



HAL
open science

Proposition d'une approche auto-organisationnelle pour le partage de ressources critiques

Frédéric Armetta

► **To cite this version:**

Frédéric Armetta. Proposition d'une approche auto-organisationnelle pour le partage de ressources critiques. Autre [cs.OH]. Université Claude Bernard - Lyon I, 2006. Français. NNT: . tel-00449502

HAL Id: tel-00449502

<https://theses.hal.science/tel-00449502>

Submitted on 21 Jan 2010

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.



Proposition d'une approche auto-organisationnelle pour le partage de ressources critiques

THÈSE

présentée et soutenue publiquement le 8 Décembre 2006

pour l'obtention du

Doctorat de l'Université Claude Bernard Lyon I
(spécialité informatique)

par

Frédéric Armetta

Composition du jury

- Président :* Olivier Boissier, Professeur, École des Mines de St Etienne
- Rapporteurs :* Vincent Chevrier, Maître de conférence, HDR (Loria, Nancy I)
Jacques Ferber, Professeur Montpellier II
Paul Valckenaers, Senior Researcher, Université catholique de Louvain (Belgique)
- Examineurs :* Olivier Boissier, Professeur, École des Mines de St Etienne
Salima Hassas, Professeur Lyon I
Philippe Mathieu, Professeur, Université de Lille I
- Invité :* Emmanuel Gonon, Directeur Industriel (Société OSLO, Lyon)

Mis en page avec la classe thloria.

Remerciements

Il est surprenant d'être spectateur de son parcours, à l'occasion de remerciements, et de constater pour quelle faible part nous intervenons sur celui-ci et à quel point il est le résultat d'une succession d'opportunités. J'ai apprécié la qualité des relations que j'ai entretenues avec les personnes qui m'ont conseillé et guidé sur ce parcours, dans un vrai élan d'altruisme et d'intégrité.

Tout commence par une intrigue, un cours de deuxième cycle universitaire sur le vaste thème de l'intelligence artificielle, proposé par Salima Hassas et Simone Pimont. À ceci succède un projet TER passionnant sur le thème de la programmation génétique, encadré par Serge Fenet que je remercie. Ayant provisoirement interrompu mes études, une première expérience professionnelle de développement dans le milieu industriel a confirmé mon intérêt pour autre chose (la recherche), je remercie donc le naturel des acteurs de cette expérience. Par la suite, j'ai eu la chance de réintégrer l'Université. J'ai pu savourer pleinement, avec grand intérêt, la thèse que je présente avec ce mémoire.

Ce travail est le fruit d'une collaboration CIFRE entre la société OSLO dirigée par Emmanuel Gonon et le laboratoire LIRIS représenté par Salima Hassas et Simone Pimont. Je remercie Emmanuel Gonon pour l'enthousiasme sincère qu'il ressent à l'égard de la recherche, ce qui nous a permis de donner un sujet et un financement pour le travail présenté. L'enjeu d'une telle collaboration consiste à mettre en accord les besoins de la recherche avec des contraintes économiques industrielles peu flexibles. Je remercie Noël Perez qui, en participant à mon encadrement, a su modérer l'enthousiasme industriel lorsqu'il a fallu prendre du recul sur les thématiques abordées. Je remercie avant tout les membres de ma famille, et particulièrement ma femme, qui ont pu me supporter, m'apporter refuge et stabilité lorsque le contexte d'étude, bien que très porteur, devenait pesant et forçait la schizophrénie.

J'ai pu bénéficier d'un soutien très précieux et complémentaire de la part de Salima Hassas et de Simone Pimont de part la richesse de leurs enseignements sur le plan professionnel et humain. D'une part, Salima Hassas a su me transmettre une partie de ses idées novatrices, ambitieuses, visionnaires et réalistes, en développant mes capacités d'abstraction et mon jugement. D'autre part, Simone Pimont a su apporter un calme et une pédagogie indispensable aux discussions passionnées que nous avons ensemble. Je remercie aussi Olivier Lefevre qui s'intègre parfaitement à l'esprit d'ouverture de notre groupe de recherche, et qui poursuit avec beaucoup d'intérêt et d'imagination le travail amorcé avec la société OSLO.

Je tiens également à remercier les différents collègues de bureau que j'ai eu la chance de côtoyer, entre autres Denis Jouvin avec qui j'ai eu beaucoup de plaisir à discuter, l'ensemble des 'habitants' du bâtiment Nautibus pour leur convivialité et la bonne humeur qu'ils manifestent, les employés de la société OSLO qui ont chacun beaucoup de qualités, notamment Mohamed Sallami qui apporte spiritualité et sagesse à notre civilisation en perte de repères.

Une des grandes satisfactions de ce travail a consisté à fournir, finalement, ce mémoire aux membres de mon excellent jury, et de constater la finesse et l'intérêt de leurs appréciations. Je les remercie pour le travail attentif qu'ils ont accepté de fournir. Je remercie Jacques Ferber pour les très précieux encouragements qu'il m'a apporté, Vincent Chevrier pour la profondeur de sa lecture et pour ses nombreux conseils. Je remercie Paul Valckenaers qui nous a fait l'honneur d'étudier mon travail, et de le confronter par son expérience à une vision industrielle. Je remercie Olivier Boissier et Philippe Mathieu pour l'intérêt qu'ils ont manifesté à partager et échanger des idées, pour ce travail de thèse en le jugeant ou au sein du groupe SyCoSMA.

*'Le bonheur est le seul but
que nous choisissons toujours pour lui-même
et jamais pour une autre fin'*
Aristote

Table des matières

I	Positionnement du problème	1
1	Introduction	3
2	Caractéristiques du problème d'un point de vue SMA	7
2.1	Différentes approches	7
2.1.1	Parcours de l'espace de recherche	8
2.1.2	Stratégie de recherche	8
2.2	Propriétés multi-agents à exploiter	10
2.2.1	Intégration des perturbations au cours du processus de résolution	11
2.2.2	Modélisation propice à la représentation des problèmes complexes	11
2.3	Modélisation complexe des problèmes de partage de ressources critiques	12
2.4	Difficultés de négociation en environnement complexe	14
2.5	Notre positionnement	15
2.5.1	Identifier des besoins	15
2.5.2	Notre proposition	16
3	État de l'art	17
3.1	Coordination par contrats	18
3.2	Coordination médiée par l'environnement	21
3.3	Heuristiques	24
3.4	Perception et action : de local à global	25
3.5	Limitations ressenties , Contributions	27
II	Démarche d'étude	29
1	Exploration et convergence en cours de thèse	31
1.1	Point de départ	31

1.2	Parcours des différents modèles	32
2	Approche par combats	35
2.1	Contexte	36
2.1.1	Contexte et objectifs de notre étude de cas	36
2.1.2	Cadre de mise en oeuvre de l'approche	37
2.2	Première expression des relations entre agents	37
2.2.1	Arbitrage de la compétition	38
2.2.2	Expérimentations, constatations et résultats	39
2.2.3	Limites du modèle	41
2.3	Enrichissement de l'expression des relations entre agents	41
2.3.1	Arbitrage de la compétition	41
2.3.2	Expérimentations et résultats	43
2.3.3	Limites du modèle	46
2.4	Conclusion	47
3	Démarche pour le contrôle de l'activité	49
3.1	Motivation : une activité agent incontrôlée	49
3.1.1	Difficulté identifiée pour le problème de partage de ressources	49
3.1.2	Propagation de l'activité par les contraintes	50
3.1.3	Critères de difficulté pour la résolution décentralisée	51
3.2	Proposition : Stratégie de gestion de l'activité	52
3.2.1	Une nouvelle architecture	52
3.2.2	Gestion de l'activité du système	56
3.2.3	Capacité de perception du système	58
3.3	Apports	60
3.3.1	Comportement constaté	62
3.3.2	Interactions utilisateur	62
3.4	Limitations	62
3.5	Conclusion	63
4	Positionnement de CESNA	65
4.1	Communication des relations complexes au sein des systèmes multi-agents	65
4.2	Équilibre entre exploration et exploitation	66
4.3	Un nouveau processus de négociation	67

III	CESNA	71
1	Description de l'approche CESNA	75
1.1	Expression du problème de partage de ressources critiques par un réseau	75
1.2	Comportement de négociation	76
1.2.1	Perception de l'environnement	77
1.2.2	Influence sur l'environnement	78
1.3	Examen des capacités de perception et d'action pour CESNA	79
1.3.1	Relations entre agents	81
1.3.2	Influencer, percevoir et calculer sa satisfaction	84
2	Architecture de réseaux et comportements associés pour des problèmes applicatifs	87
2.1	CESNA Ordonnancement	87
2.1.1	Modèle	88
2.1.2	Modèle condensé	90
2.2	CESNA Coloration de graphes	90
2.3	Quelques critères pour l'élaboration de modèles applicatifs CESNA	92
2.3.1	Paramétrage des systèmes	93
2.3.2	Pertinence des informations véhiculées	93
2.3.3	Visibilité entre entités	94
3	Mesures et analyses pour le processus de résolution	97
3.1	Paramètre d'exploration et d'exploitation	98
3.2	Influence de la taille des problèmes	99
3.3	Mesure interne des influences et des échanges	100
3.4	Capacité de mémorisation	102
3.5	Mesures de performances	103
3.5.1	Problème d'ordonnancement	103
3.5.2	Problème de coloration de graphes	103
4	Apports de CESNA	107
Conclusion		
4.1	Négociation décentralisée en environnement complexe	111
4.2	De nouveaux champs applicatifs	113
Bibliographie		115

Table des figures

2.1	Arbre de parcours de l'espace de recherche	8
2.2	Suractivité au sein du processus de négociation	14
3.1	Cardinalité des relations entre agents	20
1.1	Démarche d'élaboration de l'approche CESNA	32
2.1	Arbitrage spatial des combats entre agents <i>Tâche</i>	38
2.2	équilibre des flux de production	41
2.3	Répartition [sans/avec] informations : éloignement de niveau 1	44
2.4	Répartition [sans/avec] informations : éloignement de niveau 2	45
2.5	Répartition [sans/avec] informations : éloignement de niveau 2 + difficultés	46
2.6	Perception compliquée face à des besoins de perception complexes	48
3.1	Suractivité des agents entraînée par les décisions locales	51
3.2	Problèmes propices à la 'suractivité agent'	52
3.3	Groupes <i>Séquencement</i> et <i>Affectation</i>	54
3.4	Modèle AGR (Agent Groupe Rôle)	55
3.5	Rôles des agents	55
3.6	Relation entre groupes de résolution	56
3.7	Calcul des marges de négociation pour le groupe <i>Séquencement</i>	59
3.8	Équilibre des déplacements réalisés sur l'environnement spatial	61
4.1	Chemin d'activité des systèmes multi-agents	68
4.2	Représentation des problèmes par réseaux de négociation	69
0.3	Vue générale de CESNA	73
1.1	Illustration : un réseau de négociation spatiale	77
1.2	Illustration : satisfaction des agents	77
1.3	processus de perception/action pour un agent	78
1.4	Illustration : perception de l'environnement	79
1.5	Illustration : influence sur l'environnement	81
1.6	Différents types de communication	82
1.7	Connexions directes ou indirectes entre agents	83
1.8	Décomposition des domaines de valeurs	84
2.1	Réseau CESNA pour l'ordonnancement	89
2.2	Réseau CESNA pour l'ordonnancement (Hachage sur deux niveaux)	91
2.3	Réseau de coloration : point de vue agent	91

Table des figures

2.4	Réseau de coloration	92
2.5	Communication inefficace	94
2.6	Représentations logiques similaires par des architectures différentes	94
2.7	Deux visibilitées différentes entre entités	95
3.1	Impact du paramètre d'exploration sur les performances (mesuré à partir de 100 exécutions))	98
3.2	Impact de la taille des problèmes sur les performances (points mesurés pour 100 exécutions)	100
3.3	Changements significatifs : de local à global et de global à local	101
3.4	De local à global : Équilibre exploration/exploitation	102
3.5	Mémorisation et persistance de motifs	102
3.6	Performances de CESNA pour un problème d'ordonnancement	103
3.7	Un problème de Mycielski	104
3.8	Performances de CESNA pour des problèmes de coloration de graphes de taille moyenne	105
3.9	Performances de CESNA pour des problèmes de coloration de graphes de grande taille .	105

Première partie

Positionnement du problème

1

Introduction

Le travail présenté dans ce mémoire de thèse s'intéresse à l'étude de la résolution d'un problème de partage de ressources critiques à l'aide d'une approche à base de multi-agents. La problématique que nous abordons se situe du point de vue des multi-agents dans le cadre de la résolution collective et décentralisée de problèmes à base d'agents réactifs, et du point de vue applicatif dans le cadre de problèmes distribués de satisfaction de contraintes (DCSP) complexes, exprimé par le problème de partage de ressources critiques. Dans l'approche que nous préconisons, nous nous distinguons des approches utilisant les multi-agents comme un outil de programmation distribué et coopérative d'un CSP, dans la mouvance des travaux initiés par [Yokoo 98]. Notre démarche s'inscrit dans le cadre développé dans [Hassas 03]. Elle consiste à représenter le problème à résoudre par un système multi-agents incarnant le problème traité dans son propre environnement de résolution, et permettant d'exprimer dans ce même environnement, les caractéristiques de ce problème ainsi que les échanges qui s'établissent entre les agents au cours de la résolution. Notre objectif est alors d'amener le système à exploiter ces informations pour coordonner l'activité des agents et guider l'activité du système vers une solution au problème posé.

Le paradigme multi-agents fournit un cadre innovant pour modéliser des problèmes et les résoudre. Il s'inscrit dans le prolongement des concepts introduits par le paradigme objet qui préconise une gestion séparée des processus et données des problèmes par l'utilisation de différents objets. Les agents bénéficient de plus d'autonomie que les objets, ils peuvent ainsi suivre leurs propres objectifs, faire preuve d'initiative, etc. Ainsi, les agents de ces systèmes intègrent un environnement de résolution au sein duquel ils pourront échanger des informations, se coordonner, négocier de façon décentralisée afin d'apporter une solution à un problème posé.

Cette façon d'aborder les problèmes applicatifs fournit des avantages qui les rendent particulièrement propices à certaines configurations de problèmes. Celle-ci peut dans certains cas faciliter la modélisation. En représentant les entités d'un problème applicatif à un niveau microscopique, par des agents, on développe des comportements d'agents permettant d'atteindre un objectif global complexe. Lorsque les problèmes sont complexes, la modélisation décentralisée est quelquefois plus facile à manipuler et à faire évoluer qu'une unique fonction complexe et abstraite ayant pour objectif de synthétiser les caractéristiques complexes du problème traité. Les systèmes multi-agents manifestent par ailleurs des capacités d'adaptation qui les rendent propices à l'implantation en environnement ouvert, lorsque les paramètres des problèmes traités évoluent en cours de résolution. En effet, dans ce contexte, les agents perçoivent l'évolution du problème comme une perturbation qui se matérialise dans leur environnement (autonomie des agents), ils peuvent alors réagir dynamiquement à celle-ci et dans le meilleur des cas, permettre l'absorption locale de la perturbation. En abordant les problèmes de façon centralisée, il est souvent plus difficile de communiquer la perturbation en cours de résolution, ce qui peut mener à une réexécution globale du processus de résolution.

Le travail de recherche que nous proposons a pour objectif de trouver une modélisation multi-agents appropriée pour le problème du partage de ressources critiques. Le problème de partage de ressources critiques est un problème d'optimisation combinatoire, i.e. le nombre des combinaisons à évaluer croît exponentiellement avec la taille des problèmes. Il n'est donc pas envisageable d'évaluer l'ensemble des combinaisons qui peuvent potentiellement constituer une solution. L'enjeu de notre étude consiste donc à élaborer des mécanismes de résolution qui permettent la convergence rapide vers une configuration satisfaisante.

Aborder ce problème de façon décentralisée entraîne des nécessités que nous allons détailler. Au cours de ce mémoire, nous voyons que les approches SMA connues peinent à traiter le problème de partage de ressources critiques. Notre analyse nous permet d'identifier les limites des modèles existants, d'envisager différentes modélisations pour finalement proposer une approche innovante. C'est seulement en trouvant un modèle de résolution performant que nous pouvons mettre à profit les qualités génériques fournies par le paradigme multi-agents. Comme nous allons le voir, le problème de partage de ressources critiques est un problème difficile. Dans ce contexte, sans performance de résolution, la modélisation multi-agents perd de son intérêt.

Le problème de partage de ressources critiques est un problème complexe, qui s'exprime de façon générale par un ensemble de consommateurs, où chaque consommateur consomme des parts de ressources critiques pour se satisfaire ; un ensemble de ressources qui fournissent des services en quantité limitée et un ensemble de contraintes qui limitent les possibilités d'affectation des ressources pour les consommateurs : les consommateurs respectent des motifs d'allocation de ressources. Le problème ainsi posé peut-être abordé comme un problème d'optimisation combinatoire pour lequel on définit un espace de recherche, correspondant à l'ensemble des affectations de ressources satisfaisant l'ensemble des contraintes. Comme décrit dans la partie 2.1, il existe différentes approches dans le cadre de la résolution de problèmes combinatoires permettant d'appréhender la complexité du problème traité. Ces approches procèdent par une organisation du parcours de l'espace de recherche, ainsi qu'une stratégie de recherche qui permet un parcours pertinent de celui-ci. La performance de ces approches est définie par les informations statiques apportées pour modéliser l'espace de recherche et représenter le processus de résolution. Une approche à base de multi-agents offre plus de flexibilité car l'espace de recherche et son parcours ne sont pas définis à priori, mais sont la conséquences d'interactions complexes entre agents.

La modélisation multi-agents (SMA) que nous proposons pour ce problème est motivée par la capacité des SMA à représenter la complexité sous-jacente aux problèmes de partage de ressources. Les différentes entités manipulées (ressources et consommateurs de ressources) sont en interactions rétro-actives. En effet, affecter une ressource à une entité d'un problème rend la ressource indisponible pour le reste du système. En abordant le problème de façon décentralisée, on distribue les contraintes et les objectifs entre différents agents qui entrent en négociation. Dans ce contexte, une modélisation possible serait de représenter chaque consommateur de ressources par un agent. Chaque agent doit alors négocier des parts de ressources qui lui conviennent dans un environnement complexe. L'état de la négociation est représenté par différents accords entre les entités du problème. Lorsque chaque agent se satisfait de ses acquis, on dispose d'une solution pour le problème. Comment choisir les accords à valider entre les différentes entités du problème ? Le processus de résolution consiste à faire évoluer itérativement un ensemble d'accords entre agents, qui matérialise un état de résolution. Comment gérer alors les processus d'engagement et de désengagement des différents agents ? En effet, au cours de la négociation, un agent peut demander la rupture d'un accord qui devient pour lui obsolète dans un environnement dynamique.

Nous analysons le cours de la négociation en terme d'exploration et d'exploitation de l'espace de recherche. L'exploration aléatoire permet aux systèmes multi-agents de diversifier la recherche dans l'environnement pour découvrir de nouvelles pistes. L'exploitation permet de renforcer de façon auto-catalytique les pistes prometteuses découvertes (i.e. plus une piste est prometteuse, plus elle est exploitée). On observe chez les sociétés de fourmis qu'un bon équilibre entre les processus d'explora-

tion et d'exploitation permet de tracer le chemin le plus court entre le nid et une source de nourriture [Bonabeau 97]. Comment peut-on utiliser ces mêmes processus dans le cadre d'une négociation complexe ? Une partie de l'activité des agents consiste à valider des accords de partage s'orientant dans la dynamique de la configuration courante, une autre partie élabore des accords ayant pour objectif la diversification des accords envisagés.

Notre étude souligne que la configuration de négociation est difficile à faire évoluer, que l'activité des agents entraîne un comportement chaotique amplifié par la complexité du système. En effet, lorsqu'un agent explore une alternative (nouvel accord ou rupture d'un accord), il perturbe l'ensemble de son environnement local. Les agents percevant la perturbation adaptent eux-même leur configuration ; la perturbation se propage alors par les relations complexes du problème. [Parunak 03] qualifie ce phénomène de confusion comme un phénomène de 'suractivité agent'. En effet, l'activité des différents agents peut être observée comme exagérée et manquant de cohérence. Celle-ci interdit alors toute performance de résolution, c'est-à-dire toute convergence sur une solution au problème traité dans des temps raisonnables. L'objectif de cette étude est d'élaborer une représentation qui permet aux agents de s'approprier les relations complexes des problèmes qu'ils traitent, en vue d'organiser efficacement la convergence vers un accord global de partage qui constitue une solution. Nous verrons que résoudre le problème de 'suractivité agent' ne consiste pas à calmer les agents comme le propose Parunak, mais à leur fournir des moyens de communication d'informations complexes de coordination de local à global et vice versa. Nous étudierons la capacité de mémorisation et de renforcement entre les sous-parties des problèmes modélisés au cours du processus de construction d'un accord global de partage et proposerons une approche auto-organisée.

La première partie du mémoire présente le problème de partage de ressources critiques d'un point de vue SMA. Nous décrivons tout d'abord d'autres méthodes de résolution qui n'appartiennent pas au domaine des systèmes multi-agents et qui abordent ces problèmes d'optimisation combinatoire. Ceci nous permet de souligner les avantages que peut présenter l'approche à base d'agents que nous décrivons comme un bon candidat pour le problème. Nous détaillons les difficultés propres à la négociation multi-agents en environnement complexe et définissons le cadre dans lequel nous souhaitons les aborder. Nous voyons ensuite comment les systèmes multi-agents abordent le problème de partage de ressources critiques et soulignons des limitations qui freinent leur performance pour le problème ; les limitations décrites constituent le sujet de notre étude, nous contribuons à leur apporter des solutions.

La deuxième partie du mémoire restitue un parcours expérimental qui a permis de converger vers une approche innovante. L'investigation que nous décrivons permet de cerner le problème et les moyens de l'aborder. Partant d'une approche SMA conforme aux approches les plus courantes, nous sommes confrontés à leurs limitations ; ceci nous permet de nous approprier le problème et d'envisager plusieurs façons de repousser ces limitations, au cours de l'étude d'une nouvelle architecture et de nouveaux comportements agents. Nous introduisons ensuite les concepts d'une nouvelle approche à partir des conclusions du parcours expérimental décrit dans cette partie.

La troisième partie consiste en une description détaillée de l'approche générique que nous proposons pour le partage de ressources critiques : CESNA (*Complex Exchange between Stigmergic Negotiating Agents*). Nous verrons quelques premiers éléments méthodologiques qui permettent de l'appliquer aux problèmes applicatifs. Nous étudierons les comportements endogènes de cette approche qui manifestent des propriétés d'auto-organisation. Nous analyserons le comportement du système et verrons ce qu'il apporte par rapport à d'autres approches à base d'agents. Certaines expérimentations nous permettent de positionner les performances très prometteuses de CESNA face à d'autres méthodes de résolution.

2

Caractéristiques du problème d'un point de vue SMA

Les Systèmes Multi-Agents (SMA) traitent le problème de partage de ressources critiques comme un problème de négociation décentralisée entre agents : chaque agent prend en charge la résolution d'une sous-partie du problème.

Celui-ci est un problème d'optimisation combinatoire, i.e. la taille des configurations à étudier pour résoudre ce problème croît exponentiellement avec la taille de ses instances. Dans ce contexte, quelle que soit la méthode de résolution que l'on choisit, il n'est pas envisageable d'évaluer l'ensemble des configurations disponibles. Nous voyons dans la section 2.1 différentes approches permettant d'aborder les problèmes d'optimisation combinatoire.

Nous présentons dans la section 2.2 des propriétés des systèmes multi-agents qui les rendent pertinents pour le partage de ressources critiques. En choisissant les systèmes multi-agents, nous souhaitons identifier les difficultés génériques du problème de partage de ressources critiques en tant que problème de négociation décentralisée et cherchons à le traiter de façon générique.

Nous décrivons dans la section 2.3 le problème de partage de ressource critique comme un système complexe, et voyons comment le modéliser à partir d'agents. Ceci nous permet de positionner des difficultés spécifiques à la négociation en environnement contraint et complexe (section 2.4), pour lesquelles nous positionnons notre approche et décrivons la proposition que nous développons au cours de ce mémoire de thèse (section 2.5).

2.1 Différentes approches

Dans cette section, nous décrivons les différentes approches qui sont envisagées pour traiter le problème de partage de ressources critiques qui est un problème d'optimisation combinatoire, i.e., un problème pour lequel on se heurte à une explosion du nombre de combinaisons à évaluer.

Pour notre étude, nous introduisons la notion d'espace de recherche : l'espace de recherche est constitué de l'ensemble des combinaisons de valeurs autorisées des variables utilisées pour une représentation de problème.

Pour ces problèmes, il n'est pas envisageable d'évaluer exhaustivement l'ensemble des combinaisons qui composent l'espace de recherche (excepté pour les problèmes de petite taille). Diverses approches permettent de diminuer considérablement les temps de calcul. Nous en décrivons le mode de fonctionnement par deux propriétés :

- l'organisation du parcours de l'espace de recherche
- la stratégie de recherche

2.1.1 Parcours de l'espace de recherche

On distingue différents types de parcours : parcours organisés à l'avance ou non organisés à l'avance. Les approches complètes permettent de garantir l'existence (ou l'inexistence) d'une solution, et de garantir l'optimalité de la solution proposée. Dans cette situation, il est nécessaire d'organiser le parcours afin de ne pas omettre d'évaluer certaines configurations de l'espace de recherche. On peut pour cela utiliser un parcours par arbre tel que nous le voyons sur la figure 2.1. Différentes stratégies de parcours que nous développons plus loin permettent d'accélérer la recherche sur ce support. D'autres représentations de l'espace de recherche peuvent être envisagées. Le parcours par treillis est l'une des alternatives permettant de représenter exhaustivement l'ensemble de l'espace de recherche. Dans cette situation, les noeuds de l'arbre peuvent connaître plusieurs pères.

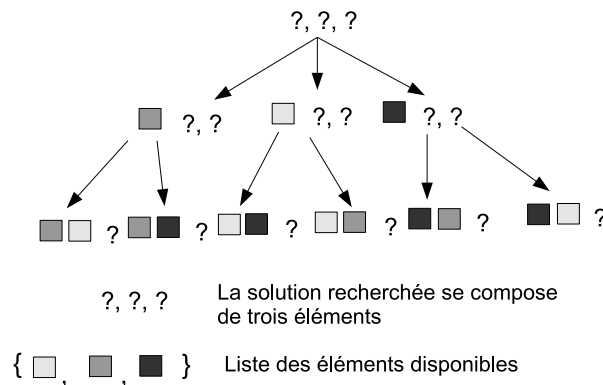


FIG. 2.1 – Arbre de parcours de l'espace de recherche

Pour certains problèmes, on se contente de solutions simplement satisfaisantes. Celles-ci sont proposées par des approches incomplètes qui ne garantissent pas l'optimalité des solutions découvertes. Comme nous l'avons vu, obtenir une solution optimale coûteuse à calculer n'a pas toujours de sens dans un environnement ouvert qui évolue en permanence. Pour obtenir une solution satisfaisante, il n'est pas obligatoire d'organiser de façon exhaustive le parcours de l'espace de recherche (aucune preuve d'optimalité à apporter). Ainsi, certaines approches telles que les approches à base d'agents font évoluer une situation sans mémoriser ni organiser à l'avance le parcours réalisé.

Pour les approches complètes ou incomplètes, les différents arrangements qui constituent l'espace de recherche sont parcourus par construction ou par voisinage. Par construction, les éléments qui composent la solution sont progressivement intégrés et évalués au cours de leur formation. Par voisinage, chaque combinaison est entièrement cohérente, les évaluations enclenchées au sein de l'espace de recherche sont réalisées sur des configurations entières.

2.1.2 Stratégie de recherche

Nous venons de voir différents types d'approches (complètes ou incomplètes), différents types de parcours (préétablis ou non) ainsi que différentes représentations de l'espace de recherche (approches constructives ou par voisinage). Ceci fournit déjà beaucoup d'alternatives pour le parcours de l'espace de recherche. Voyons maintenant les principales méthodes qui permettent de catalyser l'efficacité de la recherche.

Les méthodes de séparation et évaluation progressive (*Branch and Bound*) permettent d'éviter l'énumération de l'ensemble des solutions potentielles pour un problème. Le principe de décomposition de

l'espace de recherche (séparation) peut s'exprimer par un arbre similaire à celui que nous avons présenté précédemment. L'analyse des propriétés du problème abordé permet au cours du parcours de l'arbre de déterminer la qualité maximale que peut atteindre une configuration d'un sous-arbre (évaluation). On peut ainsi éviter l'évaluation de l'ensemble du sous-arbre si une configuration de meilleure qualité a déjà été rencontrée au cours du parcours. Les performances de cette méthode dépendent essentiellement des heuristiques utilisées pour le parcours. Ces heuristiques ont pour objectif de rencontrer des solutions de qualité élevée rapidement, ceci afin de réduire efficacement l'espace restant à l'explorer. L'analyse des capacités d'un sous-arbre peut par exemple être réalisée par des méthodes développées dans le domaine de la propagation de contraintes. Les travaux de Caseau et Laburthe décrivent comment allier ces méthodes avec la propagation de contraintes [Caseau 95].

La programmation mathématique qui provient du domaine de la recherche opérationnelle met en jeu des outils analytiques qui mènent à une configuration optimale respectant un ensemble de contraintes. Le principe appliqué pour la programmation dynamique est la décomposition du problème en plusieurs sous-problèmes plus simples de taille réduite. De cette manière, l'ensemble des sous-problèmes sont résolus à rebours afin de résoudre le problème global. Pour la résolution par programmation linéaire, on simplifie le problème afin de le rendre linéaire. Cette simplification peut intervenir par la relaxation Lagrangienne (pour un problème d'ordonnancement : relaxation des contraintes chronologiques des tâches des produits et des contraintes de capacité des ressources). Un arrangement de l'ensemble des relaxations est ensuite effectué afin de trouver une solution réalisable ne violant aucune contrainte. Dans [Wang 97], on nous propose par exemple une méthode de décomposition d'un problème de type jobshop (problème d'ordonnancement) en utilisant la relaxation Lagrangienne. L'approche analytique ne se prête pas à tous les types de problèmes. Elle permet cependant de trouver des résultats satisfaisants dans certains cas particuliers [Wang 97].

Les heuristiques permettent d'accélérer la recherche dans des domaines applicatifs spécifiques. Différentes heuristiques de parcours sont suffisamment générales pour être appliquées à plusieurs catégories de problèmes d'optimisation combinatoire, elles portent le nom de *métaheuristiques*. Voyons donc différentes approches par métaheuristiques.

L'approche constructive correspond à la création progressive des solutions sans remise en question des choix effectués. On fixe ainsi les différentes variables composant la solution recherchée jusqu'à obtenir une configuration complète. Ce mode de construction ne garantit pas l'optimalité. Grâce à son faible temps de calcul, il peut être exécuté plusieurs fois de suite. La majorité des méthodes constructives sont de type glouton : à chaque étape de la construction de ces solutions, on sélectionne la meilleure possibilité 'à court terme', on ne considère pas les conséquences de ces choix sur la solution finale. Ces solutions sont ainsi reconnues pour la myopie de leur mode de résolution [Lopez 01].

Pour les approches par recherche locale, les solutions manipulées sont entières. On définit l'ensemble du voisinage d'une solution comme étant l'ensemble des positions que l'on peut atteindre en une transformation dans l'espace de recherche. La méthode de parcours à seuil consiste à remonter vers le maximum local dans l'espace des solutions. Ainsi, si la solution atteignable est meilleure que la solution courante, on effectue le déplacement. Le recuit simulé s'inspire de la méthode d'acceptation à seuil, mais permet le déplacement par dégradation de la solution.

La méthode tabou correspond aussi au parcours local de l'espace de recherche. Suivant cette méthode, les dernières opérations de déplacement sont stockées dans une file de taille limitée. À chaque étape, on mémorise le dernier déplacement et on oublie le plus ancien. La méthode tabou interdit d'effectuer une opération de déplacement annulant une des opérations de déplacement présente dans la file des déplacements. La méthode tabou permet ainsi de parcourir l'espace de recherche avec plus de cohérence.

Contrairement aux approches constructives et locales qui font intervenir une solution unique (partielle ou non), les approches évolutives font intervenir un ensemble de solutions à chaque étape du pro-

cessus de résolution. L'algorithmie génétique est un bon exemple d'approche évolutive. Celle-ci s'inspire de l'évolution des espèces par adaptation génétique à leur environnement.

Chaque solution potentielle est représentée par un individu qui est évalué suivant l'objectif que l'on cherche à atteindre pour le problème. À chaque génération, les meilleurs individus évalués sont croisés : ils forment de nouveaux individus. Suivant ce processus, la qualité des individus s'améliore rapidement au fil des générations jusqu'à obtenir un individu associé à une solution de qualité. L'essentiel de la performance de cette approche provient d'un codage efficace des individus ainsi que des fonctions de croisement et de mutation utilisées. Dans [Xu 00] on nous présente un codage pour les individus permettant par évolution génétique d'obtenir une bonne répartition des tâches à réaliser sur les différentes ressources. Dans [Ombuki 04], on nous propose de résoudre les problèmes de type jobshop par algorithmie génétique.

Les algorithmes d'optimisation à base de colonies de fourmis font aussi partie des approches évolutives [Dorigo 97] [Pimont 00]. Pour ces algorithmes, on exploite la capacité d'émergence des systèmes à base d'agents.

Chacune des métaheuristiques décrites ci-dessus reflète des caractéristiques intéressantes pour le parcours de l'espace de recherche. Ces différents intérêts peuvent être facilement combinés par le biais d'approches hybrides [Galinier 99]. Par exemple, il est possible de commencer la recherche par un algorithme génétique pour parcourir efficacement d'importantes régions de l'espace des solutions.

On peut ensuite appliquer le recuit simulé combiné à la méthode tabou au meilleur individu trouvé dans le but de vérifier rapidement que la solution est optimale dans sa région. Dans [Ombuki 04], on combine l'algorithmie génétique avec la méthode tabou ce qui améliore les performances de résolution pour le problème du jobshop. Dans [T'Kindt 00], on nous propose de combiner le recuit simulé avec l'optimisation par colonies de fourmis.

Nous venons de décrire une large palette de possibilités pour la résolution du problème de partage de ressources. Celles-ci ne déploient pas toutes la même d'agilité, essentielle en milieu industriel. Par exemple, l'utilisation de connaissances spécifiques propres aux problèmes applicatifs (heuristiques applicatives, formalisation mathématique spécifique, moteur d'inférence sur contraintes, etc.) permet d'améliorer considérablement les performances de résolution, mais réduit aussi la généralité des approches et leur flexibilité.

2.2 Propriétés multi-agents à exploiter

Comme nous l'avons mentionné au début de la section 2.1, les approches basées sur les techniques d'optimisation se focalisent sur l'organisation à priori du parcours de l'espace de recherche et la définition à priori d'une stratégie de recherche.

Une approche à base de multi-agents, offre plus de flexibilité en mettant en oeuvre un ensemble de comportements d'agents où l'organisation de l'espace de recherche et les stratégies d'exploration et d'exploitation de cet espace de recherche ne sont pas définies à priori, mais construites par les agents en mettant en oeuvre des comportements plus ou moins complexes et en exploitant les caractéristiques du problème, qui seront inscrites dans l'environnement d'évolution des agents.

Les propriétés développées par les systèmes multi-agents (représentations adaptées aux problèmes, anthropomorphisme, décentralisation du processus de résolution, distribution, hétérogénéité, tolérance aux changements, etc.) sont propices à l'exploitation dans le milieu industriel [Parunak 96].

À l'ère des entreprises agiles, on positionne de nouveaux besoins technologiques. Nous en citons quelques uns qui nous préoccupent pour choisir une technologie propice aux configurations d'application dans le cadre d'applications industrielles.

2.2.1 Intégration des perturbations au cours du processus de résolution

Beaucoup de problèmes d'affectation de ressources critiques se positionnent dans un environnement dynamique et ouvert (c'est le cas du problème de gestion de production). Les données des problèmes (tâches à réaliser, ressources disponibles, objectifs, etc.) peuvent varier au cours du temps, ce qui correspond à une perturbation pour le processus de résolution. Les logiciels des entreprises doivent alors tolérer les aléas qui créent des différences entre les prévisions et les réalisations concrètes. Ils ont à les percevoir et à proposer des adaptations.

Étant donné le coût de la mise en application 'physique' d'une solution et la fréquence souvent élevée des perturbations, il est indispensable de privilégier l'intégration locale des perturbations plutôt que la re-génération d'une solution globale à chaque perturbation [Sanlaville 02].

En recherchant un processus de négociation qui s'approprie les relations complexes des problèmes, on souhaite fournir au système multi-agents la capacité de réaction dynamique aux perturbations. La négociation complexe pourrait alors réagir dynamiquement aux différentes perturbations, et proposer des réorganisations locales. Dans ce document, nous proposons une approche qui permet une appropriation des relations complexes par le système. Une des perspectives de ce travail consiste en l'intégration locale des perturbations. Le complément proposé pourrait alors rendre l'approche robuste aux perturbations, et capable de s'auto-organiser.

Cette forme de robustesse est plus difficile à implémenter sur des approches centralisées pour lesquelles l'intégration dynamique des perturbations semble moins appropriée. En effet, les perturbations des systèmes s'expriment le plus souvent par les entités microscopiques qui les composent. Reporter une perturbation microscopique à partir d'une représentation centralisée macroscopique est souvent une opération délicate.

De plus, l'intégration de la perturbation par le processus de résolution est elle-même délicate, ce qui entraîne le plus souvent une nouvelle exécution du processus de résolution.

Sur ce sujet, les systèmes multi-agents sont appréciés pour la robustesse qu'ils manifestent face aux perturbations. Chaque agent perçoit et réagit de façon autonome à son environnement. Les perturbations peuvent correspondre à des variations d'objectifs, des variations des composants du système, des communications bruitées entre les composants, etc. Les agents opèrent de façon asynchrone et décentralisée ce qui facilite leur adaptation aux situations rencontrées. [Tranvouez 01] décrit par exemple la robustesse de ces systèmes pour les problèmes de réparation d'ordonnancement. La modélisation proposée facilite l'intégration locale des perturbations.

2.2.2 Modélisation propice à la représentation des problèmes complexes

Une nouvelle tendance pousse les entreprises à s'adapter rapidement à un contexte économique qui évolue. L'agilité nécessaire qu'elles doivent mettre en oeuvre s'exprime en partie par les logiciels qu'elles utilisent. Plus les logiciels informatiques sont flexibles (capables de s'adapter à de nouvelles configurations), plus les entreprises sont performantes (rapidité d'action sur les marchés, coût de possession des systèmes informatiques plus faible, meilleure formation du personnel, etc.).

Pour le partage de ressources critiques, la configuration des problèmes traités est très sensible à l'évolution (nouvelles catégories d'objectifs, nouveaux problèmes, nouvelles contraintes, etc.). La capacité de personnalisation des moteurs de résolution nécessite l'appui de technologies adaptées, toutes ne le sont pas. Certaines approches non flexibles poussent à revoir les processus de résolution dans leur ensemble, à chaque modification. Pour celles-ci, dans le cas le moins favorable, la reconfiguration peut nécessiter le soutien d'équipes de docteurs en informatique. Les technologies 'boite noire' dont on perçoit mal le fonctionnement entraînent des pénalités de délais et de coût qui sont à éviter (quand c'est possible).

En modélisant les problèmes de partage à un niveau microscopique, et en apportant une solution générale aux différents problèmes rencontrés, on peut représenter les problèmes plus facilement qu'en utilisant des fonctions abstraites qui expriment les caractéristiques complexes des problèmes traités. Lorsque les besoins de l'entreprise changent, on répercute les modifications que l'on identifie facilement au sein du modèle agent associé (proximité du modèle agent avec le problème traité). Ce type de modification est moins intuitif sur un modèle centralisé.

2.3 Modélisation complexe des problèmes de partage de ressources critiques

Dans cette section, nous décrivons le problème de partage de ressources critiques comme un système complexe. Nous voyons dans la section suivante que cette complexité constitue une difficulté pour la négociation décentralisée entre agents. Un système devient complexe lorsque les évolutions des entités qui le composent sont rétroactivement liées.

Pour les systèmes complexes, une infime variation des conditions initiales mène à des résultats totalement différents. Illustrons ce phénomène par un billard virtuel sur lequel on projette une boule qui rebondit et percute d'autres boules un grand nombre de fois avant de se stabiliser. Une très faible variation des conditions initiales (angle de projection) mène à des configurations totalement différentes, résultat des interactions complexes entre les différentes boules du billard qui propagent et amplifient la faible variation initiale.

Dans le cadre de la négociation complexe que nous cherchons à développer, l'activité d'un agent a une influence sur l'évolution du cours de la négociation. Une faible variation de sa stratégie mène ainsi à des résultats totalement différents. Nous voyons dans la section suivante que les relations complexes des problèmes de partage de ressources critiques sanctionnent globalement les erreurs locales d'appréciation des agents en négociation.

Les systèmes complexes sont par ailleurs difficilement décomposables. En effet, l'évolution de chacune des entités dépend de l'évolution du reste du système. Cet autre aspect qui découle des relations de complexité nous permet de nous interroger sur la meilleure façon de mettre en oeuvre la négociation entre les agents pour ce problème. Au cours de la négociation, les agents valident/invalident des accords, ce qui a pour objectif de construire une solution pour le problème posé. Dans le cadre d'une négociation complexe, la décision d'un agent est en interaction avec la décision d'autres agents qui partagent un environnement commun. Doit-on décomposer le mécanisme de décision qui permet à un agent de valider/invalidier un accord ? Est-il plus judicieux de corréliser la complexité des relations à une activité complexe de négociation ?

Les problèmes de partage de ressources critiques mettent en jeu différentes entités, qui appartiennent à un même système. Après avoir décrit les caractéristiques générales des systèmes complexes, voyons comment se manifeste la complexité pour le partage de ressources critiques. Les illustrations que nous fournissons permettent de se faire une idée des enjeux du processus de négociation que l'on cherche à élaborer.

Dans le cadre d'une négociation pour le partage de ressources critiques, chaque agent est responsable d'une sous-partie d'un problème et dispose d'un ensemble d'accords en relation avec d'autres agents parmi lesquels certains seront validés/invalidés. L'espace de recherche correspond alors à l'ensemble des combinaisons d'état (valide / non valide) des différents accords disponibles. L'espace de recherche contient une ou plusieurs solution(s) au problème que l'on cherche à retrouver par une négociation décentralisée.

Le modèle d'accord entre agents que l'on choisit définit la taille de l'espace de recherche. Il est intéressant de favoriser des modèles d'accord de taille réduite, ce que nous avons fait dans l'approche

que nous proposons dans ce document (modèle de représentation condensée en partie III). Cependant, le modèle d'accord est aussi à étudier en vue d'y installer une activité de négociation décentralisée performante, ceci a été notre premier objectif.

Les problèmes de partage de ressources connaissent trois caractéristiques communes :

- Un ensemble de consommateurs : Chaque consommateur consomme des parts de ressources critiques pour se satisfaire
- Un ensemble de ressources : les ressources fournissent des services en quantité limitée
- Un ensemble de contraintes qui limitent les possibilités d'affectation des ressources pour les consommateurs : les consommateurs respectent des motifs d'allocation de ressources.

Ces caractéristiques générales s'expriment de différentes manières suivant le problème applicatif :

- Pour le problème d'ordonnancement, les produits (un ensemble de tâches) parcourent les machines afin d'être fabriqués. Le planning de production doit respecter les contraintes chronologiques entre les tâches du même produit (motif d'allocation de ressources). Des contraintes additionnelles telles que le respect des dates de livraisons complexifient le problème.
- Pour le problème de coloration de graphe, on associe à chacun des noeuds une couleur différente de la couleur de ses voisins. On limite le nombre de couleurs disponibles, chaque couleur est une ressource critique. Les motifs d'affectation choisis doivent respecter les contraintes représentées par la structure du réseau de coloration.
- Pour les problèmes d'affectation de ressources humaines, les tâches à réaliser (produire, former, se former, réaliser la maintenance, R&D, etc.) nécessitent des équipes adaptées (motifs d'allocation de ressources). Chaque individu possède des compétences (qui peuvent évoluer) et peut participer à différentes activités.
- etc.

