sur l'information temporelle qualitative revient à résoudre un problème de satisfaction de contraintes qualitatives temporelles. Les connaissances temporelles qualitatives sont représentées par un graphe temporel appelé réseau de contraintes qualitatives temporelles (R.C.T.Q) où les nœuds représentent les variables du problème à résoudre, généralement des tâches ou des actions à accomplir auxquelles on affecte des intervalles de temps, et les arcs représentent les relations (contraintes) temporelles de l'algèbre utilisée. Le raisonnement sur ce modèle qualitatif (R.C.T.Q) consiste à forcer la consistance d'arcs et de chemins et de chercher les scénarii possibles, c'est-à-dire les solutions qualitatives au problème considéré.

Comme nous l'avons décrit dans le premier chapitre, l'algèbre d'Allen [Allen 1983] (voir figure 1.2) comporte treize (13) relations binaires de base définies sur des intervalles de temps, qui sont : (meets (m), met\_by(mi), overlaps(o), overlapped\_by(oi), starts(s), started\_by(si), during(d), contains(di), finishes(f), finished\_by(fi), before (b), after(a), equal(e)).

L'expressivité de ce formalisme est élevée car 1) le nombre de relations possibles entre deux intervalles est très important = 213 relations, 2) permet d'exprimer le parallélisme de tâches, grâce aux relations « overlaps », « during », « equal », « starts », « finishes » ; 3) permet d'exprimer le séquençement de tâches grâce aux relations « meets » et « met\_by » ; 4) permet d'exprimer la précédence de tâches, grâce aux relations « before » et « after ».

Cette expressivité élevée engendre malheureusement un problème de complexité de calcul de NP- Hard [Vilain et Kautz 86] relatives aux tâches de raisonnement précédemment citées.

### 5.3.2 Travaux similaires

Pour tenter de résoudre ce problème de complexité de calcul, Vilain et Kautz [Vilain et Kautz 86] ont proposé une algèbre qualitative basée points ayant comme ensemble de relations de base : { <, >, = } d'où on tire les relations possibles : { <, >, =, {>, =}, {<, =}, {<, >, ? }. Dans ce formalisme, La complexité de calcul relative aux tâches de raisonnement est

polynomiale. D'autres classes de travaux, tendant à atténuer la complexité de calcul par la considération de sous classes de l'algèbre d'Allen [Nebel et Bürkert 1995].

Bien que ces algèbres aient résolu le problème de la complexité de calcul des tâches de raisonnement, elles restent tout de même moins expressives que celle d'Allen.

### 5.3.3 L'approche proposée

L'objectif visé est de proposer une approche qui permet d'atténuer la complexité de calcul liée aux tâches de raisonnement mais tout en préservant la force d'expression du langage d'Allen. Le paradigme agent semble être très approprié à prendre en charge cet objectif car il offre des concepts riches tels que la communication, la coopération, le parallélisme qui permettent d'accomplir les tâches de raisonnement sur le réseau de contraintes temporelles qualitatives (R.C.T.Q) et ce, d'une manière parallèle ou/et coopérative.

Notons que l'approche multi agents pour les réseaux de contraintes temporelles numériques (quantitatives) a fait l'objet de plusieurs travaux, citons à titre d'exemple [Jimming et al. 2002][Yoko et al. 2000][Sakamoto et al 2005][Xuan et Ryszard 2007]. Par contre, et d'après mes connaissances, aucun travail n'existe dans la littérature ayant trait au raisonnement qualitatif sur le réseau de contraintes qualitatives temporelles utilisant l'approche multi agents. L'architecture multi agents proposée est illustrée par la figure 5.7

Les agents composant l'architecture et leurs rôles sont décrits comme suit :

#### a) User Interface Agent (UIA)

Interagit avec l'utilisateur à travers une interface utilisateur graphique. Il soumet au système les données à traiter et les requêtes utilisateur qui sont :

- 1 – Vérifier les données en entrée
- 2 -Construire le modèle de la connaissance.
- 3 -Visualiser le modèle de la connaissance
- 4 – Filtrer le modèle de connaissances (vérifier et forcer la consistance d'arcs et la consistance de chemins dans le R.C.T.Q)
- 5 – Chercher les scénarios consistants
- 6- Visualiser les résultats
- 7- exécuter le processus en entier (de 1 à 6).

Pour faciliter la tâche aux autres agents de l'architecture, l'agent d'interface structure les données en entrée selon le format illustré par la figure 5.8

#### b) Data checking Agents (DCAs)

Ces agents ont pour rôle de vérifier les données en entrée et informer l'utilisateur en conséquence. Les erreurs ou les anomalies qui peuvent être détectées sont :

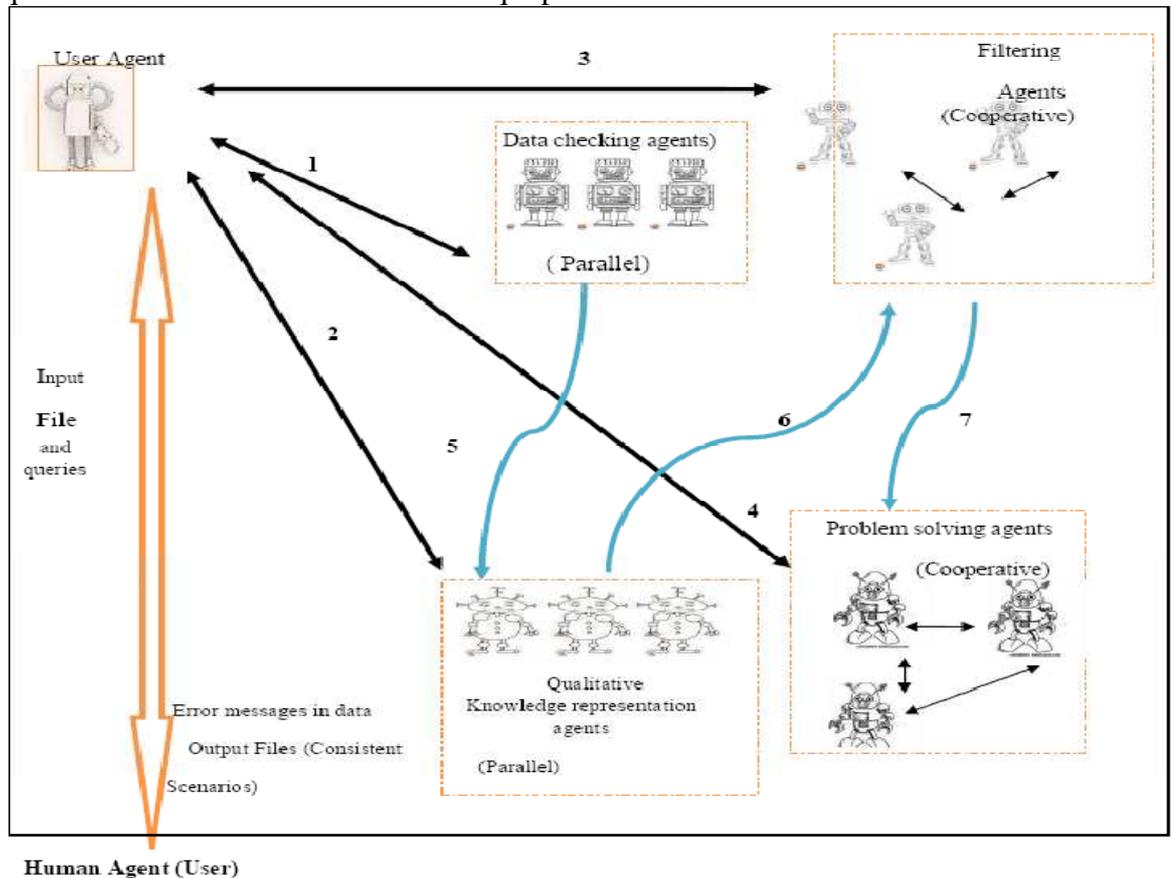


Figure 5.7 : Architecture du système multi - agents

Action avec un intervalle de temps vide,  
 La même action avec deux intervalles différents,  
 Une relation et la relation inverse définies sur le même couple d'actions,  
 Relation vide entre deux intervalles de temps,  
 Apparition de la même action avec le même intervalle plus d'une fois.

La vérification des données par l'ensemble des agents de ce groupe s'effectue d'une manière parallèle. Le nombre de ces agents est défini dynamiquement et dépend de la taille des données en entrée.

### c) Qualitative Knowledge Representation Agents (QKRAs)

Les intervalles sont des informations temporelles imprécises sur l'exécution des actions correspondantes et les contraintes  $C(i, j)$  définissent des relations temporelles entre ces intervalles ( voir figure 5.8). Sur la base de ces données, des agents de représentation de connaissances construisent la matrice d'adjacence (Voir figure 5.9) relative au réseau de contraintes qualitatives temporelles, et ce d'une manière parallèle. Le nombre de ces agents

dépend de la taille des données temporelles à représenter. Ces agents sont créés et détruits dynamiquement.

		$A_0$	$A_1$	$A_2 \dots$	$A_{N-1}$
		$I_0$	$I_1$	$I_2 \dots$	$I_{N-1}$
$A_0$	$I_0$	x	{ $C_{0,1}$ }	{ $C_{0,2}$ }	{ $C_{0,N-1}$ }
$A_1$	$I_1$	{ $C_{1,0}$ }	x	{ $C_{1,2}$ }	{ $C_{1,N-1}$ }
$A_2$	$I_2$	{ $C_{2,0}$ }	{ $C_{2,1}$ }	x	{ $C_{2,N-1}$ }
....	....	..	..	..	...
$A_{N-1}$	$I_{N-1}$	$C_{N-1,0}$	$C_{N-1,1}$	$C_{N-1,2}$	x

Figure 5.9 : La matrice d'adjacence

$A_0$	$I_0$	{ $C_{01}$ }	$I_1$	$A_1$
$A_1$	$I_1$	{ $C_{12}$ }	$I_2$	$A_2$
.....				
.....				
.....				
$A_{N-1}$	$I_{N-1}$	{ $C_{2N}$ }	$I_{N-1}$	$A_{N-1}$

Figure 5.8 : Données en entrée structurées

d) Filtering Agents (FAs)

Ces agents permettent de filtrer le modèle de connaissances construit afin d'éliminer les cas d'inconsistances d'arcs. Une inconsistance d'arc est une instance d'un couple d'intervalles qui ne vérifie pas la contrainte (relation) temporelle définie sur ces deux intervalles. Ce travail est effectué par ces agents d'une manière coopérative. Pour ce faire, Nous assignons à chaque nœud du réseau de contraintes un agent de filtrage de sorte que les agents directement liés puissent communiquer et partager des informations sur des intervalles temporels de sorte qu'ils puissent détecter les inconsistances locales et les éliminer. Le nombre d'agents de filtrage utilisés dépend également de la taille des données en entrée (exactement égal au nombre d'intervalles) et ils sont créés et détruits dynamiquement. Pour effectuer leurs tâches, les agents de filtrage utilisent l'algorithme de Mackworth décrit par la figure 5.10.

e) Qualitative Problem solving Agents (QPSAs)

Ces agents coopèrent afin de trouver des scénarii consistants au problème représenté par le réseau de contraintes qualitatives temporelles. Un scénario montre la façon dont les intervalles sont arrangés et représente une solution qualitative au problème. Ces scénarii consistants sont obtenus par ces agents en utilisant les algorithmes Q.P.C [Ladkin et Reinfeld 1992] et l'algorithme CONSISTENCY [Nebel 96], décrits par la figure 5.11 et la figure 5.12.

```

** This procedure examines the arc (ij) and deletes all instantiations of the interval Ii
** which are not consistent with the qualitative relation . Ii and Ij are respectively
** time intervals assigned to nodes i, j, Cij is the constraint ( relation) between nodes i and j
** between nodes i and j
*****
Procedure REVISE ( Vi, Vj, Cij)
1. del:= false
2. if (Vi, Vj) ∈ Ii * Ij is not consistent with Cij then do begin
3. delete Vi from Ii
4. del := true
5. return del
*****          Arc consistency algorithm          ****
AC- algorithm
** Input : an arc X with its time intervals Ii and Ij and its relation Cij      ***
** Output : a consistent arc X          ***
1. Q := { (Vi, Vj) ∈ X, i≠j}
2. repeat
3. change :=false
4. for each (Vi, Vj) ∈ Q do
5. change := ( REVISE( Vi, Vj) .OR. change
6. endfor
7. until ( not(change))
8. end

```

Figure 5.10 : Algorithme d'arc consistance de Mackworth

```

*****          path consistency algorithm          *****
QPC algorithm
** input1 : constraint matrix C (N*N)={{cij}} verifying arc consistency.      **
** input2 : Allen's composition table ( 13x13 table)          ***
** Output : a matrix C(N*N) verifying path consistency          *****
1. Repeat
2. S := C
3. For k:=1 to n do
4. for i,j:=1 to N do begin
5. Cij := Cij ⊖ ( Cik ⊗ Ckj)          ** ⊖ : set intersection operation , ⊗ : Allen's composing operation **
6. until S=T
7. end

```

Figure 5.11 : Algorithme QPC

### 5.3.4 Conclusion et futurs travaux

Malgré son expressivité, Le raisonnement qualitatif temporel en utilisant L'algèbre des intervalles d'Allen pose des problèmes de complexité de calcul, de l'ordre NP Hard. Les approches existantes visant à résoudre le problème proposent des formalismes qui réduisent considérablement cette complexité mais malheureusement ils sont d'expressivité faible.

