



De l'intelligence artificielle distribuee aux systemes multi-agents

Sofiane Labidi, Wided Lejouad

► **To cite this version:**

Sofiane Labidi, Wided Lejouad. De l'intelligence artificielle distribuee aux systemes multi-agents. [Rapport de recherche] RR-2004, INRIA. 1993. <inria-00074668>

HAL Id: inria-00074668

<https://hal.inria.fr/inria-00074668>

Submitted on 24 May 2006

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

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

*De l'Intelligence Artificielle Distribuée
aux Systèmes Multi-Agents*

S. Labidi, W. Lejouad

N° 2004

Août 1993

PROGRAMME 2

Calcul symbolique,
programmation
et génie logiciel



*Rapport
de recherche*

1993



De l'Intelligence Artificielle Distribuée aux Systèmes Multi-Agents

S. Labidi, W. Lejouad

Programme 2 — Calcul symbolique, programmation et génie logiciel
Projet SECOIA

Rapport de recherche n° 2004 — Août 1993 — 39 pages

Résumé : Les recherches en intelligence artificielle distribuée (IAD) et systèmes multi-agents (SMA) étudient la manière de répartir un problème sur un certain nombre d'entités coopérantes. Elles étudient aussi la manière de coordonner le comportement intelligent d'un ensemble d'entités selon des lois sociales. Ces entités ou *agents* sont autonomes et interagissent dans un environnement pour la résolution de problème. L'aspect collectif de ces agents nécessite l'étude de nouveaux concepts et de nouvelles théories qui émergent d'une *société d'agents*. Ces thèmes de recherche sont en cours d'exploration, l'objectif de ce rapport est de donner une vue d'ensemble sur leur évolution.

Mots-clé : IA Distribuée, Systèmes Multi-Agents et Systèmes Ouverts.

(Abstract: *pto*)

From Distributed Artificial Intelligence to Multi-Agent Systems

Abstract: Distributed Artificial Intelligence (DAI) and Multi-Agent Systems (MAS) consider how problem solving can be divided among several cooperating entities. They are also concerned with coordinating intelligent behavior among a collection of entities in respect to some social laws. These entities or *agents* are autonomous and interact in an environment to solve problems. The collective aspect of these agents requires the definition of new concepts and theories that emerge in an *agent society*. Such research topics are in progress, the purpose of this report is to give an overview on their evolution.

Key-words: Distributed AI, Multi-Agent Systems and Open Systems.

1 Introduction

À la différence de l'Intelligence Artificielle classique (IA) qui modélise le comportement intelligent d'un seul agent, l'intelligence artificielle distribuée (IAD) s'intéresse à des comportements intelligents qui sont le produit de l'activité *coopérative* de plusieurs agents.

Le passage du comportement individuel aux comportements collectifs est considéré non seulement comme une extension mais aussi comme un enrichissement de l'IA, d'où émergent de nouvelles propriétés et de nouveaux comportements.

L'objectif de ce rapport est de donner une vue d'ensemble sur les recherches en IAD et les disciplines qui en découlent, en l'occurrence, les Systèmes Multi-Agents (SMA).

Dans ce rapport, nous commençons par introduire l'IAD et présenter quelques modèles de base de ses systèmes, ensuite nous mettons l'accent sur l'évolution de cette discipline vers les SMA et nous décrivons les propriétés et lois qui émergent en passant de l'aspect individuel (le comportement d'un agent) à l'aspect collectif (son comportement dans une société d'agents), enfin nous présentons quelques applications de systèmes d'IAD pour illustrer l'état actuel de la pratique dans le domaine.

1.1 L'intelligence artificielle distribuée

L'évolution des domaines d'application de l'IA pour recouvrir des domaines complexes et hétérogènes tels que l'aide à la décision, la reconnaissance et la compréhension des formes, la conduite des processus industriels, etc., a montré les limites de l'approche classique de l'IA qui s'appuie sur une centralisation de l'expertise au sein d'un système unique.

Les travaux menés au début des années 70 sur la concurrence¹ et la distribution ont contribué à la naissance d'une nouvelle discipline : l'Intelligence Artificielle Distribuée (IAD) [11] [49] [38]. L'IAD a pour but de remédier aux insuffisances de l'approche classique de l'IA en proposant la distribution de l'expertise sur un groupe d'agents devant être capables de travailler et d'agir dans un environnement commun et résoudre les conflits éventuels. D'où la naissance de notions nouvelles en IA, telles que la coopération, la coordination d'actions, la négociation et l'émergence.