Les problèmes de partage de ressources critiques sont complexes car l'affectation d'une part de ressource à un consommateur de ressources réduit l'ensemble des possibilités restantes pour les autres consommateurs :

- Pour l'ordonnancement, réserver le planning d'une ressource de production pour une tâche rend la ressource indisponible pour le reste du système.
- Pour la coloration de graphes, affecter une couleur à un noeud rend indisponible la couleur utilisée à l'ensemble des voisins du noeud coloré.
- Pour l'affectation de ressources humaines, l'affectation d'une ressource agit sur les affectations que l'on peut/doit réaliser pour le reste des ressources humaines.
- etc.

Pour les problèmes que nous abordons, la complexité des systèmes provient des contraintes appliquées aux problèmes : ce sont en effet celles-ci qui font entrer en interactions les différentes entités au cours de la négociation. Lorsque les ressources sont disponibles en grande quantité, ou que les motifs d'allocation sont peu contraignants, les interactions entre les composants du système sont moins prononcées. On se situe alors plutôt dans le domaine du partage de ressources, et les difficultés de négociation décrites dans la section suivante s'expriment plus discrètement : les méthodes connues de négociation décentralisée fonctionnent. En effet, lorsque les problèmes sont sous-contraints, il existe beaucoup d'alternatives satisfaisantes : ceci facilite l'élaboration d'un accord global satisfaisant.

Lorsque les problèmes sont sur-contraints, on prouve facilement qu'il n'existe pas de solution. Les problèmes les plus délicats se situent à la frontière de ces deux paysages de recherche décrits, dans cette situation l'espace de recherche doit être examiné en profondeur pour trouver une des rares solutions, si elle existe.

Pour notre étude, nous étudions des configurations pour lesquelles les interactions entre les entités des problèmes (ressources et consommateurs de ressources) sont suffisamment fortes pour produire un

phénomène qualifié de ‘suractivité agent’ par [Parunak 03]). Ceci nous incite à rechercher un processus de négociation générique décentralisé entre agents qui tolère la complexité des relations des problèmes.

2.4 Difficultés de négociation en environnement complexe

[Parunak 03] observe sur différents problèmes de partage de ressources critiques un phénomène de ‘suractivité agent’ en fin de convergence des négociations décentralisées. Il semble que l’activité de certains agents ne soit pas bénéfique à la convergence vers un accord global entre agents qui matérialise une solution.

Ce problème de suractivité est intimement lié à la complexité des systèmes. Au cours du processus de négociation, lorsque un agent fait un choix, celui-ci impacte le processus global de négociation. Les agents ne perçoivent pas l’impact global de leur activité, et ne peuvent ainsi pas diminuer les nuisances qu’ils occasionnent lorsqu’ils prennent de mauvaises décisions.

Nous illustrons un phénomène qualifiable de suractivité sur la figure 2.2, pour un problème d’ordonancement. Sur la partie gauche, un agent *Tâche* prend la décision décentralisée de se déplacer d’une machine à une autre. Cette démarche pourrait se manifester par la rupture d’un accord avec la machine de départ, et un nouvel accord avec la machine d’accueil. Cette décision locale d’apparence anodine peut chambouler l’ensemble de la configuration courante. En effet, sur la machines de départ, le vide doit être comblé. Une possibilité serait de décaler l’ensemble des agents *Tâche* disponible sur la machine. Il y aurait alors un risque de violer certaines des contraintes de chronologie pour ces agents¹, ce qui aurait pour effet de propager la réorganisation à l’ensemble du système. De la même façon, les agents présents sur la machine d’accueil ont à modifier leurs positions, et peut-être réorganiser plus globalement la situation. Nous voyons sur cet exemple que le gain de satisfaction de l’agent initiateur peut être inférieur à la dégradation globale occasionnée.

L’évolution de la configuration courante doit être envisagée en coordonnant les décisions de chacun des agents, afin d’élaborer des accords de réorganisation tels que celui présenté sur la partie droite de la figure 2.2.

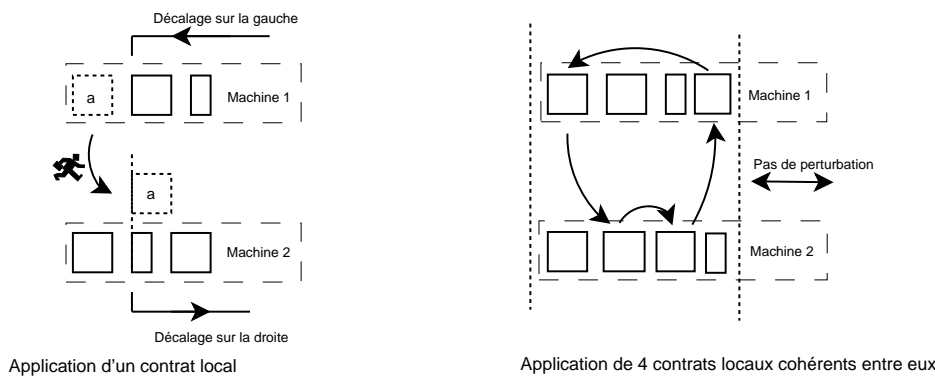


FIG. 2.2 – Suractivité au sein du processus de négociation

Lorsque les chances de dégradation sont trop fortes, le cours de la négociation ne permet plus la convergence sur une configuration satisfaisante. L’ensemble de l’étude que nous proposons consiste à remédier à ce problème qui interdit l’accès du partage de ressources critiques aux systèmes multi-agents.

¹Chaque agent *Tâche* doit respecter un placement chronologique avec les agents *Tâche* de son produit associé

[Parunak 03] propose de calmer l'activité des agents. Notre travail consiste plutôt en l'élaboration d'un nouvel environnement de négociation qui permette aux agents d'agir avec plus de pertinence, en environnement complexe de négociation.

2.5 Notre positionnement

Après avoir illustré la complexité des relations pour le problème de partage de ressources critiques, et les problèmes de 'suractivité agent' qu'elle engendre, nous positionnons la démarche que nous suivons pour la négociation complexe.

2.5.1 Identifier des besoins

Le problème de partage de ressources critiques étant un problème d'optimisation combinatoire, il est nécessaire de coordonner efficacement l'activité décentralisée de négociation des agents afin de converger rapidement sur un accord global satisfaisant tous les agents, et donc fournissant une solution au problème posé. Cette performance ne peut être atteinte sans apporter des réponses au problème de 'suractivité des agents'.

– **Vers des interactions de cardinalité $n \leftrightarrow n$**

Certains protocoles de négociation multi-agents prévoient des interactions de cardinalité $1 - n$ entre les agents. Ils restituent la complexité des relations par de multiples échanges de ce type ($n \times (n \leftrightarrow 1)$). Les systèmes complexes mettent cependant en jeu des boucles rétroactives d'activité qui les rendent inadaptés à la résolution par décomposition occasionnée.

Illustrons ce propos à partir du protocole de cardinalité $1 \leftrightarrow n$ nommé *Contract Net*, et voyons les limitations occasionnées par la décomposition.

Reprenons le cas d'un problème d'ordonnancement pour lequel un agent *Tâche* souhaite se déplacer sur une machine. Il peut par exemple interroger les différents agents *Tâche* établis sur les positions qu'il convoite. Ceux-ci ne pourront alors pas faire de proposition concrète tant qu'ils n'auront pas eux-même trouvé un nouvel emplacement d'accueil. Pour cela, ces agents propagent une nouvelle vague de négociation, elle-même soumise au même problème que la requête initiale. Suivant les mêmes principes qui engendrent une 'suractivité agent' pour le problème de partage de ressources critiques, les interactions $1 \leftrightarrow n$ se propagent rapidement à tout le système. Les différentes requêtes sont elles-mêmes en interactions complexes, sans que les processus de décisions puissent vraiment les percevoir. On assiste à un phénomène de blocage (les requêtes s'attendent entre elles) ce qui produit des latences au sein des interactions sociales. Dans ce contexte, quand valider un accord ? sur la base de quelle perception ?

Remarque : Des stratégies envisageables en utilisant des interactions de cardinalité $1 \leftrightarrow n$?

Afin de résoudre les problèmes de blocage et de latence, on peut envisager de compenser les difficultés de perception des relations complexes par des choix heuristiques qui permettent des décisions plus locales. Dans cette situation, on risque de nouveau d'être confronté à un problème de suractivité consécutive à la myopie des décisions.

Dans le cadre de notre étude, nous nous orientons vers l'élaboration d'un processus permettant la perception des relations complexes. Agir ainsi nous semble être en accord avec le problème traité.

– **Vers la coordination globale de l'activité**

L'ensemble des agents négocient au sein d'un même environnement, l'état global du système doit donc influencer leurs propres décisions. Ce travail a aussi pour objectif de permettre la coordination de l'ensemble de l'activité des agents. Les relations que nous cherchons à développer inter-

viennent comme un support de négociation dont les différents agents doivent s'imprégner afin de communiquer à travers elles des informations complexes de coordination de local à global et vice versa.

2.5.2 Notre proposition

Nous réalisons ce travail dans le cadre défini dans [Hassas 03] : des agents réactifs utilisent l'environnement comme moyen d'interaction et aussi comme support d'inscription des effets de leurs actions. Ceci exprime l'influence exercée par les effets persistants dans l'environnement, des comportements passés, sur leurs comportements futurs. Cette communication et coordination de l'activité a été d'abord observée chez les insectes sociaux par [Grassé 59], on la nomme stigmergie. Quelles sont les particularités que la communication stigmergique doit développer pour être adaptée à cette catégorie de problèmes ? Comment la mettre en oeuvre ? Comment le résultat émergent des interactions entre agents peut permettre l'émergence d'une solution complexe ?

L'activité menée par le système valide/invalide certaines alternatives parmi un ensemble de combinaisons de grande taille, et permet de converger vers une solution. L'exploration aléatoire permet aux systèmes multi-agents de diversifier la recherche dans l'environnement pour découvrir de nouvelles pistes. L'exploitation permet de renforcer de façon auto-catalytique les pistes prometteuses découvertes (i.e. plus une piste est prometteuse, plus elle est exploitée). Les systèmes réactifs sont inspirés du modèle naturel des insectes sociaux [Bonabeau 97]. Les agents dits réactifs suivent un comportement local simple (ils réagissent par 'reflex' à leur environnement) et entrent en interaction afin de faire émerger une intelligence collective à un niveau plus global (ils se différencient des agents dits cognitifs qui bénéficient de capacités de raisonnement individuel approfondies (capacités de déduction par exemple) plutôt que d'un comportement global performant fruit de leurs interactions). Comment exprimer cette exploration/exploitation supportée par une communication des relations complexes du problème traité ?

Notre travail contribue à apporter quelques éléments de réponse à ces questions, afin de donner au système un comportement de perception complexe (local/global) adapté à un comportement d'émission complexe (local/global). L'activité locale et l'activité globale sont corrélées par des informations complexes qui parcourent le système, comment laisser s'exprimer ces informations comme résultat émergent de l'activité ? Notre objectif intervient à contre-courant d'un guidage artificiel exogène au système (difficile à implémenter dans ces situations complexes et décentralisées). Il vise à l'auto-organisation du système par émergence de structures (marquées dans l'environnement) à l'origine mais aussi résultats de ses propres interprétations.

3

État de l'art

L'intelligence artificielle distribuée aborde le problème de partage de ressources critiques par élaboration d'accords entre les différents agents en interaction. Afin de profiter pleinement des qualités des systèmes multi-agents, il faut apporter des réponses aux différentes difficultés résultant de la décentralisation.

Les points suivants sont à aborder :

- Perception et action : La négociation entre agents intervient par la perception et l'action de ceux-ci. Dans le cadre des problèmes complexes, la perception/action des agents est caractérisée par la complexité des relations à rapporter.
- Temps de latence au sein des interactions sociales : Il n'est pas toujours possible de restituer le contexte global de l'état du système à chaque perception effectuée par un agent. Aussi, un agent ne perçoit pas instantanément le résultat de son activité car celui-ci est conséquent au comportement des autres agents du système.
- Le parcours de l'espace de recherche n'est pas organisé : les choix réalisés par les agents ne sont pas réversibles car on ne garde aucune trace historique de l'activité agent.

La plupart des approches que nous allons décrire se rapportent à des problèmes d'ordonnement pour lesquels la communauté scientifique prononce un vif intérêt. Ce type de problème représente fidèlement les difficultés rencontrées dans le cadre d'une résolution multi-agents.

L'étude que nous présentons souligne les choix proposés par différents travaux. Nous parcourons différentes organisations d'agents (des modèles hiérarchiques à hétérarchiques), différents protocoles de communication (communication de type contractuelle ou bien communication par marquage de l'environnement) afin de caractériser l'état de la recherche dans le domaine. La plupart des approches que nous étudions sont réactives.

Les problèmes de partage de ressources critiques sont décomposés et répartis entre agents qui négocient afin d'élaborer des accords qui constituent une solution pour les problèmes abordés. D'une part, nous étudions dans la section 3.1 ces accords ou contrats en tant que résultat d'une volonté explicite d'organiser leurs élaborations (*Contract Net*, négociation par enchères, etc.). D'autre part, nous les étudions comme le résultat d'interactions médiées par l'environnement en section 3.2.

Il est possible d'envisager la solution obtenue comme résultat de conditions favorables maintenues par différentes heuristiques. En section 3.3 nous voyons quelques possibilités de maintien de ces conditions pour des problèmes pour lesquels l'interprétation par négociation au niveau micro de l'évolution complexe du système est particulièrement délicate.

En section 3.4 nous relevons certaines approches qui nous permettent d'illustrer les enjeux de la négociation décentralisée pour le partage de ressources critiques. Pour finir, nous soulignons en section

3.5 quelques limitations des approches étudiées et montrons l'orientation de nos recherches en vue de participer à la réduction de celles-ci.

3.1 Coordination par contrats

Jennings s'intéresse aux techniques de coordination pour l'intelligence artificielle distribuée [Jennings 96]. Il propose une formalisation des buts et sous-butts de chaque agent en vue d'atteindre un état de satisfaction global. Chaque agent peut pour se satisfaire partager des intérêts avec d'autres agents, bénéficier de diverses alternatives, etc. Jennings propose que tout processus de coordination peut être intégralement décrit par les notions d'engagements (révisables ou non), de conventions sociales ainsi que de raisonnements locaux. Dans le cadre du partage de ressources critiques, les engagements sont complexes (composés de sous-parties en interaction) ce qui rend leurs élaborations délicates. Parcourons ensemble diverses méthodes pour leurs élaborations.

Wang et Usher décrivent différents schémas de négociation entre agents intelligents pour le contrôle de la production [Usher 00]. Les auteurs précisent que pour la production, la robustesse des systèmes multi-agents face aux perturbations fréquentes est un atout indéniable. De plus, lorsque la configuration des chaînes de production évolue (en taille ou en technologie), ces systèmes bénéficient de facilités d'adaptation. Les auteurs soulignent que l'architecture agent utilisée a une importance sur l'habileté des agents à communiquer. Elle peut être complètement décentralisée (hétéarchie), partiellement décentralisée (quasi-hétéarchie) ou encore centralisée (hiérarchie). Wang et Usher soulignent que l'hétéarchie peut entraîner une forme de myopie au sein de l'activité sociale : les agents ont tendance à s'ignorer. Nous illustrons à la suite les raisons de cette tendance myopie pour des protocoles de coordination tels que *Contract Net*.

Notre approche a pour objectif de réduire cette tendance en donnant aux systèmes hétéarchiques les moyens de se représenter les situations qu'ils abordent (représentation des relations complexes perçues et communiquées à travers le système).

Au contraire, l'utilisation d'une hiérarchie permet de centraliser le contrôle et de répartir explicitement les différentes responsabilités entre les différents niveaux de l'architecture. Cependant, dans cette situation, les auteurs soulignent que la situation bridée résultant de la forte modularité élargit l'étendue des possibilités portées par les agents intelligents.

Les travaux de [Tranvouez 01] proposent une décomposition hiérarchique qui permet le contrôle de la diffusion géographique des perturbations sur plusieurs niveaux (machine, groupe, atelier, usine, etc.). Suite à une perturbation, lorsque les règles de réorganisation locales échouent, la réorganisation est propagée au niveau supérieur de résolution ce qui étend le nombre de possibilités de réarrangement à envisager.

Shen et Norrie proposent MetaMorph II, une approche hiérarchique pour résoudre le problème d'ordonnancement dynamique de la production par différents mécanismes de médiations au sein d'une architecture de médiateurs [Shen 98]. Son objectif est d'intégrer les activités de production d'une entreprise (planification, ordonnancement, etc.) avec celles de ses fournisseurs et clients au sein d'un environnement ouvert et distribué. Le mécanisme de délégation de tâche proposé permet de répartir les décisions hiérarchiquement par l'utilisation du protocole *Contract Net*. La problématique d'ordonnancement est ainsi abordée hiérarchiquement : un agent médiateur *Ressource* interroge un agent médiateur *Machine* qui lui-même évalue localement les possibilités fournies par l'ensemble des machines, outils et travailleurs associés avec lesquels il communique. Les auteurs proposent de développer des mécanismes d'apprentissage des différents médiateurs, afin de leur permettre de réagir rapidement à des situations apprises. Les différents agents du système cherchent à minimiser certains critères (notamment le coût) pour la solution appliquée. Les auteurs envisagent l'approche sous une forme hybride. Une méthode de MonteCarlo

est utilisée dans la perspective de réduire les coûts, les auteurs proposent aussi d'utiliser l'algorithmie génétique ou le recuit simulé.

La coordination de l'activité de différents agents peut être envisagée suivant une stratégie cognitive de délégation. [Devigne 04] propose une solution pour l'élaboration de plans d'activité entre différents agents. Ce travail permet d'organiser l'exécution de plans mettant à profit les compétences associées à différents agents. Les auteurs décrivent des agents cognitifs ayant chacun ses propres capacités de raisonnement. Ceux-ci se regroupent en équipes au sein desquelles ils sont supervisés par un chef. Chaque agent peut réaliser certaines actions sous certaines conditions, l'élaboration de plans et leurs applications permettent le partage de la connaissance entre agents et la coopération en vue d'atteindre des objectifs complexes.

Les décompositions hiérarchiques permettent de mettre en oeuvre une grande modularité. Il nous semble cependant que les différents acteurs intermédiaires occasionnés par les architectures hiérarchiques, et les stratégies de délégation nuisent à la diffusion des caractéristiques complexes du problème abordé lorsque les différentes sous-parties du problème ne sont plus en interaction directe.

Nous nous différencions de ces travaux en recherchant une coordination décentralisée de l'activité de chacun des agents, sans privilégier les capacités de supervision d'un agent particulier. Nous souhaitons profiter des compétences du groupe plus que de celles cumulées par des individus isolés.

Les interactions hétérarchiques entre agents sont supportées par différents protocoles de négociation tel que *Contract Net* et les protocoles de négociation par enchère.

Le protocole *Contract Net* est un modèle de coopération très employé dans les systèmes multi-agents [Smith 83]. Il permet d'élaborer des contrats entre un *contractant* et un *manager*. L'élaboration des contrats intervient en trois phases :

- Un manager émet un appel d'offre pour l'exécution d'une tâche, celui-ci est reçu par différents contractants.
- Les *contractants* émettent une offre.
- Le *manager* fait part de son choix parmi les offres reçues à l'ensemble des contractants.

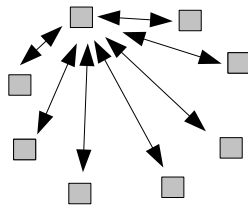
Le protocole présenté permet l'établissement de contrats entre deux entités à partir de la perception d'un manager sur des contractants : il s'agit d'une perception de cardinalité $1 - n$. Pour le problème de partage de ressources critiques en tant que système complexe, les choix devraient être réalisés sur la base d'une perception plus complexe de cardinalité $n - n$ (figure 3.1) : chaque choix réalisé par un agent impacte globalement l'évolution complexe du système. Ainsi, le protocole *contract net* utilisé seul est insuffisant pour traiter le problème de partage de ressources critiques.

Différentes stratégies de négociation utilisent pourtant le principe de ce protocole de négociation à base de marché et compensent la simplicité de sa perception. Ceux-ci étendent la négociation dans le temps afin de permettre la coordination de l'activité agent par itération. Pour cela, différentes variantes du protocole *Contract Net* autorisent l'annulation de contrat avec/sans pénalités, les engagements progressifs et tempérés, etc. Les flexibilités proposées dans les extensions de ce protocole comportent pourtant de nouvelles difficultés, souvenirs de sa contre-nature. Le fait de pouvoir remettre en cause les différents contrats élaborés risque d'entraîner des temps de latence au sein des interactions sociales. Chaque agent étant affecté à plusieurs contraintes doit en effet mener des négociations en parallèle pour chacune de ses contraintes. Lorsque les problèmes sont complexes, le temps de latence se cumule entre les négociations ce qui peut mener à un blocage du système. Dans [Aknine 04] on nous propose de réduire les difficultés propres à ce mode de négociation.

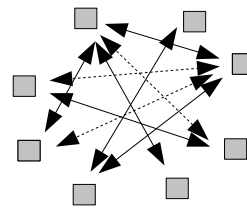
Usher et al. ([Usher 00])proposent d'aborder le problème de contrôle de la production qui est un problème de partage de ressources critiques suivant deux modes :

- SSPR (single-step production reservation) : dans cette situation, les produits sont dirigés à chaque étape de leur fabrication au sein du système de production.

Protocole de communication 1-n



Protocole de communication n-n



■ agent
 ↔ Interaction entre agents

FIG. 3.1 – Cardinalité des relations entre agents

– PR (Production reservation) : dans cette situation, on anticipe le parcours des produits dans le système de production en fournissant un planning d'utilisation des machines au cours du temps.

Le mode PR fournit des résultats de meilleure qualité car il tient compte de l'évolution du système dans une perspective plus globale. Cependant, générer un planning par anticipation est une opération qui requiert plus de calculs. Les auteurs proposent une quasi-hétéarchie pour contrôler la production selon le mode PR. Ils illustrent ce mode de réservation par une approche qui met en jeu des agents Machine communiquant par un protocole de type *Contract Net* avec des agents produits qui parcourent le système de production. L'enjeu de la négociation consiste alors à favoriser le respect des délais pour chacun des produits en concurrence. Les auteurs présentent une extension de ce travail dans [Wang 02] qui permet aux différents produits en concurrence sur les machines de collaborer : plus un produit connaît de machines d'alternatives pour sa fabrication, moins l'appel d'offre qu'il émet sur les machines est jugé prioritaire. Cette extension permet d'améliorer la fluidité de la production en préservant les ressources susceptibles de devenir des goulets d'étranglement.

Chen et al. proposent un travail intéressant dans la perspective de coordination de l'activité agent à partir d'un protocole d'enchères [Chen 03]. Le protocole d'enchères connaît lui aussi les limitations de cardinalité du protocole *Contract Net*.

Pour le problème traité, une tâche est composée de sous-tâches qui doivent chacune se satisfaire en réservant une ressource critique. Chaque ressource peut satisfaire différentes sous-tâches mais ne peut s'engager pour plus d'une d'entre elles (cet engagement est réversible, il dépend de la meilleure offre courante). À chaque tâche est associée une prime d'exécution, l'objectif est alors de répartir l'ensemble des sous-tâches sur l'ensemble des ressources afin de maximiser le total des primes de satisfaction acquises par les tâches programmées. [Weixiong 02] a prouvé que ce problème (*marble problem*) est NP-difficile lorsqu'une tâche peut se composer de plus de deux sous-tâches.

Suivant le modèle proposé, les ressources et les tâches sont représentées par des agents. Les agents *Ressource* communiquent leur coût d'utilisation courant associés à la proposition de l'agent *Tâche* ayant fourni la meilleure offre. En effet, suivant ce modèle d'enchères, c'est la tâche qui fournit la meilleure offre qui obtient la ressource, les agents peuvent revoir la valeur de leurs offres au cours du temps.

Les agents *Tâche* commencent par répartir leur 'pouvoir d'achat' (égal à la valeur de leur prime associée) entre les différentes sous-tâches. Pour chaque sous-tâche, chaque agent choisit alors la ressource ayant le coût d'utilisation le plus faible. Si un agent *Tâche* ne parvient pas à réserver l'ensemble des ressources souhaitées, il pourra modifier la répartition de son pouvoir d'achat entre ses différentes sous-

tâches afin d'accentuer son offre pour les sous-tâches qu'il n'a pu obtenir. Si ceci n'est pas suffisant, il pourra réinitialiser son état courant en libérant l'ensemble des ressources réservées et entamer une nouvelle procédure de réservation. Les réajustements et les réinitialisations sont disponibles en quantité limitée pour éviter le bouclage du processus de résolution. Le processus décrit fournit des solutions de qualité similaire à celle d'un processus de résolution centralisé, mais est plus rapide.

En utilisant tout leur pouvoir d'acquisition à chaque émission d'offres, les agents *Tâche* évitent une négociation itérative coûteuse ce qui entraîne une convergence rapide du processus de résolution. Cependant, les auteurs précisent que l'agressivité de cette forme de négociation amène des situations de sous-optimalité. Par exemple, suivant ce procédé un agent *Tâche* peut émettre une offre élevée sur une ressource et bloquer ainsi l'accès à d'autres agents *Tâches* : l'agent ne perçoit alors pas la gêne qu'il occasionne, et n'envisage pas de sa propre initiative un changement de ressource. On pourrait très bien imaginer des améliorations de ce processus de résolution permettant de parer à cette situation de sous-optimalité. Les agents pourraient par exemple assez facilement se communiquer leur détresse face à leur difficulté à se satisfaire, ainsi dans la situation décrite l'agent bloquant pourrait percevoir la gêne qu'il occasionne et agir en conséquence. Cependant, il est difficile d'étendre ce type d'aménagement sur un périmètre plus large, car le nombre de situations à se représenter croît exponentiellement avec la taille du périmètre étudié. Pour notre étude, nous choisissons de fournir au protocole de négociation la capacité de percevoir les relations complexes des problèmes abordés plutôt que d'essayer de compenser les défauts occasionnés par une perception simplifiée.

Augmenter explicitement la portée de la justification d'un contrat dans un contexte plus global peut être envisagé par un parcours décentralisé de l'espace de recherche. Chaque agent organise alors ses tentatives d'aménagement d'accords. Dans [Mathieu 00] on nous propose un modèle de négociation générique qui met en relation la validation décentralisée des différents contrats d'échange de ressources. Des règles définissent l'ensemble des possibilités à étudier et organisent le parcours des configurations. Le problème du conflit de deux négociations n'est pas vraiment résolu (la négociation avec la priorité la plus forte est réalisée en bloquant les autres négociations). Ce mode d'exploration de l'espace de recherche est entièrement guidé par un ensemble de règles statiques (retour en arrière lorsqu'une tentative échoue, règle qui détermine la règle suivante à appliquer, etc.). Afin de diminuer le coût d'exploration des différentes possibilités, on utilise une connaissance statique (ensemble de règles à appliquer) ce qui réduit considérablement la palette des motifs de configurations évaluées. Afin de réduire le temps de négociation, on néglige certaines parties de l'espace de recherche. Ce travail n'a pas pour objectif de laisser s'exprimer les relations complexes du système.

Nous avons étudié plusieurs façons d'aborder la négociation décentralisée pour le partage de ressources critiques, ces approches peinent à satisfaire pleinement les exigences du problème traité. Un des freins à la résolution provient de la limitation de cardinalité des protocoles supports de coordination. Utiliser un environnement comme support de communication est un autre moyen habile de faire émerger la coordination de l'activité agent. Comment mettre en oeuvre cette habileté pour le problème que nous traitons ?

3.2 Coordination médiée par l'environnement

L'environnement peut être un support de communication. Le biologiste Grassé a défini la notion de stigmergie en étudiant l'activité bâtisseuse des termites : 'la coordination des tâches, la régulation des constructions ne dépendent pas directement des ouvriers, mais des constructions elles-mêmes. L'ouvrier ne dirige pas son travail, il est guidé par lui' [Grassé 59]. Ce concept est utilisé dans bon nombre de travaux prenant comme métaphore le comportement des insectes sociaux : le modèle de fouragement des fourmis [Bonabeau 94], les araignées sociales [Bourjot 99], etc.

Pour le problème traité, l'environnement est un support de négociation pour l'élaboration de différents accords entre agents qui constituent la solution recherchée. Le marquage de l'environnement permet aux agents de s'influencer sans avoir pour cela à respecter des protocoles préétablis qui élaguent leurs libertés d'action. Nous pensons que ces influences permettent de représenter efficacement des relations de cardinalité $n - n$ qui font défaut au célèbre protocole *Contract Net* et à ses dérivés. Il reste cependant à donner les moyens au système de manipuler ces influences pour construire une solution.

L'écoRésolution proposée dans [Ferber 89] s'inscrit dans ce type d'approches. Elle propose une activité agent résultant de la recherche des satisfactions ainsi que de la fuite des insatisfactions par les différents agents. Les travaux de Drogoul [Drogoul 93] présentent par exemple un processus de résolution itératif qui permet la résolution d'un problème d'aménagement spatial : le problème du taquin. Pour ce problème, des palets sont positionnés sur un damier et connaissent des capacités de mouvements très limités. À partir d'une solution initiale aléatoire, le processus de résolution multi-agents permet l'évolution itérative de la solution courante et la convergence vers un agencement prédéfini satisfaisant. Chaque agent cherche à se rapprocher de sa position objectif et entraîne un comportement de fuite des agents gênant son passage. Guedira aborde un problème d'ordonnancement dans ce contexte [Ghédira 92]. On note cependant que le problème d'ordonnancement proposé ne contient pas de contraintes chronologiques entre tâches ce qui en réduit la complexité.

Il n'existe pas à notre connaissance d'approche ayant résolu intégralement le problème de partage de ressources critiques par ÉcoRésolution. Nos travaux profitent d'un mécanisme proche de l'ÉcoRésolution : chaque agent recherche une satisfaction individuelle et influence son environnement dans cet optique. Nous nous écartons cependant de modèles d'agents situés spatialement tel que celui proposé par Drogoul pour le problème du taquin (chaque agent représente un palet situé), nous verrons en effet une nouvelle modélisation de l'environnement pour le problème particulier de partage de ressources critiques.

Cicirello et al. proposent un mécanisme émergent réactif de répartition de produits sur une chaîne de production [Cicirello 01]. Les machines peuvent exercer différentes activités de production qui requièrent un temps de configuration lors des changements d'activité. Les différents produits sont dirigés en cours de fabrication à chacune des étapes de leur fabrication. Chaque machine est pilotée par un agent qui encourage les produits permettant de minimiser les temps de configuration. Ce travail s'inspire des mécanismes de spécialisation présentés dans [Campos 00]. Dans la même mouvance, [Bonabeau 97] propose un mécanisme d'apprentissage permettant de diriger différents camions sur des stands de peinture appropriés afin de minimiser les coûts de fabrication.

Ces mécanismes d'apprentissage décentralisés permettent de répartir différentes opérations de production efficacement, cependant l'approche ne propose pas de réaliser ces affectations par anticipation ce qui complique les problèmes. L'intégration de contraintes supplémentaires (dans ce cas, contraintes de chronologies entre les tâches d'un même produit) n'est pas triviale.

Valckenaers et al. proposent un système multi-agents qui coordonne son activité par marquage de l'environnement [Hadeli 03]. Appliquée à un problème de production, la coordination permet à différents produits de programmer leurs différentes tâches sur les machines de leurs parcours : l'ordonnancement est réalisé par anticipation. Pour cela, l'approche propose un marquage de l'environnement par trois types d'agents :

- Faisabilité : Certains agents parcourent le système afin de répertorier l'ensemble des chemins possibles au sein du système de production. L'information est marquée sur l'environnement des machines et suit un processus d'évaporation. L'évaporation permet le maintien en cohérence des informations recueillies face à l'état changeant du système.
- Exploration : Certains agents explorent différentes alternatives de parcours des produits. À partir de l'étape courante de fabrication d'un produit qui leur est associé, ils évaluent l'intérêt des différents chemins envisagés en considérant les prévisions de production récupérées sur les différentes

machines sondées. Chaque agent sonde un chemin unique, l'information de pertinence du chemin est communiquée au produit et subit elle aussi un processus d'évaporation.

- Intention : Certains agents réservent pour un produit les emplacements de production sur les différents plannings d'un chemin choisi. Pour réaliser des engagements pertinents, ces agents bénéficient des informations recueillies par les agents exploration.

Cette approche permet la génération d'un ordonnancement par anticipation et la perception décentralisée de la capacité de production du système qui évolue. Le marquage des intentions permet à certains agents de programmer leurs parcours au sein du système. Dans cette situation, l'objectif est certes de considérer les objectifs individuels des produits ce que prévoit l'approche proposée. Cependant, il semble difficile d'évaluer avec cette approche l'impact global des différents engagements appliqués par les produits, i.e., les caractéristiques complexes de la solution en construction ne sont pas bien intégrées au sein du processus de construction itératif. En conséquence, les auteurs recommandent de limiter les changements dans l'engagement des différents produits. Suivant ces recommandations, chaque produit doit percevoir un gain significatif (par l'intermédiaire d'un agent d'exploration) avant de prendre une décision de réaffectation. Chaque réaffectation affecte en effet globalement l'ensemble des plannings et peut entraîner des réactions en chaîne visant à revoir l'ensemble de la situation courante.

Dans le contexte de cette approche, l'influence entre les différents engagements s'exprime par l'activité chaotique qu'ils génèrent ensemble lors de leurs mises en application. Afin de mieux contrôler l'impact de ses interactions, la validation d'un engagement devrait être le résultat d'une appréciation et non le déclencheur d'un processus dont l'issue est inconnue. Nous avons été confrontés à cette même difficulté au cours des travaux décrits dans le chapitre 3 qui portent sur le contrôle de l'activité agent.

La thèse présentée par Liu aborde le problème de coordination multi-agents pour l'ordonnancement distribué de production [Liu 96]. L'auteur souligne la différence entre différents types de problèmes de résolution distribués (Decentralized Problem Satisfaction), suivant l'intensité d'interaction entre les sous-solutions recherchées, particulièrement prononcée pour l'ordonnancement et le partage de ressources critiques en général. Liu propose deux schémas de résolution agent pour le problème distribué d'ordonnancement.

Dans la première partie, un planning par anticipation est progressivement construit par des agents qui font évoluer une situation d'affectation courante. Le problème est décomposé par types de contraintes (contraintes chronologique et contraintes 'spatiales' d'affectation des tâches à réaliser sur le planning des différentes ressources). Le modèle associe un type d'agent à chacun des types de contrainte. L'activité de réarrangement des différents types d'agents autorise des situations provisoirement incohérentes, et permet par leurs résolutions à la situation courante d'évoluer. La stratégie de coordination menée a pour objectif de minimiser les perturbations conséquentes à l'activité des agents.

Des stratégies heuristiques de réarrangement mettent à profit ces informations sur la flexibilité des différentes sous-parties de solution (quelle capacité à se déplacer sur une ressource pour une tâche ?). Par ailleurs, les parties des problèmes les plus difficiles à résoudre sont utilisées comme base de résolution pour le reste de la solution et sont plus rarement revues. Celles-ci ne sont pas reconnues par le système lui-même, mais sont transmises par le paramétrage. Cette première partie du travail donne des résultats intéressants lorsque le taux de saturation des ressources reste suffisamment bas (au-dessous de 60%). L'approche se retrouve en effet rapidement confrontée au problème de perception complexe des différentes relations qui contraignent les agents.

Dans la deuxième partie du travail présenté par Liu, le modèle agent est différent : un agent est affecté à chaque ressource et doit décider de la priorité des tâches situées dans son buffer d'attente. Nous voyons que Liu a préféré relaxer le problème bloquant des contraintes chronologiques.

L'ordre d'exécution des différentes tâches en attente permet d'estimer leurs temps d'exécution. Chaque agent perçoit par mémoire partagée (*blackboard*) une estimation des temps de début et de fin des tâches suivantes et précédentes qui lui sont affectées. L'anticipation ainsi réalisée permet de calculer

un critère de retard relatif des produits face à leurs dates de sortie souhaitées. Celui-ci est intégré au processus de calcul de priorité des agents pour leurs tâches associées.

La pertinence du critère utilisé permet d'améliorer la performance des règles de répartition couramment utilisées en production. L'auteur souligne que bien que cela soit théoriquement possible, il n'existe pour l'instant aucune règle de répartition parfaite qui produise un ordonnancement optimal. On remarque que ce critère, même empiriquement efficace n'est qu'une projection des relations complexes du problème abordé, ce qui est un facteur limitant de sa pertinence. L'auteur soulève le problème de perception de l'entièreté de ces relations, cette perception est décrite comme localement très difficile.

3.3 Heuristiques

Pour certains problèmes tels que le problème de la production de composants électroniques (on le surnomme parfois 'complex jobshop'), la complexité des relations des entités en interaction est telle qu'il est très délicat de les exprimer dans leur totalité. Ces systèmes sont instables et très mal compris. On contrôle alors ces systèmes par des heuristiques plus ou moins élaborées.

Les sites de productions de composants électroniques profitent de règles heuristiques qui permettent de définir la priorité d'exécution des produits en attente devant les machines (FIFO, LIFO, règles hybrides, etc.). [Dümmler 00] étudie l'impact de la variation des méthodes d'alimentation en produit à fabriquer (en quantité, en fréquence, etc.) sur les performances du système de production. Il souligne que le système est très sensible à son mode d'alimentation en nouveaux produits à fabriquer, les perturbations peuvent persister plusieurs semaines au sein du système. Il propose d'alterner différentes règles de répartition de produits en fonction de la configuration du système de production, afin d'optimiser la performance du système (minimiser le temps de cycle² et la quantité d'encours³).

Pour le problème de production de composants électroniques, il est certes important de respecter les délais de production des différents produits, mais aussi nécessaire d'utiliser le système de production le plus efficacement possible. Les pertes financières occasionnées par l'arrêt inopportun de machines coûteuses sont très conséquentes. Ainsi, l'approche présentée par [Tsai 03] prévoit de considérer l'urgence de différents produits pour alimenter des machines susceptibles de stopper leur activité par manque de produits. Les machines les plus sensibles à la congestion informent les machines en amont de la production de l'état de leurs files d'attente. Si le niveau de file d'attente de ces machines est trop bas/haut elles encouragent/découragent la progression des produits situés sur les machines en amont de leurs productions. L'intégration de cette information dans les moteurs de règles permet une meilleure performance que celle mesurée pour des règles couramment utilisées dans le domaine de la production de composants électroniques.

La gestion de production par Kanban est une heuristique très efficace qui permet de répartir différents produits sur une chaîne de production afin de minimiser les congestions et de contrôler la quantité d'encours en production. Suivant cette approche, chaque produit qui parcourt un atelier est étiqueté, le nombre d'étiquettes par atelier est limité. Panayiotou et al. proposent une méthode permettant d'optimiser la quantité d'étiquettes à utiliser dans un tel système [Panayiotou 99].

Les propriétés développées par les heuristiques (telle que la robustesse) sont propres à un domaine applicatif (la robustesse s'exprime en effet différemment pour le problème du voyageur de commerce ou bien pour la planification de la construction d'un barrage). Les heuristiques utilisées sont elles-mêmes spécifiques aux domaines applicatifs, ce qui les rend inappropriées dans le cadre d'une approche générique. Par ailleurs, ces heuristiques sont empiriquement satisfaisantes, elles n'opèrent pas par une compréhension des relations complexes des problèmes traités qui pourrait améliorer les performances.

²Le temps de cycle d'un produit correspond à son temps de présence dans le circuit de fabrication

³L'ensemble des produits en cours de fabrication constituent les encours de la production

Elles permettent cependant d'apporter une solution pragmatique et décentralisée à certains problèmes industriels.

3.4 Perception et action : de local à global

Le travail présenté dans [Hogg 91] souligne la tendance au chaos que peuvent manifester les systèmes dans le domaine de l'intelligence artificielle distribuée. En effet, lorsque des agents informatiques en interaction agissent à partir d'une perception incomplète et asynchrone de leur environnement, leurs dynamiques peuvent devenir extrêmement complexes et donner lieu à des oscillations non linéaires et au chaos. Les auteurs étudient la capacité qu'ont des agents à se répartir entre deux ressources de façon optimale, et proposent un mécanisme de récompense permettant d'atteindre une stabilisation rapide sur la répartition recherchée. L'applicabilité d'un tel mécanisme de récompense à des situations plus générales reste un problème ouvert.

[Moyaux 06] propose une approche pour permettre la coopération de différents acteurs intervenant dans la production de biens sur un marché dont la demande évolue. Dans cette situation, les différents acteurs qui constituent une chaîne logistique (*Supply Chain*) sont eux-même soumis au risque d'oscillation chaotique de leurs achats et de leurs ventes qui mène à la sous-optimalité. Dans ce contexte, ceux-ci poursuivent un objectif collectif et un objectif individuel :

- Les acteurs doivent assurer leur rôle d'intermédiaire, et doivent pour cela maintenir un niveau de stock suffisamment haut pour répondre aux commandes de leurs clients respectifs et absorber les variations du marché.
- Les acteurs doivent maintenir le niveau de leurs stocks suffisamment bas afin de minimiser leurs coûts de stockage et de rester dynamique face à l'évolution du marché.

Sans mécanisme de coopération, chaque intermédiaire applique une stratégie individuelle et profite d'une vision locale indirecte inappropriée de l'évolution du marché. Lorsque le marché évolue, des répercussions successives de la variation de la demande s'amplifient sur l'ensemble de la chaîne de distribution de façon incohérente et entraînent des sur-coûts de stockage ainsi qu'une réduction de l'agilité de l'ensemble des acteurs. L'approche proposée permet d'empêcher cet égarement appelé 'effet coup de fouet' en instaurant la coopération en sein du système.

Les auteurs proposent que les acteurs propagent des informations de coordination au moment de passer leurs commandes :

- O : chaque acteur transmet la variation du marché
- Θ : chaque acteur transmet l'aménagement cumulé de cette variation par les différents acteurs. En effet, chaque acteur ne calque pas l'évolution du marché et peut s'il le souhaite modifier la quantité de produits qu'il maintient en stock.

Ainsi, au lieu de communiquer simplement la quantité demandée ($qt = O + \Theta$), chaque acteur transmet des informations supplémentaires relatives à la signification réelle de la commande passée ($[O, \Theta]$). Pour ce problème, les stratégies appliquées permettent alors d'optimiser le fonctionnement du système. Moyaux précise que la solution de collaboration par partage d'informations est la meilleure solution pour les entreprises, aussi bien individuellement (coût réduit pour chaque entreprise) que collectivement (aucune entreprise n'a intérêt à quitter unilatéralement cet état stable pour faire cavalier seul). Cette situation correspond donc à l'équilibre de Nash, concept souvent manipulé dans la théorie des jeux pour lequel aucun agent n'a intérêt à changer de stratégie en vue d'obtenir un gain supplémentaire.

La recherche que nous menons n'envisage pas le partage de ressources critiques dans un cadre compétitif. Ainsi, sur nos modèles, chaque agent ne ressent pas la nécessité de cacher des informations aux agents avec lesquels il communique et de mener une stratégie individuelle égoïste. L'attitude des agents ne doit donc pas être fidélisée par l'existence d'un équilibre de Nash qui les incite à collaborer.

Pour notre travail, nous nous intéressons à la collaboration par la communication. Cet article nous semble intéressant car il souligne les effets néfastes que peuvent engendrer des latences au sein des interactions sociales. Dans la situation décrite, le temps de réception des commandes est de plusieurs semaines à chaque étape de la chaîne et peut mener à un comportement incohérent en accentuant une perturbation initiale correspondant à la variation de la demande du marché. Pour parer à la myopie des acteurs du système, il est nécessaire de communiquer des informations de coordination. Les informations de coordination proposées ne sont qu'une projection de l'ensemble des informations du système (ceci permet le respect de confidentialité entre acteurs). Ainsi, chaque acteur de la chaîne de distribution ne perçoit que l'aménagement cumulé de la variation du marché (Θ), mais ne connaît pas l'aménagement appliqué par chacun des autres acteurs. Le travail de Moyaux montre que lorsque des informations partielles sont bien choisies, elles permettent la coordination de l'ensemble du système. Dans le cadre du partage de ressources critiques, la restitution globale de la situation courante n'est pas envisageable. Le travail présenté nous encourage donc à élaborer un système permettant à chaque agent de percevoir ce dont il a besoin pour agir et réagir efficacement.