Notre approche vise à atténuer cette complexité sans pour autant toucher à l'expressivité

```

Consistency algorithm
** Input: Matrix C representing an I.A based set of constraints between N nodes
** Output: True if the set is satisfiable, false otherwise.
1. C := QPC ( C )      **          use QPC          **
2. if C contains empty relation then
3. return false
4. else
5. choose an unprocessed relation Cij and
6. split Cij into R1, R2, ..., Rm
7. if no- relation can be split then
8. return true
9. endif
10. for m := 1 to k do
11. Cij := Rm
12. if consistency(C) then return true
13. endif
14. endfor
15. return false
16. endif
17. end

```

Figure 5.12: L'algorithme Consistency

du langage d'Allen. L'approche que nous avons adoptée utilise le paradigme agent car il permet d'appréhender la complexité des problèmes en offrant des concepts tels la coopération et le parallélisme.

Le système multi agents pour le raisonnement qualitatif temporel que nous avons proposé offre l'avantage de couvrir tout le processus : introductions des données qualitatives temporelles, représentation des ces données dans un modèle qualitatif pour pouvoir enfin raisonner sur ce modèle et obtenir les scénarii possibles.

Néanmoins, ce travail reste ouvert à des améliorations faisant l'objet de futurs travaux, notamment :

- Notre approche peut renvoyer seulement les solutions complètes (si elles existent) qui vérifient toutes les contraintes données. Si une solution complète n'existe pas, il serait très intéressant qu'une future extension de notre travail puisse proposer une solution optimale approximative, vérifiant un nombre maximum de contraintes..

- Prendre en considération le cas où les données en entrée sont multi sources.

**5.4 Troisième contribution :** Un algorithme pour le raisonnement qualitatif causal dans les cartes cognitives.

### 5.4.1 Problématique

L'inférence classique d'une carte cognitive consiste à trouver une réponse à la question " Qu'arrivera-t-il si ? » , mais dans la pratique , le décideur a souvent besoin d'avoir une réponse à la question " Si je veux Obtenir. . . Que dois-je faire ? ". Par exemple dans le cas du problème de l'infection virale, le décideur pose intuitivement cette question "Si je veux diminuer l'infection virale, Que dois je faire ? ". L'algorithme proposé permet de déduire les effets à appliquer aux concepts cause afin d'obtenir l'effet désiré sur les concepts effets.

### 5.4.2 Principe de l'algorithme

Après le travail d'expertise visant à construire une carte cognitive (**A**) représentant le système étudié, on obtient la matrice d'adjacence correspondante à la carte cognitive construite. Le vecteur  $\mathbf{E}_k$  de taille  $N$ , égal au nombre de concepts, contient les valeurs d'effet des  $N$  concepts appartenant à l'ensemble  $\{+, -, 0\}$  ( dont la signification est respectivement effet positif, effet négatif et aucun effet) s'appelle vecteur d'état ou d'excitation et permet d'exciter la matrice d'adjacence (**A**) pour obtenir un nouveau vecteur de  $\mathbf{E}_{k+1}$ . L'excitation de la matrice d'adjacence (**A**) est faite d'une manière itérative jusqu'à ce qu'on obtienne le même vecteur d'état. Nous disons alors que nous avons atteint un comportement typique du système étudié.

### 5.4.3 Description détaillée de l'algorithme

La figure 5.13 donne la description détaillée de l'algorithme proposé. Dans la première étape, nous commençons par construire la matrice d'adjacence (**A**) =  $[a_{ij}]$   $i, j \in [1..N]$ .  $N$  est le nombre des concepts composant la carte cognitive et les éléments  $[a_{ij}]$  appartenant à l'ensemble des valeurs qualitatives  $\{+,-, 0\}$  décrivant les relations causales entre les concepts. La deuxième étape est réalisée par l'initialisation des concepts désirés de la matrice d'adjacence par des valeurs qualitatives appartenant à l'ensemble  $\{+,-, 0\}$ . Étape 3 permet d'exciter itérativement la matrice d'adjacence par le vecteur d'excitation  $\mathbf{E}_k$  pour obtenir un nouveau vecteur d'excitation  $\mathbf{E}_{k+1}$  et le sauver dans une mémoire de travail. Finalement, dans la dernière étape 4, quand le vecteur d'excitation  $\mathbf{E}_{k+1}$  est identique à un autre vecteur appartenant à la mémoire de travail alors un comportement typique du système étudié est obtenu est l'exécution de l'algorithme se termine sinon on réinitialise les concepts désirés à la valeur (+) et les autres concepts ayant des valeurs négatives à la valeur zéro (0) puis le contrôle d'exécution de l'algorithme est transféré vers l'étape 3.

### 5.4.4 Implémentation

Par valider notre algorithme, nous avons modélisé et simulé le processus de l'infection virale par une carte cognitive floue simple. Avec l'aide d'experts virologues, nous avons énuméré vingt trois (23) concepts différents et 30 relations d'influence entre ces concepts

décrivant l'infection virale. Nous donnons ci-après ces concepts aussi bien que les relations d'influence ces concepts.

1. Extract the adjacency matrix  $A = [a_{ij}]$ ,  $i, j \in [1..N]$  /\* from the cognitive map with N concepts\*/
2. Prepare the excitation vector  $E_1(i)$ ,  $i \in [1..N]$  /\* the value nbr.  $i$  represents the state of concept  $i$ . \*/
  - a. Initialize the excitation vector  $E_1(i)$ ,  $i \in [1..N]$  to 0.
  - b. Initialize the desired concepts with a qualitative value in the set  $\{+, -, \}$  /\*\* concepts which one wants to increase /decrease them \*\*/
  - c. Add  $E_1$  to  $M$  ;  $k := 1$  /\*  $M$  is a working memory, contains all the excitation vectors \*/.
3. Make the product  $E_{k+1}(i) = [a_{ij}] * E_k(i)$  /  $i, j \in [1..N]$
4. If  $E_{k+1}$  belongs to  $M$  then End /\* a typical behavior is obtained \*/. Else
  - a. for the vector  $E_{k+1}$ , Affect the desired concepts by the beginning values (see sub-step 2.b)
  - b. Replace negative values of the vector  $E_{k+1}$  by 0.
  - c. Add  $E_{k+1}$  to  $M$
  - d. Increment  $k$  and go to the step 3.

Figure 5.13 : Description détaillée de l'algorithme

- C1. Radioactivité
- C2. Tabagisme
- C3. Conditions Socio-économiques
- C4. Malnutrition
- C5. Médicament immunosuppresseur
- C6. Désordre congénital de l'immunité
- C7. Produits chimiques
- C8. Cancer
- C9. Hygiène
- C10. Immunodépression
- C11. Infection virale
- C12. Infection bactérienne
- C13. Température du corps
- C14. Diabète
- C15. Personnes contagieuses
- C16. Abattage des animaux suspects
- C17. Réservoirs animaux
- C18. Industrie
- C19. Mondialisation
- C20. PH du milieu de virus.
- C21. Vaccination
- C22. Existence des ions positifs autour de la cellule

## C 23. Effet de serre

Les relations qualitatives entre ces concepts sont données comme suit :

- R1. Radioactivité (+) Cancer
- R2. Tabagisme (+) Cancer
- R3. Produits chimiques (+) Cancer
- R4. Cancer (+) Température du corps
- R5. Température du corps (+) Infection virale
- R6. Infection bactérienne (+) Température du corps
- R7. Abattage des animaux suspects (-) Réservoirs animaux
- R8. Réservoirs animaux (+) Infection virale
- R9. . Réservoirs animaux (+) Infection bactérienne
- R10. Industrie (+) Produits chimiques
- R11. Industrie (+) Effet de serre

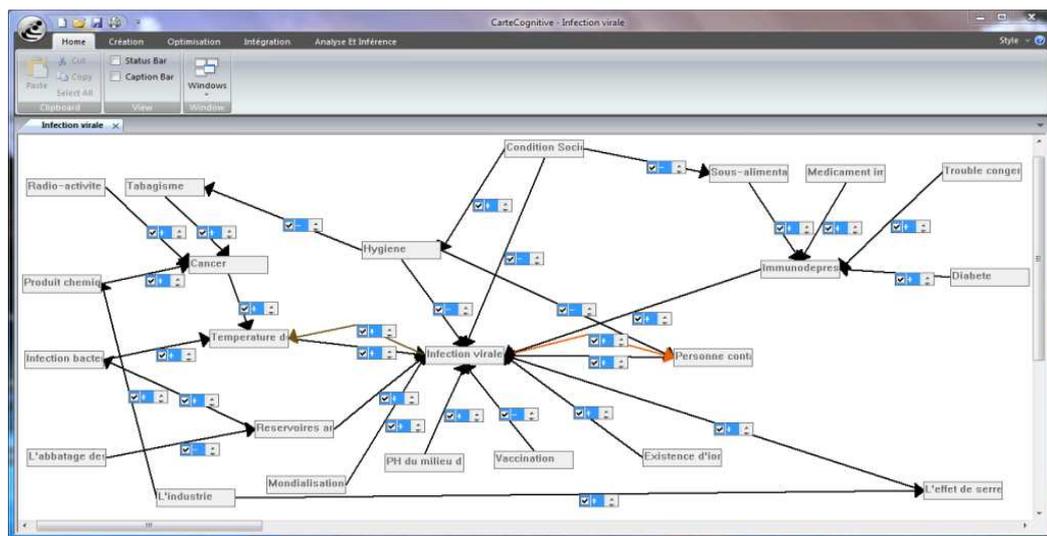


Figure 5.14 : Carte cognitive « infection virale »

- R12. Mondialisation (+) Infection virale
- R13. Hygiène (-) Tabagisme
- R14. Hygiène (-) Infection virale
- R15. Hygiène (-) Personnes contagieuses
- R16. PH du milieu du virus (+) Infection virale
- R17. Vaccination (-) Infection virale
- R18. Existence des ions positifs autour de la cellule (+) Infection virale
- R19. Infection virale (+) Température du corps
- R20. Effet de serre (+) Infection virale
- R21. Infection virale (+) Personnes contagieuses
- R22. Personnes contagieuses (+) Infection virale
- R23. Conditions Socio-économiques (-) Infection virale

- R24. Conditions Socio-économiques (-) Malnutrition
- R25. Conditions socio économiques (+) Hygiène
- R26. Malnutrition (+) Immunodépression
- R27. Médicaments immunosuppresseur (+) immunodépression
- R28. Désordre congénital de l'immunité (+) immunodépression
- R29. Diabète (+) Immunodépression
- R30. Immunodépression (+) Infection.virale

La figure 5.14 montre la carte cognitive « infection virale » construite par l’outil développé

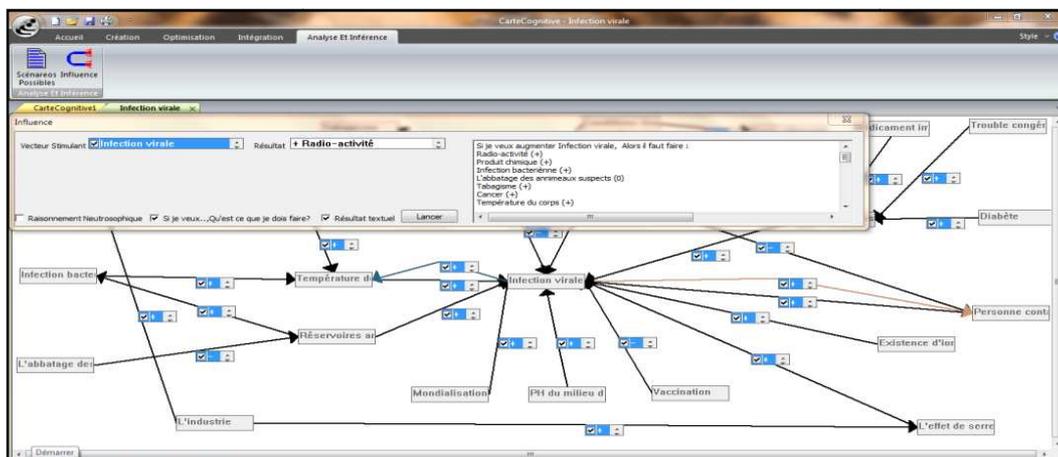


Figure 5.15 : Un scenario d'exécution de l'algorithme proposé

Excitation vector  $E1 = (0,0,0,0,0,0,0,0,0,+,0,0,0,0,0,0,0,0,0)$   
 Adj x  $E1 = (0,0,0,0,0,+,+,+,0,0,-,+,-,-,0,0,+,0,+,+,+)$  → Fonction de seuil →  
 $(0,0,0,0,0,+,+,+,0,+,+,0,0,0,0,+,0,+,+,+)$  =  $E2$   
 Adj x  $E2 = (0,0,+,0,+,+,+,+,+,+,+,+,+,+,+,+)$  → Fonction de seuil →  
 $(0,0,+,0,+,+,+,+,+,0,+,0,+,+,+,+,+,+)$  =  $E3$   
 Adj x  $E3 = (+,+,+,+,+,+,+,+,+,+,+,+,+,+,+,+,+)$  → Fonction de seuil →  
 $(+,+,+,0,+,+,+,+,+,0,+,0,+,+,+,+,+,+,+)$  =  $E4$   
 Adj x  $E4 = (+,+,+,+,+,+,+,+,+,+,+,+,+,+,+,+,+)$  → Fonction de seuil →  
 $(+,+,+,0,+,+,+,+,+,+,0,+,0,+,+,+,+,+,+,+)$  =  $E5 = E4$

Figure 5.16 : Détail du scenario d'exécution

Causes with their influence on the concept "viral infection"	
Radioactivity (+)	Socio economic conditions (0)
Chemical products (+)	Malnutrition (+)
Bacterial infection (+)	Drug immunosuppressor (+)
Demolition of the suspect animals (+)	Congenital disorder of immunity (+)
Nicotinism (+)	Immunodepression (+)
Cancer (+)	Diabetes (+)
Temperature of the body (+)	contagious persons (+)
Animal tanks (+)	Existence of positive ions around the cell (+)
Globalisation (+)	The greenhouse effect (+)
Industry (+)	
Vaccination (0)	
PH of the medium of virus (+)	
Hygiene (0)	

Figure 5.17 : Détail du vecteur d'état E4

Comme exemple de scénario d'exécution, nous allons voir quels sont les concepts-causes et leurs influences sur l'augmentation du concept effet *Infection\_virale*. La figure 5.15 montre un exemple d'exécution de l'algorithme d'inférence proposé tandis que la figure 5.16 et la figure 5.17 montrent respectivement le détail du scénario d'exécution et le détail du dernier vecteur d'état obtenu E4. Ce dernier qui correspond à un comportement typique du système, pourrait être interprété comme suit :

*« Si je veux augmenter l'infection virale (C11) alors il faut augmenter les concepts suivants : (C1, C7, C12, C16, C2, C8, C13, C17, C19, C18, C20, C5, C6, C10, C14, C15, C22, C23). ».*

#### **5.4.5 Conclusion et futurs travaux**

L'algorithme proposé permet de faire l'inférence dans une carte cognitive en répondant à la question « Si je veux... Que dois- je faire ? » qui permet de trouver les concepts - causes et leurs influences qui sont à l'origine des concepts- effets et leurs influences désirées. L'algorithme proposé constitue donc un outil d'explication et complète l'inférence classique dans les cartes cognitives basée sur la prédiction. Comme travaux futurs, nous nous suggérons tout d'abord d'étudier la complexité de cet algorithme dans le but de l'améliorer puis songer à l'appliquer dans un contexte distribué en utilisant l'approche multi agents, par exemple dans la prévision météorologique.