L'IAD peut alors être définie comme étant la branche de l'IA qui s'intéresse à la modélisation de comportement "intelligent" (jusque là c'est la définition de l'IA classique) par la coopération entre un ensemble d'agents [49].

1. La concurrence fait référence au déroulement logique simultané d'actions par opposition au parallélisme qui est le déroulement physique simultané d'actions.

1.2 Historique

Les besoins en parallélisme et distribution ont été très tôt ressentis en IA. En 1973, le premier système d'IAD a pu voir le jour. Il s'agit du système HEARSAY [31] pour la reconnaissance de la parole basé sur l'architecture de *blackboard* (cf. § 2).

Dans les recherches qui ont suivi, diverses approches liées à la distribution se sont succédées, notamment : *les acteurs* de Hewitt [45] (cf. § 3), *la société de l'esprit* de Minsky [61], le système DVMT de Lesser [54], le *Contract Net* de Smith [73], le système *Mace* de Gasser [41] et les *micro-robots* de Brooks [16]. Nous reviendrons en détail sur quelques-unes de ces approches par la suite (cf. § 2).

1.3 Thèmes de recherche de l'IAD

Après avoir défini l'IAD et la différence entre celle-ci et l'IA classique, nous présentons trois axes fondamentaux dans la recherche en IAD :

- *Les systèmes multi-agents* (SMA) [29] : Il s'agit de faire coopérer un ensemble d'agents dotés d'un comportement intelligent et de coordonner leurs buts et leurs plans d'actions pour la résolution d'un problème. C'est le thème auquel nous nous intéressons le plus dans ce rapport.
- *La résolution distribuée des problèmes* (RDP) [55] : Elle s'intéresse à la manière de diviser un problème particulier sur un ensemble d'entités distribuées et coopérantes. Elle s'intéresse aussi à la manière de partager la connaissance du problème et d'en obtenir la solution.
- *L'Intelligence artificielle parallèle* (IAP) [22] [32] : Elle concerne le développement de langages et d'algorithmes parallèles pour l'IAD. L'IAP vise l'amélioration des performances des systèmes d'intelligence artificielle sans, toutefois, s'intéresser à la nature du raisonnement ou au comportement intelligent d'un groupe d'agents. Cependant, il est vrai que le développement de langages concurrents et d'architectures parallèles peut avoir un impact important sur les systèmes d'IAD.

1.4 Problématique de l'IAD

Les problèmes que l'IAD s'attache à résoudre peuvent être divisés en deux classes : la première concerne les problèmes classiques de l'IA qui ont pris une nouvelle dimension dans le contexte multi-agents. La seconde regroupe les

nouveaux problèmes proprement liés au thème de l'IAD. Dans la première classe on peut citer :

- La modélisation de la connaissance et le problème de sa répartition sur les différents agents : comment formuler, décomposer, allouer des problèmes et synthétiser les résultats d'un groupe d'agents ?
- Les problèmes de génération de plans d'actions où il faut prendre en considération la présence d'autres agents ;
- La gestion des conflits entre les agents (points de vue différents des agents) et le maintien de la cohérence des décisions et des plans d'actions ;
- Le problème de la communication : comment permettre la communication et l'interaction entre les agents ? Quel langage et quel protocole faut-il employer ? Une communication dans les univers multi-agents n'est plus une simple tâche d'entrée-sortie, mais doit être modélisée comme un acte pouvant influencer sur l'état des autres agents.

Les problèmes de la seconde classe peuvent être divisés en deux :

- Les problèmes spécifiques au groupe d'agents, qui portent sur l'organisation, l'architecture de l'ensemble des agents et les paradigmes de coopération et d'action ;
- Les problèmes liés au comportement d'un agent au sein d'un groupe. On s'intéresse aux capacités sociales d'un agent : le partage des tâches, le partage des ressources, le raisonnement sur les autres agents (pouvoir modéliser leurs connaissances et être en mesure de connaître leurs plans d'actions et de raisonner en fonction de ces plans).

D'autres thèmes de recherche sont présents dans le contexte multi-agents, à savoir, le raisonnement temporel, le raisonnement hypothétique, la représentation de la connaissance imprécise, etc.

2 Modèles de systèmes d'IAD

Dans cette section, nous présentons quelques modèles qui ont marqué l'histoire de l'IAD.