Dans [Parunak 03] et [Brueckner 05b], on présente la tendance à la sous-optimalité des comportements agents pour les problèmes de partage de ressources critiques. L'analyse souligne les difficultés qu'ont les agents à agir de façon cohérente à partir de perceptions partielles et différées de l'état du système. Parunak et Brueckner mettent en évidence un phénomène de 'suractivité agent' pour le problème de coloration de graphe et un problème d'emplois du temps : certains agents trop actifs nuisent à l'activité de convergence vers la solution recherchée. Pour le problème de coloration de graphe, l'objectif est d'affecter à chaque noeud une couleur sachant que des noeuds adjacents ne peuvent porter la même couleur. Pour le modèle expérimental proposé, chaque noeud de graphe est associé à un agent. Un agent perçoit (avec un temps de latence) les couleurs associées à chacun de ses voisins directs. L'agent peut s'il le souhaite changer la couleur qui lui est associée afin de minimiser les conflits avec son environnement. Les difficultés répertoriées de l'IA distribuée présentées dans [Hogg 91] sont accentuées pour le problème particulièrement complexe qu'est le partage de ressources critiques.

Dans le domaine des systèmes multi-agents, on décompose l'activité des agents en terme d'exploration et d'exploitation. L'exploration permet de découvrir de nouvelles pistes de résolution, l'exploitation permet de renforcer la solution en cours d'élaboration. Les auteurs proposent d'une part de mettre en oeuvre une régulation de l'activité des agents sur la base de perceptions locales de la pertinence de leur activité. D'autre part, on propose de favoriser l'activité d'exploitation en fin de convergence. De cette façon les agents agissent de façon plus pertinente et permettent des meilleures performances de résolution. Dans [Daouas 94], les auteurs suivent la même intuition et proposent de modérer l'activité par recuit simulé.

Certains font le choix de modérer l'activité de certains agents pour résoudre le problème de suractivité : une trop forte activité ne permet pas aux agents de percevoir l'impact de leurs actions et de collaborer efficacement. Il est intéressant de s'interroger sur les autres réponses qui pourraient être apportées à ce problème de suractivité. Dans [Bonabeau 94], on décrit la capacité des agents à coordonner leur activité par marquage de l'environnement. Pour le problème de fouragement, les comportements des insectes se décomposent eux aussi en terme d'exploitation et d'exploration. Dans ce cadre, l'exploitation n'est pas ressentie comme directement nuisible à toute activité de renforcement. En effet, une fourmi peut partir explorer une partie de l'environnement sans pour autant dégrader le chemin le plus renforcé. Au contraire, suivant de nombreux modèles agent représentant des problèmes de partage de ressources (tel que le modèle proposé pour la coloration de graphe de [Parunak 03]), lorsqu'un agent explore, il impacte directement et de façon irréversible la situation courante. Son activité est en effet perçue et interprétée par son environnement et influence à court ou moyen terme l'activité de renforcement : à la différence du problème de fouragement, la modélisation ne tolère pas la persistance de situations contradictoires.

Il est intéressant de se questionner sur l'impact de cette différence entre le modèle naturel et le modèle informatique pour le partage de ressources critiques. En effet, pour qu'un système puisse percevoir une exploration comme pertinente et la faire évoluer vers de l'exploitation, il est nécessaire de ne pas l'asphyxier trop tôt. Par ailleurs, il est regrettable qu'un renforcement prometteur soit perturbé et affaibli par une exploration inopinée. Dans le cadre du partage de ressources critiques, les motifs d'affectation à élaborer sont complexes et fragiles. Comme le soulignent les auteurs, modérer l'activité des agents en augmentant sa pertinence permet de les faire évoluer plus efficacement. L'évaluation locale de la pertinence de l'activité d'un agent est cependant elle-même très difficile. Aussi, la perception proposée n'est pas générique, et peut ainsi paraître bénéficière plutôt d'un comportement empiriquement satisfaisant que d'une pleine utilisation des capacités du système.

Dans le cadre de notre étude, en partie II, après avoir constaté le problème de suractivité sur un de nos modèles (Chapitre 2), nous avons mis en place des mécanismes permettant de protéger les motifs en cours d'élaboration en favorisant la résolution locale à la diffusion globale des effets destructeurs de l'exploration (Chapitre 3) pour finalement opter pour un modèle innovant propice à une meilleure collaboration entre l'exploration et l'exploitation (Chapitre 4).

3.5 Limitations ressenties , Contributions

Beaucoup de systèmes multi-agents peinent à se représenter les relations complexes qui constituent les problèmes qu'ils abordent. Les approches visant l'élaboration de contrats ne fournissent pas aux différents acteurs une vision réaliste des situations rencontrées. De façon similaire, les approches qui proposent une coordination par l'environnement ne véhiculent pas les relations complexes des problèmes traités. Ainsi se pose un problème de transition : à partir d'une situation courante, à quel moment enclenche-t-on un engagement et sur la base de quoi ? Différentes méthodes permettent de minimiser l'impact grossier d'une activité aveugle, mais celles-ci nous paraissent tout de même limitées. Le travail que nous présentons répond à un besoin identifié : représenter efficacement les caractéristiques des problèmes complexes abordés au sein des processus de résolution, et émet une proposition dans la perspective d'y répondre.

Dans les travaux décrits se pose aussi la problématique du rapport global/local. Les architectures hiérarchiques peuvent réduire la myopie de certains modèles hétérarchiques, mais limitent alors les capacités d'auto-organisation des systèmes. Il est très intéressant d'envisager un système qui pourrait transmettre des stimuli d'activité du niveau global du système vers le niveau local des agents, tout en conservant une expression des caractéristiques complexes des problèmes traités.

Pour finir, l'activité agent peut quelquefois être destructrice en impactant négativement les constructions passées. Nous contribuons dans nos travaux à analyser le problème de 'suractivité' des agents mis en évidence par [Parunak 03] et nous y apportons une solution.

L'identification de différents problèmes qui freinent l'accès des systèmes multi-agents au problème de partage de ressources critiques nous permet de proposer des solutions dans une perspective générique.

Deuxième partie

Démarche d'étude

1

Exploration et convergence en cours de thèse

Notre étude nous a permis de constater que bon nombre de problèmes industriels se rapportent partiellement ou intégralement au partage de ressources critiques. Les systèmes multi-agents semblent être de bons candidats à l'implantation en milieu industriel en apportant des réponses aux caractéristiques des problèmes rencontrés (section 2.2 de la première partie). Cependant, au cours de l'étude de l'art, nous n'avons pas trouvé d'approche totalement satisfaisante pour le problème sous sa forme générique. [Parunak 03] soulève un problème de 'suractivité agent' pour le problème de partage de ressources critiques que nous percevons comme le symptôme d'une difficulté spécifique pour ce type de problème.

Il nous semble que les approches connues n'identifient pas de façon suffisamment claire les limites des modèles qui les supportent. Nous ne sommes pas convaincu que le fait de modérer l'activité des agents artificiellement tel que le propose [Parunak 03] soit une solution au problème de suractivité constaté.

Nous souhaitons donc approfondir les difficultés des systèmes multi-agents par l'observation de certains modèles que nous proposons au cours de cette partie. Quelles sont les limitations des approches connues ? Pourquoi ces limitations ?

Nous choisissons pour notre étude d'aborder le problème de partage de ressources critiques sous sa forme générique et de mettre à jour les difficultés des systèmes multi-agents à le traiter. Partant d'un problème applicatif spécifique (le problème d'ordonnancement), notre démarche d'investigation nous permet d'identifier différents points de blocage des approches conventionnelles limitant l'accès des systèmes multi-agents au problème de partage de ressources critiques. Ces points de blocages ont été positionnés dans un cadre générique ce qui nous a permis d'étudier une nouvelle approche (figure 1.1). Par la suite, nous avons complété et enrichi l'approche en l'appliquant à différents problèmes applicatifs. Ceci nous a permis d'améliorer les performances de résolution et d'établir des premiers éléments de méthode permettant de transposer un problème de partage de ressources dans une représentation propice au bon fonctionnement des comportements agents.

1.1 Point de départ

À la suite d'une lecture dans le domaine des systèmes multi-agents, il est souvent difficile de déterminer si l'approche présentée se rattache à des travaux d'optimisation combinatoire tels qu'ils sont abordés dans le cadre de la recherche opérationnelle par exemple. Ainsi, les chercheurs étrangers aux systèmes multi-agents peinent à connaître la capacité de ces systèmes à aborder cette catégorie de pro-

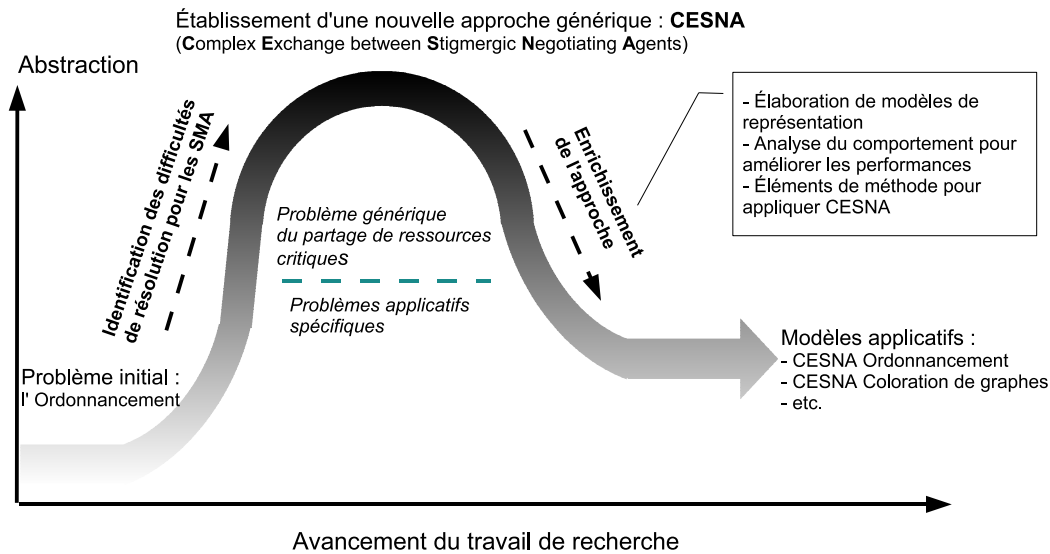


FIG. 1.1 – Démarche d'élaboration de l'approche CESNA

blèmes. Certes, les travaux multi-agents n'ont pas toujours besoin de traiter les problèmes d'optimisation les plus difficiles, mais jusqu'à quel niveau de difficulté sont-ils opérationnels ?

Pour le problème d'ordonnancement, nous avons identifié le besoin d'anticiper la production. En effet, l'anticipation permet une gestion des stocks plus fine, le respect des délais, etc. Ce besoin d'anticiper se manifeste dans les améliorations proposées par la méthode de gestion MRP2 (qui prévoit l'anticipation) par rapport à sa version antérieure. Nous avons été surpris de constater qu'en dépit d'un besoin industriel identifié, beaucoup d'approches multi-agents traitent les problèmes d'ordonnancement sans anticipation, et ne les mettent pas à profit dans une perspective plus vaste. Dans [Cicirello 01], on souligne les qualités que peut apporter l'anticipation à un système informatique dans le cadre de la production. Ce travail met en avant la capacité d'auto-organisation par la spécialisation des individus d'un système, pour en accroître les performances, mais ne traite pas l'anticipation. Des années plus tard, les auteurs abordent toujours ce problème, mais le font toujours sans notion d'anticipation [Cicirello 04]. Abordé sans anticipation, on dirige des produits qui parcourent le système de production à chacune de leurs étapes de fabrication. Avec anticipation, on prévoit le parcours des produits sur plusieurs de leurs étapes de fabrication. Dans la première partie du travail présenté par [Liu 96], on traite le problème par anticipation. L'approche se retrouve confrontée au problème de perception complexe des différentes relations qui contraignent les agents. Lui préfère relaxer le problème bloquant des contraintes chronologiques pour la deuxième partie du travail qu'il présente.

Quelles sont donc les difficultés supplémentaires apportées par cette deuxième façon de traiter le problème ? Les systèmes multi-agents connaissent de bonnes caractéristiques pour être implantés en environnement industriel, pourquoi hésite-t-on à aborder le problème d'anticipation de la production avec les systèmes multi-agents ? Ainsi, ces questionnements constituent le point de départ de notre étude.

1.2 Parcours des différents modèles

Le problème d'ordonnancement est un problème représentatif des difficultés que l'on rencontre dans le cadre de la résolution décentralisée de problèmes de partage de ressources critiques : il est particulièrement

rement complexe (contraintes de consommation de ressources et contraintes chronologiques entre tâches du même produit), il sanctionne tout défaut de représentation décentralisée. Nous nous confrontons donc aux difficultés rencontrées par [Liu 96] sur ce problème. Le problème d'ordonnancement nous sert de référence pour les deux premiers modèles que nous proposons, il nous permet d'identifier les points de blocage que rencontrent les systèmes multi-agents dans le domaine du partage de ressources critiques.

Le modèle hétérarchique décrit dans le chapitre 2 bénéficie d'une ÉcoRésolution sous forme de négociation spatiale. Son étude contribue à identifier, par un premier arbitrage de la compétition entre agents, une tendance au myopisme agent qui entraîne la diffusion incontrôlée de perturbations au sein des systèmes contraints. Nous soulignons que cette tendance ne nous permet pas de profiter pleinement des capacités de résolutions émergentes propres aux systèmes multi-agents. Nous montrons comment enrichir le processus de perception par un meilleur arbitrage de la compétition, ceci permet d'améliorer significativement les performances de résolution du système. Malgré cet enrichissement, nous ne sommes pas convaincus des capacités du modèle à aborder les instances de partage de ressources critiques plus difficiles : nous sommes confrontés aux difficultés de gestion de l'activité agent mises en évidence par [Parunak 03]. Nous identifions par ailleurs une spécialisation émergente des machines de production qui permet de répartir plus adéquatement les différentes tâches à réaliser sur l'ensemble des machines disponibles.

Dans le chapitre 3, nous montrons comment repousser les limites du modèle précédent en accroissant ses capacités à intégrer localement les perturbations. Nous proposons un modèle hybride composé de deux niveaux hétérarchiques d'Écorésolution qui communiquent de façon asynchrone par une hiérarchie. La hiérarchie intervient comme un moyen de contrôle de l'activité agent. Nous verrons que nous maintenons par ailleurs des conditions favorables à l'intégration locale des perturbations.

Ceci entraîne une meilleure stabilité du système et une capacité à aborder des problèmes plus difficiles. Nous soulignons cependant que le mécanisme de contrôle permettant de réduire la diffusion des perturbations est difficile à paramétrer et fige artificiellement le système en réduisant ses capacités d'auto-organisation. Le mécanisme de contrôle intervient par une nouvelle architecture agent qui permet de représenter facilement les objectifs du système et sa robustesse, ce qui facilite les interactions entre le système et ses utilisateurs.

À la suite de l'étude de ces deux approches, conformément à ce que l'on constate pour des approches SMA de même type, nous avons identifié la tendance au myopisme des agents pour notre premier modèle hétérogène ainsi que le frein à l'auto-organisation provoqué par la hiérarchisation de notre deuxième modèle. Ces deux constats nous poussent à approfondir tout en capitalisant l'expérience acquise.

Plutôt que de rechercher des moyens de compensation des insuffisances occasionnées par les deux modèles proposés, comme cela peut être fait avec des approches du type *Contract Net* que nous décrivons comme mal adaptées dans la partie I, nous poursuivons notre quête sans altérer nos exigences : comment représenter et véhiculer les relations complexes pour donner aux agents les moyens de coordonner leur activité vers un objectif commun ? Nous nous éloignons des approches conventionnelles et proposons un modèle innovant.

Nous présentons dans le chapitre 4 une analyse des performances des deux modèles de résolution précédents et identifions certaines limitations ce qui nous permet de converger vers un modèle agent qui envisage des négociations complexes sous un nouvel axe. Ce modèle nommé CESNA (*Complex Exchange between Stigmergic Negotiating Agents*) est l'objet d'une étude approfondie dans la partie qui suit.

2

Approche par combats

Pour cette première approche du problème, nous sommes restés cohérents avec les modèles d'agents réactifs les plus courants : des agents situés agissent directement sur la situation courante et permettent de converger progressivement vers une solution objectif. Pour cette catégorie de systèmes, la communication stigmergique s'effectue par l'intermédiaire des objets composant la solution recherchée : les objets qui constituent la solution et les objets supports de la communication stigmergique sont confondus. Nous emploierons la formulation de *convergence itérative par représentation située spatialement* pour désigner ce type de modèle agent.

Le problème du tri collectif abordé par les insectes sociaux est un des modèles les plus connus qui bénéficie de ce type de convergence. Pour ce modèle, les agents situés déplacent sur l'environnement physique différents objets à trier. On constate progressivement l'émergence de groupes d'objets de même type sur l'environnement, résultat de la stratégie collective adoptée par les différents agents.

Dans cette partie, nous abordons le problème d'ordonnancement par anticipation présenté en section 2.1. Cette étude nous permet d'étudier le comportement de l'activité sociale des agents face à la complexité des problèmes de partage de ressources critiques. Pour ce problème, nous souhaitons placer différentes tâches de production sur les différents planning limités des machines disponibles. Chaque tâche planifiée sur une machine réduit l'ensemble des possibilités complexes d'affectation pour le reste des tâches du système, les choix de placements sont en interaction.

Pour notre problème d'ordonnancement, un mécanisme de négociation par combats permet de faire évoluer une situation initiale vers une situation de qualité meilleure par itération. Le mode de calcul de l'issue des combats définit le comportement de l'écosystème en départageant les différents agents en compétition pour les mêmes ressources.

Nous avons étudié le comportement de l'activité sociale suivant deux modes d'arbitrage de la compétition entre les agents :

- En section 2.2, l'arbitrage de la compétition repose entièrement sur la qualité des motifs formés sur l'environnement spatial.
- En section 2.3, l'arbitrage prend en compte des informations supplémentaires qui parcourent le système, afin de compléter la coordination de l'activité globale du système. Ces informations parcourent un environnement dessiné par les contraintes chronologiques entre agents *Tâche* .

Suivant la négociation par combats proposée, chaque agent est égoïste et profite de situations avantageuses pour se satisfaire en remportant des combats. Voyons ensemble si la représentation proposée permet aux agents de considérer l'impact de leur activité sur le système. Les expérimentations menées sur ce modèle vont nous guider dans l'analyse de l'adéquation de cette forme de négociation face au problème complexe de partage de ressources critiques. Le modèle que nous proposons dans ce chapitre a été présenté dans [Armetta 04].

2.1 Contexte

2.1.1 Contexte et objectifs de notre étude de cas

Pour notre étude, nous abordons la production de composants électroniques qui fait partie des problèmes d'ordonnancement les plus complexes (on le surnomme parfois 'complex jobshop').

Il est caractérisé ainsi par :

- une longue séquence d'opérations (plus de 100) pour chaque produit
- de nombreux produits en production
- des flux réentrants : les produits parcourent plusieurs fois les mêmes machines pour leur fabrication
- des catégories de produits fabriqués sur la chaîne de production qui évoluent constamment
- des produits en fabrication et des produits en recherche et développement qui parcourent les mêmes machines
- des ressources très coûteuses dont il faut maximiser l'utilisation
- des pannes de machines fréquentes et imprévisibles
- des exigences commerciales évolutives
- etc.

Des objectifs (souvent contradictoires) complexifient encore la gestion de ce type de chaînes de production :

- maximiser l'utilisation des ressources, et maintenir cette maximisation durablement
- linéariser les sorties du système de production
- minimiser le temps de cycle⁴ moyen des produits en maintenant un niveau d'encours suffisamment bas.

Il n'existe pas à notre connaissance d'approches prévoyant la gestion de tels systèmes par anticipation. Les sites de productions utilisent des règles heuristiques qui permettent de définir l'ordre d'exécution des produits en attente devant les machines (FIFO, LIFO, règles hybrides, etc.). Ces systèmes sont instables et très mal compris. [Dümmler 00] a étudié l'impact de la variation des méthodes d'alimentation en produit à fabriquer (en quantité, en fréquence, etc.) sur les performances du système de production. Il souligne que le système est très sensible à son mode d'alimentation en nouveaux produits à fabriquer, les perturbations peuvent persister plusieurs semaines au sein du système. Il propose d'alterner différentes règles de répartition de produits en fonction de la configuration du système de production, afin d'optimiser la performance du système (minimiser le temps de cycle et la quantité d'encours).

Pour les problèmes que nous simulons, nous générons des ordonnancements par anticipation pour lesquels les produits parcourent plusieurs fois les mêmes machines (flux réentrants). Nous recherchons pour cette première simulation la maximisation de l'utilisation des différentes machines, nous ne considérons pas l'ensemble des objectifs industriels. Nous avons pour objectif d'identifier les différentes difficultés que rencontrent les systèmes multi-agents dans le contexte du partage de ressources critiques. Pour cette étude, nous dégagons deux problèmes dans un cadre plus générique :

- Affectation micro : quel processus choisir pour affecter les plannings des machines efficacement avec les tâches à réaliser ?
- Affectation macro : quel processus choisir pour gérer efficacement les flux de production ?

En effet, le bon fonctionnement d'un tel système n'est possible que si les tâches sont placées sur le planning des différentes machines tout en considérant plus globalement la situation de progression des différents flux de production.

⁴Le temps de cycle d'un produit correspond à son temps de présence dans le circuit de fabrication

2.1.2 Cadre de mise en oeuvre de l'approche

De part ses propriétés intrinsèques (plusieurs formes de perturbations, des interactions multiples, etc), une chaîne de production présente les caractéristiques d'un système complexe, évoluant dans un environnement incertain. Le paradigme multi-agents possède des propriétés intéressantes pour représenter un tel système.

L'approche que nous proposons s'inscrit dans le cadre de développement des systèmes auto-organisés proposé par [Hassas 03]. Ce cadre stipule que l'auto-organisation du système est obtenu par un couplage entre l'organisation sociale et l'organisation située des agents. Ce couplage s'appuie sur les éléments suivants :

- Une représentation de l'environnement du système : les différents plannings des ressources constituent ensemble la globalité de l'environnement spatial sur lequel des agents *Tâche* peuvent se placer. L'environnement correspond ainsi à une zone d'anticipation de l'ordonnancement généré. On qualifie les positions de l'environnement suivant leur proximité avec la frontière d'exécution qui correspond à une zone de l'environnement à partir de laquelle les tâches associées aux agents sont exécutées sur les machines de production. Les différents agents se positionnent ainsi en amont/aval les uns par rapport aux autres. Nous utilisons aussi, dans le modèle enrichi, un environnement plus abstrait dessiné par l'ensemble des relations chronologiques entre agents pour compléter la représentation du problème traité.

- Une intelligence comportementale : les agents *Tâche* participent à l'activité sociale du système en se plaçant le plus tôt possible sur l'environnement (le plus en aval) tout en respectant les contraintes qui leur sont associées : les agents sont associés à des produits dont ils héritent des contraintes de chronologie. Chaque agent *Tâche* recherche donc une position cohérente avec le placement d'autres agents *Tâche* qui leur sont associés, et est confronté à l'état courant d'occupation de l'environnement.

Nous ne relaxons aucune contrainte au cours de l'évolution itérative de la situation : l'environnement spatial d'affectation est un espace de placement critique non flexible, chaque agent *Tâche* placé peut maintenir sa position si et seulement si les tâches précédentes de son produit associé sont placées en son aval (nous verrons comment mettre en place une construction plus flexible dans le chapitre suivant).

- La stigmergie : utilisée comme mode de coordination distribué et asynchrone par modification de l'environnement spatial. Pour notre problématique, les agents *Tâche* sont en compétition et combattent pour se placer sur l'environnement.

Une fois placés, ces agents forment différents motifs locaux suivant différentes topologies. Nous proposons de définir des règles permettant de calculer l'issue des combats menés par les agents en compétition. Celles-ci tireront profit des caractéristiques spatiales des motifs de l'environnement.

- Équilibre exploration/exploitation : cet équilibre est entretenu par un comportement opportuniste des agents lors de la sélection des ressources sur lesquelles se placer et par un renforcement de leur placement sur les ressources, quand leurs positions sont estimées utiles par le système.

Le modèle que nous proposons est conforme à l'ÉcoRésolution proposée par [Ferber 89] pour laquelle chaque agent essaie d'améliorer sa situation courante en provoquant des comportements de fuite des agents avec lesquels il est en concurrence.

2.2 Première expression des relations entre agents

Suivant ce modèle, le calcul de l'issue des combats dépend exclusivement des motifs de l'environnement spatial perçus par les agents. Après avoir décrit l'arbitrage de la compétition que nous choisissons, nous observons le comportement du système et soulignons quelques limites du modèle.

2.2.1 Arbitrage de la compétition

Suivant notre modèle, chaque agent *Tâche* a pour objectif de se placer sur l'environnement. Lorsqu'un agent est placé, il recherche une amélioration de sa position en se déplaçant vers l'aval tout en restant positionné en amont de l'agent qui le précède pour la fabrication de son produit associé.

Pour chaque tentative de placement, un agent sélectionne aléatoirement une machine parmi celles qui peuvent accueillir la tâche à laquelle il est associé. Les contraintes chronologiques de l'agent réduisent l'ensemble des positions qui lui sont accessibles. Nous voyons sur la figure 2.1 que l'environnement local de la ressource est constitué d'agents *Tâche* placés et d'espaces vides. L'agent choisit parmi différentes positions (groupes d'éléments) sur la machine (choix aléatoire).

Arbitrage de la négociation spatiale :

- { victoire de l'attaquant, si le premier élément du groupe attaqué est un espace vide
- { défaite de l'attaquant, sinon

L'arbitrage que nous choisissons a pour objectif de permettre, par effet de bord, le déplacement des espaces vides vers l'amont de l'environnement. En effet, comme illustré sur la figure 2.1, à chaque fois qu'un agent s'insère dans l'environnement, un espace vide est repoussé vers une zone de planification plus éloignée, ce qui diminue la distance avec la solution recherchée en maximisant l'utilisation des machines.

Pour cette première approche, la situation évolue de façon cohérente : elle ne permet pas aux agents de maintenir leur position en relâchant provisoirement certaines contraintes chronologiques. Ainsi, lorsqu'un agent perd sa position sur l'environnement, les agents *Tâche* situés en amont qui lui sont associés sont forcés de quitter l'environnement, ils laissent sur l'environnement spatial un espace non utilisé.

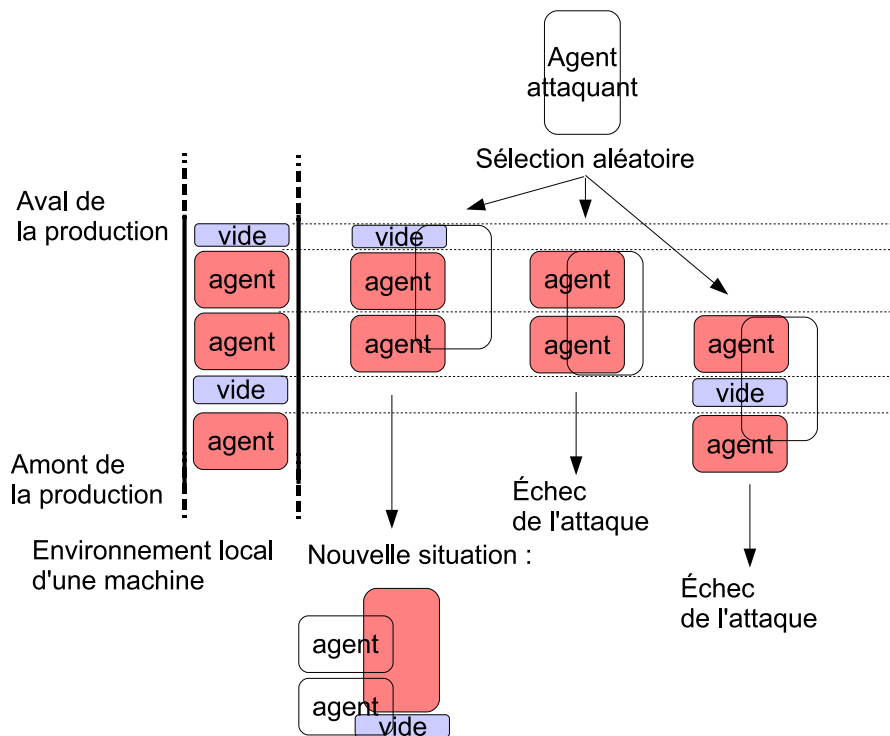


FIG. 2.1 – Arbitrage spatial des combats entre agents *Tâche*

2.2.2 Expérimentations, constatations et résultats

Dans cette partie, nous décrivons le comportement du système que nous avons observé. Ces premières constatations nous apporte une meilleure compréhension des situations complexes rencontrées par les agents dans le cadre du partage de ressources critiques. Nous voyons comment ceux-ci réagissent et participent à un mécanisme émergent qui équilibre la répartition des flux de production.

La quantité de produits disponibles dans le système est en relation avec la difficulté à générer et maintenir un ordonnancement qui maximise l'utilisation des machines de production. Plus le niveau d'encours est bas, plus les contraintes qui matérialisent la complexité du système 'tendent' l'ordonnancement à générer.

Pour ce premier contact avec le problème, nous autorisons une grande quantité d'encours (problématique simplifiée), ce qui est suffisant pour observer le comportement du système en situation complexe.

Observations

- Éjection des espaces vides

Au cours de la compétition entre les agents qui négocient le partage de l'environnement spatial, on observe des modifications de placement sur l'environnement (un agent se place, un ou plusieurs agents sont éjectés, etc.). On observe le déplacement des espaces vides déclencheurs de l'activité agent. Chaque espace vide se déplace de la position de début de groupe attaqué à la position de successeur de l'agent qui s'installe. De proche en proche, l'activité agent repousse les espaces de programmation non utilisés.

Le problème d'ordonnancement autoriserait-il une convergence itérative, telle que nous l'observons pour d'autres problématiques ?

- Étapes de convergence vers la solution

L'éjection de certains agents a des répercussions sur l'activité du système vers des zones de placement en amont ⁵. Nous constatons visuellement que le système converge itérativement depuis l'aval vers l'amont. Il parcourt une partie de l'espace de recherche local à chaque étape de sa progression que l'on peut qualifier de strate. La stabilisation d'une strate est validée par un mécanisme de contrôle résultat de l'ÉcoRésolution : lorsqu'une strate est saturée, l'activité cesse sur la strate.

- Comportement face aux perturbations

Nous avons confronté le système à des perturbations intervenant en temps réel. Nous enclenchons une panne machine sur le système multi-agents en substituant la part de l'environnement spatial de la machine. L'ensemble des agents résidant sur l'environnement local de la machine perdent leur position, d'autres environnements de machines sont affectés par la perturbation du fait des contraintes de chronologie portées par les agents.

L'ensemble des espaces vides qui apparaît sur l'environnement spatial disponible amorce une activité agent : le système modifie sa configuration et se stabilise sur une solution adaptée à la quantité de ressources disponibles suivant le processus précédemment décrit.

Lorsque la machine est réactivée, l'environnement local qu'elle représente est de nouveau disponible. Dans cette situation, ce sont les agents qui scrutent des positions meilleures (déplacement en aval) qui déclenchent l'activité agent. Lorsqu'un agent colonise la machine réactivée, il laisse un espace vide derrière lui et propage ainsi de l'activité. Ce processus déstabilise les différentes strates du système et permet la reconstruction d'une solution.

⁵Rappel : Un agent ne peut maintenir sa position sur l'environnement que s'il connaît la position d'exécution des agent *Tâche* qui le précèdent dans la nomenclature de son produit associé. Ainsi, lorsqu'un agent est éjecté, tous ces successeurs le sont aussi.

Équilibrage des flux

L'ensemble des placements réalisés sur l'environnement (à un niveau micro) constituent des flux de catégories de produits qui parcourent le système (à un niveau macro). Pour la production de composants électroniques, la complexité inhérente au système de production mène rapidement à un déséquilibre des flux de production, ce qui occasionne des pertes de productivité conséquentes.

Un goulet d'étranglement se forme sur une machine lorsque sa capacité de production est inférieure à la charge qu'on lui impose, situation qui entraîne l'apparition d'espaces de programmation non-utilisés en aval dans la chaîne de production. Un système efficace doit alors équilibrer la charge de production sur les différentes ressources afin d'éviter qu'il ne se forme des goulets d'étranglement qui nuisent à la productivité du système.

Nous voyons sur la figure 2.2 que le produit P1 peut réaliser ses différentes étapes de fabrication sur M0 ou sur M5. Par contre, P0 ne peut réaliser certaines de ses étapes de fabrication que sur M0. Pour cette expérimentation, voyons comment le système peut équilibrer les flux de production en dirigeant adéquatement le flux de produits P1 sur M0 ou sur M5 :

- Qualité de la solution obtenue

Sur la figure 2.2, nous voyons que lorsque le système ne comporte que des produits de type P1, les agents *Tâche* se placent équitablement sur M0 et M5 (30 produits P1 disponibles pour la simulation). En effet, à chaque fois qu'un agent cherche à se placer, il sélectionne aléatoirement une machine parmi les machines qui peuvent le satisfaire, ceci entraîne que nous obtenons en moyenne la moitié des tâches placées sur M0 et l'autre moitié sur M5. Nous avons constaté que plus nous chargeons le système de produits P0, plus les agents *Tâche* de produits P1 se fixent sur la machine M0.

- Émergence de spécialisation

On constate la spécialisation de la machine M0 pour les produits P0, lorsque la quantité de produits P0 augmente. Celle-ci permet le déchargement de la machine M0, l'équilibrage des flux de produits P0 et P1 sur M0 et M5. La bonne gestion de ces flux permet une productivité accrue du système de production. Ainsi, les produits de type P0 sont fabriqués sur la machine M0, les produits P1 sont réalisés plutôt sur M5.

Le phénomène de spécialisation constaté n'a pas été explicitement implémenté dans le système multi-agents : il est le fruit d'interaction entre agents aux comportements simples. Le système dans sa globalité perçoit la congestion du système et s'y adapte, sans que les agents ne perçoivent individuellement l'encombrement.

Il est toutefois possible d'interpréter le comportement qui mène à cette spécialisation : un mécanisme de contrôle valide les situations propices à la fluidité de la production. D'une part, la machine la plus convoitée qui est la plus encombrée est aussi la moins accueillante : la forte activité de compétition qui s'y déroule influence les agents qui ont le choix, à adopter un autre emplacement. D'autre part, les agents qui parviennent à s'y installer bénéficient d'opportunités leur permettant d'améliorer leurs situations en se positionnant plus en aval sur des ressources moins congestionnées.

- Opportunisme du contrôle de l'activité des agents

L'activité du système est conséquente à la présence d'espaces vides qui fragilisent l'environnement spatial ainsi qu'aux attaques des agents en quête de placement, i.e., **l'activité agent est amorcée par un environnement spatial instable, mais n'est pas conséquente à une mauvaise utilisation des ressources**.

Nous avons pu constater l'impact de ce défaut de perception sur le processus de spécialisation que nous venons de décrire : il arrive que l'environnement se fige sur une configuration non optimale à la gestion des flux de production.

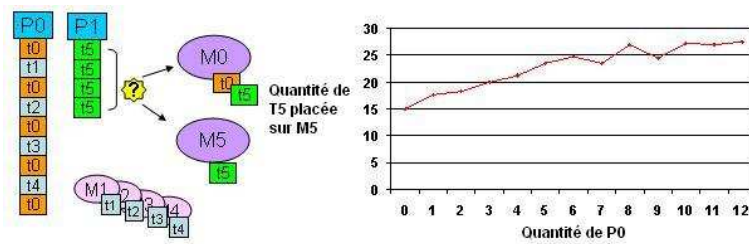


FIG. 2.2 – équilibre des flux de production

2.2.3 Limites du modèle

Nous constatons avec ce premier modèle que le mécanisme d'exploration de l'espace de recherche et le mécanisme de contrôle valide une répartition correcte des flux de produits dans le système en maximisant l'utilisation des ressources. Le système met ainsi à profit un ensemble d'informations (connaissance des zones de programmation inoccupées et perception émergente des ressources surchargées) pour se stabiliser vers une solution cohérente. Malgré ces bonnes caractéristiques, nous relevons quelques limitations que nous assimilons à la simplicité des informations mises à la disposition du système.

Le mécanisme permettant de repousser les espaces vides sur l'extérieur de l'environnement est local et manque de cohérence sur l'ensemble de l'environnement. Les comportements sociaux ne considèrent pas les relations entre les différentes strates. On constate en effet que la stabilisation s'effectue progressivement de l'aval vers l'amont : les agents de strates différentes ne collaborent pas entre eux.

Un phénomène de suractivité est propagé par le choix que nous faisons de respecter des contraintes de chronologie au cours de la construction de la solution. Nous consacrerons le chapitre suivant à l'étude d'un modèle plus flexible. Pour ce premier travail, nous souhaitons approfondir les relations complexes à communiquer aux agents.

Suivant le processus décrit, les relations de chronologie entre agents s'expriment exclusivement par l'activité générée pour leur respect. Nous avons vu que la mauvaise communication entre les différentes strates de l'environnement spatial gêne le contrôle de l'activité du système. Il est intéressant d'envisager de considérer les relations de chronologie au sein du processus d'arbitrage de la compétition, ceci ouvre des perspectives d'amélioration de la pertinence des choix du système à base d'agents.

2.3 Enrichissement de l'expression des relations entre agents

Dans cette section, nous proposons l'intégration d'informations sur les contraintes de chronologie dans le processus de calcul de l'issue des combats.

Ce modèle intervient comme une extension du premier modèle, pour lequel la communication par stigmergie s'exerce sur un environnement spatial, par l'ajout d'un environnement plus abstrait sur lequel transite des phéromones artificielles qui reportent les besoins du système d'un point de vue plus global.

Voyons donc plus en détail ce nouveau mode d'arbitrage, ses performances et ses limites.

2.3.1 Arbitrage de la compétition

L'arbitrage tel que nous l'avons proposé jusque là repose exclusivement sur l'état des motifs de l'environnement spatial. Un agent *Tâche* est cependant en relation de chronologie avec le placement des agents *Tâche* qui sont associés au même produit que lui. Nous voyons que le premier arbitrage permet de gérer les contraintes de placement spatial, mais ne gère pas les contraintes de chronologie.

L'expression des caractéristiques complexes des problèmes nous semble insuffisante car incomplète. Afin d'affiner le mécanisme de placement lors des combats entre agents, nous intégrons des informations supplémentaires qui sont considérées dans le calcul de l'issue des combats. Avec le mécanisme proposé, les agents en quête de placement (stockés dans divers plannings fictifs 'd'attente') pourront percevoir leur importance pour le système, et donc la nécessité qu'ils ont à se placer. L'activité de négociation résultante est améliorée, les informations complémentaires influencent l'issue des combats : plus un agent est perçu comme important pour le système, plus il a de chance de maintenir/obtenir une position, malgré une issue défavorable du processus de combat spatial. Le paramétrage de la pondération de l'influence de ces informations complémentaires a été réalisé empiriquement.

Un nouvel environnement

Les informations complémentaires s'expriment dans un environnement abstrait constitué des relations de chronologie entre les agents. Ce nouvel environnement est en relation avec l'environnement spatial, il permet à des informations de circuler suivant les trajets que peuvent réaliser les produits au cours de leur fabrication.

On retrouve cette même stratégie de représentation dans les travaux de [Hadeli 04] pour un problème d'ordonnancement. Pour ce travail, l'environnement permet de représenter le parcours des produits dans la chaîne de production, et de mettre les différentes tâches de même produit en relation.

Les informations sont véhiculées sous la forme de phéromones artificielles. Elles subissent un processus d'évaporation afin de rester significatives face à un système en perpétuelle évolution.

L'environnement que nous utilisons est désormais en partie spatial (configurations d'occupation) et en partie abstrait (configurations chronologiques). L'environnement ainsi recomposé permet d'exprimer l'ensemble des contraintes du problème traité, et doit permettre aux agents de mieux percevoir les relations complexes du problème. Voyons ensemble la stratégie de communication que nous proposons.

Favoriser le placement des produits les plus attendus

Une congestion apparaît au sein du système de production lorsque se forme sur une ressource une file d'attente importante, la ressource n'a alors pas un débit de production suffisant pour alimenter les ressources situées à son aval. Il est alors nécessaire de sélectionner avec pertinence quels produits fabriquer en premier sur la machine saturée, ceci en vue de minimiser l'impact de la congestion.

Anticiper les différentes opérations de production permet de percevoir à l'avance les ressources en manque de produits. Ainsi, une première stratégie de résolution serait de permettre aux ressources en manque de produits d'informer ces produits de leur importance pour le système, et donc de faciliter leur placement.

Sur une ressource, on fabrique certains produits à différentes étapes de leur fabrication. À partir d'un espace à combler sur une ressource, on génère une phéromone artificielle qui parcourt la chaîne de production vers l'amont afin d'attirer les produits attendus. Cette remontée s'effectue par simulation du trajet des produits pouvant alimenter la ressource, grâce à l'extension de l'environnement que nous proposons.

Dans ce contexte, une stratégie possible serait de faire remonter des informations concernant les espaces de planification à occuper. Suivant cette stratégie, c'est par la combinaison produit-étape que l'on définit le parcours à rebours effectué par la phéromone artificielle dans le système. Celui-ci consiste à remonter l'environnement temporel sur les machines appropriées pour rechercher les agents *Tâche* qui peuvent amener le produit jusqu'à la machine souhaitée⁶.

⁶Les marges de placement entre deux tâches d'un même produit ont été configurées empiriquement sur ce modèle

L'information remonte ainsi l'environnement en amont jusqu'à trouver un agent *Tâche* approprié : un agent *Tâche* est approprié s'il n'est pas placé dans l'environnement, mais souhaite obtenir un emplacement sur la période temporelle proposée par la phéromone artificielle. Il doit par ailleurs correspondre à l'étape de fabrication et au produit propagé par celle-ci.

L'information véhiculée pourrait parvenir à l'étape 0 de fabrication du produit. Dans cette situation, la phéromone artificielle exprime le besoin d'un nouveau produit à fabriquer. Une étude intéressante serait de considérer l'ensemble des informations de ce type pour définir la charge souhaitable du système de production.

Il est aussi envisageable d'ajouter un critère de qualité à l'information ainsi transmise, on pourra par exemple la fixer à partir de la taille du vide qui en est l'origine. En effet, plus un vide est de taille importante, plus il semble nécessaire d'attirer des produits sur l'emplacement du vide afin de dissoudre les congestions se formant dans le système.

Nous choisissons d'implémenter une stratégie plus simple qui, nous le voyons dans la partie suivante, a les mêmes effets que ceux attendus pour la stratégie que nous venons de décrire. Celle-ci bénéficie des effets de bord relatifs à l'activité des agents : au lieu de faire remonter de l'information permettant aux produits les plus attendus de se placer en priorité, on renforce la position des produits qui sont parvenus à se placer sur plusieurs opérations en amont proportionnellement au nombre d'étapes de fabrication positionnées. Ceci a pour objectif de permettre le placement des produits qui ne renforcent pas les congestions du système, en favorisant leur maintien sur l'environnement spatial. Afin de modérer l'activité des agents, des agents *Tâches* associés aux mêmes produits et aux mêmes étapes ne bénéficient pas de cette nouvelle source d'information pour se départager lorsqu'ils sont en compétition (ils utilisent dans ce cas le premier mécanisme de négociation spatiale).

2.3.2 Expérimentations et résultats

Le système considère désormais des informations complémentaires sur l'utilité des agents lors de leur mise en concurrence sur l'environnement. Cet enrichissement du modèle a pour objectif de diminuer la myopie des décisions ainsi que d'éviter les situations de blocages opportunistes pour le système.