## CONCLUSION GENERALE ET PERSPECTIVES

L'objectif visé par l'étude des approches de raisonnement qualitatif consiste à explorer les possibilités offertes par cette branche de l'IA dans le but de doter l'agent en tant qu'entité intelligente, autonome et inter-agissante avec son environnement, d'une compétence supplémentaire lui permettant de raisonner en l'absence d'informations complètes relativement à un domaine de discours donné.

Cette nouvelle compétence, si acquise par l'agent, confère certainement au paradigme agent en entier une autre dimension et lui permet de mieux appréhender la complexité des systèmes. Ceci nous a orienté à organiser notre travail au tour de trois axes différents, qui sont les approches de raisonnement qualitatif, l'approche multi agents et les systèmes complexes.

L'étude des approches de raisonnement qualitatif nous a permis d'établir une nouvelle taxinomie de ces approches composée des classes suivantes : classe des approches fondées sur la physique qualitative, classe des approches basées sur la logique non monotone et la dernière classe des approches basées sur les algèbres qualitatives.

Les approches basées sur la physique qualitative [Forbus 84] [Kuipers 86] [de Kleer and Brown 84] trouvent leur origine dans les travaux de Hayes [Hayes 79] qui a tenté de proposer des modélisations « naïves » du monde physique et l'axiomatisation des connaissances humaines de sens commun ayant trait aux objets physiques. L'étude des approches de cette première classe des approches qualitatives nous a permis d'établir des critères de comparaison entre ces approches qui sont les concepts véhiculés, les objets manipulés, la sémantique de l'état qualitatif et le comportement qualitatif. La comparaison de ces approches (Tableau 1.1) montre l'intérêt de l'approche centrée processus [Forbus 84] car elle est basée sur une idée intuitive stipulant que tout changement dans un système donné n'est que le résultat de l'interaction entre un certain nombre de processus.

La deuxième classe des approches qualitatives sont basées sur la logique non monotone. Une sous classe de cette classe d'approches est l'approche par argumentation qui renferme plusieurs modèles logiques d'argumentation. La comparaison (Tableau 1.2) de ces modèles d'argumentation sur la base d'un ensemble de critères bien choisis nous a montré que le modèle d'Amgoud et de Cayrol [Amgoud et Cayrol 2002][Amgoud et Cayrol 2005] est le plus approprié pour prendre en charge le dialogue entre deux agents interlocuteurs.

La troisième classe des approches de raisonnement qualitatif renferme des approches basées sur des algèbres qualitatives. Le raisonnement qualitative temporel utilise des algèbres qualitatives temporelles [Allen 1983][Vilain and Kautz 1986] [Nebel et Bürkert 1995], le raisonnement qualitatif spatial utilise des algèbres qualitatives spatiales [Cohen et al. 1996] et le raisonnement qualitatif causal utilise des algèbres qualitatives causales [Axelrod 1976] [Kosko 1986]. Cette classe d'approche constitue un axe de recherche actuellement très actif où

les travaux menés touchent notamment le guidage et la planification des tâches d'un robot dans un environnement réel.

L'étude de la relation systèmes multi agents et raisonnement qualitatif nous a permis de mettre en évidence l'apport de chacun par rapport à l'autre. Concernant l'apport du paradigme agent au raisonnement qualitatif, il est illustré à travers deux travaux de recherche significatifs : Dans le premier travail [Krey 2001], son auteur propose un système multi agents pour le raisonnement qualitatif spatial (SPACE) servant en tant qu'un guide touristique virtuel dont les tâches principales sont l'identification des objets, localisation des objets et description des dimensions de ces objets. Après deux années de développement et d'amélioration de ce système, l'auteur a conclu que la modélisation de son système à l'aide de l'approche basée agents a doté le système en entier de flexibilité et d'extensibilité.

Le deuxième travail de recherche [Pyka & Grebel 2003] consiste en la modélisation du réseau d'entreprenariat, qui est dynamique et ouvert et sujet à plusieurs considérations (économiques, sociales, psychologiques et culturelles) par un système multi agents afin de prédire les changements qualitatifs (comportements qualitatifs) de ce réseau à partir des données quantitatives. La simulation de ce processus et les résultats obtenus montrent encore une fois l'avantage du paradigme agent pour le raisonnement qualitatif.

Pour ce qui est de l'apport du raisonnement qualitatif au paradigme agent, l'approche par argumentation, qui est un style de raisonnement non monotone en présence d'informations incomplètes, incertaines et conflictuelles, permet à l'agent de réviser ses croyances [Fox & Ambler 1992] [Loui 1987], [Pollock 1987], [Prakken and Sartor 2001], [Simari and Loui 1992], de communiquer [Maudet and Chaib-draa 2003], de négocier [Sycara 1990] [Sillince 1994] [Kraus et al 1998] [Parsons et al 1998] [Amgoud et al 2000a] [Amgoud and Cayrol 2000] [Jung et al 2001] et de planifier ses tâches [Kakas & Moratis 2003].

D'autres travaux dans le même cadre montrent que le raisonnement qualitatif peut être utilisé pour améliorer le comportement global d'un système multi agents [Marik et al 1997] par le rééquilibrage de la charge de travail sur les agents et la réduction du besoin de communication entre ces agents.

Le plus important des avantages de l'utilisation du raisonnement qualitatif dans le paradigme agent est résumé par le travail [Frazer et al. 2004] qui consiste à doter l'agent, en tant qu'entité de base, d'un pouvoir d'abstraction élevé lui permettant de manipuler des variables qualitatives (qui correspond chacune à une infinité de valeurs quantitatives), permettant au système modélisé d'avoir des états qualitatifs (qui correspond chacun à un nombre très important d'états quantitatifs) et des scénarios qualitatifs (un scénario correspond à une classe de comportements quantitatifs). Ce pouvoir élevé d'abstraction confère à l'approche multi agents la robustesse et l'efficacité [Frazer et al. 2004].

L'approche basée sur le raisonnement qualitatif offre l'avantage de pouvoir modéliser parfaitement les systèmes complexes [Kuipers 86] [Bredeweg et al. 2009] car les informations quantitatives caractérisant ces systèmes sont souvent incomplètes et imprécises. D'autre part, l'approche multi agents offre pour les systèmes complexes un grand avantage pour modéliser et simuler ces systèmes [Bousquet et al. 1993] [Bura et al. 1993] [Drogoul 1993][Carley, et al 2006] car le paradigme agent dispose de tous les concepts (un agent est une entité active pouvant avoir un comportement réactif et adaptatif et qui interagit avec les autres agents) nécessaires pour prendre en charge cette classe de systèmes.

L'approche basée agents et l'approche basée sur le raisonnement qualitatif offrent toutes les deux des avantages pour modéliser et simuler les systèmes complexes et l'utilisation combinée de ces deux approches permettra d'appréhender cette classe de systèmes et aura certainement son impact positif sur l'évolution de la recherche dans ce domaine.

Ma première contribution intitulée « ARDGIAL : an Argumentation Based Dialogue in Multi agent Systems » se veut un exemple d'utilisation du raisonnement qualitatif dans les systèmes multi agents. Basé sur le modèle d'argumentation de Amgoud et Cayrol [Amgoud et Cayrol 2002], ce travail propose un protocole de dialogue de demande d'action basé argumentation entre deux agents cognitifs qui intègre un processus de persuasion (le demandeur d'action doit savoir justifier sa demande sinon elle sera rejetée et si l'autre agent la critique, c'est-à-dire « ne veut pas » l'exécuter car elle contredit l'un de ses buts, il doit aussi justifier sa critique). Ce protocole est spécifié et validé formellement en utilisant le langage MAUDE [Meseguer 1992]. (Pour les détails, consultez le Chapitre 5)

Ma deuxième contribution intitulé « A Multi agent Based Architecture to Qualitative Constraint Temporal Reasoning » est un exemple d'utilisation de l'approche multi agents dans le raisonnement qualitatif. L'objectif de ce travail consiste à proposer une approche multi agents pour le raisonnement qualitatif temporel basé intervalles d'Allen [Allen 1983] permettant d'atténuer la complexité des tâches de raisonnement tout en préservant toute l'expressivité du formalisme d'Allen L'implémentation de ce système est prévue en utilisant la plateforme multi agents JADE (Java Agent DEvelopment toolkit) [Fabio et al.][FIPA] ou toute autre plateforme offrant des fonctionnalités équivalentes (Pour les détails, consultez le chapitre 5).

Ma troisième contribution intitulée « An Algorithm for Qualitative Causal Reasoning in the Cognitive Maps » propose un algorithme de raisonnement causal dans les cartes cognitives permettant d'identifier les concepts - causes et leurs influences qui sont à l'origine des plusieurs concepts – effets dont les influences sont désirées ou préfixées. Cet algorithme constitue un outil d'explication et de décision et permet donc de compléter la méthode classique de raisonnement causal dans les cartes cognitive basée uniquement sur la prédiction. Pour pouvoir tester notre algorithme, nous avons développée un outil graphique de construction et de

manipulation des cartes cognitives et nous avons pris comme domaine d'application le processus biologique de l'infection virale (Pour les détails, consultez le chapitre 5).

Par ailleurs, les travaux réalisés dans le cadre de cette thèse restent ouverts à de futurs extensions et améliorations, à savoir :

➤ Concernant ma première contribution :

- Tenir compte de l'aspect temps dans le dialogue. Par exemple la durée de temps tolérée avant qu'un agent ne reçoive une réponse à sa demande et aussi le délai pour qu'un agent justifie sa demande.

- Pour des raisons de performance, la liste des arguments candidats doit être maintenue et mise à jour automatiquement après une révision de croyances de l'agent.

- Etendre notre protocole de dialogue pour une utilisation dans les systèmes multi agent ouverts où les agents sont malveillants, et peuvent « tricher » et « mentir » pour achever leurs buts.

- vérification formelle des spécifications produites du protocole de dialogue en utilisant le « model checker » du langage Maude. [Meseguer 1992].

➤ Et pour ma deuxième contribution :

- Notre approche peut renvoyer seulement les solutions complètes (si elles existent) vérifiant toutes les contraintes données. Si une solution complète n'existe pas, il serait très intéressant qu'une future extension de notre travail puisse proposer une solution optimale approximative, vérifiant un nombre maximum de contraintes.

- Prendre en considération le cas où les données en entrée sont multi sources.

➤ Et finalement , pour ma troisième contribution :

- Etudier la complexité de l'algorithme d'inférence proposé dans le but de l'atténuer, le cas échéant.

- Utilisation de l'algorithme dans un cadre distribué par l'utilisation de l'approche multi agents. Ceci constituera un très bon exemple pour la décision qualitative et collective d'un groupe d'agents.