Blackboard

Le modèle de *blackboard* [44] définit une architecture qui organise la résolution de problèmes par coopération de plusieurs modules, appelés *sources de connaissances* (KS), autour d'une base de données partagée appelée *blackboard*². Chaque KS vient lire ou écrire sur le blackboard. D'une façon générale, le rôle d'une KS est de résoudre un sous-problème particulier en fonction de l'état du blackboard. Ce modèle vise la mise en œuvre d'une résolution opportuniste, où le choix de la KS à activer est déterminé en fonction des critères de contrôle actifs.

Contract Net

Le *Contract Net* [9] est un système de résolution de problème distribué, conçu par R. Davis et R. Smith. L'objectif principal de ce système étant la distribution, il procède par allocation des tâches à un ensemble de solveurs de problèmes et utilise le concept de *négociation* pour adjuger des *contrats*. L'architecture de base contient des nœuds ayant des rôles de *chef* et d'*exécutants*.

DVMT

Distributed Vehicle Monitoring Testbed est un système d'analyse de trafic routier par synthèse des observations d'un certain nombre de capteurs. Ce système fournit un exemple d'application dans lequel la perception et la connaissance sont distribuées. DVMT utilise une collection de systèmes identiques à base de *blackboard* pour résoudre des problèmes d'interprétation des données à partir d'un ensemble de capteurs séparés géographiquement et couvrant une région. Les données sont acheminées d'un sous-ensemble de capteurs à destination de chaque nœud participant à la résolution du problème. Les nœuds coopèrent pour construire une image globale du trafic routier.

De nombreuses architectures sont insatisfaisantes quant à leur aspect centralisé, en particulier les modèles de blackboard qui donnent une vision globale du problème sous forme d'une base de données partagée. Ce sont les travaux sur le modèle *acteur* détaillé dans la section suivante, qui ont abouti à une nouvelle conception des systèmes d'IAD.

2. Le terme blackboard (ou tableau noir) désigne à la fois le modèle et la base partagée.

3 Modèle acteur

Le modèle acteur [46] est développé au MIT par l'équipe de C. Hewitt³. À l'origine, ce modèle a été conçu pour résoudre des problèmes d'IA, il a abouti à une conception des systèmes distribués, en l'occurrence, les systèmes ouverts. Le modèle acteur est un modèle sous-jacent aux systèmes ouverts.

3.1 Systèmes ouverts

Ce sont des systèmes devant fonctionner de la manière la plus indépendante possible. Selon Hewitt [47], les systèmes ouverts (SO) sont des systèmes distribués, ayant les caractéristiques suivantes :

- ils sont concurrents et asynchrones, avec un contrôle décentralisé ;
- ils sont composés d'entités indépendantes en évolution continue ;
- ils sont facilement extensibles ;
- chaque entité ne possède qu'une vision partielle du système ; elle doit négocier avec les autres pour coopérer.

3.2 Acteurs

Le modèle acteur est une extension du modèle objet par ajout de la notion d'activité. Dans les modèles objets, le modèle de calcul est un transfert de contrôle de l'objet expéditeur à l'objet receveur de message, ce qui signifie que l'activité principale est l'allocation du processeur à un objet. Un seul objet est donc actif à la fois et ce sont les messages qui activent les objets. Au contraire, dans le modèle acteur le modèle de calcul est un transfert d'information (requête) entre processus concurrents.

Dans ce modèle, un acteur est un objet actif, autonome, communiquant par envoi de messages asynchrones avec les autres acteurs. Les langages d'acteurs permettent ainsi de garder les acquis des langages à objets tout en y ajoutant la concurrence (coopération entre plusieurs objets actifs simultanément) ; on parle alors de programmation orientée objet concurrente (POOC) [1]. Un acteur se compose d'un ensemble limité de connaissances : les acteurs qu'il connaît directement appelés ses *accointances* et un comportement défini par un *script*. Un *script* est un ensemble de méthodes décrivant la manière dont il doit répondre aux événements venant de l'extérieur : l'acteur filtre les messages reçus et active les méthodes appropriées.

3. Devenue depuis le Message Passing Semantics Group.

On distingue deux types d'acteurs : les acteurs *sérialisés* et les acteurs *non sérialisés*. Les acteurs *sérialisés* sont les seuls à pouvoir changer d'état suite à l'exécution d'un message, ils ne traitent qu'un message à la fois et spécifient explicitement leur nouvel état avant de traiter le message suivant. Les acteurs *non sérialisés* peuvent traiter un nombre arbitraire de messages en parallèle [58].