Les simulations que nous reportons permettent de comparer les performances des systèmes avec/sans enrichissement de l'expression des relations entre agents que nous venons de décrire :

- Répartition des flux entre deux produits concurrents sur une ressource :

Pour cette première simulation, nous avons souhaité étudier la répartition de deux flux de produits face à la capacité d'accueil du système. La configuration physique accueille deux flux de produits (figure 2.3 : Flux P1 et P2) suivant des capacités de production différentes.

À l'écoute de la capacité d'accueil, le système doit correctement affecter les flux de produits sur les ressources et définir ainsi leur vitesse de progression. Sur la figure 2.3, nous voyons que des ateliers ont des capacités de production variables relatives au nombre de machines qu'ils comportent. L'atelier 3 a une capacité de 1, alors que les autres ateliers du système ont une capacité de 3. Le flux de produits correspondant à P2 connaît donc une capacité de production de 3 pour son étape 1 et seulement de 1 pour son étape 2. Le flux associé au produit 1 (Flux P1) peut se déplacer plus vite dans le système que le flux de produit 2. Sur sites industriels, ce type de goulet d'étranglement est difficile à reconnaître car les régimes de productions sont en constante évolution (pannes machines, variation des priorités, etc.). Ces évolutions mènent à des congestions qui persistent plusieurs semaines au sein du système de production. Il convient donc de réorganiser les flux de production pour maintenir de bonnes performances pour le système.

Nous constatons que le planning réalisé en utilisant le modèle enrichi est bien meilleur que celui réalisé avec le modèle simplifié (+ 11,6 % de tâches placées).

En considérant l'utilité pour le système des placements réalisés par les produits, le système permet la convergence vers une solution de qualité, et une répartition correcte des flux de produits.

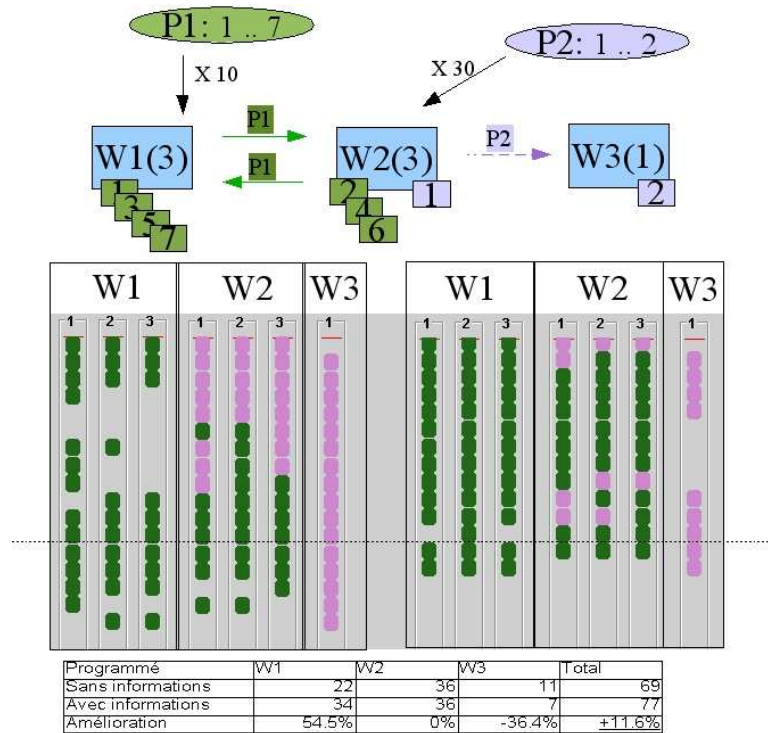


FIG. 2.3 – Répartition [sans/avec] informations : éloignement de niveau 1

– Influence du nombre d'étapes de fabrication des produits :

Les approches implantées sur les sites de production de composants électroniques bénéficient de moteurs de règles qui permettent la répartition des produits suivant les configurations de congestions perçues. Le calcul de répartition réalisé sur une machine tient compte de critères compliqués propres à la configuration de l'usine. Ces règles sont élaborées empiriquement, et requièrent une longue période de configuration (plus d'une année). Les pertes de production occasionnées par ces temps de configuration influence considérablement la compétitivité des usines (le respect des délais et des volumes définit les parts de marché). Les interactions complexes entre les différentes règles appliquées sur la chaîne de production ne sont pas connues, ceci rend compliqué et long leurs évolutions. De plus, les types de produits fabriqués sur un site de production évoluent constamment, ce qui modifie en permanence le problème de répartition abordé par l'approche empirique. Ce type d'approche peut répondre au cas précédent de la figure 2.3 qui est facilement identifiable, qu'en est-il lorsque les motifs de répartition se complexifient et lorsque les congestions sont plus difficiles à reconnaître ?

Notre modèle ne prévoit pas de frontière explicite à la quantité d'informations assimilée. Plutôt que de définir des règles de répartitions statiques appliquées à des situations identifiées, nous souhaitons donner au système la capacité de perception et d'interprétation de son environnement pour qu'il puisse agir en conséquence : donner des capacités d'auto-organisation au système.

Sur la figure 2.4, le produit P2 est ralenti à sa troisième étape de fabrication (et non à sa deuxième comme précédemment). Sur l'exemple étudié, nous allons voir que le périmètre des informations intégrées pour la résolution s'étend naturellement à plusieurs étapes de fabrication.

Sur la partie gauche de la figure 2.4⁷, nous pouvons voir la planification réalisée par le système bénéficiant du premier mécanisme de combat présenté. Nous avons pu constater que certains produits P1 ne parviennent pas à se planifier au niveau de leur étape 2 alors qu'ils peuvent être accueillis sur l'atelier 1 pour leur étape 3. A l'inverse, certains produits P2 sont planifiés sur l'atelier 2 pour leur étape 1 alors qu'ils sont bloqués à leur étape 2 car le système ne peut les accueillir. Ces produits P2 sont moins urgents mais l'opportunité de placement reconnue sur le premier modèle les autorise à maintenir leur position. Nous voyons sur la partie droite de la figure qu'une meilleure perception des relations complexes du problème permet un équilibre approprié des flux de produits. En effet, en utilisant ces informations, tous les produits P1 parviennent à se planifier, car leur utilité est désormais identifiée par le système.

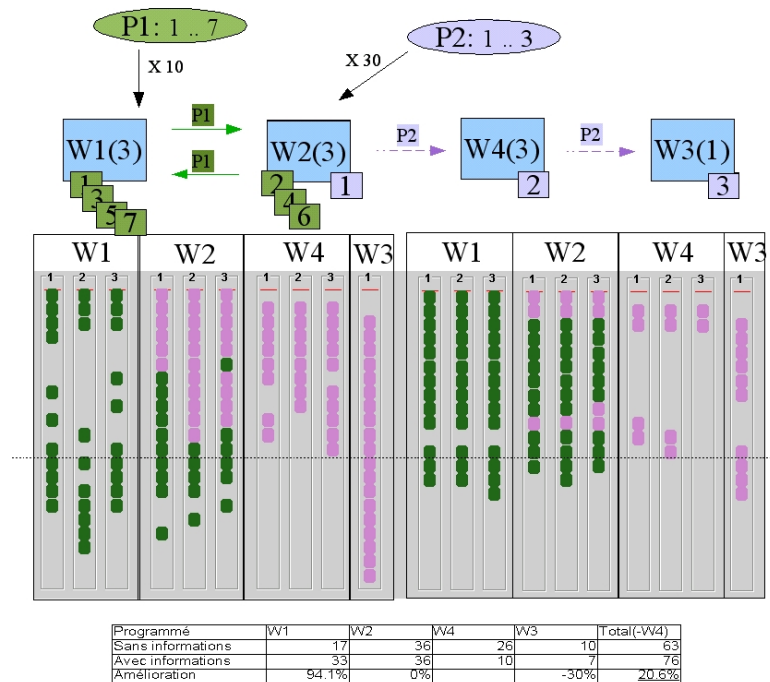


FIG. 2.4 – Répartition [sans/avec] informations : éloignement de niveau 2

– Influence du nombre de ressources communes aux produits :

Pour cette dernière expérimentation, poursuivons l'accroissement de la difficulté du problème (figure 2.5). Dans la situation proposée, le produit P1 effectue des tâches supplémentaires sur l'atelier 4. Il est ainsi mis en concurrence avec le produit P0 au niveau des ateliers 2 et 4. Sur la partie gauche de la figure 2.5, nous voyons le résultat obtenu par le premier mécanisme de combat. Par opportunité, beaucoup des tâches des produits P1 parviennent à se placer sur l'atelier 1. Ces placements ne sont pas satisfaisants car sur l'atelier 1, de nombreux espaces restent inutilisés. Les produits P1 se placent trop tard sur l'atelier 2. Les places étant une fois de plus occupées par des produits P2 bloqués à leur étape 3, et donc moins urgents. Sur la partie droite de la figure, nous voyons qu'en considérant pour la résolution l'intérêt des différents placements pour le système dans sa globalité, la solution est de nouveau sensiblement améliorée.

– Exploration et exploitation :

⁷Tous les ateliers de la figure 2.4 ne participent pas au calcul des différents indices, car leur prise en compte ne révèle pas la cohérence du placement obtenu

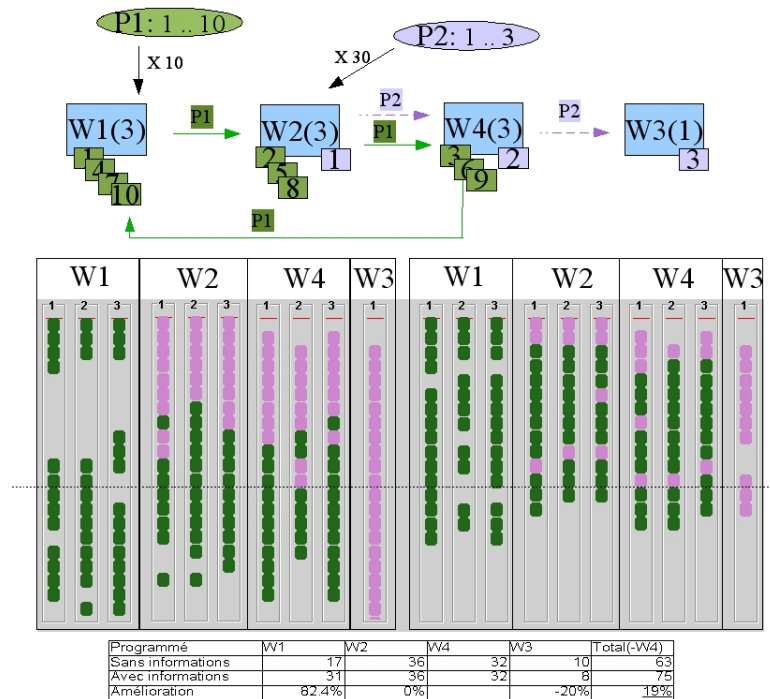


FIG. 2.5 – Répartition [sans/avec] informations : éloignement de niveau 2 + difficultés

La source d'information utilisée met en relation les différentes strates de l'environnement. Les décisions en amont dans l'environnement sont prises en fonction des situations associées en aval. Le mécanisme de contrôle est enrichi, il intègre désormais des informations sur les tentatives d'exploration : si une exploration a échoué pour un produit (les tâches du produit n'ont pas réussi à se placer), les différents agents impliqués dans l'exploration ne sont pas évalués utiles, ils risquent d'être éjectés de l'environnement. Les agents utiles au système sont eux récompensés par un renforcement de leur position. Ce mécanisme de contrôle permet la validation de l'exploration sur toutes les strates du système. Différentes tentatives d'exploration se combinent, la situation se stabilise lorsque les agents ont trouvé un emplacement qui leur est approprié : **l'amélioration de l'arbitrage des relations entre agents permet de définir les conditions de stabilité sur une solution de qualité, mais l'objectif est atteint par opportunisme** .

Lors des simulations réalisées, le temps de stabilisation s'est avéré plus important dans le cas du modèle enrichi qui permet une fouille plus approfondie de l'espace de recherche, celle-ci nous semble manquer de cohérence. L'activité est une caractéristique essentielle de notre système. Trop faible, elle n'a pas permis une exploration pertinente. Trop importante, pourrait-elle empêcher toute convergence vers une solution de qualité pour des instances de problèmes plus grandes ?

2.3.3 Limites du modèle

Les configurations simulées sont propices au bon fonctionnement du modèle proposé : nous avons majoré la quantité d'encours. Les systèmes de productions industriels profitent aussi souvent de cette majoration, car elle permet une facilité de gestion des configurations et une robustesse face aux perturbations qui interviennent de façon imprévisible (principe de la méthode *kanban*). Cependant, cette robustesse a un coût, et est plus souvent mise en place par défaut de solution alternative que par choix.

À ce stade de la recherche, nous nous questionnons sur la pertinence de la capacité de perception du système. Bien que nous avons pu communiquer des informations de coordination entre les agents, celles-ci sont-elles suffisantes pour aborder des problèmes de complexité croissante et de plus grande taille ?

Sur la figure 2.6, nous représentons le problème complexe traité, et la perception qu'en a le système. Le problème est complexe car chaque décision d'affectation impacte l'ensemble des affectations pour le problème.

On représente sur la partie gauche de la figure la complexité du problème : quels sont les produits à positionner en aval de la production afin de combler les espaces de production en amont ? On représente des chevauchements entre les parcours des différents produits. Tous les produits ne peuvent donc pas être programmés simultanément sur la période étudiée, ils sont en interaction complexe.

On représente sur la partie droite de la figure la perception biaisée de la complexité du système. La perception que nous avons proposé permet d'influencer les choix de placement réalisés en amont en fonction des besoins exprimés en aval. Les informations remontent l'espace de programmation sans aucune interaction entre elles, et sont seulement conséquentes à un calcul isolé qui définit leur parcours dans le système.

Nous soulignons donc une limite de ce modèle : la perception est compliquée alors que le problème abordé est complexe. Ainsi, lorsque le système hésite sur les choix qu'il doit réaliser, il ne perçoit pas le problème dans sa globalité. La mauvaise perception des relations complexes entraîne un parcours long et inefficace de l'espace de recherche que l'on peut associer à un problème de suractivité. Nous n'avons pas rencontré d'approche apportant des solutions à ce problème, la question reste ouverte : comment reporter les relations complexes des problèmes aux agents qui négocient de façon décentralisée ?

Les travaux réalisés par [Hadeli 04] mettent en jeu un mécanisme d'ordonnancement par anticipation proche de celui présenté. Des phéromones artificielles parcourent le système afin de réserver des emplacements de production pour les produits. Une tâche s'engage à utiliser un emplacement, sans pouvoir en percevoir l'impact complexe sur le système. Les transitions itératives entre les différents états du système peuvent aussi être saccadées puisque le système ne peut percevoir l'impact des alternatives qui se présentent à lui. Ainsi, Hadeli et al. précisent que la réaffectation d'un produit n'est envisageable que si elle entraîne des gains significatifs car elle peut diffuser de façon chaotique et imprévisible des perturbations sur plusieurs cycles d'activité. Cependant, on ne précise pas comment évaluer l'amélioration de la situation à priori. Cette évaluation doit-elle se faire localement en autarcie, par heuristique ? Cette évaluation ne devrait-elle pas plutôt se faire en considérant les relations complexes entre les affectations des produits ? Plutôt que de ne considérer que le gain pour le produit, pourquoi ne pas considérer la perte de qualité pour la solution globale ? Dans ce cas, sur quelle représentation baser cette évaluation ?

2.4 Conclusion

Nous avons présenté dans ce chapitre un travail portant sur la gestion dynamique de flux dans une chaîne de production, à travers les mécanismes de stigmergie et d'auto-organisation. Nous avons proposé un premier modèle de combat qui permet de mettre en oeuvre l'ordonnancement dynamique d'un ensemble de tâches sur un ensemble de machines ressources. Nous avons aussi observé à partir de simulations, la robustesse du système face aux perturbations émanant de son environnement incertain et dynamique ainsi que des possibilités de spécialisations émergentes. Nous avons proposé un enrichissement du modèle initial, intégrant la prise en compte d'informations rétroactives, émanant du système lui-même. Le mécanisme de coordination distribuée entre agents permet alors de contrebalancer le caractère opportuniste de la stabilité des situations en améliorant la capacité de perception du système.

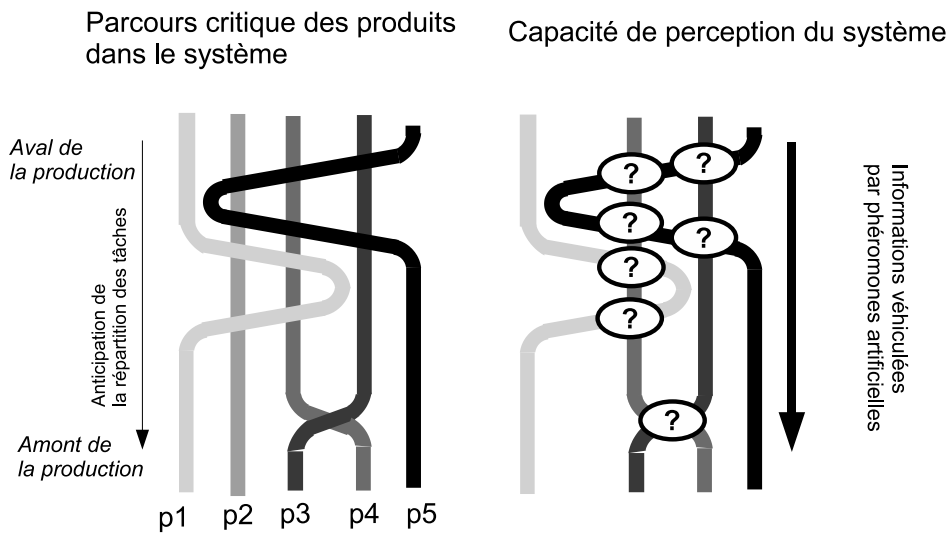


FIG. 2.6 – Perception compliquée face à des besoins de perception complexes

Avec ces premiers travaux, nous prenons contact avec un problème de partage de ressources critiques. Les problèmes sur lesquels nous avons basé cette première étude sont simplifiés face à certains problèmes difficiles rencontrés en milieu industriel. Ils nous ont tout de même permis de donner les moyens au système de percevoir les situations de congestion et de s'auto-organiser, ainsi que de mettre à jour certains défauts de perception.

Nous avons constaté dans la partie 2.3.3 des limites au modèle que nous proposons : le modèle repose sur une perception simplifiée du problème abordé. La perception biaisée mène à la propagation des erreurs à travers l'ensemble des contraintes appliquées à l'environnement. En effet, avec le modèle que nous proposons, lorsqu'un agent *Tâche* perd sa position, l'ensemble des agents situés en amont qui lui sont associés la perdent aussi, les choix réalisés par le système sont dans cette situation irréversibles. Lorsque le système commet une erreur, les répercussions se propagent dans la globalité du système. On associe cette réaction en chaîne à une 'suractivité' agent.

Face à cette première identification du problème, nous proposons deux stratégies différentes :

- Dans le chapitre 3, nous cherchons à minimiser l'impact des mauvaises décisions causées par une perception biaisée du problème. Nous proposons une nouvelle architecture qui permet de supporter un processus de validation des décisions. Conformément à ce premier modèle, le système progresse de façon itérative, par évolution de la situation spatiale courante. Des mécanismes de contrôle permettent de favoriser la résolution locale à la diffusion globale des perturbations afin d'éviter tout phénomène de 'suractivité agent'. Nous verrons aussi les mécanismes qui permettent d'engendrer des situations pour lesquelles la réorganisation locale est favorable.
- Dans le chapitre 4, plutôt que de minimiser l'impact de mauvaises décisions, nous mettons en place des mécanismes permettant au système de percevoir la complexité du problème qu'il aborde. Le travail présenté permet une meilleure communication entre les différents acteurs du problème.

3

Démarche pour le contrôle de l'activité

Dans le chapitre précédent, nous avons défini un processus de négociation pour l'amélioration itérative de la configuration courante des agents spatialement situés (modèle de combat). Nous avons intégré des informations complémentaires afin de transmettre efficacement l'objectif visé au processus de résolution (modèle de combat enrichi). Nous avons alors rencontré un problème de 'suractivité agent' qui enraille la convergence vers la configuration recherchée.

Contrôler l'activité des agents semble être bénéfique au processus de résolution. Dans [Parunak 03], chaque agent évalue localement les besoins en activité du système ce qui permet d'améliorer les performances de résolution pour le problème de partage de ressources critiques. Nous pensons que l'efficacité de cette proposition dépend des capacités de perception locale des agents. Les auteurs approfondissent peu les mécanismes visant à développer celles-ci ce qui limite la démarche de contrôle de l'activité agent'. Le chapitre qui suit introduit une nouvelle approche qui a pour objectif d'exprimer fidèlement l'échange d'informations complexes entre agents.

Dans ce chapitre, nous reprenons le concept de modération de l'activité agent' tout en l'appliquant différemment. L'objectif de cette approche est de **permettre une exploration approfondie des capacités d'aménagements locaux au cours du processus de construction ou d'adaptation de la solution afin d'éviter une diffusion anarchique et irréversible des erreurs commises par le système**. Nous reprenons le principe d'absorption locale et de diffusion contrôlée des perturbations présenté par [Tranvouez 01]. Nous nous différencions de ce travail en recherchant ces activités d'absorption et de contrôle comme le résultat d'une auto-organisation. Pour cela, le modèle que nous proposons se compose de deux groupes d'ÉcoRésolution qui collaborent par voix hiérarchique.

Nous verrons en fin de chapitre que les performances de cette approche sont malgré tout très liées au modèle de représentation des relations complexes des problèmes. Suite à l'étude des différentes perspectives visant à contrôler l'activité agent, nous nous sommes donc naturellement intéressés à un modèle de représentation des relations complexes des problèmes (chapitre suivant).

Après avoir motivé l'utilité de contrôler l'activité des agents, nous proposons une nouvelle approche, voyons ses apports et ses limites.

3.1 Motivation : une activité agent incontrôlée

3.1.1 Difficulté identifiée pour le problème de partage de ressources

Dans les travaux de [Drogoul 93], on nous présente un processus itératif qui permet la résolution d'un problème d'aménagement spatial : le problème du taquin. Pour ce problème, des palets sont positionnés sur un damier et connaissent des capacités de mouvement très limitées. À partir d'une solution initiale

aléatoire, le processus de résolution multi-agents permet l'évolution itérative de la solution courante et la convergence vers un agencement prédéfini satisfaisant. Ces travaux se situent dans le domaine de la Résolution Collective de Problèmes, et profitent d'un processus d'ÉcoRésolution développé par [Ferber 89].

Cette façon d'envisager l'activité des agents se retrouve dans beaucoup de travaux, notamment dans des travaux qui se rapportent au partage de ressources critiques ; on note toutefois des difficultés particulièrement prononcées pour cette catégorie de problèmes. Dans les travaux présentés par [Parunak 03], des agents collaborent pour résoudre des problèmes de coloration de graphe. Chaque agent est associé à un noeud du problème à colorer, et essaie de minimiser la gêne qu'il occasionne et qu'il ressent face aux agents de son voisinage, en faisant varier la couleur du noeud dont il est responsable. L'objectif étant de réduire progressivement le nombre de conflits pour atteindre progressivement une solution globale pour le problème.

Sur différents problèmes de partage de ressources critiques, Parunak constate une 'suractivité des agents' en fin de convergence. À ce stade de notre recherche, nous avons nous même pu constater ces mêmes problèmes de 'suractivité agent' et de convergence (chapitre précédent). Pourquoi ce problème s'exprime particulièrement pour le partage de ressources critiques ?

3.1.2 Propagation de l'activité par les contraintes

Comme nous le décrivons dans la section 2.4 de la partie I, les problèmes de partage de ressources critiques sont complexes, i.e., les différentes affectations de ressources réalisées entrent en interaction rétroactive les unes avec les autres. Le taux d'interaction est consécutif à la quantité et à la qualité des contraintes appliquées aux problèmes. Nous avons pu constater sur le premier modèle étudié que plus les contraintes sont nombreuses et sévères, plus les agents font preuve de 'suractivité', conséquence d'un comportement global inefficace.

Afin d'éclaircir le rapport local/global inhérent au processus de résolution itératif proposé, considérons que la solution finale recherchée se compose d'un ensemble de motifs. Ces motifs sont compatibles avec les contraintes du problème, ils sont stables sur un périmètre limité. Ils requièrent une succession plus ou moins longue d'évènements propices à leur formation. Lorsque tous les motifs, compatibles entre eux, sont formés, on atteint une solution pour le problème traité. Voyons ce qui perturbe l'élaboration de ces motifs complexes et mène à la 'suractivité agent' constatée.

Un des apports de ce travail consiste à permettre le maintien de motifs incomplets, 'un certain temps', afin d'explorer les possibilités locales de les compléter, plutôt que de diffuser des perturbations relatant l'échec de leurs formations à l'ensemble du système. Étudions le phénomène de propagation des perturbations par les contraintes pour le problème d'ordonnancement et de coloration de graphe.

Contraintes pour l'ordonnancement

Pour le problème d'ordonnancement, on représente sur la partie gauche de la figure 3.1 la propagation d'une perturbation lorsqu'un agent *Tâche* prend la décision de se déplacer d'une machine à une autre machine. Dans cette situation, la machine de départ supporte un emplacement non utilisé qu'elle doit combler. Une première stratégie peut consister à décaler les différentes tâches programmées sur la machine. Cependant, le décalage risque de violer certaines contraintes chronologiques appliquées aux tâches programmées sur la machine, et donc de propager la perturbation à l'ensemble du système. Les processus de résolution agent par construction itérative de la solution perçoivent mal l'impact des décisions appliquées à la situation courante.

Nous voyons sur la partie gauche de la figure 3.1 que ces mêmes perturbations se propagent sur la machine d'accueil. Le processus de résolution que nous souhaitons élaborer devra permettre l'élaboration

de motifs complexes comme celui présenté sur la partie droite de la figure 3.1. Dans cette situation, chaque agent améliore sa satisfaction tout en respectant une cohérence globale et ne génère pas d'activité chaotique injustifiée.

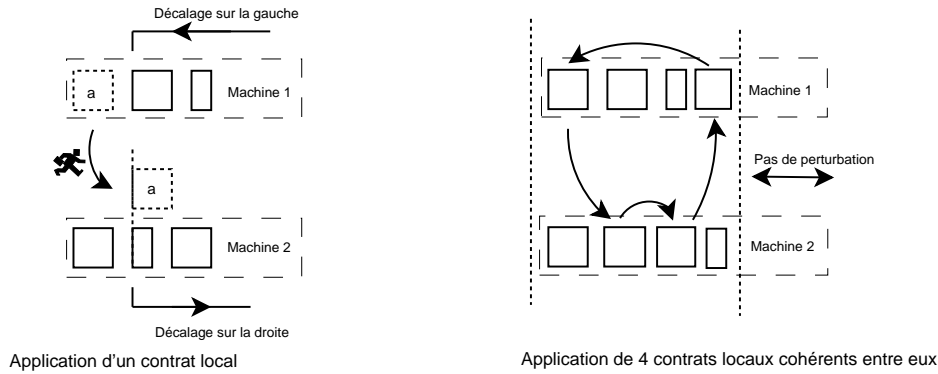


FIG. 3.1 – Suractivité des agents entraînée par les décisions locales

Contraintes pour la coloration de graphe

Dans le cas du problème de coloration de graphes, chaque agent est associé à un noeud du graphe à colorer et choisit une couleur. Chaque agent peut alors calculer son état de satisfaction à partir des couleurs affectées à ses voisins et de la couleur qu'il a lui-même choisie. Si une des couleurs de ces voisins est identique à sa propre couleur, l'agent insatisfait entamera une stratégie de résolution au cours de laquelle il pourra choisir une autre couleur. Chaque agent bénéficie d'une vision locale de son environnement. Pour sa stratégie, l'agent pourra choisir une couleur qui minimise le nombre de contraintes violées localement. L'agent peut pour cela estimer sur la base de ce qu'il perçoit l'amélioration ou la dégradation de la solution courante. Cette estimation est réalisée localement, mais perturbe globalement la situation de résolution. Les différentes contraintes du problème matérialisent sa connectivité, et entraînent la répercussion des décisions prises localement à un niveau plus global.

3.1.3 Critères de difficulté pour la résolution décentralisée

L'ÉcoRésolution permet à chaque agent de faire évoluer son état afin de maximiser sa satisfaction tout en diminuant l'insatisfaction de ses voisins avec lesquels il est en interaction. De proche en proche, ce processus permet par effet de bord de converger vers la solution recherchée. Pour certains problèmes, ce n'est pas le cas, quels sont donc les critères de difficulté qui empêchent cette convergence ?

Nous avons pu constater que les contraintes étaient responsables du phénomène de 'suractivité agent'. Plus celles-ci sont nombreuses, plus l'activité des agents propage des perturbations à l'intérieur du système et nuit à la formation des motifs complexes que l'on recherche. Nous avons par ailleurs pu remarquer que lorsque les possibilités de réarrangements locaux sont nombreuses, les perturbations peuvent être absorbées localement plus facilement. Les facilités de réarrangement locaux sont aussi liées à la rareté des solutions autorisées pour le problème abordé (figure 3.2).

Un problème de 'suractivité agent' intervient lorsque les chances de dégradation (par propagation) de la solution sont trop grandes, ce qui rend l'activité d'exploration des agents destructrice des motifs complexes composant la solution recherchée.

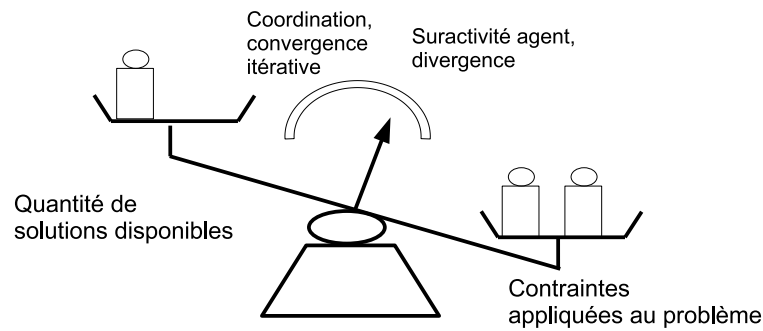


FIG. 3.2 – Problèmes propices à la ‘suractivité agent’

3.2 Proposition : Stratégie de gestion de l'activité

La nouvelle stratégie que nous proposons consiste à **modérer la propagation des perturbations à l'ensemble du système en favorisant leur absorption locale**. Cette stratégie s'exprime par les deux composantes suivantes :

- Une architecture plus flexible qui tolère des états provisoirement incohérents : Notre intuition est qu'en accroissant les capacités de résolution locale, tout en diminuant les propagations trop systématiques de perturbations, le système multi-agents peut converger vers l'élaboration de motifs complexes tel que celui présenté sur la partie droite de la figure 3.1.

Pour cela, nous proposons une architecture hiérarchique présentée dans la section 3.2.1 qui permet un contrôle flexible entre le respect des contraintes de chronologie et le respect des contraintes de placements spatiaux gérés respectivement par deux groupes de résolution.

- Le maintien de conditions favorables au réarrangement local : En autorisant des états provisoirement incohérents, nous fournissons au processus de résolution plus de liberté, mais devons néanmoins, à terme, respecter la cohérence de la solution.

Notre effort se porte d'une part, sur la coordination des échanges de ressources critiques réalisés entre les agents au cours de leur activité : ceci permet de ne pas trop s'éloigner d'une solution respectant toutes les contraintes. Par exemple, pour le problème d'affectation spatiale des agents *Tâche*, les quantités déplacées sur différentes zones de l'environnement spatial contraint doivent rester complémentaires. Dans ce chapitre, nous proposons des mécanismes heuristiques d'estimation de ces échanges qui nous permettent de guider l'évolution du système.

D'autre part, nous proposons de maintenir des situations robustes qui facilitent l'intégration locale de perturbations. La robustesse des configurations donne aux systèmes multi-agents la capacité de réagir favorablement aux perturbations.

3.2.1 Une nouvelle architecture

La nouvelle architecture que nous élaborons a pour objectif d'établir un meilleur contrôle de 'l'activité agent' pour favoriser la résolution locale à la diffusion globale des perturbations en cours de résolution. Nous utilisons le modèle Aalaadin ([Ferber 98]) pour la représenter. Ce modèle permet de représenter des agents qui interviennent dans différents groupes par différents rôles (*Agent Groupe Rôle*). Nous avons appliqué cette architecture sur le problème d'ordonnancement.

L'architecture proposée permet d'améliorer les capacités de perception du système. En effet, le système que nous décrivons autorise le maintien provisoire d'états incohérents ce qui entraîne une perception étalée dans le temps des capacités de réorganisation du système. Auparavant, sans cette décomposition, la

construction par itération n'autorisait pas d'états incohérents : à chaque départ d'un agent *Tâche* de l'environnement spatial une partie des agents associés au même produit perdaient leur position. Le modèle ne prévoyait pas d'étaler les décisions de propagation de perturbations, celles-ci étaient instantanées.

Décomposition par catégories de contrainte

Nous proposons de gérer de façon dissociée les deux catégories de contraintes du problème d'ordonnement : les contraintes de chronologie et les contraintes d'affectation spatiale. L'approche présentée en première partie de [Liu 94] consiste en une décomposition similaire.

Nous associons donc un groupe d'agents à chaque catégorie de contraintes à gérer :

- Groupe *Séquencement* : Ce groupe a pour objectif de définir les positions souhaitables pour chacune des tâches des produits à fabriquer. Cette répartition prend en compte le respect des contraintes chronologiques entre les tâches d'un même produit et intègre des objectifs de robustesse et de respect de délais.
- Groupe *Affectation* : Ce groupe a pour objectif d'affecter les différentes tâches aux différentes machines disponibles en respectant la répartition élaborée par le groupe *Séquencement*.

L'architecture de supervision que nous proposons pourrait être appliquée suivant des décompositions différentes. Pour le problème de coloration de graphe, on peut envisager de superviser la diffusion de perturbation entre sous-parties du problème. La supervision a alors pour objectif de favoriser les réarrangements géographiques locaux. Dans ce contexte, une négociation entre agents superviseurs permet d'équilibrer les ressources consommées entre les différentes régions du problème.

Les travaux de [Tranvouez 01] proposent une décomposition hiérarchique qui permet le contrôle de la diffusion géographique des perturbations sur plusieurs niveaux (machine, groupe, atelier, usine, etc.). Suite à une perturbation, lorsque les règles de réorganisation locales échouent, le périmètre de réorganisation est élargi.

Flexibilité appliquée aux contraintes de chronologie

Pour le problème d'ordonnement, nous identifions deux types de contraintes par lesquelles les perturbations peuvent se propager au cours de la résolution : les contraintes chronologiques et les contraintes de capacité de production des machines. Nous avons constaté que les contraintes de chronologie constituent un facteur important de 'suractivité agent'. On note en effet que les problèmes d'ordonnements abordés sans ce type de contrainte ne produisent pas autant de problèmes de 'suractivité agent' ([Ghédira 92]).

L'architecture que nous proposons permet de 'flexibiliser' la relation entre le groupe *Séquencement* et le groupe *Affectation* : toutes les séquences de placement élaborées ne sont pas toujours respectées par le groupe *Affectation*. Ceci autorise des configurations provisoirement incohérentes mais permet de transiter avec plus de souplesse entre les états du système.

Nous voyons sur la figure 3.3 la relation de 'supervision flexible' d'un groupe *Séquencement* sur le groupe *Affectation*. Nous détaillons dans la partie 3.2.2 les relations de perception et d'action qui s'exercent entre ces deux niveaux.

Nous choisissons de maintenir l'environnement spatial du groupe *Affectation* cohérent en interdisant toute distortion de celui-ci : la somme des espaces occupés par l'ensemble des agents reste constante. Cette cohérence facilite la perception du groupe *Séquencement* sur le groupe *Affectation*. La flexibilité de l'architecture intervient seulement entre le groupe *Affectation* et *Séquencement*. **La performance du processus de résolution dépend de la capacité de ces groupes à se synchroniser, et donc à converger vers une configuration satisfaisante .**

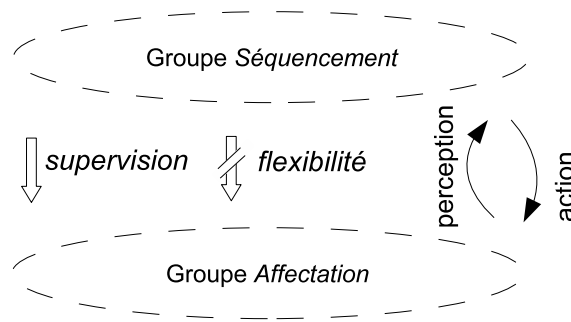


FIG. 3.3 – Groupes Séquencement et Affectation

Représentation AGR (Agent Groupe Rôle)

L'architecture que nous proposons pour le problème d'ordonnancement a été élaborée pour permettre une gestion séparée, mais cohérente, de deux types de contraintes : les contraintes de positionnement spatial et les contraintes chronologiques de séquencement.

Sur la figure 3.4, le modèle de représentation *Agent Groupe Rôle* permet de décrire l'architecture agent composée des groupes *Séquencement* et *Affectation* sur lesquels des agents jouent des rôles.

Nous définissons trois types d'agents dont les rôles sont présentés dans le tableau 3.5 :

- Agents *Étape* : Ces agents jouent un rôle dans les deux groupes de l'architecture. Dans le groupe de séquencement, les agents *Étape* associés au même produit collaborent afin de trouver une séquence de placement appropriée aux objectifs. Au terme de cette négociation, chaque agent a pour objectif de s'associer avec un agent *Emplacement* du groupe *Affectation*. L'agent *Étape* délègue alors les responsabilités de négociation spatiale à cet agent.
- Agents *Emplacement* : Ces agents jouent un rôle de négociation spatiale dans le groupe *Affectation*. Ils représentent un espace de l'environnement machine. Ils sont créés par une machine puis se déplacent sur les plannings en fonction de leurs objectifs. Ces agents peuvent exister dans deux états :
 - Lorsqu'ils sont affectés à un agent *Étape*, ils essayent de le satisfaire en atteignant ou maintenant une position sur l'environnement.
 - Lorsque ces agents sont libres (non associés à un agent *Étape*), ils ont pour objectif de fluidifier l'activité spatiale sur les différents plannings de machine (déplacements conciliants, fusions ou scission d'agents *Emplacement*).
- Agents *Ressource* : Les agents *Ressource* n'ont de rôle que dans le groupe *Affectation*. Chaque agent *Ressource* a la capacité de générer et d'associer des agents *Emplacement* aux agents *Étape*.

Relations entre les groupes

Une activité d'ÉcoRésolution s'exerce dans chacun des groupes. Des séquences de placement sont élaborées par le groupe Séquencement qui les soumet au groupe Affectation. Nous voyons, sur la figure 3.6, qu'une proposition émise peut être acceptée lorsqu'un emplacement spatial est trouvé (partie gauche) ou bien rejetée ce qui conduit à une renégociation de la séquence.

Nous détaillons dans la partie qui suit la gestion de l'activité de chacun des groupes.

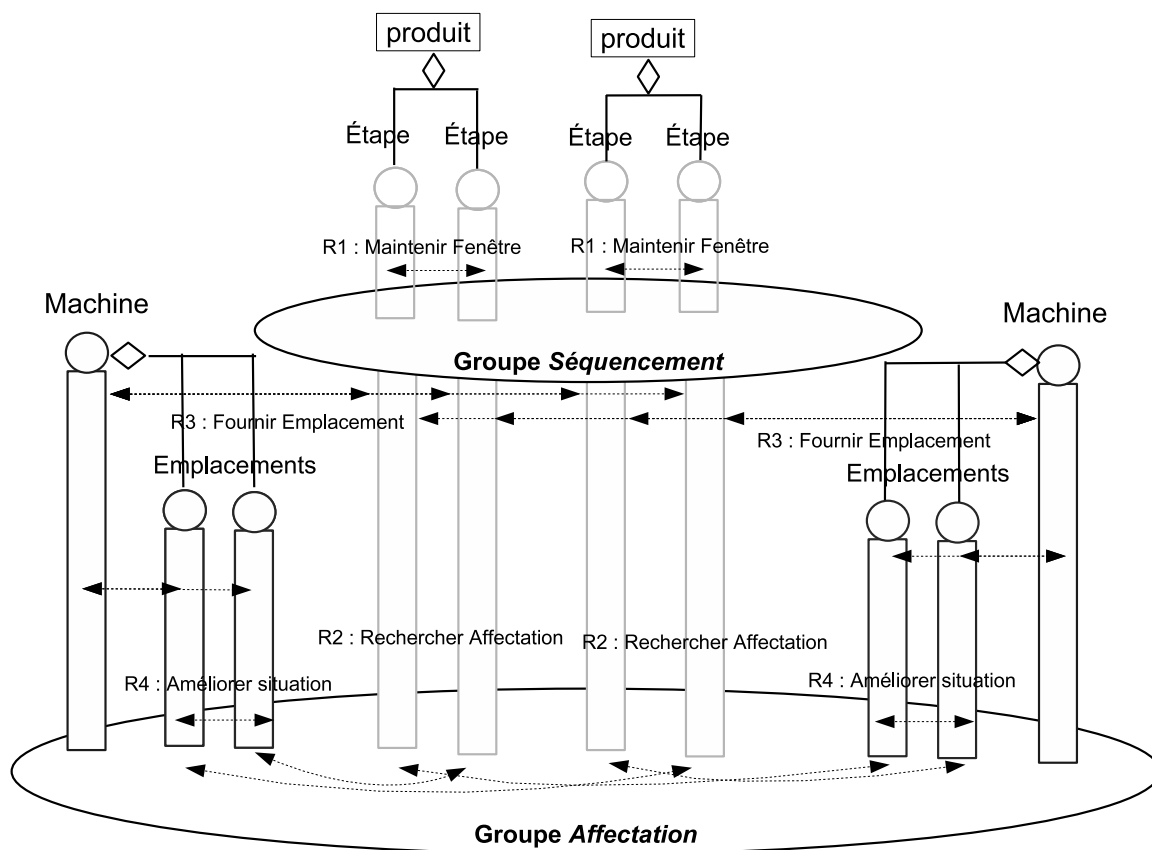


FIG. 3.4 – Modèle AGR (Agent Groupe Rôle)

Agents/Groupes	Groupe Séquencement	Groupe Affectation
Agent <i>Étape</i>	Maintenir une fenêtre de placement	Rechercher une affectation
Agent <i>Ressource</i>		Fournir des agents <i>Emplacement</i>
Agent <i>Emplacement</i>		Améliorer sa position

FIG. 3.5 – Rôles des agents

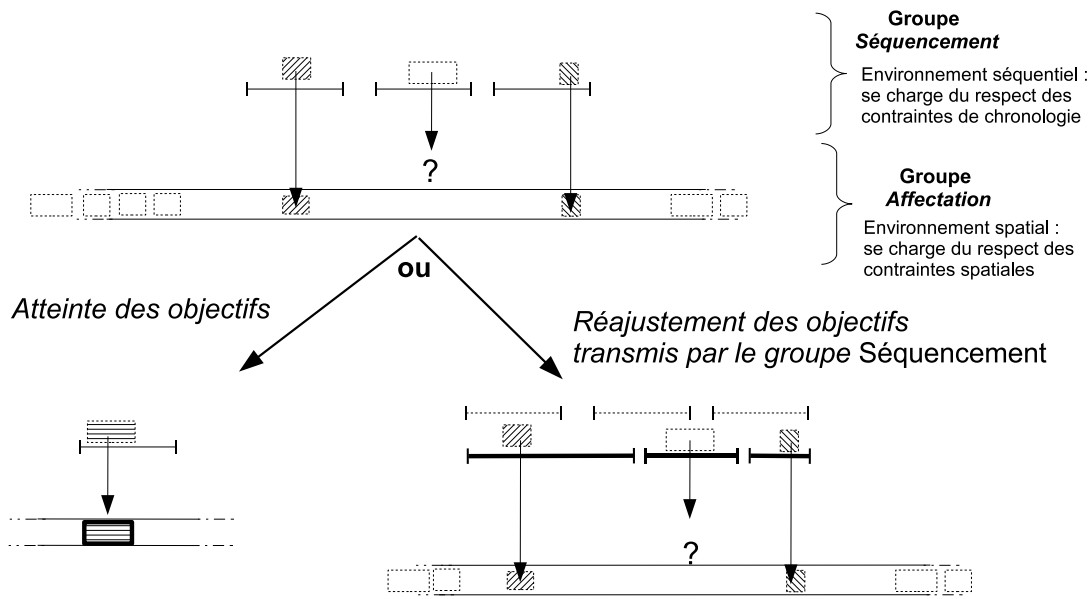


FIG. 3.6 – Relation entre groupes de résolution

3.2.2 Gestion de l'activité du système

Suivant le modèle que nous proposons, l'activité agent est consécutive à l'instabilité des agents *Étape* du groupe *Séquencement* qui adaptent les objectifs transmis au groupe *Affectation*. Chaque agent *Étape* recherche une fenêtre de placement qui satisfait certains critères, il doit pour cela trouver un accord avec les agents *Étape* associés à son produit. On définit $Fenetre(e_i) = \{tDeb \in N, tFin \in N\}$ pour un agent *Étape* e_i . Une fois la fenêtre définie, chaque agent *étape* transmet des objectifs de placement au groupe *Affectation*.