## REFERENCES BIBLIOGRAPHIQUES

- [Aknine et al 2000] Aknine S., Pinson S., Shakun M. F. "Des méthodes de coordination Multi-agent par Formations de coalitions" 8e Journées Francophones d' Intelligence Artificielle et Systèmes Multi-Agents (JFIADSMA'00), Paris, Hermes., 2000.
- [Aknine 2002] Aknine S. "Strategies and Behaviours of Agents in Multi-phased Negotiations" EC-Web, pp.17-26., 2002.
- [Amgoud et al 1998]. Amgoud, L, N. Maudet, and S. Parsons, "Modelling dialogues using argumentation" . in E. Durfee (ed.), Proceedings of the 4th International Conference on Multi-Agent Systems (ICMAS 1998), IEEE Press: Boston MA, USA, pp. 31–38, 1998.
- [Amgoud and Cayrol 1998]: Amgoud L. and Cayrol C., "On the acceptability of arguments in preference based argumentation". In proceedings of the 4<sup>th</sup> conference on Uncertainty in Artificial Intelligence ( UAI 1998), San Fransico, CA. Morgan., 1998.
- [Amgoud et al 2000a] : Amgoud L., Nicolas M., and Parsons S., "Modelling dialogues using argumentation" . In Dufree. E (ed.) Proceed. of the 4<sup>th</sup> International Conference on Muti agent systems ( ICMAS 2000). Boston MA. IEEE Press. Pp31-38, Kauffman pp 1-7, 2000.
- [Amgoud et al 2000b]: Amgoud L., Parsons., and Maudet Nicols. "Arguments, dialogues and argumentation" In Horn W.(Ed.), proceedings of the European Conference on Artificial Intelligence ( ECAI 2000), Amsterdam, Netherlands, IOS press, pp338-342,2000
- [Amgoud & Cayrol 2002] : Amgoud L., Cayrol C., "A reasoning model based on the production of acceptable arguments". In Annals of Mathematics and Artificial Int, V. 34 - N. 1-3, p. 197-215, 2002.
- [Amgoud 2003] Amgoud, L. "A formal framework for handling conflicting desires," in T. D. Nielsen and N. L. Zhang (eds.), Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 7th European Conference (ECSQARU 2003), Vol. 2711 of Lecture Notes in Computer Science, Springer Verlag: Berlin, Germany, pp. 552–563, 2003.
- [Amgoud and Kaci 2004] Amgoud L. and S. Kaci, "On the generation of bipolar goals in argumentation-based negotiation," in Proceedings of the 1st International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2004), Lecture Notes in Computer Science. Springer Verlag: Heidelberg, Germany, 2004.
- [Amgoud & Prade 2005] Amgoud L, H. Prade, "Handling threats, rewards and explanatory arguments in an unified setting", International Newspaper of Intelligent Systems, Wiley periodical Inc, V 20 N 12, p. 1195-1218, 2005.
- [Axelrod 1997] Axelrod R. "Advancing the Art of Simulation in the Social Sciences" Simulating Social Phenomena, Hegselmann R. and Terna P. (Eds.) Berlin: Springer, pp. 21-40.; 1997.

- [Baeijs et Demazeau 1996] Baeijs C., Demazeau Y. "Les organisations dans les systèmes multi-agents", 1996.
- [Beer et al 1998] Beer M., d'Inverno M., Luck M., Jennings N., Preist C., Schroeder M. "Negotiation in Multi-Agent Systems" Workshop of the UK Special Interest Group on Multi-Agent Systems (UKMAS'98), 1998.
- [Bagnoli 1998] Franco Bagnoli. "Cellular Automata », 1998. arXiv: cond-mat/9810012v1
- [Bourjot et al 1999] Bourjot C., Chevrier V., Bernard A., Kraft B. "Coordination par le Biais de l'Environnement: une Approche Biologique" Journées Francophones d'Intelligence Artificielle Distribuée et Systèmes Multi-Agents (JFIADSMA'99), Hermès. 1999
- [Boissier et al 1999] Boissier O., Guessoum Z., Ocello M. (1999) "Plates-Formes de développement de systèmes multi-agents" Bulletin AFIA., Vol. 39.
- [Boissier 2001] Boissier O. "Modèles et architectures d'agents" Principes et architectures des systèmes multi-agent., J-P Briot et Y.Demazeau, (Eds.) Hermes, Vol. 1, pp. 71-107. , 2001.
- [Bond et Gasser 1988] Bond A. H. and Gasser L. "Readings in distributed artificial intelligence". Kaufmann M., (Eds.) San Mateo, California., 1988.
- [Bonneaud 2009] S. Bonneaud, Des agents modèles pour la modélisation et la simulation de Systèmes complexes, Thèse de Doctorat- Université de Bretagne Occidentale, France 2009.
- [Bondarenko et al 1997]: Bondarenko, A. Dung, P., Kowalski, R. and Toni F. 1997. "An abstract argumenatation theoretic approach to default reasoning". Artificial Intelligence 93, 63-101, 1997.
- [Bousquet et al. 1993] Bousquet F., Cambier C., Mullon C. "Simulating the Interaction between a Society and a Renewable Resource". Journal of Biolog. Systems. 1(2), 1993
- [Bousquet et le Page 2001] Bousquet F., et Le Page C. "Systèmes multi-agents et écosystèmes" Principes et architectures des systèmes multi-agents, J-P Briot et Y.Demazeau, (Eds.) Paris : Hermes, Vol. 1, pp. 235-266, 2001.
- [Boussetta et al 1998] Boussetta S., El Fallah-Seghrouchni A., Haddad S., Moraitis P., Taghelit M. (1998) "Coordination d'agents rationnels par planification distribuée" Revue d'Intelligence Artificielle, Vol 12, n°1. 1998.
- [Bouزيد 2006] : Bouزيد M., " Temps, Agent et vers une extension à la dimension spatial " ; Mémoire H.D.R , UFR des sciences, Département d'Informatique , Université de Caen. 2006.
- [Brandolese et al., 2000] Brandolese A., Brun A., Portioli-Staudacher A. "A Multi-Agent approach for the capacity allocation problem" International Journal of Production Economics, Vol. 66, pp. 269-285, 2000.

- [Bratman et al., 1988] Bratman M. E., Israel D. J., and Pollack M. "Plans and resource-bounded practical reasoning". *Computational Intelligence*, Vol. 4, pp. 349-355, 1988.
- [Bredeweg & Salles 2005] Bredeweg, B. and Salles, P. "The Ants' Garden: Complex interactions between populations and the scalability of qualitative models", *AI Communications*, 18(4), pp 305-317, 2005.
- [Bredeweg et al. 2009] Bredeweg, B., Linnebank, F., Bouwer, A., and Liem, J.,. " Garp3 - Workbench for Qualitative Modelling and Simulation". *Ecological Informatics* 4(5-6), 263-281.2009
- [Busetta et Ramamohanarao, 1998] Busetta P, Ramamohanarao K. "The BDIM Agent Toolkit Design"., Department of Computer Sciences, Victoria. Australia. 1998.
- [Bura et al. 1993] Bura S., Guérin-Pace F., Mathian H., Pumain D. et Sanders L. "Multi-Agents Systems and the Dynamics of a Settlement System." In *Simulating Societies Symposium*, Siena, C. Castelfranchi (Ed.).1993.
- [Cambier 1994] Cambier C. SIMDELTA: Un système multi-agent pour simuler la pêche sur le Delta Central du Niger. Thèse de l'Université de Paris 6., 1994.
- [Carley et al. 2006] Carley, K.M., Fridsma, D.B., Casman, E., et al.: "BioWar: Scalable Agent-Based Model of Bioattacks". *IEEE Transactions on Systems, Man and Cybernetics – Part A: Systems and Humans* 36, pp. 252–265, 2006.
- [Carlos et al 2000]: Carlos Ivan Chesnevar, Ana Gabriela Maguitman, and Ronald Prescott Loui. *Logical Models of arguments*, *ACM computing surveys*, 32(4):337-383, 2000.
- [Chaib-Draa, 1996]: Chaib-Draa, B. "Interaction between Agents in Routine, Familiar and Unfamiliar; situation" *International Journal of Cooperative Information Systems*, Vol 5, pp. 1-25., 1996.
- [Chaib Draa et al. 2001] Chaib-Draa B., Jarras I., Moulin B., "Systèmes multi-agents : principes généraux et applications" *Principes et architectures des systèmes multi-agents*, J.-P. Briot et Y. Demazeau (Eds.) Hermès., 2001.
- [Chaib Draa et gaguet 2002] Chaib-Draa B., Gageut L. "Aspects formels des Systèmes Multi-Agents" *Organisation et applications des systèmes multi-agents*, Hermès, pp 109-132., 2002.
- [Chaib Draa et al. 2003] Brahim C., Marc André Labrie and Nicholas Maudet., "Request for Action Reconsidered as a Dialogue Game Based on Commitments", *LNCS*. Vol 2650/2003, pp 284 -299, 2003.
- [Chang et Woo, 1992] Chang M. K., Woo C. C. "SANP: A Communication level Protocol for Negotiation" *Decentralized A.I.* 3, E. Werner et Y. Demazeau (Eds.), pp. 31-51. ;1992.

- [Cohen et al 1996] : Cohen, A. *Calculi for Qualitative Spatial Reasoning*. In *Artificial Intelligence and Symbolic Mathematical Computation*, L NCS 1138, eds: J Calmet, J A Campbell, J Pfalzgraf, Springer Verlag, 124-143, 1996.
- [Collinot et al. 1996] Collinot A., Ploix L., Drogoul A. "Application de la méthode Cassiopée à l'Organisation d'une équipe de Robots" *Journées Francophones d'Intelligence Artificielle Distribuée et Systèmes Multi-Agents (JFIADSMA'96)*, Hermès, 1996.
- [Clavel 2005 ] : M. Clavel, F. Durán, S. Eker, P. Lincoln, N. Martí-Oliet, J. Mesenguier, and C. Talcott. *Maude Manual (Version 2.1.1)*, April 2005
- Davis R., Smith R. G. "Negotiation as a Metaphor for Distributed Problem Solving" *Readings in Distributed Artificial Intelligence*. Bond A. H. and Gasser L., (Eds.), pp. 333-356.,1980.
- [de Kleer and Brown 1984] J. de Kleer and J.S. Brown. A qualitative physics based on confluences. *Artificial Intelligence*, 24:7–83, 1984.
- [Dechter 2003] R. Dechter. *Constraint Programming*. Morgan Kauffmann Publishers., 2003. Book Elsevier Science. (ISBN 1-55860-890-7).
- [Demazeau 1995] Y. Demazeau., "From interactions to collective behaviour in multi-agents systems" *First European Conference on Cognitive Science*, Saint-Malo, France. 1995.
- Dignum et al. 2001] Dignum, F., B. Dunin-Keplicz, and R. Berbrugge, "Creating collective intention through dialogue," *Logic J. IGPL* vol. 9, no. 2, pp. 289–303, 2001.
- [Drogoul 1993] : Drogoul A. « De la simulation multi-agent à la résolution collective de problèmes. Une étude de l'émergence de structures d'organisation dans les systèmes multi-agents », . Thèse de doctorat, Université Paris 6., 1993.
- [Dung 1993] : Phan Minh Dung. « An argumentation semantics with logic programming with explicit negation ». In *ICLP*, pages 616-630,1993
- [Dung 1995] : Dung P. M . "On the acceptability of arguments and its fundamental role in non - monotonic reasoning and logic programming and n-person games". *Artificial Intelligence- 77*, pp 321-357, 1995.
- [Durfee et Moss 1997] : Durfee E. H., Moss S. "Modelling Bounded Rationality using Evolutionary Techniques" *AISB'97 workshop on Evolutionary Computation*, Manchester, UK, 1997.
- [Doyle 1979] : Doyle. J. « A truth maintenance system ». *Artificial Intelligence*.23,3 (nov)-231-272, 1979.
- [Drogoul 1993] Drogoul A. « De la simulation multi-agent à la résolution collective de problèmes. Une étude de l'émergence de structures d'organisation dans les systèmes multi-agents », Thèse de l'Université Paris 6, 1993.

- [Epstein et Axtell, 1996] Epstein M.E, Axtell R. "Growing Artificial Societies-Social Science from the Bottom Up". Washington: Brookings Institution Press. Cambridge, MA: MIT Press, 1996.
- [Erceau et Ferber, 1991] Erceau J., Ferber J. "L'Intelligence Artificielle Distribuée" La Recherche, Vol. 233., 1991.
- [Fabio et al. ] Fabio Bellifemine, Giovanni Caire, Tiziana Trucco (TILAB, formerly CSELT) Giovanni Rimassa , JADE : Java Agent Development toolkit ., (University of Parma) –Italie.
- [Faratin et al., 2000] Faratin P., Jennings N. R., Buckle P., Sierra C. "Automated negotiation for provisioning virtual private networks using FIPA-Compliant agents" PAAM'00., 2000.
- [Faratin et al., 2001] Faratin P., Klein M., Sayama H., Bar-Yam Y. "Simple Negotiating Agents in Complex Games: Emergent Equilibria and Dominance of Strategies" 8th International Workshop on Agent Theories, Architectures and Languages (ATAL-2001), Seattle, USA.,2001.
- [Ferber 1995] Ferber J. « Les systèmes multi-agents. Vers une intelligence collective ». Paris. 1995
- [Fisher 1995a] Fischer K., Müller J.P., Pischel M. "Unifying control in a layered agent architecture"., IJCAI95, Agent Theory, Architecture and Language Workshop 95, pp. 240-252.Frasson. C.( eds),. Proceedings of the 5<sup>th</sup> international conference on autonomous agents, 324-331, Monreal- Canada, 1995.
- [Fisher 1995b] Fischer K., Müller J. P., Pischel M. "Cooperative transportation scheduling: an application domain for DAI" Journal of Applied Artificial Intelligence (Special Issue on Intelligent Agents – Eds. M. J. Wooldridge & N. R. Jennings). 1995.
- [Fox & Ambler 1992] J. Fox, P. Krause, and S. Ambler, "Arguments, contradictions and practical reasoning," in B.Neumann (ed.), Proceedings of the 10th European Conference on Artificial Intelligence (ECAI-1992), Vienna, Austria, pp. 623–627, 1992.
- [Fraser et al. 2004]: Gordon Fraser, Gerald Steinbauer, and Franz Wotawa. "Application of qualitative reasoning to robotic soccer"., In *18th International Workshop on Qualitative Reasoning.*, Illinois, USA, August 2004.
- [Gardenfors et al 1988] P. Gardenfors, Knowledge in Flux: Modeling the Dynamics of Epistemic States, MIT Press: Cambridge MA, USA, 1988.
- [Georgeff 1987]. M. Georgeff, "Planning," Annu. Rev. Computer Sci. Vol. 2, pp. 359–400, 1987.
- [Georgeff 1995] Rao A., Georgeff M. "BDI Agents: From Theory to Practice" Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95), Menlo Park, California, AAAI Press, pp. 312-319, 1995.