3.3 Communications

L'activité d'un acteur consiste à recevoir des messages et à élaborer des réponses. Un acteur peut *déléguer* les messages auxquels il ne peut pas répondre. Dans les sections suivantes, nous présentons les communications dans un modèle acteur.

3.3.1 Envoi de messages

Dans un monde d'acteurs, tout est acteur et le seul événement qui peut se produire est l'envoi de message : un acteur émetteur transmet à un acteur récepteur un message. Ceci se fait normalement d'une manière asynchrone, mais certains langages d'acteurs (comme ABCL/1 [81]) offrent aussi les communications synchrones ou anticipées⁴.

Les messages contiennent des *continuations* qui sont des informations supplémentaires permettant de désigner l'acteur auquel sera envoyé le résultat du message. Une continuation peut être un autre acteur ou un acteur créé spécialement pour recevoir ce résultat. Par défaut, la continuation est l'émetteur du message.

3.3.2 Délégation

Il s'agit d'un mécanisme permettant à chaque acteur ne pouvant pas traiter un message de le *déléguer* à un autre acteur appelé *proxy* : le *proxy* prend en charge l'exécution du message et l'acteur déléguant est immédiatement disponible pour exécuter d'autres messages. Le *proxy* connaît l'acteur déléguant pour le cas où il aurait besoin d'informations supplémentaires.

Pour éviter une régression à l'infini dans le processus de délégation, il y a des acteurs qui ne délèguent jamais : ce sont les acteurs primitifs (appelés *rock-bottom* dans des langages comme Act1 [57]).

4. Une communication anticipée permet à un acteur d'envoyer un message dont il n'utilisera le résultat qu'ultérieurement : l'acteur continue son activité et sollicitera le résultat au moment où il en aura besoin, et si le résultat n'est pas encore disponible, il se mettra en attente.

La délégation est un moyen de partage de connaissances et de comportements plus souple que la notion d'héritage : un acteur choisit dynamiquement son *proxy* alors que le partage de connaissance décrit par un graphe d'héritage est statique. La délégation se caractérise donc par son aspect asynchrone, dynamique et non bloquant.

3.4 Le modèle de Gul Agha

Gul Agha a donné une définition formelle du modèle acteur tout en étendant sa sémantique : un acteur est désormais défini par l'adresse de sa boîte aux lettres où seront envoyés les messages qui lui sont destinés, et un comportement définissant le traitement de ces messages. Il a aussi ajouté le mécanisme de délégation qui n'existait pas chez Hewitt. Au niveau de la communication entre acteurs, il n'a retenu que la transmission asynchrone, toutefois les transmissions synchrones et anticipées peuvent s'exprimer en terme de transmissions asynchrones [82].

3.5 Conclusion

Les systèmes d'acteurs sont caractérisés par la communication par envoi de messages et un traitement local : un agent (acteur) détient des informations locales, et son comportement est indépendant de celui des autres. Ils présentent une distribution à la fois des connaissances, des résultats partiels et des méthodes utilisées pour la résolution du problème. En revanche, le comportement global d'un système d'acteurs est difficile à analyser, et les langages d'acteurs restent peu diffusés et sont loin d'être des produits finis. Les travaux menés dans ce sens tiennent plus de la conception de langages que de systèmes de résolution de problèmes⁵.

Une nouvelle approche en IAD se développe actuellement, elle essaye de faire coopérer des entités auxquelles sont rattachées des caractéristiques de haut niveau, tout en essayant de respecter les principes des systèmes ouverts. Ces entités seront désormais nommées *agents* et les systèmes correspondants seront appelés *systèmes multi-agents*. Nous étudions, dans ce qui suit, l'univers des systèmes multi-agents.

4 Évolution vers les systèmes multi-agents

Contrairement à l'approche centralisée de l'IA, l'intelligence artificielle distribuée vise la distribution de l'expertise sur un ensemble de composants qui

⁵. Peut-être, avec l'arrivée des architectures parallèles et des réseaux de processeurs, prendront-ils une autre dimension ?

communiquent pour atteindre un objectif global (élaboration d'un diagnostic, résolution d'un problème, etc.). Cette approche nécessite la division du problème en sous-problèmes. Mais cette hypothèse n'est pas toujours réaliste car beaucoup de problèmes ne peuvent être partitionnés de cette manière [42].