Dans ce contexte, à quel moment déclencher une activité de renégociation pour le groupe *Séquencement*? Il existe de nombreuses possibilités de gestion de cette activité, nous n'avons pas arrêté une stratégie particulière mais après plusieurs tentatives, nous soulignons les difficultés à combiner efficacement les critères d'activité que nous allons décrire. Cette constatation nous a permis d'éclaircir les difficultés rencontrées que nous proposons de résoudre dans le chapitre qui suit.

Lorsqu'un agent *Étape* ne parvient pas à obtenir un agent *Emplacement* qui le satisfait⁸, sa contrariété augmente jusqu'au déclenchement d'une renégociation de la séquence pour le produit associé. Ainsi, **nous proposons un seuil de contrariété comme déclencheur de l'activité.**

Dans [Liu 94], la stratégie de résolution choisie a elle-même pour objectif de minimiser la diffusion des perturbations à l'ensemble du problème abordé. De la même manière, on contrôle la propagation de perturbations par différents seuils de contrariété agent. La contrariété d'un agent dépend de son temps d'insatisfaction. On autorise la perturbation du système par les agents les plus contrariés et affectés aux parties de problème les plus critiques.

Il est intéressant d'envisager d'affiner le niveau du seuil de contrariété en considérant les opportunités d'amélioration de satisfaction perçues par l'agent *Étape*. Nous proposons d'établir cette perception suivant trois critères qui sont détaillés par la suite : $S_{e_i}^{Contrainte}$ et $S_{e_i}^{Robustesse}$ sont des critères évalués

⁸Un agent *Étape* est satisfait lorsqu'il obtient un agent *Emplacement* associé à la tâche du produit qu'il représente dans la fenêtre de placement qu'il propose

localement par l'agent e_i , $S_{e_i}^{Affectation}$ est une estimation de la capacité d'accueil perçue par le groupe *Séquencement* sur le groupe *Affectation*.

Groupe séquencement

Fidèlement au principe d'écorésolution, chaque agent recherche une amélioration de sa propre satisfaction. L'enjeu de la négociation repose sur la définition d'une fenêtre de placement pour chacun des agents *Étape*.

Dans le groupe *Séquencement*, chaque agent *Étape* peut évaluer sa satisfaction pour une configuration, i.e., pour une fenêtre de placement, comme il suit :

$$S_{e_i \in p_j} = S_{e_i}^{Contraintes} \oplus S_{e_i}^{Robustesse} \oplus S_{e_i}^{Affectation}$$

Sachant que :

- e_i est un agent *Étape* associé au produit p_j
- $S_{e_i}^{Contrainte}$ est la mesure de respect des objectifs de délais pour le produit associé.
- $S_{e_i}^{Robustesse}$ est une mesure heuristique de la robustesse pour l'agent *Étape*
- $S_{e_i}^{Affectation}$ est la mesure de satisfaction d'affectation pour l'agent *Étape*.
- \oplus est un opérateur de composition des différents paramètres de la fonction.

On définit l'état de satisfaction pour un produit dans une configuration donnée S^{p_j} , tel que $S^{p_j} = \sum_{e_i \in p_j} S_{e_i}$. Le processus de négociation entre agents *Étape* a pour objectif de maximiser la valeur de S^{p_j} , qui est le résultat d'une composition de la satisfaction des différents agents *Étape*. Il peut être envisagé de façon décentralisée ou même centralisée, suivant des stratégies très variées.

Groupe Affectation

Le groupe affectation est responsable de l'aménagement spatial de l'environnement des différentes machines. Les objectifs du groupe *Séquencement* sont communiqués au groupe *Affectation* par l'attribution de fenêtre de placements à certains agents *Emplacement*.

Nous avons développé plusieurs stratégies satisfaisantes de réorganisation locales pour le groupe *Affectation* (stratégies limitant les situations de blocage, stratégie de génération de nouveaux agents *Emplacement* pour les agents *Étape*, etc.). Nous ne les décrivons pas de façon exhaustive car elles ne sont pas nécessaires à la compréhension du travail réalisé, les conclusions consécutives au modèle proposé sont d'ailleurs dissociées de la capacité de réarrangement constatée sur le groupe *Séquencement*.

L'activité agent réalisée sur l'environnement des machines met en jeu des situations de fuite pour lesquelles certains agents *Emplacement* dégradent leur satisfaction pour permettre le positionnement adéquat d'autres agents *Emplacement*. Un agent *Emplacement* non affecté à un agent *Étape* tend à se regrouper avec d'autres agents similaires jusqu'à obtenir une taille suffisante pour être candidat à l'affectation avec un agent *Étape*. Ces agents sont libres et n'occasionnent aucune gêne de placement pour les agents *Emplacement* affectés à un agent *Étape*.

Comme nous le voyons par la suite, l'approche décrite repose en partie sur le contrôle de la robustesse au sein du système de production. Celui-ci permettant de faciliter les réorganisations locales et de modérer l'activité des agents. Dans le groupe *Affectation*, chaque agent *Emplacement* affecté à un agent *Étape* a pour objectif de se placer et de se maintenir dans une autorisée région de l'environnement. Afin de maximiser la robustesse des situations, on définit la satisfaction d'un agent *Emplacement* comme maximale lorsqu'il obtient une position proche du centre de la fenêtre de placement. Dans cette situation l'agent connaît autant de capacité de mouvement en amont et en aval de sa position, ce qui accroît ses capacités de réorganisation.

Nous considérerons pour la suite de ce chapitre que la stratégie de réorganisation appliquée au groupe *Affectation* permet à chaque agent *Étape* de concrétiser sa situation en obtenant un agent *Emplacement*

correctement positionné lorsque les objectifs de placement fixés par le groupe *Séquencement* sont correctement établis. L'essentiel de la problématique provient alors de la pertinence de l'activité du groupe de *Séquencement*.

3.2.3 Capacité de perception du système

Cette description de l'approche a pour objectif de dégager des perspectives de gestion de l'activité des agents. Dans cette partie, nous approfondissons les critères de satisfaction des agents *Étape*, mais ne fixons pas la méthode permettant de les combiner. D'une part, les agents *Étape* recherchent des fenêtres de placement permettant le respect de délais, ainsi que le respect d'un critère de robustesse. D'autre part, ceux-ci sont à l'écoute des capacités d'organisation perçues sur le groupe *Affectation*.

Expression du respect de délais et de robustesse

Les critères de satisfaction de robustesse sont évalués localement (sans considérer le groupe *Séquencement*) pour différentes séquences. Une première étape de calcul permet de définir les marges de négociation des séquences (il est inutile de laisser le groupe *Séquencement* estimer des configurations non autorisées qui ne respectent pas les contraintes de chronologie). Le calcul permet de garantir le respect d'un ensemble de contraintes de délais définies. Lorsque le système ne parvient pas à trouver de solution intégrant toutes les contraintes de délais, on peut envisager d'en relaxer certaines et d'élargir ainsi le domaine des séquences autorisées pour la négociation. La dégradation occasionnée est alors perçue par l'intermédiaire de $S_{e_i}^{Contrainte}$.

Définition des marges de négociation pour le groupe *Séquencement* :

La méthode PERT est une méthode d'ordonnancement qui repose sur un processus de recherche du plus long chemin dans un graphe de contraintes. Elle introduit pour chaque tâche une plage d'exécution autorisée qui permet de garantir le respect des contraintes du problème. Nous utilisons pour définir les marges de négociation un concept 'de fenêtre de placement' semblable à celui de la méthode PERT.

Dans notre situation, pour appliquer le concept de la méthode PERT, nous considérons les fenêtres de placement temporelles comme des tâches à réaliser. Ces fenêtres ont un 'temps d'exécution' variable (il sera l'enjeu de la négociation au sein du groupe *Séquencement*), la méthode PERT ne peut donc pas être appliquée directement. Sur la figure 3.7, nous représentons le résultat de deux calculs PERT intermédiaires qui définissent les temps de début au plus tôt et au plus tard dans le cas de robustesse minimale et maximale (taille de fenêtre minimale et maximale). À partir de ces résultats, nous pouvons définir les plages horaires sur lesquelles les fenêtres pourront prendre place tout en respectant l'ensemble des contraintes du problème (figure 3.7). Les cumuls des différences entre marges amont et aval mènent à de larges marges de négociation pour les agents du groupe *Séquencement*.

Maintenant que nous avons définis l'ensemble des marges de négociation pour le groupe *Séquence-ment*, voyons comment le critère de robustesse intervient au sein du processus de négociation du groupe *Séquencement*.

$S_{e_i}^{Robustesse}$ estime la qualité de la robustesse pour un agent *Étape* e_i . Pour le problème d'ordonnancement, comme pour de nombreux problèmes, la robustesse permet d'augmenter les capacités de réorganisation locales. Pour le problème traité, on associe à chaque agent *Étape* une taille fenêtre temporelle objectif, on contrôle la capacité d'absorption des perturbations pour chaque étape de fabrication. Plus la fenêtre de placement est grande, plus la réorganisation spatiale de l'environnement est facilitée (robustesse). Plus la taille de fenêtre envisagée diffère d'une taille objectif dite de robustesse maximale, plus $S_{e_i}^{Robustesse}$ retourne une valeur faible.

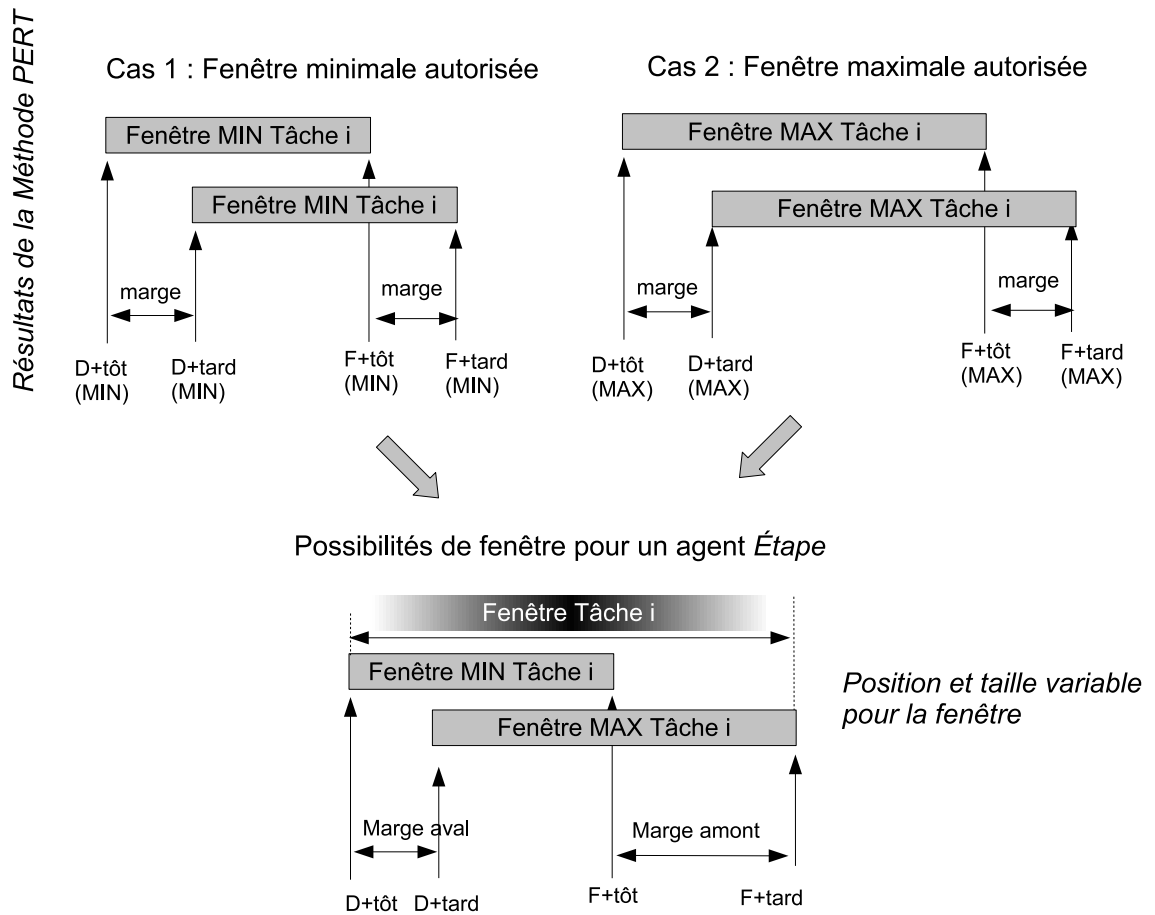


FIG. 3.7 – Calcul des marges de négociation pour le groupe *Séquencement*

Estimation de la capacité d'accueil du système

Le groupe Séquencement doit développer des capacités de perception des opportunités d'aménagement du groupe Affectation.

Nous avons envisagé des mesures d'estimation de la capacité d'accueil ($S_{e_i}^{Affectation}$) suivant plusieurs axes :

- Une **mesure heuristique de charge de l'environnement** : Chaque agent *Étape* peut sonder l'environnement spatial afin d'en estimer directement la capacité d'accueil. Ainsi, plus il existe d'espaces vides sur la partie de l'environnement visée, plus les chances de s'y positionner sont nombreuses. L'estimation n'est pas seulement calculée à partir d'un taux d'occupation au niveau de la fenêtre de placement évaluée, mais l'est sur un périmètre plus large. En effet, la capacité de réorganisation de l'environnement spatial manifeste des capacités de réorganisation (celles-ci sont en partie configurées par le niveau de robustesse défini par le groupe *Séquencement*).
- Une **mesure pour la complémentarité des échanges au cours du temps**, mémoire de la charge de l'environnement au cours du temps : Comme nous le voyons sur la figure 3.8, l'insertion forcée d'un agent Emplacement sur l'environnement spatial créé un déséquilibre qui se propage de proche en proche. La robustesse du groupe Affectation peut alors permettre une réorganisation locale. Cependant, la perturbation occasionnée réduit les capacités de réorganisation future. La mesure complémentaire a pour objectif de favoriser la compensation des déficits/excédents occasionnés par les migrations passées des agents *Étape* afin de maintenir la robustesse du système au cours du temps.
- Une **mesure de coordination** : Cette mesure de coordination est la plus ambitieuse, mais la plus pertinente. Elle fait l'objet d'un approfondissement dans la troisième partie du travail présenté. Les constatations faites sur cette première version vont guider l'amélioration de l'approche visant à coordonner l'activité des agents.

Pour cette version, chaque agent *Étape* marque sur l'environnement des intentions d'acquisition et de libération d'agents Emplacement sur différentes plages de l'environnement spatial. Ces intentions sont le résultat des configurations étudiées lors de la négociation entre agents *Étapes*. Lorsque des intentions complémentaires sont mises en relation sur l'environnement (intention d'acquisition - intention de libération), elles ont tendance à se renforcer : ce renforcement se matérialise par l'augmentation de la valeur $S^{Affectation}$ des agents concernés.

La coordination de l'activité agent intervient par la réaction en chaîne occasionnée par ces différents renforcements. En effet, le renforcement peut se propager entre les différents agents *Étape* du produit par l'intermédiaire de la négociation au sein du groupe *Séquencement*, pour ensuite propager des influences au groupe *Affectation*. Nous voyons que la propagation décrite semble calquer la complexité des problèmes. Lorsque le renforcement arrive à maturité, différentes séquences sont validées et transmises au groupe *Affectation*. Pour cette première version, la maturité est estimée par un seuil de renforcement, défini empiriquement. Nous voyons en conclusion que la gestion par seuil entraîne des biais de perception qui ne sont pas propices à l'auto-organisation de ces systèmes.

3.3 Apports

Nous avons implémenté différents agencements des critères que nous venons de décrire, nous discutons dans cette partie des différentes alternatives proposées pour la gestion de l' 'activité agent'. La palette de comportements décrits représente des mécanismes très variés dont certains ont été développés dans d'autres approches. Nous nous sommes confrontés aux mêmes difficultés que celles-ci rencontrent.

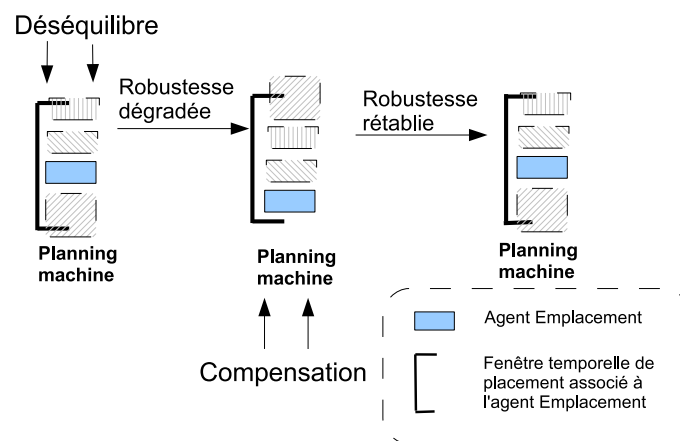


FIG. 3.8 – Équilibre des déplacements réalisés sur l'environnement spatial

La proposition de gestion de l'activité suivant l'architecture dégage des perspectives qui pourront être approfondies et complétées par de futurs travaux.

3.3.1 Comportement constaté

Nous avons pu constater une amélioration sensible de la performance du système : Le système permet la réorganisation locale ainsi qu'une convergence rapide vers une situation satisfaisante. L'approche permet une fouille approfondie des possibilités de réarrangements locaux : les perturbations qui interviennent au cours du processus de résolution sont absorbées localement lorsque cela est possible. Nous constatons cependant des difficultés de résolution lorsque les problèmes se complexifient (plus de contraintes et moins de réarrangements locaux disponibles). Lorsque la résolution locale échoue, les perturbations sont tout de même propagées au sein du système qui semble alors manquer d'une perception aiguisée des différents arrangements possibles en environnement complexe.

3.3.2 Interactions utilisateur

Ce nouveau mode de négociation est aussi l'occasion de faciliter les interactions utilisateur sur la solution générée. L'étude de marché menée dans le domaine du partage de ressources critiques a souligné un manque de flexibilité d'interaction entre l'utilisateur et le logiciel qu'il utilise. D'une part, les modèles informatiques utilisés parviennent difficilement à s'adapter aux besoins changeants des utilisateurs. D'autre part, les solutions fournies par les outils informatiques sont difficiles à remettre en cause et à faire évoluer. Une fois de plus, nous avons souhaité profiter de la flexibilité de modélisation apportée par les systèmes multi-agents, qui permet une facilité d'expression d'objectifs dynamiques. Ainsi, nous avons donné la possibilité à l'utilisateur d'ajouter ou de relâcher dynamiquement des contraintes de délais appliquées à l'ordonnancement généré. Le processus inspiré de la méthode PERT peut alors redéfinir rapidement les enjeux de la négociation pour le groupe *Séquencement*.

3.4 Limitations

Nous remarquons que plus les problèmes que nous abordons sont contraints, plus le système peine à se stabiliser et à fournir une solution. En d'autres termes, favoriser la résolution locale ne permet pas de définir le chemin qui y mène. Les résultats que nous obtenons ne nous satisfont pas totalement, nous constatons toujours une 'suractivité agent', bien que celle-ci s'exprime de façon différente par une diffusion contrôlée des perturbations. Les mécanismes mis en jeu modèrent les risques de 'suractivité agent' (architecture qui privilégie la réorganisation locale, contrôle du niveau de robustesse, favoriser les échanges cohérents), mais ne les annule pas.

Les processus permettant de limiter la diffusion des perturbations sont très utiles, mais ne répondent pas à la globalité de la problématique abordée. Ceux-ci doivent en effet être complétés par un processus de coordination performant qui permet la construction de motifs complexes d'échange cohérent sur l'ensemble de l'environnement. Celui que nous avons implémenté ne semble pas suffisant, analysons donc ses carences.

Nous n'avons pas décrit le processus de coordination en détail, mais en avons simplement explicité les principes. Ces principes ne sont pas exempt de tout défaut :

- Utilisation de seuil de déclenchement de l'activité :

Notre modèle profite d'un processus de décision à seuil à deux niveaux d'activité. D'une part, les agents Étape tolèrent une insatisfaction passagère pendant un certain laps de temps, puis sont progressivement poussés à entamer une négociation pour réaménager leurs objectifs. D'autre part, la mesure de coordination présentée dans la partie 3.2.3 permet un renforcement en chaîne jusqu'à

atteindre un seuil de déclenchement. Qu'il s'agisse d'un seuil basé sur le temps ou sur la qualité, le principe d'action par seuil est dans les deux cas pénalisant. L'occasion nous est donnée de nous questionner sur la contradiction que l'utilisation de seuils peut représenter pour les systèmes auto-organisés.

Par défaut, ces seuils sont paramétrés de façon empirique. Dans notre situation, l'utilisation de seuil est d'une certaine façon pratique : c'est une façon d'explorer en partie un espace de recherche pour ensuite décider et appliquer une décision de propagation de perturbation. Cependant, le comportement global d'un système complexe peut-il être projeté dans une simple valeur pour être comparée à une valeur fixée seuil ? Les difficultés que nous avons rencontrées pour fixer des valeurs de seuil pour la mesure de coordination nous mène à penser que l'utilisation de seuil crée un biais qui est très handicapant pour le problème de partage de ressources critiques. Le seuil peut être vu comme une frontière artificielle qui ne permet pas aux agents de percevoir les caractéristiques du problème complexe qu'ils abordent. Il peut aussi être analysé comme paralysant pour l'auto-organisation d'un système qui perd toute autonomie de décision. Nous rencontrons avec ce modèle des difficultés proches de celles rencontrées lors de notre précédent modèle : comment donner la capacité de perception des relations complexes du problème abordé aux agents ?

- Coordination avec engagements contradictoires : Smith et Davis ont proposé un protocole de négociation très utilisé : *contract net* ([Smith 83]). Pour certains de ces modèles d'application, le protocole prévoit l'engagement réversible des agents pour les différents contrats. Pour la perception du groupe *Séquencement* sur le groupe *Affectation*, les agents *Étape* peuvent exprimer des intentions de migration contradictoires entre elles sur différents environnements de machines. On se retrouve alors dans le contexte de gestion d'engagement réversibles.

Dans [Sauter 99], on propose un modèle selon lequel chaque agent peut exprimer plusieurs intentions contradictoires entre elles. On propose d'estimer l'utilisation des ressources résultantes selon une approche statistique. Cette estimation permet de faire évoluer la négociation afin de converger vers une répartition adéquate des différents consommateurs de ressource. Une fois de plus, nous avons rencontré des difficultés à élaborer de telles stratégies. En effet, l'engagement contradictoire des agents représente un risque de perturbation lors de la mise en application partielle d'agents doublement engagés. Ceci nous incite à revoir le mode d'engagement des agents dans les différentes coordinations.

3.5 Conclusion

Pour le modèle implémenté, nous avons mis en place des mécanismes permettant à chaque agent d'améliorer sa position tout en considérant le contexte global dans lequel il s'intègre. Pour cela chaque agent perçoit l'amélioration de sa situation grâce aux différents paramètres décrits, ce qui l'amène à négocier avec les agents par lesquels il est lié par des contraintes.

Nous soulignons le rôle fondamental des négociations menées dans le groupe *Séquencement*. Dans ce groupe, les deux premiers paramètres considérés permettent de représenter les contraintes de délais du problème et de maintenir le système dans un état robuste. Nous ne remettons pas en cause le fait que ceux-ci soient utiles à l'organisation d'un tel système. La modélisation proposée permet l'intégration de contraintes (contraintes chronologiques, contraintes de délais, contraintes utilisateur, etc.) tout en considérant un niveau de robustesse souhaité pour le système. Le paramétrage du système permet d'accentuer le respect de l'une ou l'autre de ces composantes. Le dernier paramètre que nous avons considéré permet d'évaluer la capacité d'accueil du système afin de réaliser des échanges de position cohérents sur l'ensemble du système.

Bien que le contrôle de la robustesse soit un outil de contrôle de l'activité agent, elle ne suffit pas à résoudre le problème de suractivité. Par ailleurs, celle-ci a un coût ce qui la rend inappropriée pour certains problèmes de partage de ressources critiques (une solution robuste n'est pas toujours optimale). Les mécanismes de flexibilité visant à favoriser la réorganisation locale ne règlent eux non plus pas tous les problèmes. En effet, lorsqu'une réorganisation locale n'est pas réalisable, une perturbation associée est amenée à perturber le système plus globalement. Ainsi, nous avons pu constater que le nouveau modèle proposé n'a pas permis de réaliser des échanges cohérents permettant de favoriser la réorganisation globale. Nous poursuivons notre étude par l'approfondissement de la capacité de coordination de l'activité des agents, ceci afin de permettre des échanges cohérents sur l'environnement.

Nous n'avons pas ressenti la nécessité de parcourir de façon exhaustive l'ensemble des stratégies que l'on pourrait envisager sur ce modèle, l'analyse que nous faisons de ces résultats nous mène à penser le problème sous un angle différent : nous poursuivons l'approche en développant la capacité de coordination par l'élaboration de motifs d'échanges complexes cohérents.

Positionnement de CESNA

La construction itérative d'une solution complexe par décentralisation est particulièrement délicate. Il est très difficile de guider le système de résolution dans un espace de recherche de grande taille.

Pour que la situation courante évolue efficacement, il est souhaitable de fournir de bonnes capacités de perception au système. Notre premier modèle n'a pas permis d'exprimer avec suffisamment de précision les problèmes traités pour permettre un parcours cohérent de l'espace de recherche. Comme dans d'autres approches, nous avons alors été confrontés à un problème de 'suractivité agent'. L'objectif de notre deuxième modèle fût de contrôler la diffusion de perturbations génératrices de suractivité afin de favoriser la résolution et l'intégration locale de celles-ci. Cependant, le modèle proposé segmente (hiérarchie) artificiellement (seuil artificiel d'activité) les différentes sous-parties des problèmes ; ceci nuit à la communication des relations complexes à travers le système.

Il est aussi souhaitable de maintenir une bonne coordination des activités d'exploration et d'exploitation. Sur le premier modèle étudié, les combats engendrent une activité destructrice qui n'est pas favorable à un tel équilibre. Le deuxième modèle quant à lui compense explicitement ce défaut de fonctionnement en minorant les destructions de l'exploration sur l'exploitation.

Voyons donc quelles améliorations sont à apporter pour que le système puisse agir efficacement face aux relations complexes des problèmes qu'il aborde en maintenant un équilibre performant entre exploration et exploitation.

4.1 Communication des relations complexes au sein des systèmes multi-agents

Au cours de notre étude, nous avons choisi un mode de communication par marquage de l'environnement ou stigmergie. L'environnement peut être un environnement spatial dans lequel des agents situés évoluent. Il peut aussi être plus abstrait et permettre d'exprimer des configurations abstraites.

Les contraintes s'expriment de façon plus ou moins appropriée dans un environnement spatial. Pour le problème du taquin traité par [Drogoul 93], les contraintes de déplacement s'expriment naturellement sur l'environnement qui calque une configuration physique associée. Pour le problème d'ordonnancement, ces contraintes ne s'expriment pas aussi directement, elles requièrent une modélisation abstraite appropriée. Ainsi, le premier modèle proposé prévoit la remontée de phéromones artificielles entre les différents agents *Tâche*. Ces phéromones constituent un signal qui parcourt un environnement artificiel consécutif aux différentes contraintes de chronologie entre les produits à fabriquer. La continuité de l'environnement spatial ne semble en effet pas appropriée pour exprimer ce type de relations entre agents *Tâche*. Les différents types de contraintes ont donc été exprimés dans des environnements différents, avec des modes de marquage et de propagation différents, qu'en est-il donc de la transition entre

ces environnements ? Comment les relations complexes s'influencent entre représentations différentes ? La transition est souvent réalisée de façon totalement synthétique ce qui est un frein à la possibilité d'échange d'informations au sein du système et d'auto-organisation de celui-ci. Par exemple, nous avons présenté dans le chapitre précédent un mécanisme qui régit la relation entre le séquençement des agents et leur affectation spatiale sur les différentes ressources de fabrication. Pourquoi le système devrait avoir une approche différente pour traiter des contraintes de chronologie ou bien des contraintes de réservation de ressources de production ?

Pour traiter le problème de partage de ressources critiques de façon générique, trouver un modèle de représentation commun à tous les types de contraintes envisageables nous semble être une solution à la bonne diffusion des caractéristiques complexes des problèmes abordés dans leur intégralité.

4.2 Équilibre entre exploration et exploitation

L'activité des agents permet le parcours de l'espace des possibilités via des mécanismes d'exploration et d'exploitation. L'exploration permet d'envisager de nouvelles pistes alors que l'exploitation renforce les pistes qui semblent les plus prometteuses. Pour le problème de fouragement, les comportements sociaux se décomposent eux aussi en terme d'exploitation et d'exploration [Bonabeau 94]. Dans ce cadre, l'exploitation n'est pas ressentie comme directement nuisible à toute activité de renforcement. En effet, une fourmi peut partir explorer une partie de l'environnement sans pour autant dégrader le chemin le plus renforcé. Au contraire, suivant de nombreux modèles d'agents appliqués aux problèmes de partage de ressources, lorsqu'un agent explore, il impacte directement et de façon irréversible la situation courante : l'exploration n'est pas libre. Ainsi chaque erreur d'exploration peut détruire des motifs fragiles en cours d'élaboration.

Pour le problème que nous traitons, les modèles agents ne tolèrent pas de contradiction d'où un antagonisme entre exploration et l'exploitation. On pourrait envisager de permettre le maintien de situations contradictoires en parallèle par la multiplication des espaces de marquage. Ainsi, une exploration réalisée dans un espace de marquage ne serait pas confrontée à l'état des autres espaces. Une stratégie très intéressante serait alors de partager les différentes connaissances accumulées dans les différents espaces de marquage. Pour cela, la gestion des renforcements ou des diversifications entre espaces de marquage nous confronterait de nouveau à un problème d'exploration et d'exploitation. Ce problème peut donc se retrouver à différents niveaux, le reporter à un niveau supérieur n'est pas la seule alternative disponible.

En effet, il est possible de faire cohabiter des activités d'exploration et d'exploitation sur un même espace de marquage pour le problème de partage de ressources critiques : ceci est le cas pour des approches d'optimisation par colonies de fourmis inspirées des comportements émergents que l'on constate chez les insectes sociaux pour l'activité de fouragement. Dans [Dorigo 97] et [Solnon 06], des graphes sont des supports de résolution qui mémorisent un marquage phéromonal sur une seule dimension. Les influences de chacun des marquages entrent en interaction par la rétroaction entre l'activité (simulation de parcours d'agents) et l'état de l'environnement. Ces approches ne bénéficient cependant pas d'un processus de résolution décentralisé⁹ ce qui restreint considérablement leurs capacités de modularité, d'ouverture et de robustesse.

Constater que les activités d'exploration et d'exploitation peuvent collaborer au sein d'un même espace de marquage nous encourage à approfondir l'expression de cette relation dans ce contexte, pour le partage de ressources critiques.

Nous représentons sur la figure 4.1 un modèle simplifié de la progression itérative des systèmes multi-agents en cours de construction de solution, celle-ci nous permet d'illustrer les améliorations que

⁹Pour les approches d'optimisation par colonies de fourmis, le retour perçu du système sur son activité (*feedback* du système) est calculé par un processus global centralisé

nous pourrions apporter à la corrélation exploration/exploitation dans le cadre du partage de ressources critiques. Les systèmes multi-agents évoluent itérativement et convergent vers une situation particulière associée à la solution du problème traitée. Pour cette illustration, nous représentons sur un axe la distance entre l'état courant du système et la solution recherchée comme étant la quantité minimale d'activité à réaliser pour rejoindre une solution. À chaque état de progression, le système multi-agents réalise des choix qui lui permettent de progresser vers la solution ou qui le pénalisent. Pour cette représentation, nous simplifions les probabilités de changement d'état (par exemple $p(d_4 \rightarrow d_3)$ n'est pas constant pour une distance à la solution, mais dépend de la configuration précise dans laquelle se trouve le système).

À chaque étape de progression de la situation courante, le système multi-agents peut se rapprocher de la solution recherchée ou bien s'en éloigner. Une probabilité permet de définir la possibilité d'améliorer la situation courante, un chemin probabiliste permet de définir la quantité de dégradation enclenchée par le système en cas de mauvaise décision.

La capacité à prendre les bonnes décisions dépend de la capacité du système à percevoir le problème qu'il aborde (rapport local/global, perception des caractéristiques complexes, etc.). Le risque de dégradation dépend du problème traité. Pour le problème de partage de ressources critiques, plus les contraintes sont nombreuses et sévères, plus elles propagent les pénalités lorsque le système commet une erreur. L'un comme l'autre de ces facteurs est en relation avec le problème applicatif, ce qui peut justifier que pour certains types de problème on ne rencontre pas de suractivité agent.

Nous souhaitons améliorer la capacité de perception du système par une nouvelle représentation des problèmes. Cependant, comment bénéficier efficacement d'une exploration qui peut être particulièrement préjudiciable pour le problème de partage de ressources critiques ? Comment minimiser les pénalités de dégradation ?

Ces pénalités sont propagées par les contraintes. En autorisant des états provisoirement incohérents, nous pourrions dégrader une solution progressivement et non radicalement. Nous avons pu constater avec notre premier modèle que respecter l'ensemble des contraintes au cours du processus d'évolution itératif dissipe toute la flexibilité du système. Le nouveau système que nous cherchons à élaborer prendra l'initiative de sa propre activité (nous n'utiliserons plus de seuils de déclenchement artificiels propres au deuxième modèle). Ainsi, à chaque étape de dégradation, il pourra rapidement percevoir le chemin à suivre pour se diriger vers la bonne solution ce qui minimisera la longueur du chemin de dégradation.

Au cours de notre étude, nous pourrions analyser la notion de perception à un niveau supérieur : perceptions entre motifs. Les motifs locaux sont en effet en interactions ce qui peut accroître leurs renforcements ou leurs affaiblissements dans un contexte plus global. Ainsi, un motif local pertinent globalement sera soutenu par son environnement en cas d'exploration infructueuse ce qui augmentera ses chances de survie. Ce type d'interaction entre motifs peut aussi se reconnaître pour les algorithmes de colonies de fourmis. Cependant, dans cette approche, le processus de perception est centralisé. En choisissant un processus de perception décentralisé, nous accroissons la capacité de modularité, d'ouverture et de robustesse du système que nous proposons.

4.3 Un nouveau processus de négociation

CESNA (*Complex Exchange between Stigmergic Negotiating Agents*) est une approche qui a pour objectif d'exprimer les caractéristiques complexes de problèmes au sein d'un processus de résolution.

Après avoir constaté qu'une bonne communication de ces caractéristiques complexes s'exprimerait plus facilement par une expression commune des différentes catégories de contraintes qui peuvent être rencontrées, nous proposons de représenter le problème de satisfaction distribué par un réseau. Ce réseau permet à des agents de connaître localement leurs capacités et leurs besoins (figure 4.2) et de représenter

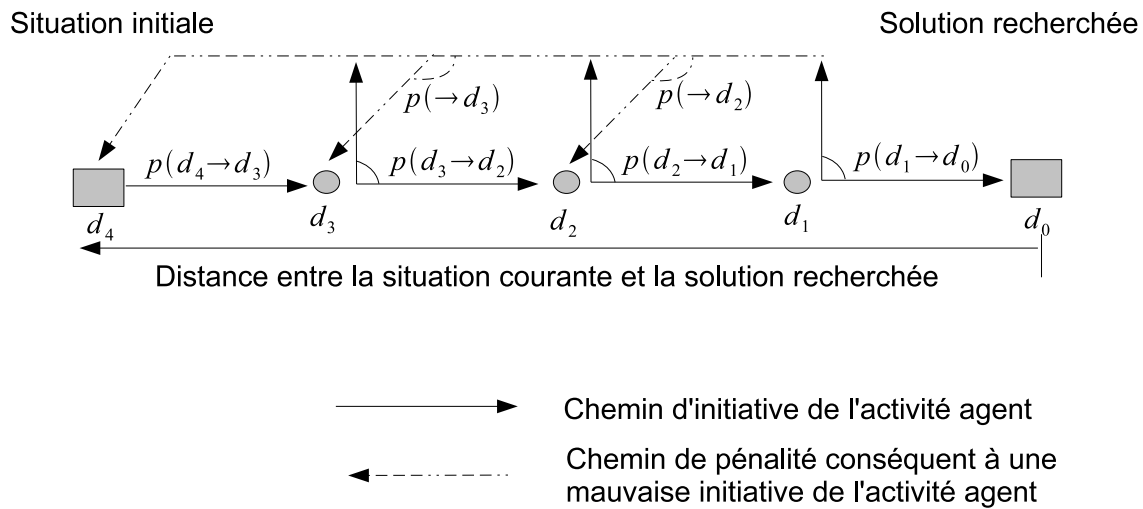


FIG. 4.1 – Chemin d’activité des systèmes multi-agents

ainsi fidèlement les problèmes traités. Suivant le principe d’ÉcoRésolution, les agents cherchent à se satisfaire tout en considérant les contraintes de leur environnement.

Suivant cette représentation (figure 4.2), chaque agent est connecté à un réseau de négociation par l’intermédiaire d’un arbre qui supporte des relations logiques (capacités de l’agent et conditions de satisfaction de l’agent). Les agents entrent en interactions par les différents contrats associés aux feuilles de leurs arbres respectifs. Du point de vue local d’un agent, l’environnement local est constitué par l’ensemble des contrats potentiels par lesquels l’agent est connecté au réseau. Les agents ainsi situés, perçoivent leur environnement local et négocient la validation des contrats qui peuvent les satisfaire par marquage de leur environnement. Ce nouveau processus de négociation est décrit en détail dans la partie qui suit.

À la différence des modèles situés précédents qui ne différenciaient pas les informations véhiculées dans le système en tant qu’actes de communication et l’activité exercée sur la configuration située spatialement courante, ce nouveau modèle procède différemment : la communication stigmergique est réalisée sur un espace abstrait propice à représenter la complexité des relations. L’enjeu de la négociation consiste à élaborer un accord global entre tous les agents afin que chaque agent soit satisfait. C’est seulement une fois cet accord global atteint que la solution est concrétisée.

Nous verrons que les mécanismes d’exploration et d’exploitation s’expriment plus facilement suivant cette représentation abstraite, nous décrirons celle-ci comme propice à représenter la complexité des problèmes du niveau local au niveau global : nous observerons des informations complexes la parcourir.

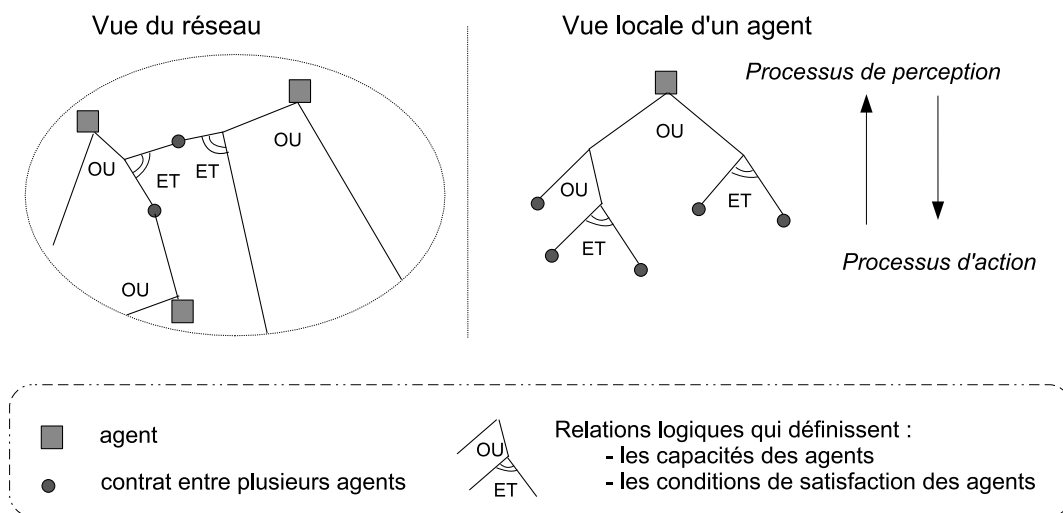


FIG. 4.2 – Représentation des problèmes par réseaux de négociation

Troisième partie

CESNA

CESNA

Complex Exchanges between Stigmergic Negotiating Agents

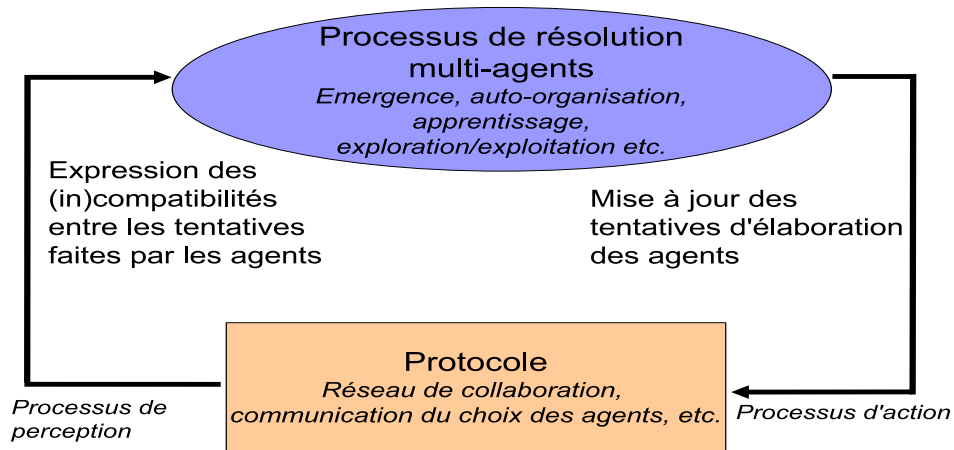


FIG. 0.3 – Vue générale de CESNA

Le processus de résolution CESNA bénéficie d'un protocole de communication supporté par un réseau (figure 0.3). On distingue d'une part un protocole de communication qui permet aux agents de communiquer/percevoir leurs intentions et leurs accords/désaccords, d'autre part un processus de résolution qui met en oeuvre différentes propriétés d'auto-organisation des systèmes multi-agents (émergence, rapport local/global, exploration v.s. exploitation, etc.).

Le réseau est un environnement de négociation qui exprime les relations entre les différents agents et fournit un espace de marquage. Suivant cette représentation, les combinaisons de validation des différentes relations du réseau représentent l'espace de recherche pour le problème traité.

Nous décrivons formellement dans le chapitre 1 le mode de représentation des problèmes de partage de ressources ainsi que les comportements de négociation qui sont mis en oeuvre avec CESNA. Nous illustrons l'utilisation de CESNA sur deux applications et voyons des premiers éléments de méthodes pour l'élaboration de réseaux de comportements de négociation dans le chapitre 2. Nous mesurons et analysons les performances et le fonctionnement de CESNA dans le chapitre qui suit. Les travaux que nous présentons ont été l'objet des articles [Armetta 06b] et [Armetta 06a].