- [Guessoum et Ocelllo 2001] Guessoum Z., Ocelllo M. (2001) "Environnements de développement", Principes et architectures des systèmes multi-agents, Briot J.-P. and Demazeau Y., (Eds.) Paris: Hermes Sciences., 2001.
- [Hage & Verheij 1995]: Hage J and Verheij, B "Reason based logic: a logic for reasoning with rules and reasons". *Computers and Artificial Intelligence*.3,2. 195, 1995.
- [Hart 1951]: Hart H. the ascription of responsibility and rights. In A. Flew Ed., *logic and language* Blackwell, 1951.
- [Hayes 1979] Hayes, P. The naive physics manifesto. In D. Michie (Ed.), *Expert Systems in the Micro-Electronic Age*. Edinburgh University Press. 1979.
- [Huget et Pinson 1998] Huget M.P., Pinson S., "Une typologie des Modèles d'Agents" Document du Lamsade, n° 110. 1998
- [Heidtke, et al. 1998] K. R. Heidtke and S. Schulze-Kremer. "Design and implementation of a qualitative simulation model of lambda phage infection" . *Bioinformatics* 14(1): 81-91. 1998.
- [Hulstij, 2001] J. Hulstij, "Dialogue models for enquiry and transaction," Ph.D. thesis, Universiteit Twente, Enschede, The Netherlands, 2001.
- [Hayes 1985] Hayes, P. "Naive Physics 1: Ontology for liquids". In Hobbs, R., & Moore, R. (Eds.), 1985.
- [Hulstij et Van der Torre 2004] J. Hulstij and L. van der Torre, "Combining goal generation and planning in an argumentation framework," in A. Hunter and J. Lang (eds.), *Proceedings of the Workshop on Argument, Dialogue and Decision at the International Workshop on Non-monotonic Reasoning (NMR 2004)*, Whistler, Canada, 2004.
- [Iwazaki 1997] Y. Iwazaki. "Real-World Applications of Qualitative Reasoning", *IEEE Expert: Intelligent Systems and Their Applications* issue pp. 16-21, May 1997.
- [Jennings et al. 2000] Jennings N. R., Parsons S., Sierra C., Faratin P., "Automated Negotiation" 5<sup>th</sup> International Conference on The Practical Application of Intelligent Agent and Multi-Agent Systems (PAAM-2000), Manchester, UK. 2000.
- [Jennings et al 2001] Jennings N. R., Faratin P., Lomuscio A.R., Parsons S., Sierra C., Wooldridge M. "Automated Negotiation: prospects, methods and challenge" *International Journal of Group Decision and Negotiation (GDN)*, Vol 10, pp. 99-215, 2001.
- [Jimming et al. 2002] Jimming Liu, Han Jing; Y.Y tang . 2002. Multi Agent oriented Constraint Satisfaction., *Artificial Intelligence* (136), pages 104-144., Elsevier., 2002.
- [Jong, et al. 2001] H. de Jong, M. Page, C. Hernandez and J. Geiselman. "Qualitative simulation of genetic regulatory networks: method and application". *Proceedings of IJCAI-2001*.

- [Jung et al 2001]: jung. H., Tambe. M., & Kulkarni. S. , “Argumentation as distributed constraint satisfaction : Applications and results”. In Muller, J. P, André. E., Sen. S. 2001.
- [Kakas and Moraitis 2003]. A. Kakas and P. Moraitis, “Argumentation based decision making for autonomous agents,” in Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2003), Melbourne, Australia, pp. 883–890, 2003.
- [Kari 2005] Jarkko Kari. Theory of cellular automata: A survey. Theoretical Computer Science, 334:3–33, april 2005. ( DOI: 10.1016/j.tcs.2004.11.021. )
- [Karlins and Abelson 1970]: Karlins M and Abelson H. I, “Persuasion: How opinions and attitudes are changed”, 2<sup>nd</sup> edn. Lockwood, 1970.
- [Koning et al. 1998] Koning J.-L., François G., Demazeau Y. "An Approach for Designing Negotiation Protocols in a Multi-Agent System" IFIP'98, Vienne, Austria Budapest, Hungary. 1998.
- [Kraus et al 1995] : Kraus, S., Ambler P, ELvang –Goransson, M., & Fox. J.,1995 “A logic of argumentation for reasoning under uncertainty”, Computational intelligence 11, 113-131. 1995.
- [Kraus et al 1998] : Kraus, S., Sycara, K., & Evenchik, A “Reaching agreements through argumentation : A logical model and implementation”. Artificial intelligence 104(1-2), 1-69, 1998.
- [Kraus 2001] Kraus S. (2001) "Automated Negotiation and decision Making in MultiAgent Environments", Multi-Agent systems and Applications 9<sup>th</sup> ECCAI Advanced Course, ACAI 2001 and Agent Link's 3rd European Agent Systems Summer School, EASSS 2001.
- [Kreifelts & von Marital 1991] Kreifelts T., Von Martial F. "A Negotiation Framework for Autonomous Agents", Decentralized A.I. 2, Demazeau Y. and Müller J-P., (Eds.), pp. 71-87. 1991.
- [Krey 2001] Christian Krey, “ The benefits of multi agent systems in spatial reasoning”, Centre allemand de recherche en intelligence artificielle. DFKI ,2001.
- [Krishna et Ramesh 1998] Krishna V., Ramesh V.C. "Intelligent Agents for Negotiations in Market Games, Part1: Model" IEEE Transactions on Power Systems, Vol. 13, No. 3., 1998.
- [Kumar et Shapiro 1994] "The OK BDI Architecture" IJAIT, International Journal Of Artificial Intelligence Tools, Vol. 3, pp. 349-366.,1994
- [Kurka, et Maass 1997] F. Blanchard, P. Kurka, and A. Maass. “Topological and measure-theoretic properties of one-dimensional cellular automata”. Physica D: Nonlinear Phenomena, 103(1-4):86–99, 1997.
- [Ladkin et Reinfeld 1992] Ladkin, P., and A. Reinfeld.. Effective solution of qualitative interval constraint problems. *Artificial Intelligence*, 57:105-124., 1992.

[Lander et Lesser 1993] Lander S. E. and R. Lesser. V. "Understanding the Role of Negotiation in Distributed Search Among heterogeneous Agents" International Joint Conference on Artificial Intelligence (IJCAI-1993).

[Langton 1988] C. Langton (Ed.) Artificial Life. Addison-Wesley, 1998.

[Le Strugeon 1995] Le Strugeon E. « Une méthodologie d'Auto-Adaptation d'un Système Multi-Agents ». Thèse de doctorat, Université de Valenciennes., 1995.

[Lee, et al. 2008] Lee, B., et al.: Virtual Epidemic in a Virtual City: Simulating the Spread of Influenza in a United States Metropolitan Area. Unpublished manuscript, 2008.

[Leperhoff 2002] Leperhoff N. "SAM- Simulation of Computer-mediated Negotiations" Journal of Artificial, 2002.

[Loui 1987] R. Loui, "Defeat among arguments: A system of defeasible inference," Computational Intelligence, vol. 3, pp. 100–106, 1987.

[Maes 1994] Maes P. "Agents that Reduce Work and Information Overload" Communications of the ACM, Vol. 37, pp. 31-40., 1994.

[Mandiau et al. 2002] Mandiau , R., Grislin-Le Strugeon, E., Péninou, A. « Organisation et applications des SMA ». Paris, Hermes., 2002.

[Marik et al 1997] : Vladimir MARIK, Olga STEPANKOVA, Jin LAZANSKY. « Role of qualitative reasoning in a multi-agent system” , LNCS, Springer, volume 1333, EUROCAST, pp 380---393,1997.

[Matos et al. 1998] Matos N., Sierra C., Jennings N. R. "Negotiation Strategies: An Evolutionary Approach" International Conference on MultiAgent Systems (ICMAS'98), Paris, 1998, pp. 182- , 1998.

[Maudet and Chaib-draa 2003] N. Maudet and B. Chaib-draa, "Commitment-based and dialogue-game based protocols—new trends in agent communication language," Knowledge Eng. Rev. vol. 17, no. 2, pp. 157–179, 2003.

[Meurisse et Vanbergue 2001] Meurisse T., Vanbergue D. (2001) "Et maintenant, à qui le tour? Aperçu de problématiques de conception de simulations Multi-Agents" Laboratoire d'Informatique de Paris 6 -OASIS, Paris. 2001.

[McBurney et al. 2003]. P. McBurney, R. M. van Eijk, S. Parsons, and L. Amgoud, "A dialogue-game protocol for agent purchase negotiations," J. Autonomous Agents and Multi-Agent Systems, vol. 7, no. 2, pp. 235–273, 2003.

[Michel 2004] Fabien Michel. « Formalisme, outils et éléments méthodologiques pour la modélisation et la simulation multi-agents » . PhD - thesis, Université Montpellier II, 2004.

[Moraitis 1994] Moraitis P. «Paradigme Multi-Agent et Prise de Décision Distribuée ». Thèse de Doctorat Université Paris Dauphine., 1994.

- [Moss 2001] Moss S. "Game Theory: Limitations and an Alternative" *Journal of Artificial Societies and Social Simulation*, Vol. 4, 17 Pp., 2001
- [Moulin et al. 2002] : Moulin B., Irandoust. H., Belanger. M., & Desbordes G. "Explanation and argumentation capabilities : towards the creation of more persuasive agents". In *Artificial Intelligence review 17* : Kluwer academic publishers in Holland , pp 169-222, 2002.
- [Müller 1996] Müller J. H. "Negotiation Principles" *Foundations of Distributed Artificial intelligence*, 1996.
- [Müller 1998] Müller J.P. "Vers une méthodologie de conception de systèmes multi-agents de résolution de problème par émergence" *Journées Francophones d'Intelligence Artificielle Distribuée et Systèmes Multi-Agents (JFIADSMA'98)*. Hermès. 1998.
- [Müller 2002] Müller J.P. "Des systèmes autonomes aux systèmes multi-agents: Interaction, émergence et systèmes complexes" *Habilités à Diriger des Recherches*, Univ. Montpellier II. 2002.
- [Nakamura et Kurumatani 1995] Nakamura M., Kurumatani K. "A mathematical model for the foraging of an ant colony and pattern formation of pheromone trail" *Fundamental Theories of Deterministic and Stochastic Models in Mathematicla Biology*, Institute of Statistical Mathematics, pp120-131.1995.
- [Nash 1953] J. F. "Two Persons Cooperative Games" *Econometrica*, Vol. 21, pp. 128-140. Von Neumann J., Morgenstern (1944) *The Theory of Games and Economic Behaviour*. Princeton University Press., 1953.
- [Nebel & Bürkert 1995] B. Nebel and H-J. Btirckert. Reasoning about Temporal Relations: A Maximal Tractable Subclass of Allen's Interval Algebra. *Journal of the ACM*, 42(1):43--66, 1995.
- [Nebel 1996] Nebel N. Solving Hard Qualitative Temporal Reasoning Problems: Evaluating the Efficiency of Using the ORD-Horn Class'. In: W. Wahlster (ed.): *Proceedings of the 12<sup>th</sup> E.C.A.I ,96* . pp. 38-42., 1996.
- [O'Hare & Jennings 1994] O'Hare G. M. P., Jennings N. R. "Foundations of Distributed Artificial Intelligence". John Wiley & Sons, Inc., New York, USA. Osborne M. J., Rubinstein A. *A Course in Game Theory*. MIT Press. 1994.
- [Padgham & Winikoff 2002] Padgham L., Winikoff M. "Prometheus: A Methodology for Developing Intelligent Agents" *Proceedings of the third International Workshop on Agent-Oriented Software Engineering, (AAMAS 2002)*, Bologna, Italy. 2002.
- [Parsons et al 1998] :Parsons S., Sierra C., and Jennings, N., Agents that reason and negotiate by arguing, *J. Logic Comput.* vol. 8, no. 3, pp. 261–292, 1998.

[Parunak et al. 1998] Parunak H.V.D., Savit R., Riolo R.L. "Agent-Based Modelling vs. Equation-Based Modelling: A Case Study and Users' Guide" Proceedings of Workshop on Multi-agent systems and Agent-based Simulation, Springer, pp. 10-25., 1998.

[Pollack 1992 ] Pollack M. "The uses of plans" Artificial Intelligence, Vol. 57, pp. 43-68.,1992.

[Pollack & Horty 1999] M., Horty J. "There's more to life than making plans: plan management in dynamic, multi-agent environments", Vol. 20, pp. 71-84. 1999.

[Pollock 1974]: J. Pollock., Knowledge and justification. Princeton. 1974.

[Pollock 1987] J. L. Pollock, "Defeasible Reasoning," Cognitive Sci. vol. 11, pp. 481–518, 1987.

[Pomerol 1997] Pomerol J-C "Artificial Intelligence and Human Decision Making" European Journal of Operations Research, Vol 99, pp 3-25. 1997.

[Prakken and Sartor 2001] H. Prakken, and G. Sartor, "The role of logic in computational models of legal argument: a critical survey," in A. Kakas, and F. Sadri (eds.), Computational Logic: From Logic Programming into the Future (In honour of Bob Kowalski), Vol. 2048 of Lecture Notes in Computer Science, Springer Verlag: Berlin, Germany, pp. 342–343, 2001.