Une extension des systèmes d'IAD est proposée: les composants doivent être capables de raisonner sur les connaissances et les capacités des autres dans le but d'une coopération effective [50]. Pour ce faire, ils doivent être dotés de capacités de perception et d'action sur leur environnement et doivent posséder une certaine autonomie de comportement, on parle alors d'agents⁶ et par conséquent de systèmes multi-agents [26].

Les systèmes multi-agents se caractérisent alors par l'autonomie et l'intelligence des composants impliqués. Toutefois, un agent ne dispose pas d'une vision globale de son environnement.

4.1 Concept d'agent

Un agent peut être défini comme une entité (physique ou abstraite) capable d'agir sur elle-même et son environnement, disposant d'une représentation partielle de cet environnement, pouvant 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 [38].

Les agents ont deux tendances: une *tendance sociale* tournée vers la collectivité (les mécanismes et connaissances associés concernent les activités du groupe) et une *tendance individuelle* avec des mécanismes et des connaissances contenant les règles de fonctionnement interne de l'agent. On peut caractériser un agent par: son rôle, sa spécialité, ses objectifs et ses fonctionnalités, ses croyances, ses capacités décisionnelles, ses capacités de communication et éventuellement ses capacités d'apprentissage. Nous détaillons l'architecture et le fonctionnement d'un agent dans le § 5.

4.2 Déterminant d'un agent

On appelle *déterminant* d'un agent l'ensemble nécessaire et suffisant de ses caractéristiques structurelles, environnementales et comportementales qui permet d'expliquer ses façons d'agir.

Les caractéristiques environnementales d'un agent sont liées à la représentation que se fait l'agent de son environnement et de lui-même. Les caractéristiques structurelles d'un agent déterminent l'ensemble de ses composants, alors que les caractéristiques comportementales contraignent l'ensemble

6. La terminologie dans le domaine n'est pas encore clairement définie par la communauté IAD, et le terme agent reste encore un terme générique.

de ses comportements, en accord avec les caractéristiques environnementales. Par exemple, un avion est un agent dont les caractéristiques structurelles sont la capacité, la vitesse, etc. Les caractéristiques comportementales sont le pilote et le plan de vol et sa caractéristique environnementale est sa position [59].

4.3 Agents cognitifs vs réactifs

La granularité⁷ des agents impliqués dans une application varie selon deux écoles coexistantes aujourd'hui : l'école cognitive et l'école réactive. Suivant le type d'agent utilisé, on parlera de systèmes cognitifs ou de systèmes réactifs.

4.3.1 Agents cognitifs

Les systèmes d'agents cognitifs sont fondés sur la coopération d'agents capables à eux seuls d'effectuer des opérations complexes. Un système cognitif comprend un petit nombre d'agents qui disposent d'une capacité de raisonnement sur une base de connaissances, d'une aptitude à traiter des informations diverses liées au domaine d'application, et d'informations relatives à la gestion des interactions avec les autres agents et l'environnement [24]. Chaque agent est assimilable, suivant le niveau de ses capacités, à un système expert plus ou moins sophistiqué, on parle d'agent de forte granularité (*coarse grain*).

4.3.2 Agents réactifs

Les défenseurs de cette approche partent du principe suivant : *dans un système multi-agents, il n'est pas nécessaire que chaque agent soit individuellement "intelligent" pour parvenir à un comportement global intelligent*. En effet, des mécanismes simples de réactions aux événements peuvent faire émerger des comportements correspondant aux objectifs poursuivis. Cette approche propose la coopération d'agents de faible granularité (*fine grain*) mais beaucoup plus nombreux.

Les agents réactifs sont de plus bas niveau, ils ne disposent que d'un protocole et d'un langage de communication réduits, leurs capacités répondent uniquement à la loi stimulus/action.

Les premiers travaux relatifs à cette approche ont été réalisés au MIT en 1986 par R. Brooks [16]. D'après lui, plusieurs milliers de micro-robots identiques, d'une taille aussi petite que possible, travaillant ensemble sur une tâche donnée pourront être plus efficaces qu'un seul gros robot spécialisé⁸. Dans le

7. On entend par *granularité* le degré de détail des connaissances de l'agent. La granularité exprime la complexité des fonctionnalités d'un agent.

8. La construction de ces petits agents simples et dotés d'une autonomie réelle ne sera pas plus difficile que la fabrication des circuits intégrés d'aujourd'hui !