1

Description de l'approche CESNA

1.1 Expression du problème de partage de ressources critiques par un réseau

Suivant notre modèle, le problème général est représenté par un réseau de négociation dans lequel chaque agent intervient au travers d'une sous-partie du réseau, incluant les contrats auxquels il participe. Chaque agent a pour objectif de valider un sous-ensemble de ses contrats qui lui apporte satisfaction, tout en respectant les objectifs des agents avec lesquels il est en concurrence.

Plus formellement, on définit le réseau comme un graphe $G = \langle N, R \rangle$ sachant que $N = B \cup C$. Soit $A = \{a_1, a_2, a_3, \dots, a_n\}$ un ensemble d'agents. À chaque agent est associé un noeud de l'ensemble B , nous confondons chaque agent avec son noeud associé pour la suite du mémoire.

- Soit $C = \{c_1, c_2, c_3, \dots, c_m\}$ un ensemble de noeuds *Contrat*.
- Soit $B = \{B^{ou} \cup B^{et} \cup \dots\} = \{b_1, b_2, b_3, \dots, b_n\}$ un ensemble de noeuds de connexion qui expriment des relations logiques. Ils permettent d'exprimer les conditions de satisfaction des agents par rapport aux contrats disponibles.
- Soit $R = \{r_1, r_2, r_3, \dots, r_p\}$ un ensemble d'arcs entre les noeuds du réseau. On note $suc(w_i) = \{y | (w_i, y) \in R\}$ l'ensemble des successeurs du noeud w_i , et $pred(w_i) = \{x | (x, w_i) \in R\}$ l'ensemble des prédécesseurs du noeud w_i .

Ce réseau respecte les règles suivantes :

- $\forall a \in A, pred(a) = \emptyset$
- $\forall b \in B, suc(b) \in \{B \cup C\}$
- $\forall c \in C, suc(c) = \emptyset, pred(c) \in B$

Les contrats et les agents sont reliés de la manière suivante :

- L'ensemble des contrats associés à l'agent a est noté : $C(a), a \in A, C(a) \subset C$.
- L'ensemble des agents associés au contrat c est noté : $A(c), c \in C, A(c) \in A$. Un contrat est associé à un agent s'il lui est connecté par des noeuds de l'ensemble B .
- L'état de validation d'un contrat pour un agent est noté : $\varphi(c, a), c \in C, a \in A, \varphi(c, a) \in \{vrai, faux\}$. L'état de validation d'un contrat est noté $\varphi(c) = \bigwedge_{a_i \in A(c)} \varphi(c, a_i) = (\forall a_i \in A(c), \varphi(c, a_i) = vrai), c \in C, \varphi(c) \in \{vrai, faux\}$.

L'ensemble du réseau exprime le problème dans sa globalité :

- Un ensemble d'états accessibles : suivant notre modèle, l'espace de recherche est représenté par l'ensemble des combinaisons d'état de validation des différents contrats du réseau.
- Un ensemble de contraintes à respecter : ces contraintes sont exprimées par des relations logiques de l'ensemble B . Elles permettent de calculer la satisfaction d'un agent, en fonction de l'état de validation de ces différents contrats suivant la fonction 2 *getSatisfaction*.

Fonction 1 $Liste \subset C$ getContrats($b \in B$)

```

Liste retour  $\leftarrow$  listeVide()
si  $b \in C$  alors
    retour  $\leftarrow$  retour +  $b$ 
sinon
    pour tous  $w \in suc(b)$  faire
        retour  $\leftarrow$  retour + getContrats( $w$ )
    fin pour
fin si
retourner retour

```

Fonction 2 $\{vrai, faux\}$ getSatisfaction($n \in N$)

```

si  $n \in C$  alors
     $c \leftarrow n$ 
    retourner  $(\nexists a \in A(c) | \varphi(c, a) = faux)$ 
sinon si  $n \in B^{et}$  alors
    retourner  $(\nexists s \in suc(n) | getSatisfaction(s) = faux)$ 
sinon si  $n \in B^{ou}$  alors
    retourner  $(\exists s \in suc(n) | getSatisfaction(s) = vrai)$ 
sinon si  $n \in N^{\dots}$  alors
    ... {comportement spécifique pour le noeud de connexion}
fin si

```

Suivant notre modèle, nous laissons libre le choix de représentation pour les agents (tâche, ressource, etc.), ainsi que le type des contrats représentés (échange de ressource, mouvement coordonné, changement cohérent, etc.). La structure du réseau peut être étendue afin d'exprimer des contraintes spécifiques entre agents, propres à un problème applicatif.

Nous illustrons la capacité du réseau décrit à représenter les relations complexes entre agents sur la figure 1.1.

Pour ce problème, on représente à travers un réseau différentes positions d'intégration sur un environnement spatial que peut envisager un agent. De plus, on représente les aménagements consécutifs à chaque position envisageable, en effet la réorganisation est étudiée localement mais a des incidences plus globales conséquentes à la complexité du problème traité. Dans cette situation, l'agent devra alors négocier afin de percevoir à travers le réseau les implications de ses choix et agir ainsi en pertinence avec son environnement.

Le réseau proposé permet d'exprimer les situations de satisfaction possibles pour un agent. Chaque agent doit valider un sous-ensemble de ses contrats afin de trouver une solution globale, c'est à dire qu'il faut pour chaque agent définir $C_{valid} \subset C | \{\forall c \in C_{valid}, \forall a_i \in A(c), \varphi(c, a_i) = vrai\}$, $\{\forall a \in A, getSatisfaction(a) = vrai\}$.

Dans le cadre de la négociation spatiale, l'agent cherchant à se placer négocie afin d'obtenir un ensemble d'accord qui lui permettent de s'insérer localement, et qui sont cohérent dans un contexte global de négociation (figure 1.2).

1.2 Comportement de négociation

Le système évolue en suivant la fonction 3. On définit une période d'activité comme une procédure de vie perception/action exécutée pour chaque agent dans un ordre aléatoire. À chaque période d'activité,

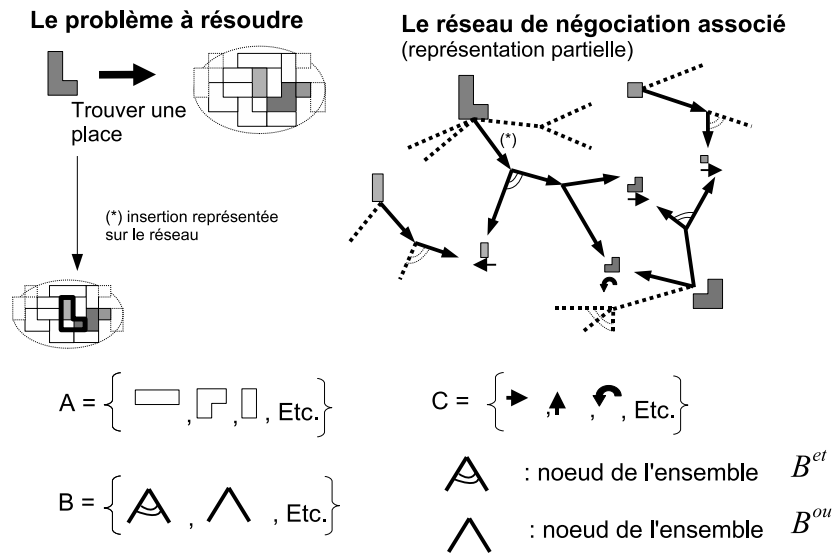


FIG. 1.1 – Illustration : un réseau de négociation spatiale

Processus de calcul de satisfaction : *getSatisfaction*

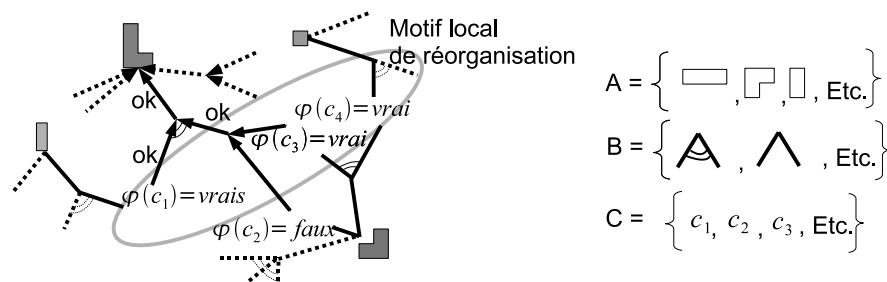


FIG. 1.2 – Illustration : satisfaction des agents

chacun des agents perçoit son environnement puis influence les autres agents en communiquant son choix parmi les différents contrats qui lui sont associés (figure 1.3). Chaque agent profite d'une perception par stigmergie de son environnement : quels sont les contrats les plus prometteurs ? Ensuite, chacun des agents fait un choix réactif : il sélectionne un sous-ensemble de contrats pouvant le satisfaire et marque son choix sur son environnement.

1.2.1 Perception de l'environnement

Chaque agent perçoit et communique via l'environnement. On associe à chaque noeud du réseau une valeur $\lambda(n|n \in N) \in \mathbb{R}$. Pour un contrat, cette valeur exprime le taux d'attraction/répulsion, elle est initialisée à zéro. Pour les noeuds de connexion, elle permet une mémorisation de la perception, elle est mise à jour par la fonction λ *perçoitEnvironnement*.

Sur la figure 1.4, la perception permet à l'agent d'estimer la facilité d'intégration sur chacune des positions qu'il envisage. On remarque que chacun des noeuds de l'ensemble B^{ou} propagent la valeur maximale de ses successeurs, car l'agent considère alors la piste la plus prometteuse parmi les choix pro-

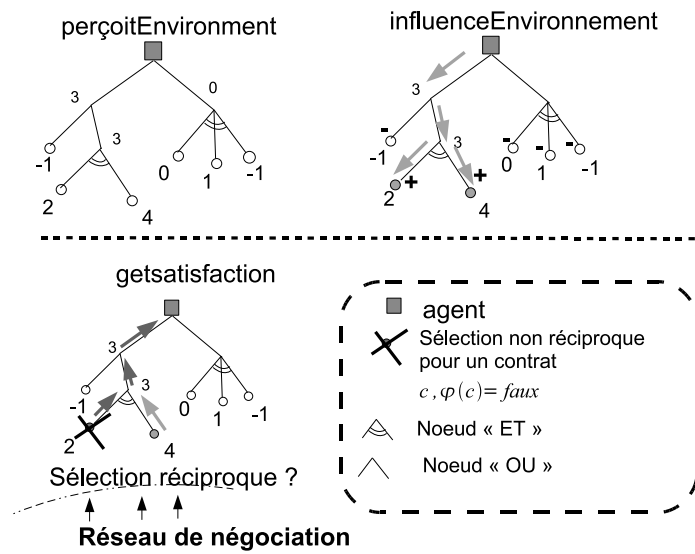


FIG. 1.3 – processus de perception/action pour un agent

Fonction 3 Vie des agents

tant que $\exists a \in A | getSatisfaction(a) = faux$ **faire**
pour tous $a \in A$ choisi aléatoirement **faire**
perçoitEnvironnement(a)
influenceEnvironnement(a, a, vrai)
fin pour
fin tant que

posés. Pour les noeuds de l'ensemble B^{et} , il est par contre nécessaire de considérer l'état de l'ensemble des successeurs.

1.2.2 Influence sur l'environnement

Dans la fonction 5 *influenceEnvironnement*, chaque agent considère les valeurs perçues lors de la procédure précédente pour choisir les contrats qui pourraient le satisfaire. La fonction *influence(c ∈ C, state ∈ {vrai, faux})*, détaillée par la suite, marque le choix de l'agent sur chaque valeur λ de contrat. Si l'agent sélectionne le contrat, il en augmente la valeur λ afin d'attirer l'attention des autres agents associés avec ce contrat. Au contraire, lorsque l'agent rejette un contrat, il en diminue sa valeur λ. Les différents agents en relation avec cet agent perçoivent l'acte de communication lors de leur procédure de vie (perception/action) suivante.

La fonction *expressionExploration* permet l'équilibre exploration/exploitation pour le système. Elle permet à un agent d'initier l'essai de nouveaux contrats suivant un taux d'exploration qui influence les performances du système (voir section 3.1).

La figure 1.5 restitue graphiquement l'expression de l'influence de l'agent cherchant à se positionner sur son environnement, conformément à ce que notifie la fonction 5 *influenceEnvironnement*.

Fonction 4 \mathbb{R} `perçoitEnvironnement`($n \in N$)

```

si  $n \in C$  alors
  retourner  $\lambda(n)$ 
sinon si  $n \in B$  alors
  Liste buffer  $\leftarrow$  listeVide()
  pour tous  $s \in \text{suc}(n)$  faire
    buffer  $\leftarrow$  buffer + perçoitEnvironnement( $s$ )
  fin pour
  si  $n \in B^{et}$  alors
     $\lambda(n) \leftarrow$  moyenne(buffer) {retourne la valeur moyenne des successeurs}
  sinon si  $n \in B^{ou}$  alors
     $\lambda(n) \leftarrow$  maximum(buffer) {retourne la valeur maximale des successeurs}
  sinon si  $n \in B^{\dots}$  alors
    ... {comportement spécifique pour le noeud de connection}
  fin si
  retourner  $\lambda(n)$ 
fin si
  
```

Processus de perception : `perçoitEnvironnement`

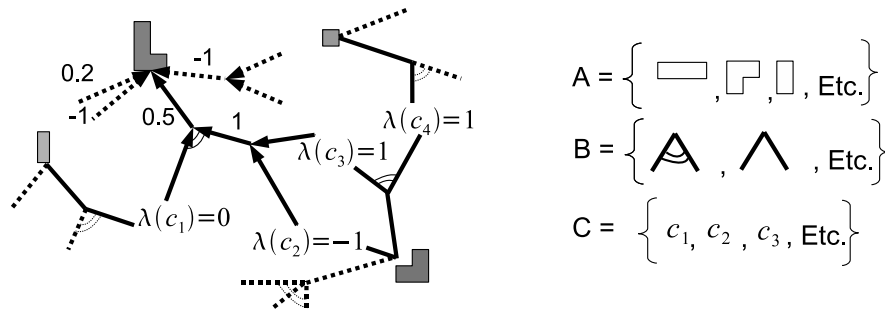


FIG. 1.4 – Illustration : perception de l'environnement

1.3 Examen des capacités de perception et d'action pour CESNA

Après avoir présenté le mode opératoire du protocole de communication établi avec CESNA, nous développons dans cette partie des enrichissements que nous avons jusque là envisagés. Ceux-ci dégagent des perspectives d'utilisation des capacités de communication pour CESNA.

Nous commençons par développer le type de relations que peuvent connaître les agents au sein du réseau (nous en avons jusque là envisagés qui se déclinent sur deux axes). Nous approfondissons ensuite les différents comportements agents supports de ces relations. Nous identifions des extensions suivant les trois fonctionnalités agent présentées dans la section précédente :

- influencer son environnement (fonction 5)
- percevoir l'influence de son environnement (fonction 4)
- calculer sa satisfaction (fonction 3)

Chacun des noeuds CESNA est une composition hybride de différentes variations de chacune de ces fonctionnalités.

Fonction 5 $influenceEnvironnement(n \in N, a \in A, state \in \{vrai, faux\})$

```
si  $n \in C$  alors
   $c \leftarrow n$ 
   $\varphi(c, a) \leftarrow state$ 
   $\lambda(c) \leftarrow influence(c, state)$  {la fonction influence communique l'intérêt de l'agent pour le contrat, elle est
  étudiée dans la section qui suit}
sinon si  $n \in B^{et}$  alors
  pour tous  $s \in suc(n)$  faire
     $influenceEnvironnement(s, a, state)$ 
  fin pour
sinon si  $n \in B^{ou}$  alors
  si  $state = vrai$  alors
     $sucMax = Max_{s \in suc(n)} \lambda(s)$  {en cas d'égalité, Max retourne une choix aléatoire parmi des objets
    en égalité}
     $sucMax \leftarrow expressionExploration(sucMax, getSuc(n))$  {expressionExploration peut changer
    sucMax pour un autre noeud}
     $influenceEnvironnement(sucMax, a, vrai)$ 
    pour tous  $s \in suc(n) | s \neq sucMax$  faire
       $influenceEnvironnement(sucMax, a, faux)$ 
    fin pour
  sinon
    pour tous  $s \in suc(n)$  faire
       $influenceEnvironnement(s, a, state)$ 
    fin pour
  fin si
sinon si  $n \in B^{..}$  alors
  ... {comportement spécifique pour le noeud de connection}
fin si
```

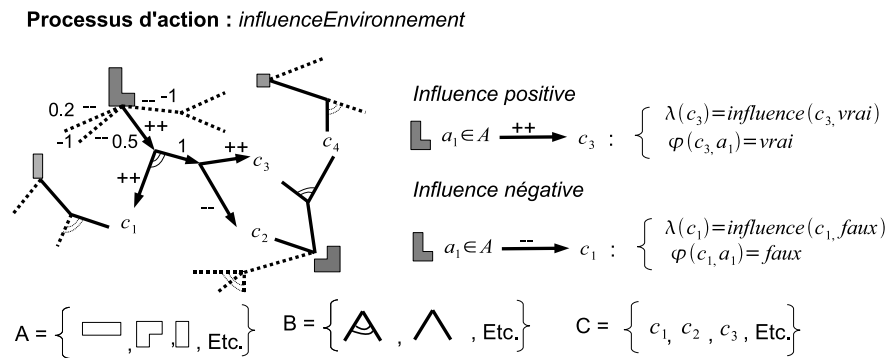


FIG. 1.5 – Illustration : influence sur l'environnement

Les capacités de perception et d'action que nous développons sont illustrées dans le chapitre suivant sur différents problèmes applicatifs. Ces illustrations nous permettent d'entamer une discussion sur des éléments méthodologiques visant à employer les capacités de perception et d'action de CESNA avec efficacité.

1.3.1 Relations entre agents

Les relations entre agents permettent d'exprimer deux types de négociation : précise ou diffuse. Après avoir justifié l'intérêt de cette diversité, nous proposons un mode de connexion représentatif pour chaque type de relation. Nous présentons ensuite le principe d'expression condensée de la communication entre les agents, celui-ci permet de réduire les informations véhiculées à leur essentiel ce qui, d'après nous, limite aussi les risques de confusion au sein des interactions sociales.

Négociation précise ou diffuse ?

Lorsqu'un agent entre en négociation précise avec un autre agent, il focalise son activité sur un contrat unique. Dans le cas d'une négociation diffuse, un agent influence d'autres agents et vise plusieurs contrats simultanément, nous allons voir que ce type de négociation requiert quelques adaptations des fonctions de perception, d'influence et de calcul de satisfaction. L'un ou l'autre de ces modes d'interaction a des avantages/inconvénients et sont à justifier dans un contexte donné.

Nous avons présenté avec la fonction 5 *influence*Environnement l'expression de l'influence par négociation précise d'un agent sur un noeud *contrat*. L'expression de cette influence est symétrique car chacun des agents impliqués transmet un signal par l'intermédiaire du dit contrat. Ainsi, chacun des agents doit donner son accord pour valider un contrat, ce qui est cohérent avec la notion d'engagement des deux intervenants.

Pour certaines configurations, il est intéressant d'exprimer des influences plus simplifiées entre agents (c'est le cas pour le problème d'ordonnancement étudié dans le chapitre suivant). La notion d'engagement peut être envisagée différemment avec ce mode de communication : un agent influence l'orientation des choix d'un autre agent, mais ne participe pas forcément à la fonction de satisfaction de celui-ci. Ceci permet à un agent d'exprimer par influence son intérêt pour différentes alternatives qui peuvent être contradictoires entre elles (partie droite de la figure 1.6). Il pourra ainsi percevoir en peu de périodes

d'activité¹⁰ les possibilités les plus pertinentes à envisager sur un spectre de possibilités potentiellement très large. La même opération envisagée sous l'axe d'une négociation spécifique entraîne un parcours itératif du spectre de possibilités ce qui ne permet pas à l'agent de saisir toutes les opportunités dans un contexte de négociation dynamique.

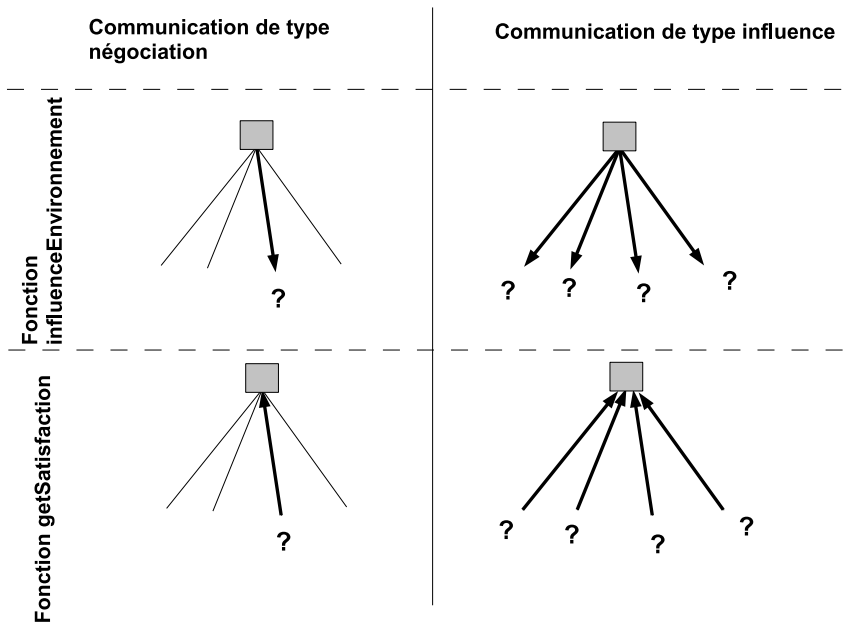


FIG. 1.6 – Différents types de communication

Les architectures de réseau sont elles-mêmes à adapter aux relations choisies pour les agents. Par commodité et par souci de performance, l'influence entre agents n'est pas nécessairement exprimée par un contrat mais peut transiter entre des noeuds de l'ensemble B associés à des agents différents. Nous définissons pour cela différents sous-ensembles de l'ensemble B :

- $B^{infl_{OUT}} \subset B$: un ensemble de noeuds qui diffusent leur influence en vue d'adapter le comportement de noeuds de l'ensemble $B^{infl_{IN}}$ à leurs intérêts individuels.
- $B^{infl_{IN}} \subset B$: un ensemble de noeuds qui perçoivent l'influence de noeuds de l'ensemble $B^{infl_{OUT}}$.

Ces ensembles de noeuds de connexion particuliers pourront porter des fonctionnalités de perception et d'influence de type 'ou', 'et', etc. Ils permettent des négociations diffuses, mais peuvent aussi être utilisés dans le cadre de négociation précises suivant les besoins du concepteur.

Nous voyons sur la figure 1.7 comment une connexion directe entre agents permet ainsi d'exprimer l'influence d'une négociation diffuse d'un agent sur l'autre facilement tout en modérant la taille du réseau. Nous représentons sur cette même figure l'expression d'une connexion précise entre agents.

Nous illustrons dans la partie 1.3.2 quelques possibilités d'implémentation offertes par ces deux perspectives.

¹⁰On définit une période d'activité comme une procédure de vie perception/action exécutée pour chaque agent dans un ordre aléatoire (cf. fonction 3 *Vie des agents*). À chaque période d'activité, chacun des agents perçoit son environnement puis influence les autres agents en communiquant son choix parmi les différents contrats qui lui sont associés (figure 1.3).

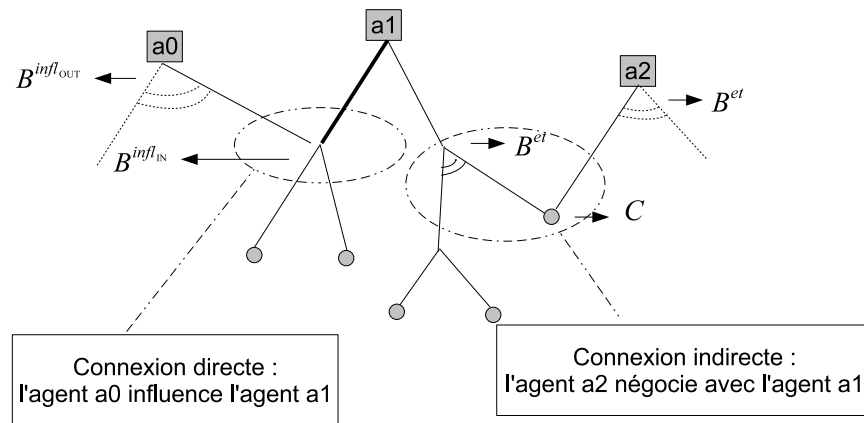


FIG. 1.7 – Connexions directes ou indirectes entre agents

Perception et action condensée

Les problèmes de partage de ressources critiques mettent en jeu des variables dont les espaces de valeurs accessibles peuvent être de grande taille. Ceux-ci sont manipulés facilement par des approches symboliques telles que l'approche par propagation de contraintes qui permet de réduire très efficacement l'ensemble des possibilités à étudier. Ces approches mettent en jeu des représentations symboliques telles que : 'a est plus petit que b', plutôt qu'une énumération de l'ensemble des possibilités telle qu'elle pourrait être envisagée sous CESNA : 'a = 1, ou a = 2, ou ...'. Lorsque les domaines des valeurs manipulés sont très grands, les représentations sous CESNA tendent à être du même ordre de grandeur. Ces difficultés se retrouvent dans les approches d'optimisation par colonies de fourmis dont le processus de résolution est supporté par un graphe.

Lorsque les réseaux sont de grande taille, les algorithmes de perception et d'action exécutés par les agents prennent plus de temps et occupent plus de mémoire. Pour remédier à ce problème, on peut envisager d'appliquer le processus de résolution à un graphe dynamique dont la structure pourrait évoluer. En effet, de cette façon, l'exploration et l'exploitation propres aux approches réactives pourraient s'exécuter sur un niveau supplémentaire. Par ailleurs, dans le cadre d'applications industrielles, il n'est pas nécessaire de représenter l'intégralité de l'espace de recherche lorsque l'on élabore une solution satisfaisante en milieu perturbé. Ainsi, les domaines de valeurs associés aux différentes variables pourraient être artificiellement réduits.

Dans cette partie, nous étudions les possibilités de représentation de l'intégralité de l'espace de recherche par un réseau de taille minimale. D'une part, des réseaux de plus petite taille réduisent la complexité des opérations de perception et d'action des agents, d'autre part ceux-ci permettent d'éviter des renégociations inutiles causées par des redondances au sein du réseau.

Le principe que nous appliquons est de permettre à chaque agent de percevoir et d'agir seulement sur des orientations (qui peuvent être symboliques) qui les concernent directement. Par exemple, un agent a négocie avec un agent b un transport de marchandises. L'agent a n'a pas besoin de percevoir le type de véhicule que l'agent b choisit. Afin d'implémenter une communication pertinente, nous recherchons des moyens d'exprimer les influences entre agents à différents niveaux d'abstraction au sein du réseau.

Ainsi, nous proposons de décomposer les domaines de valeurs des problèmes abordés pour que les influences puissent s'exprimer avec pertinence et sans égarement. Nous voyons sur la figure 1.8 que ces décompositions peuvent s'exprimer hiérarchiquement ou bien transversalement. Nous les utilisons dans le chapitre suivant pour le problème d'ordonnement et de coloration de graphes.

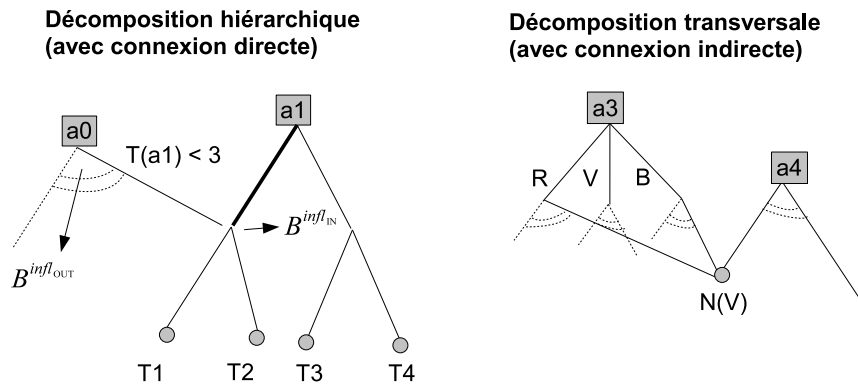


FIG. 1.8 – Décomposition des domaines de valeurs

Sur la partie gauche de la figure, nous voyons que l'agent a_0 cherche à satisfaire $T(a_1) < 3$, ce qui est vérifié pour les contrats T_1 et T_2 . L'agent a_0 peut alors se connecter directement sur l'arbre de négociation associé à a_1 qui rassemble les alternatives T_1 et T_2 . Quelques fois, la décomposition n'est pas aussi triviale et s'exprime alors volontairement de façon transversale comme nous le voyons sur la partie droite de la figure 1.8.

1.3.2 Influencer, percevoir et calculer sa satisfaction

Comme nous l'avons décrit, la négociation entre agents intervient de façon précise ou diffuse. Nous illustrons la mise en oeuvre de ces deux types de relation par les fonctions qui nous ont apportées satisfaction pour les problèmes applicatifs décrits dans le chapitre qui suit.

Illustration : Négociation précise

La fonction 6 *influence* qui est utilisée dans la fonction précédente 5 *influenceEnvironnement* permet à un agent source d'influencer le comportement futur d'un agent cible en attirant/repoussant son attention. Celle-ci donne de très bons résultats dans le cadre d'une négociation précise.

Avec cette fonction d'influence, nous proposons de restreindre l'ensemble des valeurs autorisées d'attraction/répulsion pour un contrat : $\forall c \in C, \lambda(c) \in \{-1, 0, 1\}$. Ainsi, chaque contrat qui est utilisé comme support de communication par les agents émet un signal négatif, neutre ou bien positif. Suivant la fonction 6, les transitions entre ces valeurs sont instantanées ce qui permet au système de changer facilement l'orientation de la situation courante. Nous avons implémenté d'autres fonctions d'influence pour lesquelles les valeurs accessibles n'étaient pas restreintes, et pour lesquelles les transitions s'inscrivaient plus dans la durée par un renforcement positif/négatif plus progressif. Nous n'avons pas obtenu de résultats satisfaisants avec celles-ci, nous avons donc envisagé qu'il ne pouvait y avoir de renforcement sous cette forme, que celui-ci s'exprimerait à d'autres niveaux au sein du système (voir l'analyse du chapitre 3). Ce constat nous a permis d'accentuer la nervosité du processus de résolution, et mené jusqu'à la fonction d'influence qui nous a fournies des résultats empiriquement satisfaisants pour chacun des problèmes que nous avons traités (ceux-ci sont décrits par la suite).

Fonction 6 $\{-1, 0, 1\}$ *influence*($c \in C, state \in vrai, faux$)

Précondition $\lambda(c) \in \{-1, 0, 1\}$

retour \leftarrow 1

si *state* **alors**

si $\lambda(c) = -1$ && *Random*(1) ≥ 0.5 **alors**

 { *Random*(1) retourne aléatoirement un nombre entre 0 et 1 }

 retour \leftarrow 0

fin si

sinon

 retour \leftarrow -1

fin si

retourner retour

Illustration : Négociation diffuse

Un agent peut émettre une recommandation sans entrer dans un jeu de négociation. Dans cette situation nous proposons une connexion directe entre agents telle qu'elle est présentée sur la figure 1.7.

Pour le mécanisme que nous avons implémenté, contrairement à la fonction 6, l'agent qui influence ne considère pas l'état d'attraction/répulsion courant du noeud pour calculer le poids de son influence, il émet une recommandation inconditionnelle.

Nous illustrons à travers les fonctions 7 et 8 la relation entre un(ou plusieurs) noeud(s) de l'ensemble $B^{infl_{OUT}}$ et un noeud de l'ensemble $B^{infl_{IN}}$.

On définit $B(a|a \in A) \subset B$, une fonction qui retourne l'ensemble des noeuds associés à un agent (ces noeuds sont la propriété de l'agent). Les fonctionnalités d'influence et de perception exécutées sur le noeud de l'ensemble $B^{infl_{IN}}$ sont dépendantes du noeud d'appel, que l'on détermine par le paramètre $a \in A$ grâce à la fonction B^{11} .

D'une part, la fonction 7 *influenceEnvironnement* permet à un noeud de l'ensemble $B^{infl_{OUT}}$ d'influencer un noeud de l'ensemble $B^{infl_{IN}}$ par l'intermédiaire d'une variable de stockage $\delta(b)$. Cette influence est perçue au cours de la fonction 8 *perçoitEnvironnement* exécutée par l'agent propriétaire du noeud $B^{infl_{IN}}$.

D'autre part, la fonction 7 *influenceEnvironnement* permet à l'agent influencé de mémoriser son souhait par l'intermédiaire de la variable *memoState*. Celui-ci pourra être perçu par les différents agents qui l'influencent.

Enfin, la fonction 2 *getSatisfaction* connaît elle-même une adaptation dont nous énonçons le principe.

Les noeuds de l'ensemble $B^{infl_{OUT}}$ perçoivent, pour le calcul de satisfaction, l'état de satisfaction des noeuds influencés. D'autres alternatives peuvent être envisagées, telles la perception de l'exécution positive de la fonction *influenceEnvironnement* émise par l'agent influencé.

Par contre, les agents influencés calculent de façon indépendante leur satisfaction (ils exécutent leur fonctionnalité de satisfaction courante). Pour ces agents, le calcul de satisfaction n'est donc pas modifié.

¹¹Pour la fonction 8 *perçoitEnvironnement*, on ajoute un paramètre $a \in A$ à la fonction de perception courante, ce qui permet d'identifier l'agent appelant la fonction

Fonction 7 *influenceEnvironnement*($n \in N, a \in A, state \in \{vrai, faux\}$)

```
si  $n \in B^{infl_{IN}}$  alors
   $b \leftarrow n$ 
  si  $b \notin B(a)$  {l'agent  $a$  n'est pas propriétaire de  $b$ , on marque ici l'influence d'un noeud de l'ensemble  $B^{infl_{OUT}}$ } alors
    si  $state$  alors
       $\delta(b) \leftarrow 1$ 
      {l'agent communique son influence par l'intermédiaire de la variable  $\delta$ }
    fin si
  sinon
     $memoState \leftarrow state$  {mémorisation de l'intention de l'agent propriétaire du noeud, utile à la perception enclenchée par des noeuds de l'ensemble  $B^{infl_{OUT}}$  décrite à la suite}
    ... {processus courant d'influence associé au noeud de l'ensemble  $B^{infl_{IN}}$ }
  fin si
fin si
```

Fonction 8 *perçoitEnvironnement*($n \in N, a \in A$)

```
si  $n \in B^{infl_{IN}}$  alors
   $b = n$ 
  si  $b \notin B(a)$  alors
    si  $memoState$  alors
       $\lambda(b) \leftarrow 1$  {l'agent propriétaire du noeud  $b$  exprime le fait qu'il a souhaité, lors de sa précédente influence, sélectionner la branche associée à  $b$ }
    sinon
       $\lambda(b) \leftarrow -1$ 
    fin si
  sinon
     $\lambda(b) \leftarrow \dots$ 
    {exécution de la fonctionnalité de perception standard}
     $\lambda(b) \leftarrow \lambda(b) + \delta(b)$ 
    {l'agent perçoit l'influence}
     $\delta(b) \leftarrow 0$ 
    {réinitialisation de  $\delta(b)$ }
  fin si
fin si
retourner  $\lambda(n)$ 
```

2

Architecture de réseaux et comportements associés pour des problèmes applicatifs

Au cours de nos expérimentations, nous avons élaboré des réseaux CESNA pour différents problèmes applicatifs. Dans ce chapitre, nous décrivons les architectures que nous avons sélectionnées pour le problème d'ordonnancement ainsi que pour le problème de coloration de graphes. L'objectif de notre étude est de montrer que CESNA peut, en représentant les relations complexes de problèmes différents, permettre leur résolution.

Nous avons pu constater dans le chapitre précédent que CESNA permet une multitude d'architectures de réseaux associables avec de nombreux comportements différents. Il est alors difficile de sélectionner la combinaison la plus pertinente. Pour chaque problème traité, nous avons élaboré différents réseaux pour arriver à ceux que nous présentons. Les systèmes multi-agents sont difficiles à élaborer car le comportement de différents éléments en interaction est difficile à prévoir et à interpréter. Il convient alors d'extraire de différentes constatations empiriques des hypothèses qui permettent d'améliorer notre capacité à les modéliser. Un travail très intéressant s'inscrit dans cette perspective [Parunak 05]. À partir de différents problèmes applicatifs, les auteurs étudient la vitesse de convergence de différents systèmes à base d'agents réactifs. Il reste cependant beaucoup d'avancées à réaliser dans le domaine pour pouvoir profiter pleinement de la technologie agent. En fin de chapitre, nous tirons quelques premiers éléments de méthode consécutifs à notre étude dans le cadre de l'élaboration de solutions applicatives CESNA.

2.1 CESNA Ordonnancement

Nous choisissons un problème d'ordonnancement défini ainsi :

- différents produits sont à fabriquer dans un temps limité
- chaque produit se compose d'un ensemble de tâches soumises à différentes contraintes chronologiques
- les ressources de production sont critiques, leur capacité à produire est limitée

Le problème choisi exprime les composantes complexes de la plupart des problèmes d'ordonnancement : d'une part les tâches sont en concurrence sur les mêmes machines, d'autres part elles subissent entre elles des contraintes chronologiques d'exécution. Le fait que les tâches partagent le même environnement a déjà été efficacement modélisé avec les systèmes multi-agents dans le cadre de travaux menés par [Ghédira 92] suivant une approche par ÉcoRésolution. Notre principale difficulté pour ce problème fût d'intégrer efficacement les contraintes chronologiques entre tâches pour le processus de résolution CESNA.

2.1.1 Modèle

Nous présentons sur la figure 2.1 une représentation du problème d'ordonnancement sous CESNA pour laquelle les agents instancient les tâches à réaliser. Nous pouvons lire le problème du point de vue d'un agent : chaque agent *Tâche* étudie les possibilités de placement sachant qu'un placement valide comprend l'autorisation de la ressource de production ainsi qu'une cohérence chronologique de placement avec d'autres agents *Tâche* . Le réseau ainsi représenté exprime fidèlement les différentes contraintes.

Le contact contraint des agents s'exerce entre les agents *Tâches* d'un même produit, ainsi qu'entre les agents *Tâche* partageant les mêmes ressources.

D'une part, lorsqu'un agent a sélectionne une position sur une ressource, il diffuse son influence et incite les agents qui le contraignent chronologiquement à choisir une position compatible.

Illustrons sur la figure 2.1 l'influence chronologique de l'agent $t2$ sur l'agent $t1$. L'influence est réalisée par inondation conformément au processus d'influence décrit dans la section 1.3.2 du chapitre précédent : $B^{ouInond} \subset B^{inflow}$. Nous voyons clairement l'intérêt de ce mode de communication par influence : l'agent $t2$ n'a pas besoin de connaître la position précise que choisit l'agent $t1$ mais l'incite à se positionner avant lui sur le planning des ressources de production.

Nous avons étudié le processus de négociation du point de vue de l'agent $t2$. Celui-ci perçoit similairement le résultat des influences exercées pour le respect des contraintes chronologiques par les agents $t1$ et $t3$ ce qui lui permet d'orienter ses choix tout en gardant une marge de manoeuvre sur la position précise qu'il souhaite choisir (dans ce cas, cette position dépend aussi de la perception réalisée sur l'ensemble des ressources). Dans cette situation, une négociation précise n'a pas d'intérêt et apporterait beaucoup de confusions au processus de résolution.

D'autre part, l'agent négocie sa position sur la ressource. Chaque ressource est gérée par différents noeuds de connexion spécifiques de l'ensemble B^{PR} , qui permettent le découpage de la capacité de production de la ressource dans le temps en parts de ressources. Ce type de noeud n'est pas associé à un agent tel que nous l'avons présenté car il n'a pas de processus propre de vie. Ce type de noeud ne fait que réagir aux sollicitations émises par les autres agents (fonctions 9 *influenceEnvironnement* , 10 *perçoitEnvironnement* et 11 *getSatisfaction*). Ce noeud répond une valeur positive par la fonction de perception *perçoitEnvironnement* lorsque la part de ressource qu'il représente est disponible à la production, une valeur négative dans le cas contraire. En agissant ainsi, le noeud se signale comme disponible/indisponible à son environnement. Les agents connectés peuvent alors réserver la portion de ressource par la fonction *influenceEnvironnement* , et tester que celle-ci est bien effective par la fonction *getSatisfaction* . Nous présentons un approfondissement de ce choix de modélisation de ressource dans la section 2.3.

Fonction 9 \mathbb{R} *perçoitEnvironnement*($n \in N, a \in A$)

```

si  $n \in B^{PR}$  alors
  si estLibre( $n$ ) alors
    retourner 1
  sinon
    retourner -1
  fin si
fin si

```

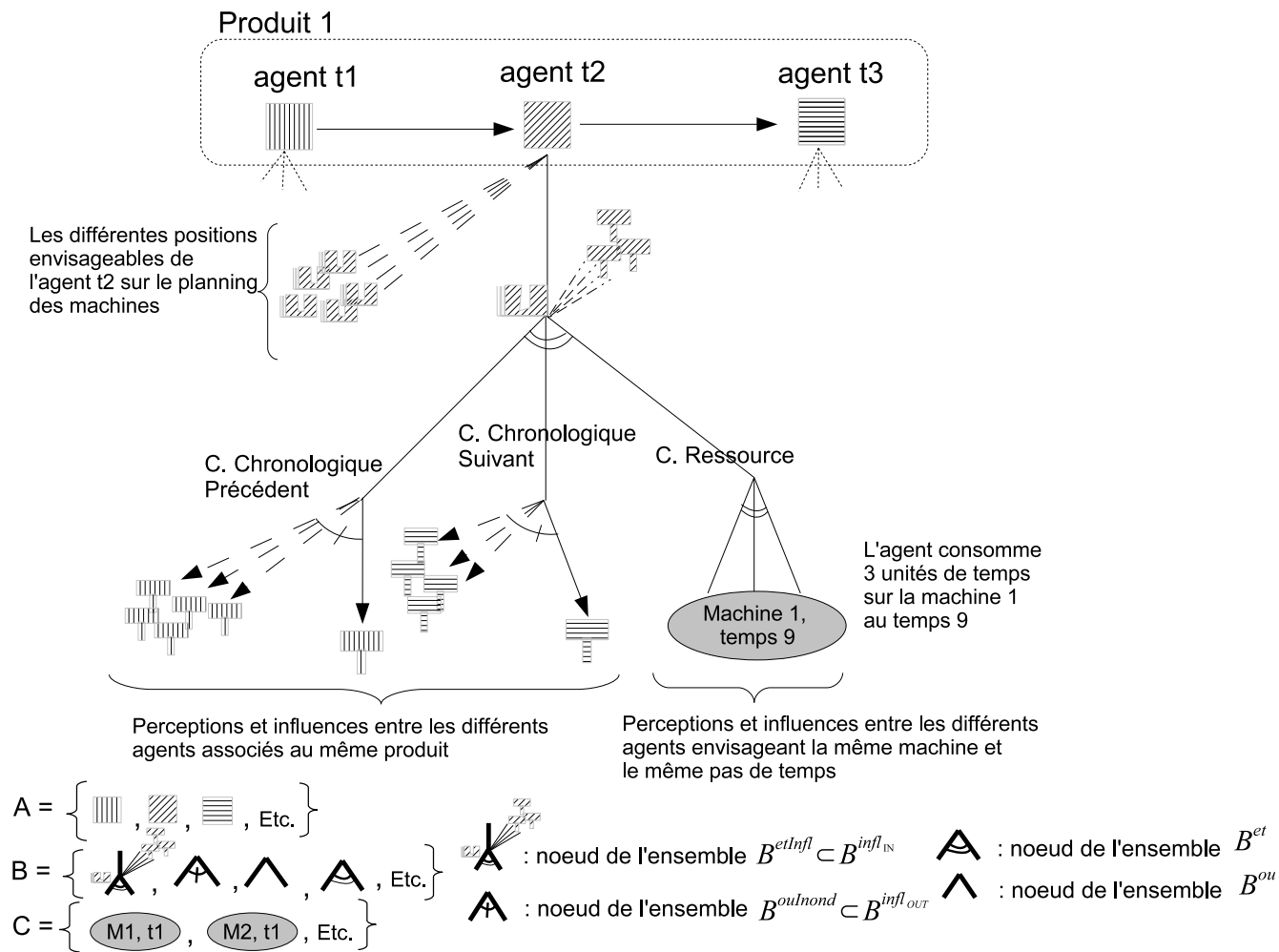


FIG. 2.1 – Réseau CESNA pour l'ordonnancement

Fonction 10 $influenceEnvironnement(n \in N, a \in A, state \in \{vrai, faux\})$

si $n \in B^{PR}$ **alors**

si $state$ **alors**

si $estLibre(n)$ **alors**

$estLibre(n) \leftarrow faux$

$agentAccord(n) \leftarrow a$

fin si

sinon

si $agentAccord(n) = a$ **alors**

$estLibre(n) \leftarrow vrai$

$agentAccord(n) \leftarrow null$

fin si

fin si

fin si

Fonction 11 $\{vrai, faux\}$ `getSatisfaction`($n \in N, a \in A$)

si $n \in B^{PR}$ **alors**
 retourner $agentAccord(n) = a$
fin si

2.1.2 Modèle condensé

Le modèle que nous venons de voir permet de représenter l'ensemble des combinaisons de placement, mais est très volumineux : pour chaque position, les noeuds assurant le respect des contraintes chronologiques sont connectés à l'ensemble des positions possibles des agents en relation. Un agent n'a en réalité pas la nécessité de percevoir la position précise des agents en relation, mais doit simplement percevoir si ceux-ci se placent adéquatement, avant ou après la position qu'il envisage. Il est donc dans cette situation pertinent d'utiliser une représentation hiérarchique condensée présentée dans la section 1.3.1 du chapitre précédent. Le modèle amélioré calque le modèle précédent tout en intégrant une décomposition hiérarchique des positions des agents. Sur la figure 2.2 nous voyons que le réseau représente toujours l'ensemble des combinaisons de placement mais réduit les capacités de perception et d'action des agents à leur essentiel. Cette décomposition est réalisée par subdivision récursive de l'espace temps, et reprend les principes de segmentation par hachage employé par de nombreux systèmes d'indexation en informatique.