[Pyka & Grebel 2003] : A. Pyka , Thomas Grebel . "Agent based modeling, A Methodology for the analysis of the qualitative developpement processes " Discussion Paper Series 251, University of Augsburg, Institute for Economics, Germany, 2003.

[Rahwan et al 2003] : Rahwan, I., Sonenberg , L., & Dignum, F. "On interest based negotiation." In Dignum, F.. (e.) , Advances in Agent communication ( Lecture notes in artificial intelligence, vol 2922. Berlin: Springer Verlag. 2003.

[Rahwan et al. 2004] : Rahwan, I., Ramchurn , D. R., Jennings, N. R., Peter, M., Parsons, S & Sonenberg. L. 2004. Argumentation based negotiation, The Knowledge Engineering review, Cambridge university press, vol 18:4, 343-375, 2004.

[Rahwan 2004] I. Rahwan, "Interest-based negotiation in multi-agent systems," Ph.D. thesis, Department of Information Systems, University of Melbourne, Melbourne, Australia, 2004.

[Ramchurn et al . 2003]: Ramchurn, S. D., Jennings, N. R. & Sierra, C. "Persuasive negotiation for autonomous agents: a rhetorical approach", In Reed, C., Grasso, F. (ed), 2003.

[Reiter 1980] : R. Reiter. A logic for default reasoning. Artificial Intelligence, 13 (1-2) : 81-1, 1980.

[Renz 2002] J. Renz "Qualitative Spatial Reasoning with Topological Information ». Lecture Notes in Computer Science 2293, Springer Verlag, 2002.

[Rueda et al. 2002] S.V. Rueda, A. J. Garcia and G. R. Simari, “Argument-based negotiation among BDI agents,” *Computer Sci. Technol.* vol. 2, no. 7, pp. 1–8, 2002.

[Russel et Norvig 1995] Russel S., Norvig P. « Artificial Intelligence. A Modern Approach”. Prentice hall, Series in Artificial Intelligence.,1995.

[Sadri et al 2001]: Sadri F., Toni F., & Torroni P. “Logic agents, dialogues and negociation : an abductive approach”. In Stathis, K., Schroeder, M. (ed.), *Proceedings of the AISB 2001 Symposium on information agents for E-commerce.* 2001.

[Sawamoto et al 2005] Jun Sawamoto,Hidekazu, Tsuji, Dilmurat Tilwalidi, Hisao Koizumi. “A proposal of multi-Agent system development framework for cooperative problem solving and its experimental evaluation”, *Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA'05).*, 2005.

[Sillince 1994]: Sillince J., A. “Multi agent conflict resolution : A computational framework for an intelligent argumentation program” . *Knowledge base systems* 7,2 (june), 75-90,1994

[Simari and Loui 1992] G. R. Simari and R. P. Loui, “A Mathematical Treatment of Defeasible Reasoning and its Implementation,” *Arti. Intelligence* vol. 53, pp. 125–157, 1992.

[Simon 1969] Simon H. E. “The Sciences of the Artificial”. Cambridge: MIT Press. 1969.

[Simos 1990] J. Simos "Evaluer l'impact sur l'environnement" . Presses polytechniques et universitaires romandes, 1969.

[Schroeder 2000]: Schroeder. M., Plewe D., A., & Raab .A. “ULTIMA RATIO : A visual language for argumentation”. University of Magdeburg- Germany. 2000.

[Smith 1988] Smith R. G. "The Contract net Protocol: High-level Communication and Control in a Distributed Problem Solver". *Readings in Distributed Artificial Intelligence*, Bond A. H.and Gasser L., (Eds.), pp. 357-366,1988.

[Sycara 1988] Sycara K. "Resolving goal conflicts via negotiation" 7th National Conference on Artificial Intelligence, 1988.

[Sycara 1990] Sycara., K. “The Persuader”. In Shapiro, D (ed.), *The encyclopaedia of artificial intelligence.* New York: Jhon Wiley and Sons, 1994.

[Sycara 1992] Sycara K. "The PERSUADER" *The Encyclopedia of Artificial Intelligence*, Shapiro J., (Eds.) New York, 1992

[Terna 1998] Terna P. "Simulation Tools for Social Scientists: Building Agent Based Models with SWARM" *Journal of Artificial Societies and Social Simulation*, Vol. 1, 9 Pp. 1998.

[Toulmin 1958]: Toulmin, S. "The uses of argumentation". Cambridge Univ.Press. 1958.

[Treuil et Mullon 1997] Treuil J.P., Mullon C. Expérimentation sur mondes artificiels : pour une réflexion méthodologique Editions Scientifiques et Médicales ELSEVIER., 1997.

[van Eemeren et al. 1996] F. H. van Eemeren, R. F. Grootendorst, and F. S. Henkemans, "Fundamentals of Argumentation Theory: A Handbook of Historical Backgrounds and Contemporary Applications". Lawrence Erlbaum Associates: Hillsdale NJ, USA, 1996.

[Vercouter et al. 1998] Vercouter L., Franchesquin N., Treuil J.-P., Pinson S. "Simulation multi-agents des processus de négociation en agronomie : un exemple de gestion locale sous contrainte collective." vol. 115, Pinson S., Ed. Document du LAMSADE. Université Paris-Dauphine, 1998.

[Vreeswijk 1993] : Vreeswijk .G. A. "Studies in defeasible argumentation".. Phd thesis- Vrije Univ- Holland, 1993.

[Walton and Krabbe 1995 ] D. N. Walton and E. C. W. Krabbe, "Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning". SUNY Press, Albany NY, USA, 1995.

[Weiss 1999] Weiss G. "Multi agent Systems", MIT Press., 1999 [Wolfram 84] Stephen Wolfram. Universality and complexity in cellular automata.Physica D: Nonlinear Phenomena, 10:1–35, January 1984. DOI: 10.1016/0167-2789(84)90245-8.

[Winikoff et Padgham 2001] Winikoff M., Padgham L., Harland J. "Simplifying the Development of Intelligent Agents" 14th Australian Joint Conference on Artificial Intelligence (AI'01), Adelaide,Australia, 2001.

[Winograd and Flores 1987]: T. Winograd and F. Flores, "Understanding Computers and Cognition: A New Foundation for Design" , Addison-Wesley Professional, 1987.

[Wooldridge et Jennings 1994] Wooldridge M, Jennings N.R. "Agents Theories, Architecture and Language: A Survey" In Intelligent Agents, ECAI'94, M. Wooldridge and N.R. Jennings (Eds.) LNAI 890, Springer Verlag, pp 1-32, 1994.

[Wooldridge 1999] Wooldridge M. "Intelligent Agents" Multiagent systems, Weiss G., Ed.: MIT Press, 1999.

[Wooldridge et Ciancarini 2000] Wooldridge M., Ciancarini P. (2000) "Agent-Oriented Software Engineering: The State of the Art" Lecture Notes in AI., Vol. 1957: Springer-Verlag.

[Wooldridge et al 2000] Wooldridge M., Jennings N.R., Kinny D. "The Gaia Methodology for agent-oriented analysis and design" Autonomous Agents and multi-agent systems", Vol. 3, 2000.

[Xuan et et Ryszard 2007] Xuan Thang Nguen and Ryszard Kowalczyk. " On solving distributed fuzzy constraint satisfaction with agents". IEEE/WIX/ACM. International Conference On intelligent Agent Technology , pages 387-390, 2007.

[Yoko et al. 2000] M. Yokoo and K. Hiramaya. Algorithms for distributed constraint satisfaction: A review. *Autonomous Agents and Multi-Agent Systems*, 3(2):185–207, 2000.

[Zeng et Sycara 1997] Zeng D. D., Sycara K. (1997) "Benefits of Learning in Negotiation" AAAI-97.

[Zlotkin et Rosenschein 1991] Zlotkin G., Rosenschein J. S. "Negotiation and goal relaxation" *Decentralised A.I.*, v 2, 1991.

[Zlotkin et Rosenschein 1993] Zlotkin G., Rosenschein J. S. "A domain theory for task oriented negotiation". In 13<sup>th</sup> International Joint Conference on Artificial Intelligence., 1993.

### Références Web

[Briot 2002] <http://www.poleia.lip6.fr/~briot/cours/intro-agents-iarfa01-02.pdf>

[FIPA] : "Specification: Agent communication language". Foundation for Intelligent Physical Agents; <http://www.fipa.org/spec/fipa99spec.htm>

[Monet] : site web: <http://monet.aber.ac.uk>

[ QRG ] : site web : <http://userweb.cs.utexas.edu/~qr/>

SWARM (2001) "Tutorial" 2001.[www.swarm.org/csss-tutorial/toc.html](http://www.swarm.org/csss-tutorial/toc.html)

## ANNEXE : Quelques Détails d'Implémentation

### 1. Présentation de l'outil d'édition et de manipulation des cartes cognitives développé.

Le but escompté est le développement d'un outil graphique permettant de mettre en œuvre les techniques de construction et de manipulation des cartes cognitives pour modéliser et de raisonner qualitativement sur les systèmes dynamiques indépendamment du domaine de discours. Cet outil nous donnera par la suite la possibilité, d'une part d'étudier un domaine d'expertise qui est l'infection virale, un processus biologique qualifié de système complexe, et d'autre part de tester l'algorithme d'inférence que nous avons proposé.

L'environnement de développement utilisé est Microsoft Visual C++ qui est un environnement de développement intégré (IDE) ; conçu par Microsoft pour le langage de programmation C++, afin de développer et déboguer du code en C++. L'architecture du logiciel est donnée par la Figure a.3. Les fonctionnalités offertes par cet outil sont regroupées en catégories (onglets) fonctionnelles qui sont :

- **Accueil** : contient des fonctionnalités permettant l'exécution des tâches de presse papier



Figure a.1 : Capture d'écran pour la catégorie « Accueil »

(Sélectionner, Couper, Coller), la gestion des options d'affichages et la gestion des fenêtres (Figure a.1).

- **Création** : contient les fonctionnalités qui permettent de créer une carte cognitive (Voir

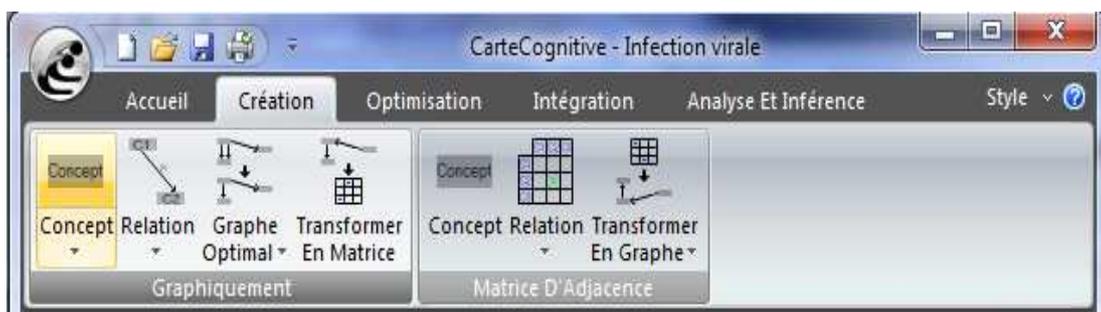


Figure a.2 : Capture d'écran pour la catégorie « Création »

Figure a.2). Cette création se fait de deux manières : soit graphiquement, c'est-à-dire, dessiner manuellement le graphe (concepts et relations) ou bien à partir de la matrice d'adjacence.

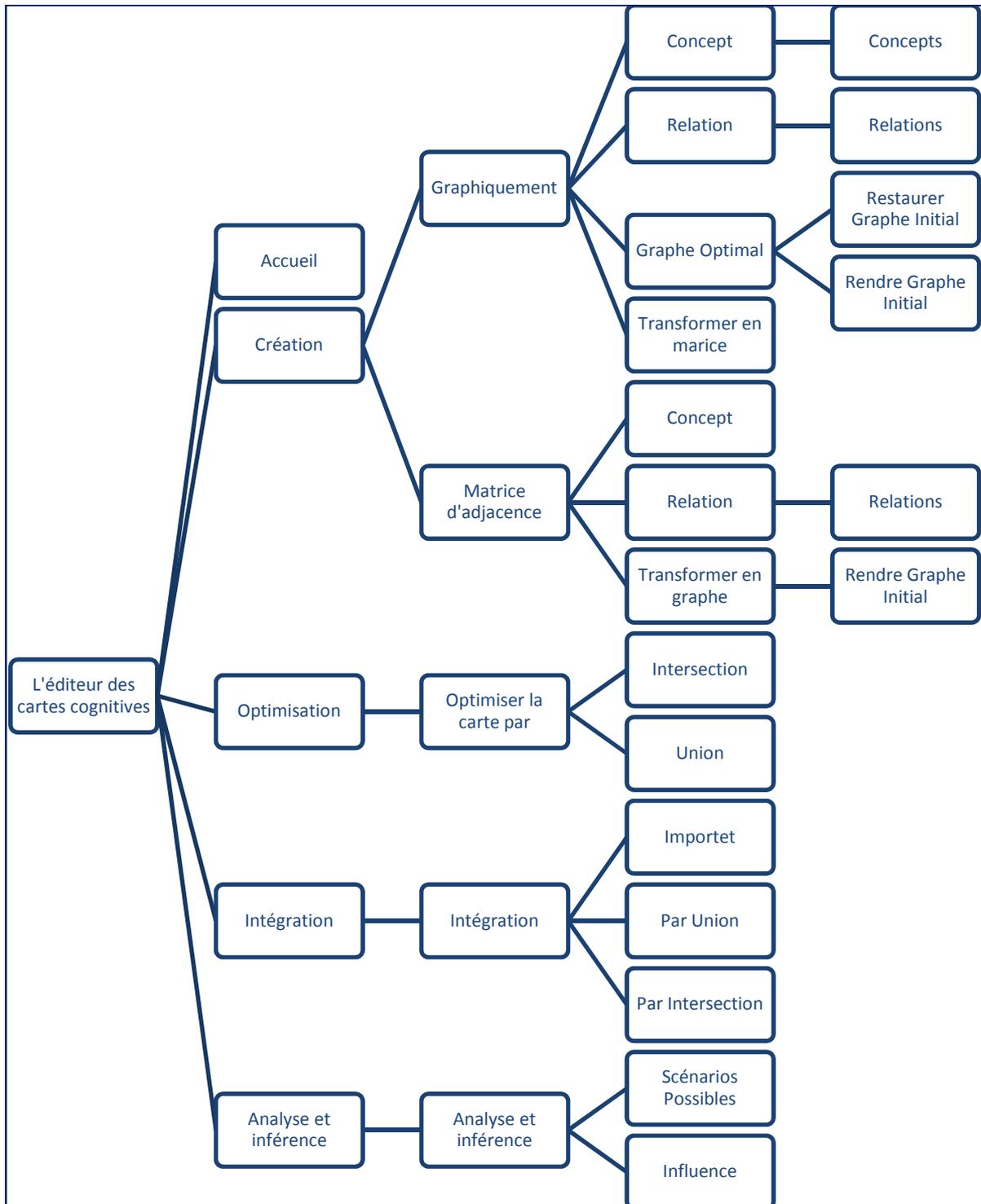


Figure a.3 L'architecture de l'outil logiciel développé

- **Optimisation** : contient deux fonctionnalités pour manipuler les relations entre deux concepts. Deux fonctionnalités pour manipuler sont prévues pour optimiser les relations qui existent entre les concepts d'une carte cognitive : optimisation par Intersection et optimisation par union (Voir la Figure a.4).