<i>Systèmes d'agents cognitifs</i>	<i>Systèmes d'agents réactifs</i>
Représentation explicite de l'environnement	Pas de représentation explicite
Peut tenir compte de son passé	Pas de mémoire de son historique
Agents complexes	Fonctionnement stimulus/réponse
Petit nombre d'agents	Grand nombre d'agents

TAB. 1 - les agents cognitifs vs réactifs

même sens, une équipe de l'université de Bruxelles développe des sociétés de micro-robots destinés à l'utilisation en milieu hostile ou au recueil d'échantillon sur d'autres planètes [30]. Une fois lancés dans la nature, ces micro-robots doivent se débrouiller seuls, s'adapter à l'environnement rencontré et coordonner leurs actions. Si l'un d'entre eux tombe en panne, le relais est pris par d'autres et le fonctionnement global ne sera pas condamné à l'arrêt (ceci n'est pas le cas si on utilise un seul gros robot et qu'un de ses membres tombe en panne). Le nombre important de robots utilisés permet de supporter des pertes sans remettre en cause l'objectif final.

En France on peut citer les travaux de J. Ferber à l'Université de Paris VI, qui a travaillé sur le langage MERING⁹ [33] [34]. Plusieurs versions se sont succédées depuis 1982, la dernière, MERING IV [37], est un langage concurrent pour la représentation des connaissances qui peut être utilisé comme une plateforme pour la conception de systèmes multi-agents réactifs. D'autres travaux plus récents portent sur l'éco-résolution considérée comme exemple d'agents réactifs [36].

Conclusion

Dans le tableau 1, nous résumons les différences entre les deux approches. C'est l'école cognitive qui, jusqu'à maintenant, a donné lieu aux applications les plus avancées. La complexité des systèmes réactifs exige le développement de nouvelles théories dans le domaine de la coopération, de la communication et de la compréhension de nouveaux phénomènes telle que l'émergence. Toutefois, il est possible de concevoir des systèmes hétérogènes comportant les deux types d'agents. Il est aussi possible de doter les agents cognitifs de capacités de réactions aux événements : on parlera alors d'agents hybrides¹⁰.

9. Premier langage hybride français, il a introduit la notion de réflexion.

10. À citer dans ce sens les travaux de l'équipe d'IA de l'université de Laval [18].

4.4 Caractéristiques d'un agent

L'avancement des travaux en IAD et SMA a conduit les chercheurs à définir non seulement la notion d'agent mais aussi quelques-unes de ses caractéristiques¹¹.

Intentionnalité

Un agent intentionnel est un agent guidé par ses buts. Une intention [70] [77] [69] est la déclaration explicite des buts et des moyens d'y parvenir. Elle exprime donc la volonté d'un agent d'atteindre un but ou d'effectuer une action. Notons que la notion d'intentionnalité n'existe pas chez Minsky¹².

Rationalité

Un agent rationnel est un agent qui suit le principe suivant (dit principe de rationalité) [63] :

“Si un agent sait qu'une de ses actions lui permet d'atteindre un de ses buts, il la sélectionne”.

Les agents rationnels disposent de critères d'évaluation de leurs actions, et sélectionnent selon ces critères les meilleures actions qui leur permettent d'atteindre le but. De tels agents sont capables de justifier leurs décisions.

La notion de rationalité se rapporte au comportement cognitif de l'agent. Ce terme qualifie l'utilisation efficace des ressources par l'agent. Les critères d'évaluation de cette rationalité sont étudiés dans [60].

Engagement

La notion d'engagement [6] [10] est l'une des qualités essentielles des agents coopératifs. Un agent coopératif planifie ses actions par coordination et négociation avec les autres agents. En construisant un plan pour atteindre un but, l'agent se donne les moyens d'y parvenir et donc s'engage à accomplir les actions qui satisfont ce but : l'agent croit qu'il est en mesure d'exécuter tout le plan qu'il a élaboré, ce qui le conduit (ainsi que les autres agents) à agir en conséquence.

Adaptativité

Un agent adaptatif est un agent capable de contrôler ses aptitudes (communicationnelles, comportementales, etc.) selon l'agent avec qui il interagit. Un agent adaptatif est un agent d'un haut niveau de flexibilité.

11. Les caractéristiques énoncées dans cette section se rapportent aux agents cognitifs.

12. Minsky utilise la notion d'*objectif* [61].