2.2 CESNA Coloration de graphes

Le problème de coloration de graphes permet la modélisation d'une grande variété de problèmes applicatifs (certains problèmes d'ordonnancement, problèmes d'allocation de fréquence en téléphonie mobile, problèmes de sudoku, etc.). Il nous a semblé pertinent d'étudier ce problème qui représente fidèlement les problèmes de partage de ressources critiques.

Nous avons réalisé cette étude en collaboration avec Olivier Lefevre lors du stage qu'il a effectué au sein du LIRIS à l'occasion d'un master recherche. Le travail qu'Olivier a entamé se poursuit dans le cadre d'une nouvelle thèse CIFRE en collaboration avec la société OSLO. Celle-ci permet de renforcer l'activité de recherche portée sur l'approche CESNA.

Résoudre un problème de coloration de graphes consiste à affecter à chaque noeud d'un graphe une couleur différente de celles de ses voisins. Nous traitons le problème particulier de k coloration . Pour ce problème, on connaît le nombre de couleurs disponibles pour colorer le graphe.

Pour la modélisation que nous proposons, nous n'utilisons pas de connexion directe et de négociation diffuse ($\forall b \in B, |pred(b)| \leq 1$). Nous profitons par contre de la décomposition transversale que nous avons présentée sur la figure 1.8.

Nous voyons sur la figure 2.3 le réseau pour la coloration de graphes d'un point de vue agent. Voyons ensemble comment chaque agent est connecté à l'ensemble du réseau pour la coloration. Suivant notre modèle, chaque agent est associé à un noeud du graphe à colorer. Un agent sélectionne donc tout d'abord une couleur. Pour qu'il soit satisfait dans son choix, la couleur sélectionnée doit être cohérente avec l'environnement de l'agent, i.e., les différents voisins associés au graphe à colorer (conséquence externes du choix). Par ailleurs, l'agent communique son choix à son environnement (conséquences internes du choix). Ainsi, un agent choisissant la couleur B (bleue) communique le fait qu'il ne porte pas le R(rouge) : N(R), ni le V(vert) : N(V). Cette décomposition transversale permet aux voisins de l'agent de vérifier qu'ils ne sont pas en conflit, sans pour autant connaître ni intervenir sur l'état précis du choix de coloration réalisé par l'agent.

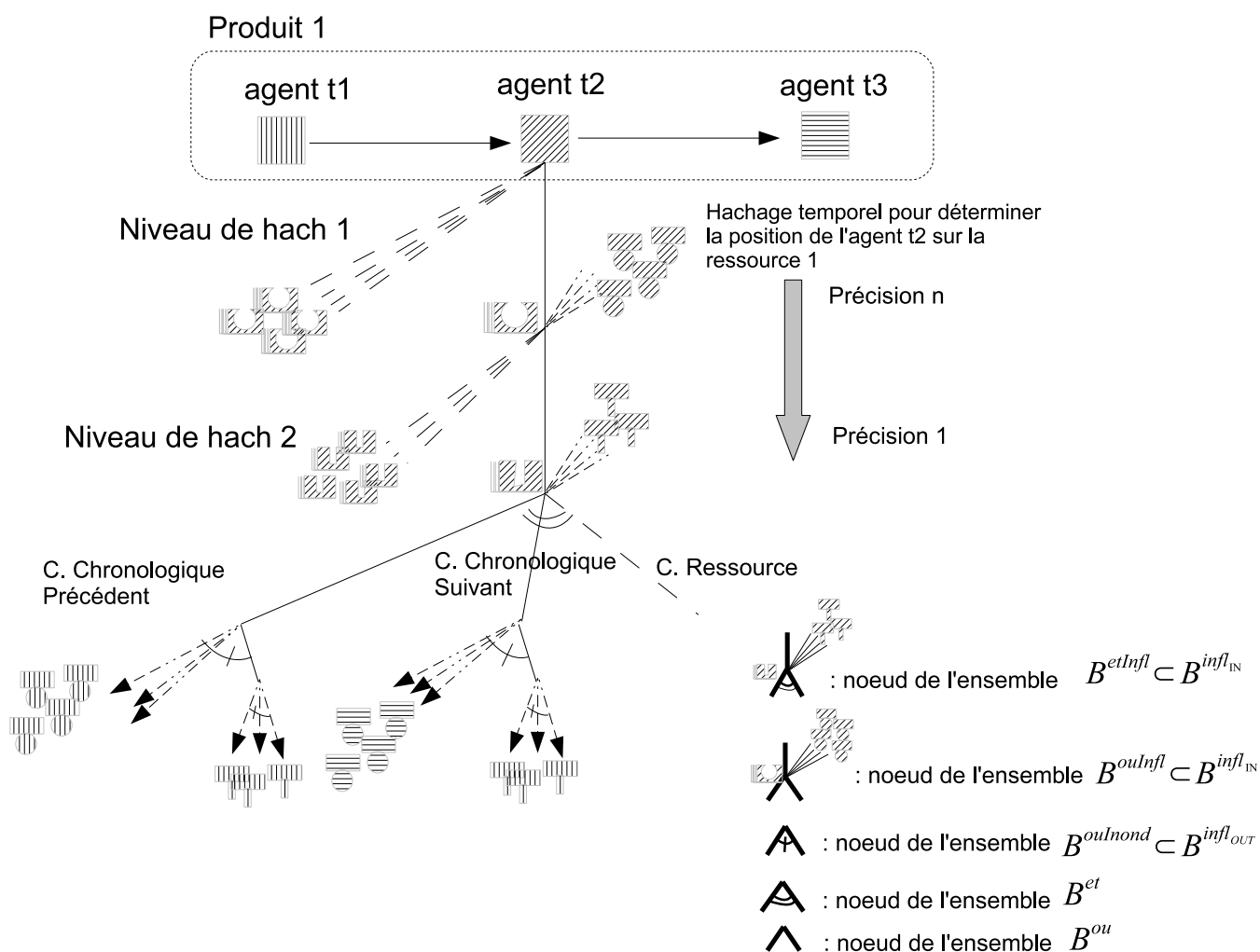


FIG. 2.2 – Réseau CESNA pour l'ordonnancement (Hachage sur deux niveaux)

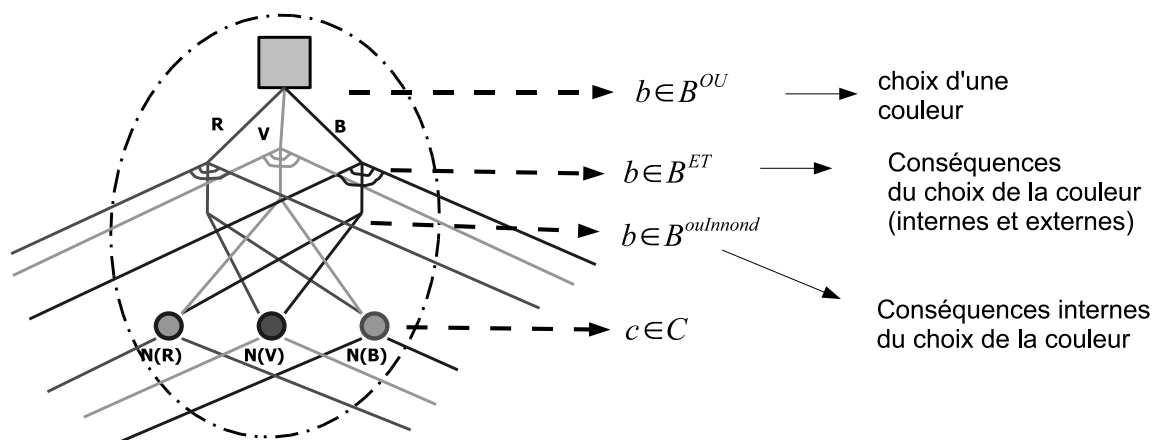


FIG. 2.3 – Réseau de coloration : point de vue agent

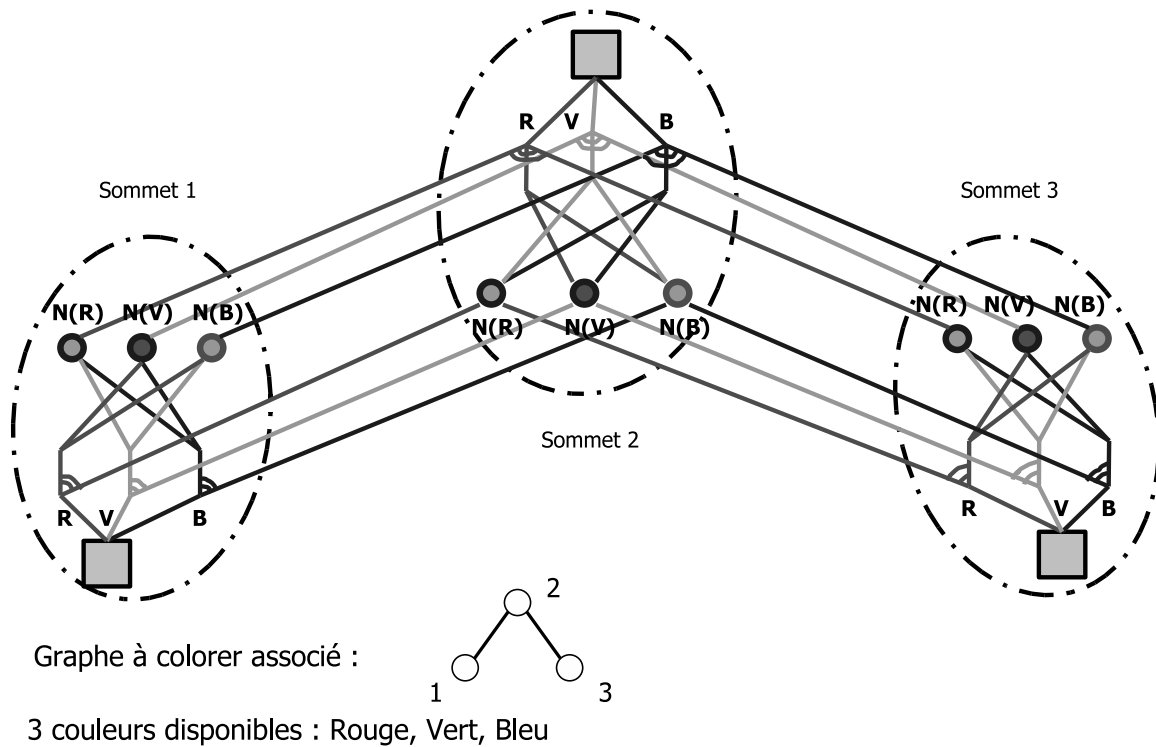


FIG. 2.4 – Réseau de coloration

Sur la figure 2.4, on perçoit plus précisément les connexions entre les agents sur une instance complète de modèle. Chaque agent exprime son choix (conséquences internes) et perçoit l'état de son environnement (conséquences externes). Les deux canaux de communication sont cependant tous deux symétriques : les agents perçoivent et influencent sur les deux canaux de communication. Nous avons utilisé les noeuds les plus classiques des ensembles B^{OU} , B^{ET} et C . Une adaptation a été réalisée pour les noeuds de l'ensemble $B^{ouInnond}$. Pour ce type de noeud, les préférences sont diffusées par inondation (fonction 5 *influenceEnvironnement*) conformément à ce qui est réalisé pour les noeuds de l'ensemble B^{ET} . Par contre, la satisfaction de ce type de noeud est assurée si le noeud totalise autant d'accords qu'il connaît de fils (fonction 2 *getSatisfaction*). En effet, l'agent ne se préoccupe pas de la répartition des accords entre ses fils, mais seulement du nombre d'accords qui permet de valider qu'il n'est pas en contradiction avec son environnement.

2.3 Quelques critères pour l'élaboration de modèles applicatifs CESNA

Nous avons évalué empiriquement différentes combinaisons réseaux-comportement pour chacun des problèmes applicatifs décrits afin de proposer les meilleurs découverts. Dans cette section, nous essayons de capitaliser l'expérience d'investigation que nous avons acquise dans ce contexte. Cette expérience est restituée par quelques premiers éléments méthodologiques qui pourront être approfondis.

Nous discutons tout d'abord des facilités/difficultés d'élaboration de modèles CESNA suivant les exigences du problème applicatif abordé. Nous avons identifié deux situations pour lesquelles des hésitations inutiles sont intégrées au processus de résolution, celles-ci peuvent être généralisées à travers les

notions de pertinence des informations véhiculées entre agents et de visibilité entre entités (nous verrons que les objets qui composent un problème applicatif ne sont pas toujours représentés par des agents).

2.3.1 Paramétrage des systèmes

Nous avons présenté deux problèmes applicatifs sur lesquels l'approche CESNA a été appliquée. L'élaboration de ces deux modèles a permis de découvrir une grande diversité de topologies de réseaux et de comportements.

Afin d'étudier les capacités de CESNA, les problèmes applicatifs que nous avons choisis sont propices à la suractivité agent (beaucoup de contraintes, solutions rares). Ceux-ci ont nécessité un paramétrage approfondi parmi un vaste ensemble de possibilités.

Il est assez facile de constater qu'une architecture associée à certains comportements ne fonctionne pas. Par contre, il est difficile d'interpréter les causes de l'échec : les phénomènes recherchés émergent d'un paramétrage subtil qui met en oeuvre différents processus en interaction (exploration/exploitation, rapport local/global, etc.), ceux-ci ne peuvent être étudiés séparément. Notre stratégie a été de profiler attentivement les processus implémentés afin de vérifier qu'ils ne sont pas source d'hésitations.

Tous les problèmes que CESNA peut traiter ne sont pas aussi complexes : les solutions s'expriment quelquefois plus localement que globalement. Dans ces configurations, la transmission des caractéristiques complexes des problèmes sur un périmètre plus réduit sont plus immédiates, CESNA tolère alors un paramétrage moins travaillé.

Il est alors envisageable de fournir à l'utilisateur du système la liberté de modéliser lui-même les problèmes qu'il aborde. Les relations logiques 'ET', 'OU', etc. permettent d'exprimer facilement les besoins de différentes entités en interaction qui miment celles qui composent les problèmes industriels.

Un de nos objectifs applicatifs est de majorer la capacité d'autonomie des utilisateurs. Nous identifions pour cela des motifs complexes de problèmes pour lesquels nous proposons des solutions. L'utilisateur peut alors reconnaître ces motifs applicatifs préparamétrés, les intégrer et les faire évoluer au sein de son propre environnement. Par exemple, la modélisation du problème de coloration de graphes que nous proposons fournit un motif d'exclusion entre consommateurs de ressources qui pourrait être adapté rapidement à d'autres problèmes. L'ensemble de ces connaissances pourrait alimenter une méthodologie pour l'élaboration de modèles CESNA.

2.3.2 Pertinence des informations véhiculées

Il est intéressant d'observer le sens des informations de communication échangées entre agents pour valider leur pertinence. Pour le problème de coloration de graphes étudié, nous avons rencontré une situation particulièrement intéressante présentée sur la figure 2.5 parmi la douzaine d'architectures que nous avons testées et souhaitons illustrer avec celle-ci une situation contreproductive que l'on peut rencontrer.

Sur la figure 2.5, on représente par un réseau CESNA la relation entre deux noeuds d'un graphe à colorer. Chaque noeud est associé à un agent. Nous soulignons sur la figure 2.5 la transition de couleur de (R)ouge à (B)leu de l'agent a_2 . Dans cette configuration, l'agent a_1 a sélectionné la couleur (V)erte, il ne devrait donc pas être affecté par cette transition. Cependant, l'architecture proposée enclenche dans cette situation une renégociation inutile. On représente en effet par deux contrats différents des relations similaires entre les deux agents ($a_1 : V, a_2 : R$ et $a_1 : V, a_2 : B$). La renégociation enclenche alors une activité qui peut facilement se propager à l'ensemble des voisins de l'agent puis à l'ensemble du système. Cette constatation souligne la nécessité de s'intéresser à l'information qui est véhiculée entre les agents. Les modèles condensés permettent, comme le montre l'architecture que nous avons finalement sélectionnée, d'exprimer les relations entre agents avec plus d'efficacité. Par ailleurs, en synthétisant les relations à leur essentiel, on diminue aussi la taille des réseaux qui est aussi un critère d'efficacité. Les

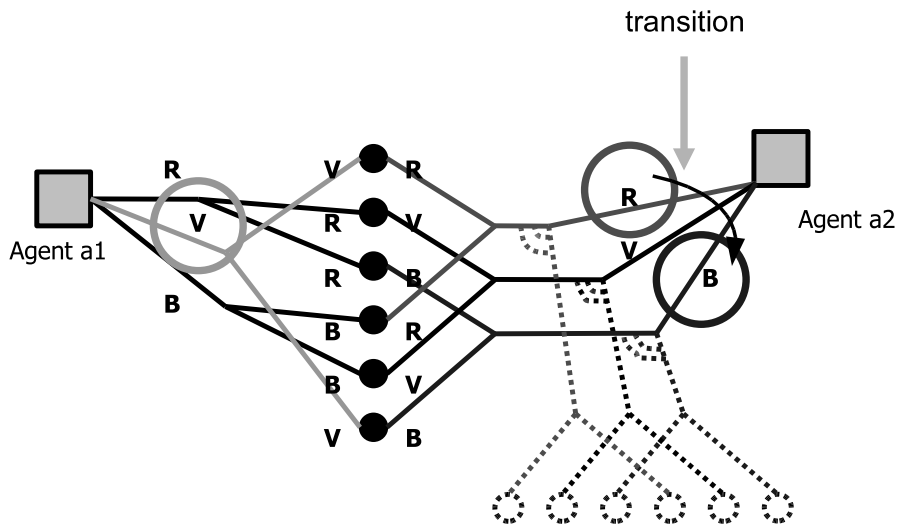


FIG. 2.5 – Communication inefficace

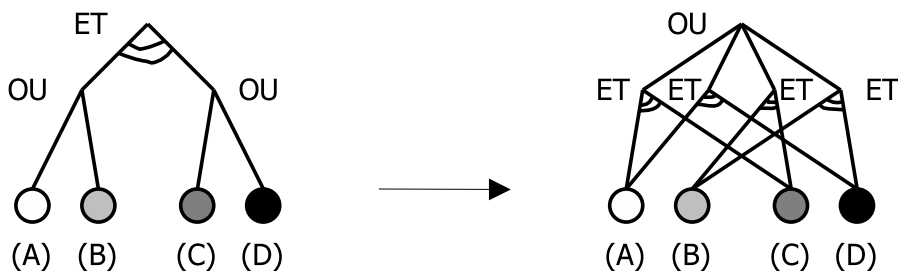


FIG. 2.6 – Représentations logiques similaires par des architectures différentes

performances que nous avons obtenues avec le modèle condensé sont bien meilleures que celles obtenues avec le modèle redondant que nous venons de décrire.

2.3.3 Visibilité entre entités

L'élaboration d'un réseau CESNA requiert d'identifier les agents responsables des problèmes. Les agents sont à la source de l'activité enclenchée par leurs processus de perception et d'action. Nous avons pu constater au cours de l'élaboration du réseau pour l'ordonnancement qu'il n'est pas nécessaire de saturer le réseau en agent, que certains noeuds peuvent faire preuve d'anthropomorphie sans pour autant être représentés par des agents (contrats de type ressource). On pourrait parler dans cette situation d'agents passifs.

On met en relation les agents d'un système à travers différentes relations logiques. Il faut être attentif à la taille des réseaux résultant qui peut parfois compromettre la performance de résolution. Nous voyons sur la figure 2.6 le type d'accroissement inutile qui peut se rencontrer en élaborant un réseau CESNA¹².

¹²Nous n'excluons toutefois pas qu'une architecture plus volumineuse puisse dans certaines situations être bénéfique au processus de résolution

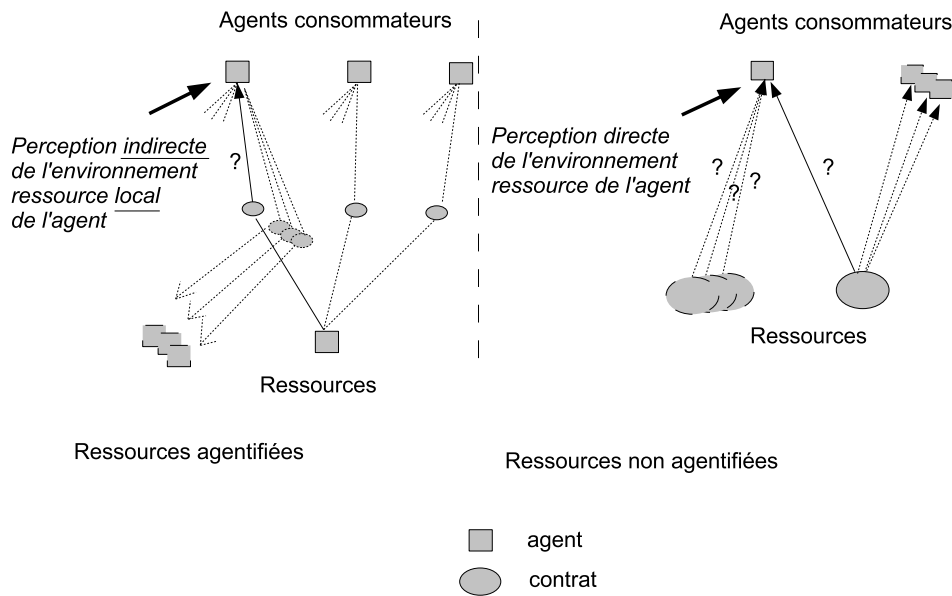


FIG. 2.7 – Deux visibilité différentes entre entités

Sur la figure 2.7, on propose deux modes de relation entre Ressources et Agents consommateurs de ressource. En représentant chaque ressource par un agent, une ressource communique avec les agents consommateurs de ressources par l'intermédiaire de contrats. Ainsi, la ressource répond à une sollicitation au cours de sa période d'activité suivante. Suivant ce procédé, les agents consommateurs de ressources ne peuvent pas solliciter des ressources par inondation (en effet, des engagements contradictoires pourraient rapidement renforcer un processus de suractivité). Avec cette représentation, chaque agent consommateur doit donc questionner ses différentes ressources alternatives une à une, et subit par ailleurs les latences occasionnées par la communication indirecte entre les deux agents. Au contraire, lorsque la ressource n'est pas agentifiée (partie droite de la figure), l'agent peut percevoir directement les ressources de son environnement (cette fois, l'inondation est possible car il ne s'agit pas d'un processus d'engagement pour l'agent), pour ensuite sélectionner la ressource qui lui convient parmi les différentes alternatives disponibles, sans aucune latence. Nous voyons une fois de plus que certaines subtilités mènent rapidement à un blocage ou à une mise à profit des réseaux CESNA.

3

Mesures et analyses pour le processus de résolution

Les mesures et analyses que nous réalisons sont de deux catégories :

Certaines ont pour objectif de cerner les processus qui s'établissent au sein des réseaux CESNA. On souhaite s'appuyer sur une meilleure compréhension des comportements internes de nos modèles pour guider nos futurs travaux. Nous avons commencé cette étude sur des réseaux générés aléatoirement (mesures des sections 3.1, 3.2 et 3.3). Le contexte d'étude des problèmes générés aléatoirement est exposé ci-dessous.

D'autres mesures et analyses permettent de se faire une idée des performances de CESNA face à d'autres modèles SMA et à d'autres approches. Nous utilisons pour nous comparer les problèmes d'ordonnancement et de coloration de graphes (mesures de la section 3.5).

Description : Contexte d'étude pour les problèmes générés aléatoirement.

Les problèmes que nous choisissons de générer sont constitués de noeuds de type N^{ou} et N^{et} tel que nous les avons décrits dans le chapitre 1. Nous avons utilisé :

- $\forall n \in B, 1 < suc(n) < 5$
- $\forall a \in A, \forall c \in C, 1 < \delta(a, c) < 6$, avec $\delta(c, a)$ la distance entre a et c

Nous garantissons une solution pour chacun des problèmes générés. Quelle que soit la taille du problème, le processus de résolution retrouve la solution explicitement constituée. Pour chaque problème, les solutions sont rares parmi l'ensemble des arrangements infructueux. Il est probable que les solutions que nous générons explicitement soient uniques, ce qui est propice à l'étude de la coordination de l'activité agent en environnement contraint.

Pour les problèmes générés aléatoirement, chaque point de mesure est réalisé à partir de 100 exécutions. Afin de réduire le temps de calcul pour les mesures réalisées, nous arrêtons le processus de résolution après 1000 périodes d'activité : nous considérons que le processus a échoué, enregistrons l'échec et comptons 1000 périodes d'activité pour la mesure courante.

C'est dans ce contexte que nous mesurons, dans la section 3.1, l'impact du paramètre d'exploration sur les performances de résolution. Dans la section 3.2, nous mesurons l'influence de la taille des problèmes sur le temps de résolution. Dans la section 3.3 nous mesurons les informations échangées au sein du réseau de collaboration.

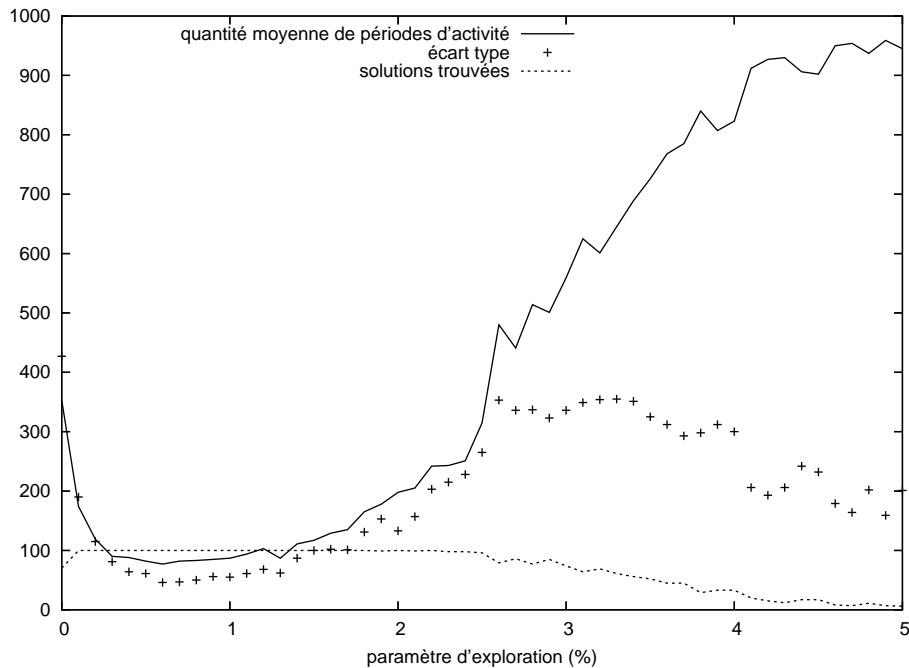


FIG. 3.1 – Impact du paramètre d'exploration sur les performances (mesuré à partir de 100 exécutions)

3.1 Paramètre d'exploration et d'exploitation

Dans la procédure 5, la fonction *expressionExploration* permet de sélectionner aléatoirement un autre noeud que le successeur perçu comme le plus prometteur, suivant une probabilité de $parametreExploration/100$. C'est avec ce paramètre que nous faisons varier la part d'exploration du processus de résolution.

Sur le graphique de la figure 3.1, on représente l'influence du paramètre d'exploration sur le nombre moyen de périodes d'activité pour résoudre un problème. Les mesures sont réalisées à partir du graphe aléatoire suivant : 21 agents ont à valider 85 contrats (solution garantie) parmi 374 contrats dans un graphe de 847 noeuds.

L'aspect général de la courbe de la figure 3.1 montre que les performances du système sont intimement liées à la valeur du paramètre d'exploration utilisé. Sur cette courbe, le comportement optimal est atteint pour une exploration aux alentours de 1%. Le réseau étudié se compose d'environ 240 noeuds de l'ensemble B^{ou} sur lesquels s'exprime l'exploration conformément à la fonction 5 *influenceEnvironnement*.

Pour les problèmes de coloration de graphes que nous étudions en fin de chapitre, dans la partie 3.5, les meilleurs résultats sont cette fois relevés suivant un paramètre d'exploration plus faible aux alentours de 0.1%. On note cependant que pour les problèmes de coloration, l'exploration s'exprime par une quantité supérieure de noeuds de l'ensemble B^{ou} qui composent les réseaux de négociation (environ 5000 pour le problème de coloration *school*). Il semble qu'il y ait une relation entre la taille des problèmes traités et la valeur optimale du paramètre d'exploration. Voyons quelles en sont les raisons.

Le paramètre d'exploration ne définit pas l'exploration du système de façon absolue. Des réseaux CESNA différents sont composés de quantités variables de noeuds de l'ensemble B^{ou} , ce qui produit des comportements d'exploration différents malgré un paramètre d'exploration constant. Nous ne pouvons par ailleurs pas affirmer que l'exploration absolue réalisée par le système dépend exclusivement de la quantité de noeud de l'ensemble B^{ou} et du paramètre d'exploration choisit. En effet, l'expression de

l'exploration dépend aussi de la topologie des réseaux. Approfondissons ce propos en se positionnant d'un point de vue agent. Chaque agent est connecté à l'ensemble du réseau CESNA par un arbre. La proportion d'élargation réalisée par la fonction *influenceEnvironnement* (qui dépend de la forme de l'arbre et donc du réseau CESNA) définit elle aussi la capacité d'expression des différents noeuds de l'ensemble B^{ou} .

Nous ne pouvons par ailleurs pas assurer que l'exploration absolue injectée dans le système évolue linéairement avec la valeur du paramètre d'exploration. En effet, l'exploration absolue résulte de la complexité des interactions et des influences qui s'expriment au sein du réseau (voir la section 3.3). Toutes ces considérations soulignent l'intérêt de donner à ses systèmes des capacités d'auto-organisation plutôt que de fixer artificiellement certains de leur paramètre sans pour autant en percevoir entièrement leur impact. Une amélioration de l'exploration telle que nous l'avons implémentée serait de formaliser celle-ci de façon à considérer les caractéristiques complexes du problème dans le calcul de sa diffusion. Il est cependant possible, dans un premier temps, de fixer empiriquement la valeur de ce paramètre par catégorie de problème.

Nous avons pu constater que pour certains problèmes simples (tels que les problèmes de Mycielski détaillés à la suite), nous obtenons de bons résultats avec un paramètre d'exploration à zéro. Comment expliquer que certains autorisent une exploration nulle ?

Sur le graphique de la figure 3.1, nous observons deux cas lorsque le paramètre d'exploration est proche de zéro : la solution est trouvée rapidement, ou stagne dans une mauvaise configuration de résolution et conduit à un échec. Au contraire, une forte valeur d'exploration conduit à une communication très bruitée, et crée un dysfonctionnement. L'écart type des performances mesurées se réduit alors car les différentes exécutions excèdent uniformément le temps limite imparti (1000 périodes d'activité).

Nous pensons que des problèmes de partage de ressources différents, ou des représentations CESNA différentes ne nécessitent pas forcément le même taux d'exploration absolu. La capacité de perception des relations complexes par CESNA peut dans certaines situations être entière et ne donner lieu à aucune raison de douter et de se détourner de l'attraction la plus forte. C'est le cas des problèmes de Mycielsky détaillés en fin de chapitre. Dans d'autres situations (perception plus floue, latences entre les signaux émis et ceux perçus par l'activité sociale), il est nécessaire de bruite le comportement du système afin de ne pas s'égarer sur des voix sans issue, et de ne pas bloquer le système sur des minima locaux.

Les différences importantes de performances soulignent la nécessité pour les systèmes multi-agents réactifs de garantir un bon équilibre entre exploration et exploitation.

3.2 Influence de la taille des problèmes

Sur le graphique de la figure 3.2, nous relevons l'évolution du temps de résolution par rapport à la taille des problèmes générés aléatoirement. Nous présentons l'évolution de la quantité moyenne de perceptions et d'actions des agents (= nombre moyen de périodes d'activité * nombre d'agents) par rapport à l'espace de recherche des problèmes traités. Les problèmes présentés mettent en jeu des quantités variables de contrats (100 → 1000). On représente les espaces de recherche sur une échelle logarithmique. Pour ces mesures à partir de problèmes aléatoires, nous fixons le paramètre d'exploration à 1%.

Nous avons mené une étude similaire dans [Armetta 05a] et [Armetta 05b], sur un modèle précurseur de CESNA. Ce premier modèle, plus simple, ne prévoyait pas la diversité des relations logiques que propose CESNA, le décrire en profondeur n'apporte rien d'essentiel et de nouveau au travail présenté. Nous montrons dans ce document que ce modèle précurseur établissait des résultats de coordination similaire lorsque la taille des problèmes augmentait.

Les problèmes générés aléatoirement et étudiés sur le graphique de la figure 3.1 nous permettent d'étudier l'impact de la taille des problèmes sur les performances de résolution. Nous notons sur le

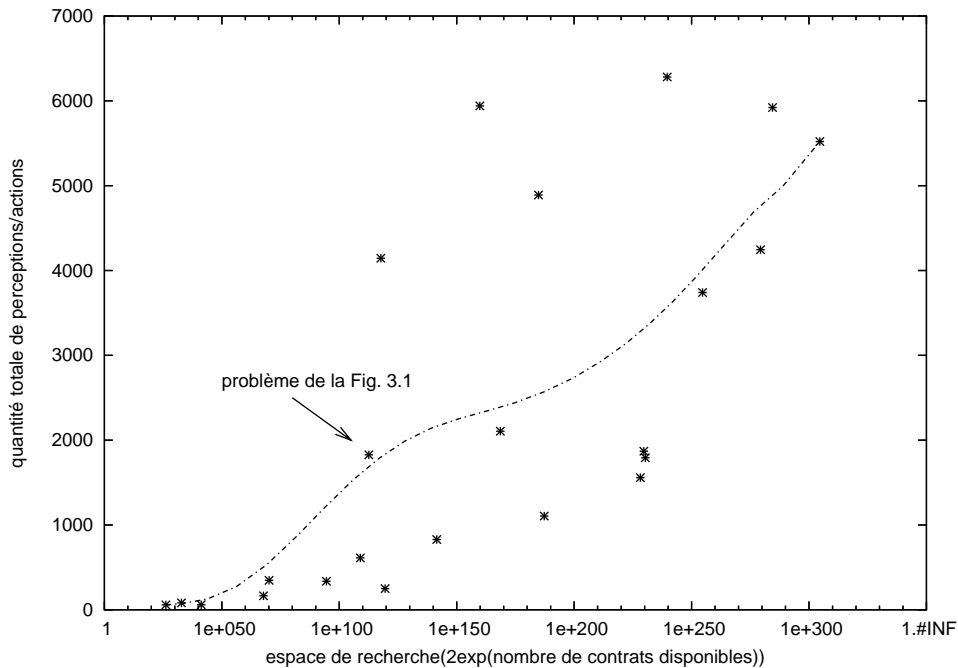


FIG. 3.2 – Impact de la taille des problèmes sur les performances (points mesurés pour 100 exécutions)

graphique certains points de mesures isolés de la tendance générale. Ceux-ci montrent que le temps de résolution ne dépend pas exclusivement de la taille des problèmes traités : un autre facteur topologique intervient, certains des réseaux générés aléatoirement matérialisent des paysages de recherche plus difficiles. Pour les problèmes générés aléatoirement les plus difficilement résolus par CESNA, le temps d'exécution n'excède pas quelques secondes, le taux de succès avoisine 90 %.

La tendance générale linéaire de l'approximation par courbe de bézier fait apparaître un accroissement logarithmique du nombre moyen de perceptions/actions face à la taille des problèmes traités. Pour chaque problème généré, nous garantissons une seule solution (il s'avère que nous ne retrouvons que celle explicitement générée). Cette solution est cachée dans un espace de recherche de très grande taille. Nous pouvons donc conclure que le système que nous proposons permet de percevoir les caractéristiques complexes du problème traité et de retrouver des solutions rares. Si le système ne percevait pas les contraintes du problème, il ne convergerait pas aussi rapidement et s'égarerait dans l'immense espace de recherche. D'autre part, cette perception est réalisée de façon décentralisée ce qui souligne la capacité de coordination de l'ensemble de l'activité agent. D'autres observations complètent cette première estimation de la capacité de convergence de CESNA par comparaison avec une approche hybride qui combine l'algorithmie génétique et le recuit simulé pour le problème de coloration de graphes (cf. section 3.5).

3.3 Mesure interne des influences et des échanges

L'activité des agents permet le parcours de l'ensemble des états accessibles au système, jusqu'à trouver un motif globalement satisfaisant. Il est difficile d'analyser la trace brute des transitions d'état de $\varphi(c) \in \{vrai, faux\}, c \in C$. On ne perçoit pas de cohérence sur ce signal brut qui semble refléter une évolution chaotique. Les expériences que nous avons présentées montrent cependant que le système converge pour des problèmes difficiles : il existe donc une cohérence de l'activité des agents.

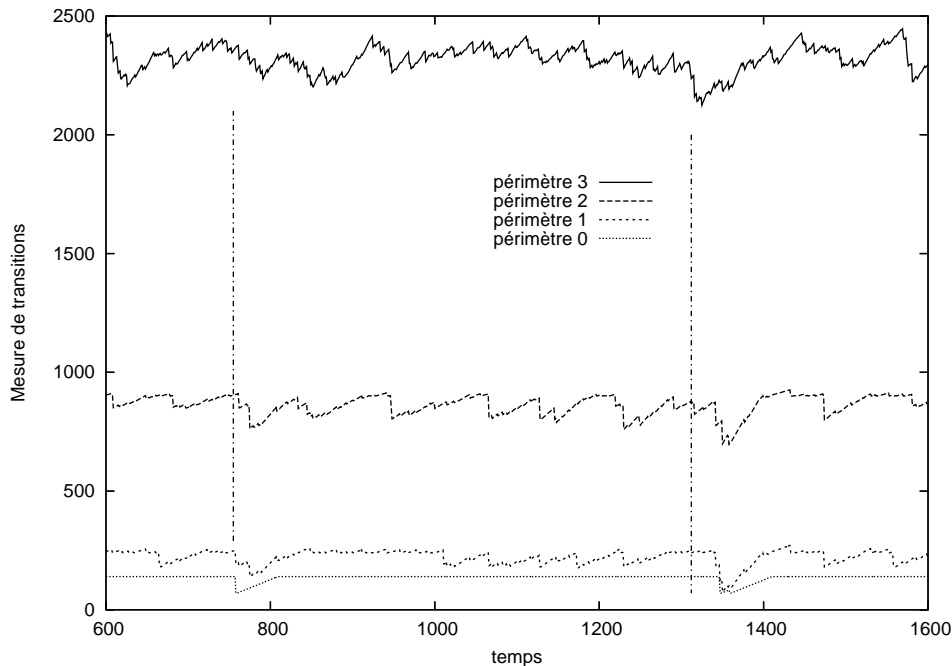


FIG. 3.3 – Changements significatifs : de local à global et de global à local

Quelle est donc la dynamique entre ordre et désordre qui permet la coordination de décisions décentralisées ? Notre démarche consiste à mettre en relief des éléments de la trace brute pour faciliter son étude, ceci nous permet de valider partiellement des premières hypothèses relatives au comportement émergent du système.

Nous mesurons les transitions d'état de $\varphi(c) \in \{\text{vrai}, \text{faux}\}$, $c \in C$. Pour compléter les informations perçues, nous mémorisons et rapportons aussi les transitions de $\varphi(b) \in \{\text{vrai}, \text{faux}\}$, $b \in B$ qui représentent l'état de validité propagé par les différents agents dans le réseau au cours de la procédure 5. Sur le graphique de la figure 3.3, nous mesurons des changements d'état significatifs sur différents périmètres à partir d'un point central du réseau :

- Changements d'état significatif : On affecte à chaque noeud un compteur qui est positionné à zéro en cas de transition, incrémenté dans le cas contraire. Pour chaque périmètre représenté sur le graphique 3.3, on somme les différents compteurs des noeuds appartenant au même périmètre.
- Différents périmètres : À partir d'un noeud central de périmètre 0, sélectionné aléatoirement, le périmètre est incrémenté à chaque parcours d'un noeud de connexion.

La mise en relief choisie permet de mesurer l'influence de transitions à faible période entre différentes parties du réseau, de local à global, et de global à local. Certaines de ces transitions expriment une bifurcation dans l'exploration de l'espace de recherche. Sur le graphique 3.3, on observe l'influence de ces transitions sur différents périmètres. Nous notons qu'une transition peut se propager de local à global (tracé vertical gauche), ou de global à local (tracé vertical droit). Ces mesures confortent notre intuition que la représentation des relations complexes par un réseau autorise une communication non explicite entre agents, sur leur état de résolution. Ces relations sont asynchrones et de cardinalité $n - n$.