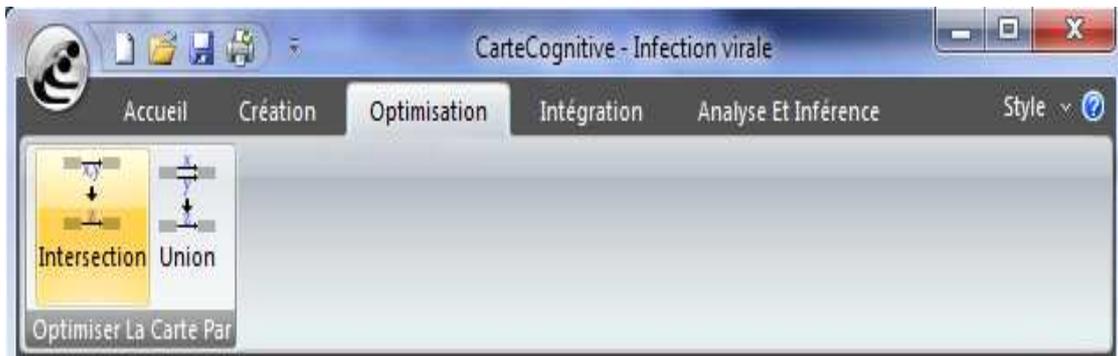


Figure a.4 : Capture d'écran pour la catégorie « Optimisation ».

- **Intégration** : contient des fonctionnalités pour intégrer plusieurs cartes en une seule. Ceci permettra d'avoir un modèle unifié (vision globale) à partir de plusieurs modèles (visions partielles) émanant de plusieurs experts. (Voir Figure a.5).



Figure

a.5 : Capture d'écran pour la catégorie « Intégration »

- **Analyse et inférence** : offre des fonctionnalités qui servent à extraire des connaissances d'aide à la prise de décision. Cet onglet représente le composant de raisonnement causal offert par cet outil (Voir Figure a.6). La première fonctionnalité donne la possibilité d'avoir tous les scénarios possibles sous forme textuelle. La sélection de cette fonctionnalité permet d'avoir une fenêtre illustrée par la figure a.7.

La deuxième fonctionnalité, appelée « Influence » nous donne la réponse à la question « qu'arrivera-t-il si ? ». Après un clic sur ce bouton, La sélection de cette option nous donne la fenêtre illustrée par la Figure a.8.

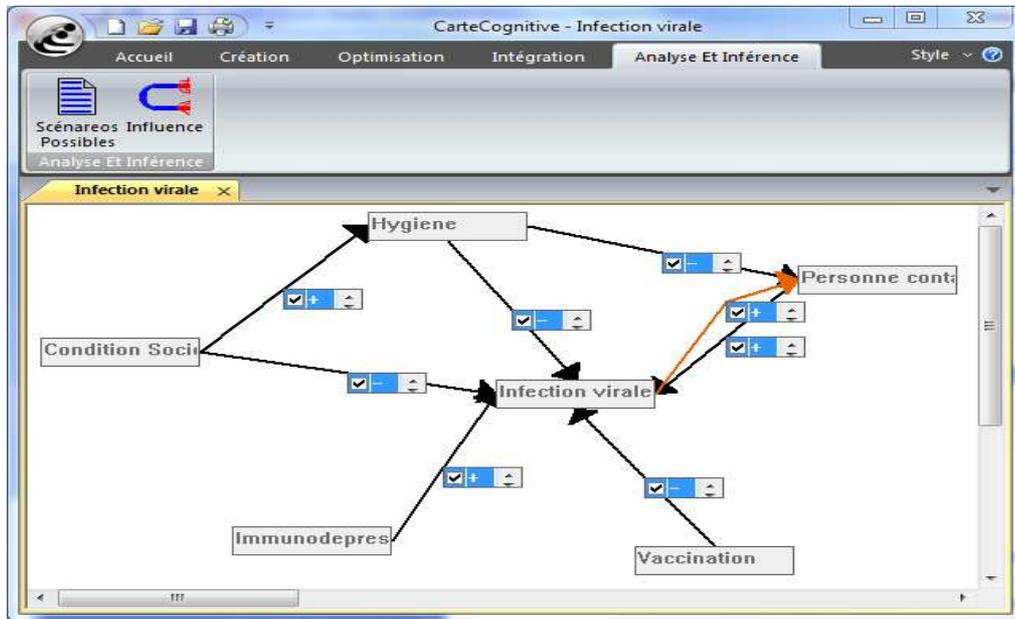


Figure a. 6 : Capture d'écran pour la catégorie « Analyse et Inférence ».

Dans le cas de manipulation d'une carte cognitive neutrosophique, choisir l'option « Neutrosophique » avant de Lancer l'opération.

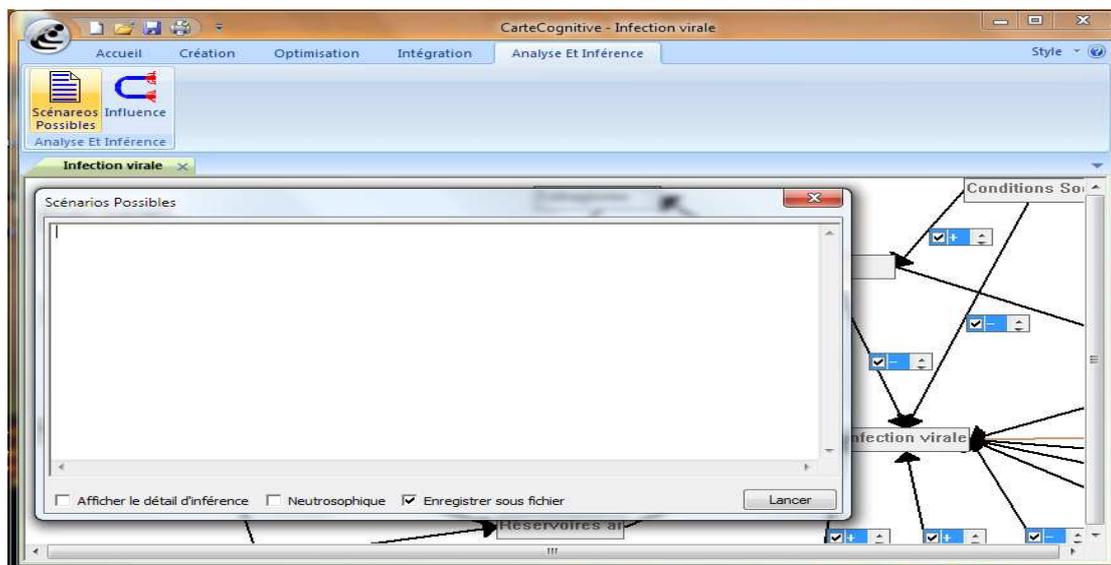


Figure a.7: Capture d'écran pour la fenêtre « Scénarios Possibles ».

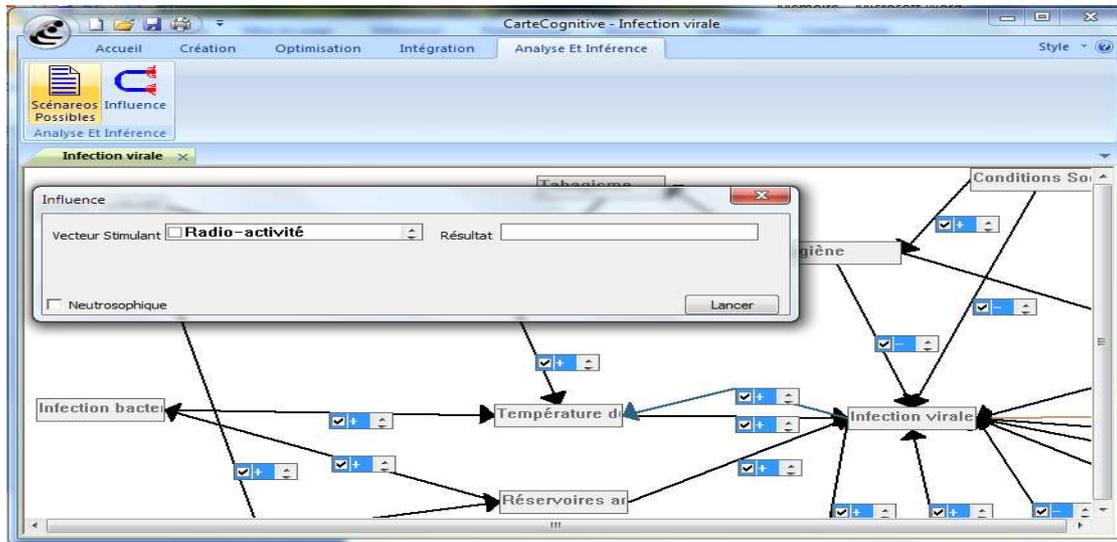


Figure a.8: Capture d'écran pour la fenêtre « Confluences ».

## 2. Spécification en Langage Maude du protocole de dialogue Inter- agents « ARGDIAL »

Avant de donner la spécification de notre protocole de dialogue, voici un aperçu général sur le langage Maude.

### 2.1 Bref aperçu sur Le langage Maude :

Maude est un langage de programmation concurrente orienté objet basé sur la logique de réécriture [Meseguer 1992]. Il offre l'avantage de pouvoir spécifier formellement les systèmes concurrents et simuler leurs comportements par l'exécution des spécifications fournies. Aussi, ces dernières peuvent être vérifiées formellement grâce à son « model checker » basé sur la logique temporelle linéaire LTL.(Linear Temporal Logic). [ Clavel 2005]

Dans ce langage, l'unité de spécification de base s'appelle un « module ». On distingue trois sortes de modules : les modules fonctionnels (functional modules, notés *fmod*), les modules système (system modules, notés *mod*) et les modules objet (object modules, notés *omod*). Un module fonctionnel contient des informations sur les données, des équations, des informations de typage et d'appartenance. En plus de ces informations, un module système contient, les règles de réécriture définissant les transitions d'états du système étudié ou une partie constituante de ce système. Les modules objet confèrent à Maude une autre dimension

lui permettant de profiter pleinement de la richesse des fonctionnalités et des concepts du paradigme objet. A noter que, en langage Maude un module objet peut contenir des règles de réécriture pour définir les transitions dans l'état de l'objet.

## 2.2 Spécification formelle en Maude du protocole de dialogue « ARGDIAL »

```

***** Début spécification*****
fmod STATE-DIALOG is
sort StateKind StateName DialogState .

ops Initial Final Intermediate : -> StateKind .

op _;_ : StateName StateKind -> DialogState .
ops S1 S2 S3 S4 S5 S6 S7 S8 S9 : -> StateName .
endfm

*****

mod ACTION is
sort Action .
ops Ac1 Ac2 : -> Action .
endm

*****

*****

mod GOAL is
inc CONFIGURATION .
pr STRING .
sort Goal .

op NoGoal : -> Goal .

*****User's part*****
ops G1 G2 G3 G4 G5 G6 : -> Goal .
endm

*****

mod CONTRADICTION is
pr ACTION .
pr GOAL .

op InContradiction : Goal Action -> Bool .

*****Partie utilisateur*****

```

```

eq InContradiction(G1, Ac1) = false .
eq InContradiction(G1, Ac2) = false .
eq InContradiction(G2, Ac1) = true .
eq InContradiction(G2, Ac2) = false .

eq InContradiction(G3, Ac1) = true .
eq InContradiction(G3, Ac2) = false .
eq InContradiction(G4, Ac1) = true .
eq InContradiction(G4, Ac2) = false .

endm

*****
mod GOAL-LIST is
pr CONTRADICTION .
pr GOAL .
sorts GoalList .

subsort Goal < GoalList .

op HeadGL : GoalList -> Goal .

op TailGL : GoalList -> GoalList .

op _:_ : Goal GoalList -> GoalList .

op ErrorGoal : -> Goal .

op ErrorGoalList : -> GoalList .

op EmptyGoalList : -> GoalList .

op IsEmptyGoalList : GoalList -> Bool .

var G : Goal .

var GL : GoalList .

***
***

eq G : EmptyGoalList = G .

***
***

eq IsEmptyGoalList(EmptyGoalList) = true .

eq IsEmptyGoalList(G : GL) = false .

```

\*\*\*

\*\*\*

eq HeadGL(EmptyGoalList) = ErrorGoal .

eq HeadGL(G : GL) = G .

eq HeadGL(G) = G .

\*\*\*

\*\*\*

eq TailGL(EmptyGoalList) = ErrorGoalList .

eq TailGL(G : GL) = GL .

eq TailGL(G) = EmptyGoalList .

endm

red InContradiction(HeadGL(G4 : (G2 : G3)), Ac2) .

\*\*\*\*\*

mod KNOWLEDGE-BASE is

sort KnowledgeBase .

endm

\*\*\*\*\*

mod ARGUMENT is

pr KNOWLEDGE-BASE .

pr GOAL-LIST .

pr INT .

subsort Int &lt; Strength .

sorts Argument Support Conclusion ArgContent Strength .

op NoArgument : -&gt; Argument .

op Supp : KnowledgeBase GoalList GoalList Argument -&gt; Support .

op Concl : ArgContent Strength -&gt; Conclusion .

op Arg : Support Conclusion -&gt; Argument .

op ReceivedArgument : Support -&gt; Argument .

op ArgumentConclusion : Argument -&gt; Conclusion .

op ArgumentSupport : Argument -&gt; Support .

op ArgumentContent : Conclusion -> ArgContent .

op ArgumentStrength : Conclusion -> Strength .

op MyGoalList : Support -> GoalList .

op ItsGoalList : Support -> GoalList .

\*\*\*\*\*

var Ar : Argument . var S : Support . var C : Conclusion .

var ArC : ArgContent . var St : Strength .

vars G GO : Goal . var K : KnowledgeBase .

eq ReceivedArgument(Supp(K, G, GO, Ar)) = Ar .

eq ArgumentSupport(Arg(S, C)) = S .

eq ArgumentConclusion(Arg(S, C)) = C .

eq ArgumentContent(Concl(ArC, St)) = ArC .

eq ArgumentStrength(Concl(ArC, St)) = St .

\*\*\*\*\*Partie utilisateur\*\*\*\*\*

ops Sup1 Sup2 Sup3 Sup4 Sup5 Sup6 : -> Support .

ops Arg1 Arg2 Arg3 Arg4 Arg5 Arg6 : -> ArgContent .

endm

\*\*\*\*\*

mod ARGUMENT-LIST is

pr ARGUMENT .

sorts ArgumentList .

subsort Argument < ArgumentList .

op HeadArL : ArgumentList -> Argument .

op TailArL : ArgumentList -> ArgumentList .

op \_::\_ : Argument ArgumentList -> ArgumentList .

op ErrorArgument : -> Argument .

op ErrorArgumentList : -> ArgumentList .

op EmptyArgumentList : -> ArgumentList .

op IsEmptyArgumentList : ArgumentList -> Bool .

op ThereIsMoreStongArgument : Argument ArgumentList -> Bool .

op ArgumentMoreStong : Argument ArgumentList -> Argument .

\*\*\*\*

\*\*\*\*

var Ar : Argument .

var ArL : ArgumentList .

\*\*\*

\*\*\*

eq Ar :: EmptyArgumentList = Ar .

\*\*\*

\*\*\*

eq IsEmptyArgumentList(EmptyArgumentList) = true .

eq IsEmptyArgumentList(Ar :: ArL) = false .

\*\*\*

\*\*\*

eq HeadArL(EmptyArgumentList) = ErrorArgument .

eq HeadArL(Ar :: ArL) = Ar .

eq HeadArL(Ar) = Ar .

\*\*\*

\*\*\*

eq TailArL(EmptyArgumentList) = ErrorArgumentList .

eq TailArL(Ar :: ArL) = ArL .

eq TailArL(Ar) = EmptyArgumentList .

\*\*\*

\*\*\*

eq ThereIsMoreStongArgument(Ar, ArL) = if IsEmptyArgumentList(ArL) == true then false

    else if ArgumentStrength(ArgumentConclusion(HeadArL(ArL))) >

ArgumentStrength(ArgumentConclusion(Ar)) then true

    else ThereIsMoreStongArgument(Ar, TailArL(ArL))

    fi fi .

eq ArgumentMoreStong(Ar, ArL) = if IsEmptyArgumentList(ArL) == true then NoArgument

    else if ArgumentStrength(ArgumentConclusion(HeadArL(ArL))) >

ArgumentStrength(ArgumentConclusion(Ar)) then HeadArL(ArL)

    else ArgumentMoreStong(Ar, TailArL(ArL)) fi fi .

endm

red ThereIsMoreStrongArgument(Arg(Sup4, Concl(Arg1, 3)), (Arg(Sup1, Concl(Arg1, 1)) :: (Arg(Sup2, Concl(Arg2, 3)) :: Arg(Sup3, Concl(Arg3, 5)))) .

red HeadArL(Arg(Sup1, Concl(Arg1, 1)) :: (Arg(Sup2, Concl(Arg2, 3)) :: Arg(Sup3, Concl(Arg3, 5)))) .  
 \*\*\*\*\*

```
mod DIALOG-PROTOCOL is
inc CONFIGURATION .
protecting STATE-DIALOG .
pr GOAL-LIST .
pr ARGUMENT-LIST .
sorts Dialog Agent .
subsorts Dialog Agent < Cid .
op Dialog : -> Dialog .
op Agent : -> Agent .
```

```
op Initiator : _ : Oid -> Attribute .
op Participant : _ : Oid -> Attribute .
op State : _ : DialogState -> Attribute .
```

```
op GList : _ : GoalList -> Attribute .
op ArgList : _ : ArgumentList -> Attribute .
```

```
op InformOpen : Oid Oid -> Msg .
op Request : Oid Oid Action -> Msg .
op Agree : Oid Oid Action -> Msg .
op Criticise : Oid Oid Argument -> Msg .
op Retract : Oid Oid Action -> Msg .
op Argue : Oid Oid Argument -> Msg .
op InformClose : Oid Oid -> Msg .
```

```
vars I P : Oid .
vars GLI GLP : GoalList .
var ArLI ArLP : ArgumentList .
```

```
vars D : Oid .
var A : Action .
var Ar : Argument .
```

```
rl [C1] : InformOpen(I, P)
  < D : Dialog | Initiator : I, Participant : P, State : (S1 ; Initial) >
=>
  < D : Dialog | Initiator : I, Participant : P, State : (S2 ; Intermediate) >
  Request (P, I, Ac1) .
```

```
crl [C2] : Request (P, I, Ac1)
  < I : Agent | GList : GLI, ArgList : ArLI >
  < P : Agent | GList : GLP, ArgList : ArLP >
  < D : Dialog | Initiator : I, Participant : P, State : (S2 ; Intermediate) >
=>
```

```

    < I : Agent | GList : GLI, ArgList : ArLI >
    < P : Agent | GList : GLP, ArgList : ArLP >
    < D : Dialog | Initiator : I, Participant : P, State : (S3 ; Intermediate) >
    Agree(I, P, Ac1)
    if InContradiction(HeadGL(GLI), Ac1) = false .

crI [C2] : Request (P, I, Ac1)
    < I : Agent | GList : GLI, ArgList : ArLI >
    < P : Agent | GList : GLP, ArgList : ArLP >
    < D : Dialog | Initiator : I, Participant : P, State : (S2 ; Intermediate) >
=>
    < D : Dialog | Initiator : I, Participant : P, State : (S3 ; Intermediate) >
    < I : Agent | GList : GLI, ArgList : TailArL(ArLI) >
    < P : Agent | GList : GLP, ArgList : ArLP >

    Criticise(I, P, HeadArL(ArLI))
    if InContradiction(HeadGL(GLI), Ac1) = true .

rl [C71] : Agree(I, P, Ac1)
    < D : Dialog | Initiator : I, Participant : P, State : (S3 ; Intermediate) >
=> < D : Dialog | Initiator : I, Participant : P, State : (S4 ; Intermediate) >
    InformClose(I, P) .

rl [C61] : InformClose(I, P)
    < D : Dialog | Initiator : I, Participant : P, State : (S4 ; Intermediate) >
=> < D : Dialog | Initiator : I, Participant : P, State : (S9 ; Final) > .

crI [C31] : Criticise(I, P, Ar)
    < I : Agent | GList : GLI, ArgList : ArLI >
    < P : Agent | GList : GLP, ArgList : ArLP >
    < D : Dialog | Initiator : I, Participant : P, State : (S3 ; Intermediate) >
=>
    < I : Agent | GList : GLI, ArgList : ArLI >
    < P : Agent | GList : GLP, ArgList : ArLP >
    < D : Dialog | Initiator : I, Participant : P, State : (S5 ; Intermediate) >
    Retract(I, P, Ac1)
    if ArgumentMoreStong(Ar, ArLP) = NoArgument .

    ***if ThereIsMoreStongArgument(Ar, ArLP) = false .

crI [C32] : Criticise(I, P, Ar)
    < I : Agent | GList : GLI, ArgList : ArLI >
    < P : Agent | GList : GLP, ArgList : ArLP >
    < D : Dialog | Initiator : I, Participant : P, State : (S3 ; Intermediate) >
=>
    < I : Agent | GList : GLI, ArgList : ArLI >
    < P : Agent | GList : GLP, ArgList : ArLP >
    < D : Dialog | Initiator : I, Participant : P, State : (S5 ; Intermediate) >

```

```

Argue(P, I, ArgumentMoreStong(Ar, ArLP))
if ArgumentMoreStong(Ar, ArLP) /= NoArgument .
***if ThereIsMoreStongArgument(Ar, ArLP) = true .

```

```

crI [C51] : Argue(P, I, Ar)
  < I : Agent | GList : GLI, ArgList : ArLI >
  < P : Agent | GList : GLP, ArgList : ArLP >
  < D : Dialog | Initiator : I, Participant : P, State : (S5 ; Intermediate) >
=>
  < I : Agent | GList : GLI, ArgList : ArLI >
  < P : Agent | GList : GLP, ArgList : ArLP >
  < D : Dialog | Initiator : I, Participant : P, State : (S6 ; Intermediate) >
  Argue(I, P, ArgumentMoreStong(Ar, ArLI))
  if ArgumentMoreStong(Ar, ArLI) /= NoArgument .

```

```

crI [C512] : Argue(P, I, Ar)
  < I : Agent | GList : GLI, ArgList : ArLI >
  < P : Agent | GList : GLP, ArgList : ArLP >
  < D : Dialog | Initiator : I, Participant : P, State : (S5 ; Intermediate) >
=>
  < I : Agent | GList : GLI, ArgList : ArLI >
  < P : Agent | GList : GLP, ArgList : ArLP >
  < D : Dialog | Initiator : I, Participant : P, State : (S6 ; Intermediate) >
  Agree(I, P, Ac1)
  if ArgumentMoreStong(Ar, ArLI) == NoArgument .

```

```

crI [C52] : Argue(I, P, Ar)
  < I : Agent | GList : GLI, ArgList : ArLI >
  < P : Agent | GList : GLP, ArgList : ArLP >
  < D : Dialog | Initiator : I, Participant : P, State : (S6 ; Intermediate) >
=>
  < I : Agent | GList : GLI, ArgList : ArLI >
  < P : Agent | GList : GLP, ArgList : ArLP >
  < D : Dialog | Initiator : I, Participant : P, State : (S5 ; Intermediate) >
  Argue(P, I, ArgumentMoreStong(Ar, ArLI))
  if ArgumentMoreStong(Ar, ArLI) /= NoArgument .

```

```

rI [C4] : Retract(I, P, Ac1)
  < D : Dialog | Initiator : I, Participant : P, State : (S5 ; Intermediate) >
=> < D : Dialog | Initiator : I, Participant : P, State : (S8 ; Intermediate) >
  InformClose(I, P) .

```

```

rI [C62] : InformClose(I, P)
  < D : Dialog | Initiator : I, Participant : P, State : (S8 ; Intermediate) >
=> < D : Dialog | Initiator : I, Participant : P, State : (S9 ; Final) > .

```

```

rI [C72] : Agree(I, P, Ac1)

```

---

```
< D : Dialog | Initiator : I, Participant : P, State : (S6 ; Intermediate) >  
=> < D : Dialog | Initiator : I, Participant : P, State : (S7 ; Intermediate) >  
  InformClose(I, P) .
```

```
rl [C63] : InformClose(I, P)  
  < D : Dialog | Initiator : I, Participant : P, State : (S7 ; Intermediate) >  
=> < D : Dialog | Initiator : I, Participant : P, State : (S9 ; Final) > .
```

```
endm
```

```
rew InformOpen(I, P)  
  < I : Agent | GList : (G1 : (G2 : G3)), ArgList : (Arg(Sup1, Concl(Arg1, 1)) :: (Arg(Sup2, Concl(Arg2, 13))  
  :: Arg(Sup3, Concl(Arg3, 5)))) >  
  < P : Agent | GList : (G4 : (G5 : G6)), ArgList : (Arg(Sup4, Concl(Arg4, 14)) :: (Arg(Sup5, Concl(Arg5,  
  11)) :: Arg(Sup6, Concl(Arg6, 12)))) >  
  < D : Dialog | Initiator : I, Participant : P, State : (S1 ; Initial) > .
```

```
*****Fin spécification*****
```