Afin d'illustrer l'évolution du système, nous introduisons la notion plus abstraite de motifs en compétition. Les sélections réalisées par les agents matérialisent des conflits ou des accords qui se propagent ou se renforcent à travers le réseau (figure 3.3). On peut ainsi tracer le périmètre de différentes relations d'accord qui matérialisent différents motifs en émergence. Similairement et consécutivement au

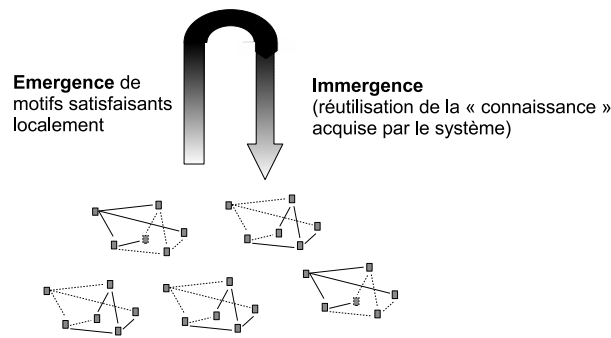


FIG. 3.4 – De local à global : Équilibre exploration/exploitation

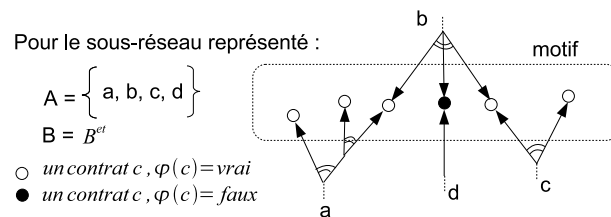


FIG. 3.5 – Mémorisation et persistance de motifs

comportement des agents, les différents motifs s’influencent les uns les autres. Sur la figure 3.4 nous représentons ces différents motifs en émergence qui sont en interaction. Ils perçoivent une influence de l’état de résolution global du système, que l’on nomme immergence.

Nous discutons dans la partie qui suit des influences entre motifs émergents au sein du réseau, et des conditions de leurs survies.

3.4 Capacité de mémorisation

Dans l’état de l’art que nous avons présenté, on reporte que l’hyperactivité des agents peut nuire à la convergence du processus de résolution. À chaque tentative d’exploration, un agent peut détruire irrémédiablement un motif pertinent en cours d’élaboration. Notre modèle permet à un agent de percevoir les caractéristiques complexes d’un problème et l’évolution de la situation courante. Il permet aussi une influence entre les sous-parties du réseau. Cependant, sachant que l’exploration de l’espace de recherche n’est pas mémorisée, celle-ci n’autorise pas de retour en arrière : comment se protéger alors des effets parfois destructeurs et irrémédiables de l’exploration ? Nous discutons dans cette partie de la capacité du système à s’auto-organiser.

Un système est capable d’auto-organisation lorsqu’il est capable de créer et maintenir des structures internes stables sans contrôle extérieur. Nous représentons sur la figure 3.5 une partie du réseau sur laquelle un agent d est en conflit avec un motif porté par les agents a, b, c . Nous ne représentons sur la figure que les noeuds de connexion appartenant à B^{et} : ce sont eux qui caractérisent un motif (les noeuds appartenant à B^{ou} représentent des ‘bifurcations’ entre motifs).

Étudions le processus d’activité agent dans cette situation. L’agent d exprime son désaccord, le contrat entre d et b est invalidé ce que l’agent b perçoit (procédure 4). L’agent b perçoit aussi le soutien des agents a et c (l’agent b perçoit par l’environnement la moyenne des influences de a, d, c suivant

la procédure 4). En fonction de la force, et de la fréquence des influences perçues, l'agent b convaincra l'agent d de rejoindre le motif représenté (par marquage de l'environnement), ou bien participera à la disparition du motif porté par a et c et à l'émergence d'un nouveau motif. Nous pensons que la chance de survie d'un motif dépend de sa pertinence dans son environnement pour un état de résolution donné. L'activité du système permet le parcours de l'espace de recherche, ce qui donne l'occasion à des motifs pertinents de croître.

Le procédé décrit permet des facilités d'adaptation qui sont difficiles à envisager par une approche ne bénéficiant pas d'une auto-organisation. Suivant le processus d'auto-organisation décrit, on envisage facilement que deux motifs puissent fusionner en cas de compatibilité, ou bien être partiellement revus. Ce type de réorganisation envisagée par un contrôle extérieur au système n'est pas trivial.

3.5 Mesures de performances

Dans cette section, nous mesurons les performances de CESNA pour l'ordonnancement et la coloration de graphes afin de positionner l'approche face à d'autres modèles SMA et d'autres méthodes de résolution.

3.5.1 Problème d'ordonnancement

Nous avons réalisé quelques mesures des capacités de CESNA à gérer l'activité agent sur des problèmes concrets d'ordonnancement par anticipation (avec intégration de contraintes de chronologie) et de coloration de graphes. Nous avons pour cela utilisé les architectures et comportements présentés dans le chapitre précédent.

Pour le problème d'ordonnancement étudié, nous choisissons de placer des produits de trois tâches consécutives sur un espace de ressources limitées. Nous décomposons l'espace temps en différents *slots* qui ont chacun une capacité de production. Chaque tâche consomme alors une unité production sur un des *slots* disponibles tout en respectant les contraintes chronologiques du problème traité.

Nous relevons sur le tableau 3.6 le nombre de périodes d'activité nécessaires pour obtenir une répartition convenable des tâches sur les ressources pour deux problèmes.

Quantité de produits de 3 tâches	Quantité de slots	capacité des slots	taux d'utilisation associé	nombre de périodes d'activité
16	7	7	98 % (48/49)	≈ 3
33	10	10	99 % (99/100)	≈ 3

FIG. 3.6 – Performances de CESNA pour un problème d'ordonnancement

Les problèmes d'ordonnancement choisis occasionnent une saturation des ressources critiques (98 % et 99% d'utilisation) qui nécessite une bonne coordination de l'activité agent. La convergence rapide que nous relevons sur le tableau 3.6 montre que cette nouvelle approche tolère la complexité introduite par les contraintes de chronologie. En effet, nous ne constatons pas les difficultés de gestion de l'activité agent que rencontrent les approches à base d'agents pour le problème de partage de ressources critiques.

3.5.2 Problème de coloration de graphes

Nous avons appliqué CESNA à différents problèmes de coloration de graphes de difficulté variable. Les différentes mesures ont été réalisées à partir d'un Intel Pentium-M 1.7 Ghz. sur une implémentation en java dont le code n'est pas optimisé.

Pour les problèmes qui ne sont pas identifiés comme des problèmes difficiles, CESNA permet de converger rapidement sur une solution. C'est le cas pour le problème de Mycielsky présenté sur la figure 3.7 pour lequel nous obtenons la solution optimale après une période d'activité en moyenne. Pour un autre problème de Mycielsky de plus grande taille (*myciel7* : 191 noeuds, 2360 arrêtes, 8 couleurs), on relève une dizaine de périodes d'activité en moyenne pour un temps d'exécution inférieur à 2 secondes.

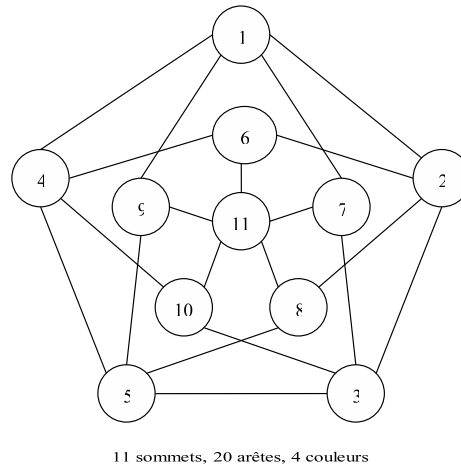


FIG. 3.7 – Un problème de Mycielski

Nous mesurons et comparons les performances de CESNA pour deux catégories de problèmes. D'une part, nous nous comparons à l'approche ABAC présentée dans [Bui 07] sur le tableau 3.8 pour des problèmes de taille moyenne. L'approche ABAC (*An Ant-Based Algorithm for Coloring Graphs*) utilise les algorithmes d'optimisation par colonies de fourmis dans le cadre de problèmes de coloration de graphes.

Afin d'étudier le comportement de notre système sur des instances de grande taille, nous avons repris différents graphes étudiés dans [Fotakis 01]. Les auteurs proposent un algorithme EVA (*Evolutionary Annealing algorithm*) spécifique au problème de coloration de graphes qui bénéficie d'une représentation centralisée des problèmes. L'approche hybride présentée combine l'algorithmie génétique et le recuit simulé pour la coloration de graphes. Nous présentons quelques résultats de comparaison à cette approche sur le tableau 3.9.

Chaque mesure a été réalisée à partir de 50 exécutions. Nous établissons un seuil de 150 périodes d'activité au-delà duquel on compte un échec de résolution (les exécutions qui n'aboutissent pas à la solution ne sont pas considérées dans le calcul des moyennes et des écarts types). Nous fournissons ces mesures à titre indicatif (machines différentes, langages d'implémentation utilisés différents, mesures différentes), mais ceux-ci permettent de se faire une idée des capacités de résolution de l'approche CESNA.

Pour la première série de mesures présentée, nous obtenons des performances comparables à celles mesurées pour l'approche ABAC. Pour les instances de problèmes plus grandes, CESNA parvient à obtenir des résultats pour certains des problèmes, mais n'atteint pas les performances de certaines autres approches. En particulier, des instances reconnues comme étant particulièrement difficiles (graphes de Leighton *le450_5a* et *le450_15c*) ne sont actuellement pas résolus par le modèle CESNA que nous proposons pour la coloration de graphes.

Problèmes de k-coloration				performances de CESNA				performances d'ABAC
désignation	noeuds	arrêtes	couleurs	taux réus-site	périodes d'activité en moyenne	écart type	temps moyen (Intel Pentium-M 1.7 Ghz.)	temps moyen (Mobile Pentium4 3.2 Ghz.)
david.col	86	406	11	50/50	27	31.2	1.2 s.	0.38 s.
anna.col	138	494	11	33/50	45.5	40.2	3.9 s.	1.14 s.
homer.col	561	1628	13	42/50	15.3	53.9	20 s.	20.75 s.

FIG. 3.8 – Performances de CESNA pour des problèmes de coloration de graphes de taille moyenne

Problèmes de k-coloration				performances de CESNA				performances d'EVA
désignation	noeuds	arrêtes	couleurs	taux réus-site	périodes d'activité en moyenne	écart type	temps moyen (Intel Pentium-M 1.7 Ghz.)	meilleur temps moyen (SUN Sparc 10)
miles1000.col	128	6432	42	46/50	70.3	35	135.8 s.	N.C.
school1.col	385	19095	14	28/50	50.7	33.6	161.4 s.	12 s.
school1_nsh.col	352	14612	14	5/50	76.6	29.1	173.3 s.	29 s.
le450_5a	450	5714	5	0/50				366 s.
le450_15c	450	16680	15	0/50				73392 s.

FIG. 3.9 – Performances de CESNA pour des problèmes de coloration de graphes de grande taille

En se comparant à des problèmes étudiés dans la communauté scientifique qui traite de l'optimisation combinatoire, nous souhaitons positionner les performances de calcul de CESNA sur des problèmes difficiles. L'objectif premier de CESNA n'en reste pas moins de traiter de façon générique une large palette de problèmes industriels associés à des problèmes de partage de ressources critiques, dont la difficulté combinatoire n'atteint pas toujours les niveaux de complexité de calcul présentés, et dont les difficultés sont réparties dans d'autres domaines (systèmes ouverts sujets à des perturbations, problèmes traités qui évoluent, etc.).

L'objectif applicatif que nous nous fixons n'est ainsi pas de dépasser les performances d'approches par optimisations spécifiques, mais de permettre le déploiement de la technologie multi-agents en milieu industriel pour le partage de ressources critiques. La problématique qui nous intéresse est avant tout 'multi-agents' à savoir : comment permettre à un SMA de se construire (de façon autonome et décentralisée) les 'connaissances' qui lui permettent de contrôler le cours de son activité de résolution pour un problème, ce à quoi l'approche CESNA répond par la construction d'un réseau d'échanges complexes et la mise en place de son processus de résolution.

La résolution rapide et efficace des problèmes de taille moyenne montre que CESNA peut dès lors s'approprier les relations complexes de ces problèmes. Nous n'avons pas encore eu le temps d'exploiter toutes les capacités de représentation de la complexité des problèmes traités et n'excluons pas d'améliorer ces premiers résultats qui nous semblent dès lors très prometteurs. En effet, l'évolution régulière des modèles applicatifs que nous utilisons apporte des gains de performance significatifs.

Par exemple, une voie de développement très intéressante consiste à rechercher une nouvelle représentation permettant toujours la perception des caractéristiques du problème tout en réduisant les volumes d'information manipulés. En effet, le réseau que nous utilisons actuellement pour les problèmes

de coloration de graphes occupe beaucoup d'espace ce qui constitue un handicap pour les problèmes de grande taille. En réduisant ainsi l'espace mémoire utilisé et le temps d'exécution de chaque période d'activité (supérieur à 2 secondes pour les problèmes de grande taille), on peut alors envisager de traiter efficacement des problèmes reconnus difficiles et de grande taille.

4

Apports de CESNA

L'approche générique CESNA que nous proposons permet de résoudre différents types de problèmes complexes. Elle permet de retrouver une solution rare au sein des réseaux générés aléatoirement présenté dans la section 3.2. En section 3.5, nous voyons qu'elle permet de résoudre des problèmes d'ordonnement et de coloration de graphes. Les problèmes d'ordonnement choisis occasionnent une saturation des ressources critiques (98 % et 99% d'utilisation) qui nécessite une bonne coordination de l'activité agent. L'approche CESNA a des performances qui peuvent être comparées à celles de certains algorithmes spécifiques de coloration de graphes.

Les différentes opportunités d'apparition de 'suractivité agent' semblent se retrouver sur les différents problèmes abordés (rareté des solutions, saturation des ressources, relations complexes) sans pour autant s'y développer. Voyons donc comment l'approche CESNA se distingue des modèles agents connus dans ce domaine et des deux précédents modèles que nous avons développés dans la partie précédente.

Le premier atout de CESNA est une perception et une influence réactive performante des différents agents dans des configurations complexes. Le processus de négociation ne définit pas de façon statique les choix à réaliser au cours du processus, mais les laisse s'exprimer sous la forme d'influences entre agents. Cette influence peut désormais s'exprimer de façon similaire entre contraintes de types différents. Nous avons présenté dans la section 1.3 différentes variantes pour laisser s'exprimer ces influences. Nous avons souligné dans le chapitre précédent qu'une bonne capacité de perception des caractéristiques complexes d'un problème était une voix que nous souhaitions explorer, ce que nous avons fait.

Nous voyons dans la section 3.3 que l'approche CESNA permet à des influences locales de se propager globalement au sein du réseau de collaboration. Cette constatation justifie la bonne coordination de l'ensemble de l'activité agent qui est nécessaire pour éviter tout problème de suractivité.

Cependant, cette coordination ne pourrait être réalisée efficacement sans une capacité de mémorisation présentée dans la section 3.4 qui permet de supporter le poids de l'exploration qui a des vertus destructrices. Avec l'approche proposée, les motifs les plus résistants sont pertinents dans leur environnement et vice versa. Les contacts entre motifs permettent l'émergence par renforcement, accroissement et fusions des motifs pertinents qui mènent à une solution au problème posé.

Toutes ces facultés corrélées nous semblent être une solution à la réduction considérable des problèmes de suractivité que nous avons souligné dans nos précédentes approches, et reconnus par [Parunak 03] pour le problème de partage de ressources critiques. Cette manière de traiter le problème de suractivité dégage des axes d'approfondissement et nous rend optimiste sur les champs de développement des systèmes multi-agents dans le domaine.

Conclusion

Pour ce travail, nous avons étudié le comportement des systèmes multi-agents confrontés aux problèmes de partage de ressources critiques. Bien que ces systèmes soient attendus pour composer les logiciels agiles de demain, nous notons que les approches applicatives spécifiques sont souvent confrontées à une mauvaise gestion de l'activité des agents pour ce problème. Cette constatation nous a incitée à nous confronter aux exigences que l'on rencontre pour la résolution générique de problèmes de partage de ressources critiques dans un contexte de résolution décentralisée par un système à base d'agents :

Une nécessité de coordonner l'activité décentralisée des agents négociant en environnement complexe :

- *comment véhiculer des informations complexes de coordination de local à global et de global à local au sein des systèmes multi-agents ?*
- *comment permettre l'émergence de motifs complexes résultat des interactions entre agents, afin de composer la solution recherchée pour les différents problèmes applicatifs ?*

Le travail présenté contribue à répondre à ces deux thèmes de recherche indissociables car corrélés au sein d'une même activité agent, elle-même complexe.

4.1 Négociation décentralisée en environnement complexe

Le premier thème de recherche s'est porté sur l'élaboration d'une représentation permettant la décentralisation de la résolution des problèmes de partage de ressources critiques. La représentation choisie doit permettre à chaque agent d'agir localement et coordonner son activité à celle du système dans sa globalité.

Au cours de ce document, nous avons parcouru différents modèles qui nous ont permis d'évaluer la pertinence de différentes représentations et de différents modèles de comportement pour les agents. Les deux premières représentations étudiées dissocient la représentation de contraintes de catégories différentes (pour le problème d'ordonnancement : contraintes chronologiques et contraintes d'affectation spatiale). Pour le problème complexe du partage de ressources critiques, il est nécessaire de permettre aux agents de s'appropriier la complexité des relations complexes exprimée par les contraintes. Pour cela, la représentation choisie doit favoriser le parcours d'informations complexes de coordination. En introduisant différents niveaux de représentation des relations, par catégorie de contraintes, on risque de freiner l'expression transparente des influences portées par des contraintes de catégories différentes.

- Pour le modèle de combat, le contact entre les représentations distinctes des catégories de contraintes intervient dans le calcul de l'arbitrage des combats : l'arbitrage spatial est influencé par les informations qui remontent par l'environnement des contraintes de chronologie.
- Pour le modèle de gestion de l'activité agent, celui-ci intervient par l'utilisation de seuils qui permettent de confronter des séquençements chronologiques aux capacités d'aménagement de l'environnement spatial.

Les moyens d'expression des relations entre contraintes que nous avons proposés nous semblent nuisibles à la communication local/global et global/local des relations complexes du problème. Une approche qui maintiendrait une représentation distincte par catégorie de contraintes devrait développer de nouveaux moyens d'expression des contacts entre les représentations, pour ne pas segmenter artificiellement les parties du problème.

L'expérience apportée par les deux premiers modèles agent proposés nous incite à choisir une représentation homogène de contraintes de catégories différentes. Pour cela, nous avons opté pour un réseau de collaboration qui permet de décentraliser le problème de partage : des agents sont connectés au réseau par l'intermédiaire de relations logiques et de contrats qui sont l'objet de négociations.

Le réseau garantit qu'une solution au problème traité est atteinte lorsque chacun des agents connectés atteint un état de satisfaction. Nous avons proposé un protocole de communication qui joue un rôle dans l'expression des capacités et des besoins des agents connectés au réseau.

Nous avons étudié quelques représentations (problème d'ordonnancement et problème de coloration de graphes), ce qui nous a permis de dégager quelques premiers éléments de méthodologie pour élaborer ces réseaux.

À partir de la représentation décentralisée du problème, il reste à mettre en oeuvre des moyens de communication entre les agents, afin que ceux-ci puissent coordonner leur activité vers un objectif commun. Nous avons proposé une communication par marquage de l'environnement (stigmergie) comme moyen de coordination de l'activité décentralisée des agents. Nous avons vu que ce type de communication autorise des interactions asynchrones entre agents, ce qui nous paraît favorable à l'expression de relations complexes de cardinalité $n-n$. La communication par marquage de l'environnement intervient comme support d'expression des influences entre les agents qui négocient, les agents incitent/inhibent la validation des contrats qui leur semblent appropriés/inappropriés. De proche en proche, des premières mesures montrent la propagation de ces influences de local à global et vice versa.

La proposition que nous avons faite nous semble permettre de représenter les relations complexes du problème, est-ce pourtant la seule exigence pour atteindre une performance de résolution pour le problème de partage de ressources critiques ?

Une fois la représentation choisie et le mode de communication installé, notre effort a porté sur le processus d'exploration de l'espace de recherche matérialisé par le réseau de négociation : quelle stratégie de recherche pour que le système valide une combinaison pertinente de contrats parmi ceux disponibles ?

Le premier modèle que nous avons proposé permet d'identifier une configuration comme adaptée lorsque le système la parcourt. Lorsque l'ensemble des configurations accessibles par le système croissent, ceci n'est pas suffisant : il faut que le système perçoive les moyens de converger rapidement vers cet état de satisfaction plutôt que de l'atteindre par opportunisme. Nous avons reconnu sur ce premier modèle une difficulté à gérer l' 'activité agent', celle-ci intervient par une propagation de perturbations au cours du processus de résolution.

Les perturbations qui éprouvent la robustesse des systèmes multi-agents sont de deux types. Nous les avons principalement considérées dans le cadre du processus de résolution : lorsque l'état du système change, les évolutions brutales se propagent par l'ensemble des contraintes du problème ce qui chamboule la configuration courante de façon désordonnée. Les modifications dynamiques des paramètres du problème traité sont aussi une forme de perturbation, celle-ci est perçue de façon similaire par le système. Quelque soit le type de perturbation appliquée au système multi-agents, il est nécessaire que celui-ci puisse s'adapter efficacement sans manifester de phénomènes de 'suractivité agent' qui nuisent à la convergence vers les solutions/adaptations recherchées. En nous intéressant à l'intégration des perturbations, nous contribuons à améliorer le comportement des systèmes dans les deux cas décrits.

Le deuxième modèle propose de modérer les effets destructeurs des perturbations appliquées aux systèmes. Nous avons proposé de superviser la diffusion des perturbations en favorisant leurs absorptions locales et en minimisant ainsi les impacts sur la globalité des configurations.

Nous avons constaté une meilleure stabilité du système mais une sous-optimalité des solutions qu'il propose. Pour le deuxième modèle, nous évaluons le comportement du système comme étant en désaccord avec la complexité du système : le processus de résolution ne doit pas favoriser différentes absorptions locales mais doit mettre celles-ci en relation d'un point de vue global afin qu'elles s'équilibrent. Finalement, l'intégration de ces perturbations ne se pose plus comme un problème de supervision, mais comme un problème de perception des capacités d'absorption.

Avec le modèle CESNA proposé, les capacités de communication accrues semblent représenter plus fidèlement les influences qui s'exercent entre les agents, nous avons vu que la représentation homogène par un réseau de négociation est favorable à la communication entre les agents.

La représentation proposée permet par ailleurs une construction progressive, non saccadée de la solution. Le système ne perçoit et ne manipule plus des événements/perturbations composées, décrites comme propres au domaine applicatif (par exemple : déplacement d'une tâche => décalage de toutes les tâches, etc.). Il perçoit ces événements/perturbations de façon décomposée par l'ensemble des relations complexes qui les matérialisent. Ainsi, les validations effectuées par le système sont plus progressives ce qui permet de les remettre en cause plus facilement. Nous avons vu que les chemins de dégradation de la configuration courante, conséquents aux erreurs commises par le système, sont moins pénalisant en adoptant une représentation microscopique des relations complexes constituant le problème.

Nous avons aussi souligné un phénomène de renforcement entre motifs qui permet la mémorisation d'un état de résolution. Ainsi, un motif (ensemble de contrats validés cohérents sur un périmètre) ne disparaît pas suite à une mauvaise décision du système, mais suite à une activité hostile prolongée qui le décrit comme non pertinent dans son environnement, et donc comme inadapté à la solution recherchée. Les influences ainsi représentées entre les motifs permettent un parcours plus cohérent de l'espace de recherche et une convergence plus directe sur la solution. La modèle CESNA permet donc une meilleure gestion de l'activité en environnement complexe mais aussi une perception émergente du chemin de convergence permettant l'obtention rapide d'une solution. Cette perception permet un équilibre efficace des activités d'exploration et d'exploitation au sein d'un même espace de marquage : le réseau de négociation.

Les performances de résolution de l'approche CESNA ont motivé un dépôt de brevet aux USA. Celles-ci peuvent être comparées aux performances de modèles applicatifs spécifiques pour les problèmes de coloration de graphes. Ces premiers résultats sont très encourageants et laissent présager une nouvelle façon d'aborder les problèmes de partage de ressources critiques en utilisant les systèmes auto-organisés à base d'agents.

4.2 De nouveaux champs applicatifs

L'approche CESNA porte sur l'expression complexe des influences entre agents pour les problèmes de partage de ressources critiques. L'étude des problèmes de partage de ressources critiques nous a permis d'identifier une difficulté particulière pour la négociation décentralisée en environnement complexe. Les propositions faites sont à approfondir dans le domaine du partage de ressources critiques, nous encourageons à les envisager dans d'autres domaines applicatifs.

Nous avons traité le problème de partage sous la forme d'un problème de satisfaction de contraintes selon lequel l'objectif consiste à trouver une solution que satisfasse toutes les contraintes :

- Pour le problème d'ordonnancement : trouver un ordonnancement qui respecte les contraintes de délais de fabrication des différents produits.
- Pour le problème de coloration : trouver une coloration possible avec un nombre de couleurs prédéfini.

Il est intéressant d'envisager ces problèmes en tant que problèmes de minimisation de critère(s). Les objectifs se complexifient de la façon suivante :

- Pour le problème d'ordonnancement : minimiser les pénalités commerciales occasionnées par les retards de production, minimiser le coût de réorganisation de la configuration courante suite à une panne machine, etc.
- Pour le problème de coloration : minimiser le nombre de couleurs à fournir pour colorer le graphe.

Cette catégorie de problème abordée comme une optimisation exacte est particulièrement difficile, même pour les méthodes qui autorisent la centralisation du processus de résolution. Une fois encore l'objectif des systèmes multi-agents intervient pour l'élaboration d'une solution satisfaisante en environnement ouvert et dynamique. Dans ce contexte, de la même manière que nous nous sommes intéressé à véhiculer des informations complexes pour résoudre les problèmes de partage, quel support permettrait de véhiculer des informations sur les difficultés multi-critères perçues par les agents de façon décentralisée ? Certains concepts que nous avons proposé sont-ils transposables pour ce type de représentation ?

Après avoir traité le problème des relations complexes entre agents, est-il possible d'étendre le travail présenté à d'autres catégories de problèmes ? En effet, tous les problèmes complexes ne sont pas associés à des problèmes de partage, n'en restent-ils pas moins complexes ? Les difficultés particulièrement prononcées rencontrées pour le problème de partage de ressources critiques ne s'exprimeraient-elles pas plus discrètement pour d'autres problèmes complexes ? Serait-il alors possible d'envisager une meilleure collaboration entre les activités d'exploration et d'exploitation des agents pour ces problèmes ?

Bibliographie

- [Aknine 04] Samir Aknine, Suzanne Pinson & Melvin F. Shakun. *An Extended Multi-Agent Negotiation Protocol*. Autonomous Agents and Multi-Agent Systems, vol. 8, no. 1, pages 5–45, 2004.
- [Armetta 04] Frédéric Armetta & Salima Hassas and Simone Pimont and Emmanuel Gonon. *Managing dynamic flow in production chains through self-organization*. In Brueckner et al. [Brueckner 05a], pages 240–255.
- [Armetta 05a] Frédéric Armetta, Salima Hassas & Simone Pimont. *A new protocol to share critical resources by self-organized coordination*. In Brueckner et al. [Brueckner 06], pages 90–103.
- [Armetta 05b] Frederic Armetta, Salima Hassas, Simone Pimont & Emanuel Gonon. *Un nouveau protocole pour le partage de ressources critiques*. In Hermès, editeur, Actes des Journées Francophones sur les Systèmes Multi-Agents, pages 83–86, Calais, Novembre 2005.
- [Armetta 06a] Frédéric Armetta, Salima Hassas & Simone Pimont. *Coordination émergente de l'activité décentralisée d'agents pour le partage de ressources critiques*. In Hermès, editeur, Actes des Journées Francophones sur les Systèmes Multi-Agents, Annecy, Octobre 2006.
- [Armetta 06b] Frédéric Armetta, Salima Hassas, Simone Pimont & Olivier Lefevre. *Towards the control of emergence by the coordination of decentralized agent activity for the resource sharing problem (selected and invited paper)*. In Engineering Self-Organising Systems, volume 4335 of *Lecture Notes in Computer Science*. Springer Verlag, 2006. à paraître.
- [Bonabeau 94] Eric Bonabeau & Guy Theraulaz. Intelligence collective. numéro 2-86601-447-2. Edition Hermès, 1994.
- [Bonabeau 97] Eric Bonabeau, Andrej Sobkowski, Guy Theraulaz & Jean-Louis Deneubourg. *Adaptive Task Allocation Inspired by a Model of Division of Labor in Social Insects*. In Dan Lundh, Björn Olsson & Ajit Narayanan, editeurs, BCEC, numéro 981-02-3262-4, pages 36–45. World Scientific, 1997.
- [Bourjot 99] Christine Bourjot, Vincent Chevrier, Alexandre Bernard & Bertrand Krafft. *Coordination par le biais de l'environnement, une approche biologique*. In Hermès, editeur, Ingénierie des systèmes multi-agents, actes des 7èmes Journées Francophones sur l'Intelligence Artificielle Distribuée et les Systèmes Multi-agents, pages 237–250, St Gilles les bains, La Réunion, 1999.
- [Brueckner 05a] Sven Brueckner, Giovanna Di Marzo Serugendo, Anthony Karageorgos & Radhika Nagpal, editeurs. Engineering self-organising systems, methodologies and applications [revised versions of papers presented at the engineering selforganising applications (esoa 2004) workshop, held during the autonomous agents and multi-agent systems

- conference (aamas 2004) in new york in july 2004, and selected invited papers], volume 3464 of *Lecture Notes in Computer Science*. Springer, 2005.
- [Brueckner 05b] Sven A. Brueckner & H. Van Dyke Parunak. *Information-driven phase changes in multi-agent coordination*. In Brueckner et al. [Brueckner 06], pages 104–119.
- [Brueckner 06] Sven A. Brueckner, Giovanna Di Marzo Serugendo, David Hales & Franco Zambonelli, editeurs. Engineering self-organising systems, third international workshop, esoa 2005, utrecht, the netherlands, july 25, 2005, revised selected papers, volume 3910 of *Lecture Notes in Computer Science*. Springer, 2006.
- [Bui 07] Thang N. Bui, ThanhVu H. Nguyen, Chirag M. Patel & Kim-Nah T. Phan. *An Ant-Based Algorithm for Coloring Graphs*. to appear in the Journal of Discrete Applied Mathematics, 2007.
- [Campos 00] Mike Campos, Eric Bonabeau, Guy Théraulaz & Jean-Louis Deneubourg. *Dynamic Scheduling and Division of Labor in Social Insects*. Adaptive Behaviour, vol. 8, no. 2, pages 83–96, 2000.
- [Caseau 95] Yves Caseau & Francois Laburthe. *Improving Branch and Bound for Jobshop Scheduling with Constraint Propagation*. In Combinatorics and Computer Science, pages 129–149, 1995.
- [Chen 03] Jinbo Chen, Alejandro Bugacov, Pedro Szekely, Martin Frank, Min Cai, Donghan Kim & Robert Neches. *Coordinated Aggressive Bidding in Distributed Combinatorial Resource Allocation*. In AAMAS Workshop on Representations and Approaches for Time-critical Decentralized Resource/Role/Task Allocation, Melbourne, Australia, 2003.
- [Cicirello 01] Vincent Cicirello & Stephen Smith. *Wasp-like Agents for Distributed Factory Coordination*. Rapport technique CMU-RI-TR-01-39, Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, December 2001.
- [Cicirello 04] Vincent A. Cicirello & Stephen F. Smith. *Wasp-like Agents for Distributed Factory Coordination*. Autonomous Agents and Multi-Agent Systems, vol. 8, no. 3, pages 237–266, May 2004.
- [Daouas 94] Thouraya Daouas, Khaled Ghedira & Jean-Pierre Müller. *Une approche multi-agents avec recuit simulé pour l’ordonnancement dans un atelier flexible*. In Rencontres nationales sur le résolution pratique de problèmes NP-complets, Montpellier, 1994.
- [Devigne 04] Damien Devigne, Philippe Mathieu & Jean Christophe Routier. *Gestion simple d’équipe d’agents cognitifs*. In MANifestation des JEunes Chercheurs STIC (MAJECS-TIC), Calais, Octobre 2004.
- [Dorigo 97] Marco Dorigo & Luca Maria Gambardella. *Ant Colony System : A Cooperative Learning Approach to the Traveling Salesman Problem*. IEEE Transactions on Evolutionary Computation, vol. 1, no. 1, pages 53–66, April 1997.
- [Drogoul 93] A. Drogoul. *De la simulation Multi-Agents à la Résolution Collective de Problèmes : Une Etude de l’Emergence De Structures Dans les Systèmes Multi-Agents*. Thèse de doctorat, Université Paris 6 (Jussieu), 1993.
- [Dümmler 00] Mathias A. Dümmler. *Cycle time and capacity planning : analysis of the instationary behavior of a wafer fab during product mix changes*. In Winter Simulation Conference, pages 1436–1442, 2000.

-
- [Ferber 89] Jacques Ferber. *Objet et Agents : une étude des structures de représentation et de communication en Intelligence Artificielle*. Thèse de doctorat, Université Paris 6 (Jussieu), 1989.
- [Ferber 98] Jacques Ferber & Olivier Gutknecht. *A meta-model for analysis and design of multi-agent systems*. In International Conference on Multi-Agent Systems, pages 128–135, 1998.
- [Fotakis 01] Dimitris Fotakis, Spiridon D. Likothanassis & Stamatis Stefanakos. *An Evolutionary Annealing Approach to Graph Coloring*. In Egbert J. W. Boers, Stefano Cagnoni, Jens Gottlieb, Emma Hart, Pier Luca Lanzi, Gunther Raidl, Robert E. Smith & Harald Tijink, éditeurs, Applications of Evolutionary Computing. EvoWorkshops 2001 : EvoCOP, EvoFlight, EvoIASP, EvoLearn, and EvoSTIM. Proceedings, volume 2037 of *Lecture Notes in Computer Science*, pages 120–129, Como, Italy, 2001. Springer-Verlag.
- [Galinier 99] Philippe Galinier & Jin-Kao Hao. *Hybrid evolutionary algorithms for graph coloring*. Journal of Combinatorial Optimization, vol. 3, no. 4, pages 379–397, 1999.
- [Ghédira 92] Khaled Ghédira & Gérard Vertfaillie. *A Multi-Agent Model for the Resource Allocation Problem : A Reactive Approach*. In ECAI, pages 252–254, 1992.
- [Grassé 59] Pierre-Paul Grassé. *La reconstruction du nid et les coordinations inter-individuelles chez Bellicositermes natalensis et Cubitermes sp. La théorie de la Stigmergie : Essai d'interprétation du comportement des Termites Constructeurs*. Insectes sociaux, vol. 6, pages 41–80, 1959.
- [Hadeli 03] Karuna Hadeli, Paul Valckenaers, Constantin B. Zamfirescu, Hendrik Van Brussel, Bart Saint Germain, Tom Holvoet & Elke Steegmans. *Self-Organising in Multi-Agent Coordination and Control Using Stigmergy*. In Giovanna Di Marzo Serugendo, Anthony Karageorgos, Omer F. Rana & Franco Zambonelli, éditeurs, Engineering Self-Organising Systems, volume 2977 of *Lecture Notes in Computer Science*, pages 105–123. Springer, 2003.
- [Hadeli 04] Karuna Hadeli, Paul Valckenaers, Bart Saint Germain, Paul Verstraete, Constantin B. Zamfirescu & Hendrik Van Brussel. *Emergent Forecasting Using a Stigmergy Approach in Manufacturing Coordination and Control*. In Brueckner et al. [Brueckner 05a], pages 210–226.
- [Hassas 03] Salima Hassas. *Systèmes Complexes à base de Multi-Agents Situés*. Habilitation à Diriger des Recherches, Université Claude Bernard - Lyon 1, Laboratoire d'InfoRmatique en Images et Systèmes d'information, Décembre 2003.
- [Hogg 91] Tad Hogg & Bernardo A. Huberman. *Controlling chaos in distributed systems*. IEEE Transactions on Systems, Man, and Cybernetics (Special Section on DAI), vol. 21, no. 6, pages 1325–1332, 1991.
- [Jennings 96] Nick R. Jennings. *Coordination Techniques for Distributed Artificial Intelligence*. In G. M. P. O'Hare & N. R. Jennings, éditeurs, Foundations of Distributed Artificial Intelligence, pages 187–210. John Wiley & Sons, 1996.
- [Liu 94] Jyi-Shane Liu & Katia P. Sycara. *Collective Problem Solving Through Coordinated Reaction*. In International Conference on Evolutionary Computation, pages 575–578, 1994.
- [Liu 96] Jyi-Shane Liu. *Coordination of Multiple Agents in Distributed Manufacturing Scheduling*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, April 1996.

- [Lopez 01] Pierre Lopez & François Roubellat. *Ordonnancement de la production*. Hermès, 2001. numéro 2-7462-0184-4.
- [Mathieu 00] Philippe Mathieu & Alain Taquet. *Une forme de négociation pour les systèmes multi-agents*. In Hermès, éditeur, Journées Francophones pour l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents, pages 133–147, 2000.
- [Moyaux 06] Thierry Moyaux, Brahim Chaib-draa & Sophie D'Amours. *Information Sharing as a Coordination Mechanism for Reducing the Bullwhip Effect in a Supply Chain*. IEEE Trans. on Systems, Man, and Cybernetics, 2006. (à paraître).
- [Ombuki 04] Beatrice M. Ombuki & Mario Ventresca. *Local Search Genetic Algorithms for the Job Shop Scheduling Problem*. Appl. Intell., vol. 21, no. 1, pages 99–109, 2004.
- [Panayiotou 99] Christos G. Panayiotou & Christos G. Cassandras. *Optimization of Kanban-Based Manufacturing Systems*. Automatica, vol. 35, no. 9, pages 1521–1533, 1999.
- [Parunak 96] H. Van Dyke Parunak. Applications of Distributed Artificial Intelligence in Industry, chapitre 4, pages 139–164. G. M. P. O'Hare / N. R. Jennings (Eds.) : Foundations of Distributed Artificial Intelligence, 1996.
- [Parunak 03] H. Van Dyke Parunak, Sven A. Brueckner, Robert Matthews & John Sauter. *How to calm hyperactive agents*. In The Second International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS), numéro 1-58113-683-8, pages 1092–1093. ACM Press, 2003.
- [Parunak 05] Van Parunak, Sven Brueckner, John Sauter & Robert Matthews. *Global Convergence of Local Agent Behaviors*. In Frank Dignum, Virginia Dignum, Sven Koenig, Sarit Kraus, Munindar P. Singh & Michael Wooldridge, éditeurs, AAMAS, pages 305–312. ACM, 2005.
- [Pimont 00] Simone Pimont & Christine Solnon. *A Generic Ant Algorithm for Solving Constraint Satisfaction Problems*. In 2nd International Workshop on Ant Algorithms (ANTS 2000), pages 100–108, Bruxelles, 2000.
- [Sanlaville 02] Eric Sanlaville. *Flexibilité et Robustesse en Ordonnancement (Article collectif du groupe flexibilité du groupe de travail GOTH)*. Bulletin de la Société Française de Recherche Opérationnelle et d'Aide à la décision, Juin 2002.
- [Sauter 99] John A. Sauter & H. Van Dyke Parunak. *ANTS in the Supply Chain*. In Workshop on Agent Based Decision Support for Managing the Internet-Enabled Supply Chain, Seattle, WA, 1999.
- [Shen 98] Weinming Shen & Douglas H. Norrie. *An Agent-Based Approach for Dynamic Manufacturing Scheduling*. In Gianni Jacucci, Gustav J. Olling, Kenneth Preiss & Michael J. Wozny, éditeurs, PROLAMAT, volume 132 of *IFIP Conference Proceedings*, pages 579–590. Kluwer, 1998.
- [Smith 83] Reid G. Smith & Randall Davis. *Negotiation as a Metaphor for Distributed Problem Solving*. Artificial Intelligence, vol. 20, no. 1, pages 63–109, January 1983.
- [Solnon 06] Christine Solnon. *Des fourmis pour le problème d'ordonnancement de voitures*. In deuxièmes Journées Francophones de Programmation par Contraintes (JFPC 2006), pages 305–316, 2006.
- [T'Kindt 00] Vincent T'Kindt, Nicolas Monmarché, Daniel Laugt & Fabrice Tercinet. *Combining Ants Colony Optimization and Simulated Annealing to solve a 2-machine flowshop bi-criteria scheduling problem*. In 13th European Chapter on Combinatorial Optimization (ECCO XIII), pages 129–130, Mai 2000.

-
- [Tranvouez 01] Erwan Tranvouez. *IAD et ordonnancement : une approche coopérative du réordonnement par systèmes multi-agents*. Thèse de doctorat, Université de Droit d'Economie et des Sciences d'Aix-Marseille III, 2001.
- [Tsai 03] Chih-Hung Tsai, Yun-Min Feng & Rong-Kwei Li. *A Hybrid Dispatching Rules in Wafer Fabrication Factories*. International journal of the computer, the internet and management, vol. 11, no. 1, pages 64–72, January 2003.
- [Usher 00] John M. Usher & Yi-Chi Wang. *Negotiation between intelligent agents for manufacturing control*. In 4th International Engineering Design and Automation Conference, Orlando, Florida, USA, 2000.
- [Wang 97] Jihua Wang, Peter B. Luh, Xing Zhao & Jinlin Wang. *An Optimization-based Algorithm for Job Shop Scheduling*. Sadhana (Academy Proceedings in Engineering Sciences), vol. 22, no. 2, pages 241–256, 1997.
- [Wang 02] Yi-Chi Wang & John M. Usher. *An Agent-Based Approach for Flexible Routing in Dynamic Job Shop Scheduling*. In 11th Industrial Engineering Research, 2002.
- [Weixiong 02] Zhang Weixiong. *Modeling and solving a resource allocation problem with soft constraint techniques*. Rapport technique, School of Engineering & Applied Science, Washington University in St. Louis, May 2002.
- [Xu 00] Yuefei Xu, Robert W. Brennan, Xiaokun Zhang & Douglas H. Norrie. *A Genetic Algorithm-based Approach to Holon Virtual Clustering*. In in Proceedings of World Multiconference on Systemics, Cybernetics and Informatics (SCI'2000), volume 3, pages 380–385, 2000.
- [Yokoo 98] M. Yokoo, E. H. Durfee, T. Ishida & K. Kuwabara. *The distributed constraint satisfaction problem : formalization and algorithms*. Knowledge and Data Engineering, IEEE Transactions on, vol. 10, no. 5, pages 673–685, 1998.

Résumé

Pour ce travail, nous nous intéressons à la modélisation de systèmes à base d'agents (situés) pour la résolution de problèmes de partage de ressources critiques. Nous voyons que les systèmes multi-agents manifestent des caractéristiques qui les rendent appropriés à l'implantation industrielle en environnement ouvert et dynamique.

Les problèmes de partage de ressources critiques sont des systèmes complexes : les différentes parties de ces problèmes sont en interactions rétroactives. Ils sont aussi qualifiés de problèmes d'optimisation combinatoire : on assiste à un accroissement exponentiel du nombre de combinaisons à évaluer pour les résoudre lorsque la taille des problèmes augmente. Dans le cadre de la modélisation générique que nous proposons pour ces problèmes, nous devons donc corréliser la complexité des relations héritées du problème avec la complexité des relations décentralisées des agents en interaction. Nous devons aussi permettre une convergence rapide du système modélisé vers une configuration satisfaisante qui matérialise une solution pour le problème traité, parmi un grand ensemble de configurations disponibles.

Les systèmes multi-agents abordent le problème de partage de ressources critiques comme un problème de négociation décentralisée. En décentralisant le processus de résolution, on se confronte à la difficulté de corrélation de comportements locaux par rapport aux propriétés globales du système. Pour les problèmes de partage de ressources critiques, une mauvaise coordination de l'ensemble de l'activité des agents entraîne un dysfonctionnement qualifié de 'suractivité agent', celui-ci enraille la convergence vers une solution recherchée : les agents enclenchent et diffusent des perturbations au sein du processus de résolution de façon incontrôlée. Nous choisissons la communication par marquage de l'environnement (stigmergie) comme moyen de coordination de l'activité des agents. Celle-ci permet d'exprimer l'influence exercée par les effets persistants dans l'environnement, des comportements passés des agents, sur leurs comportements futurs.

Ainsi, une partie de notre étude consiste à trouver un environnement de marquage qui permet une représentation décentralisée des problèmes de partage traités, tout en autorisant une diffusion décentralisée d'informations complexes de coordination.

À partir de la modélisation choisie, nous cherchons à mettre à profit la perception émergente de tels systèmes afin d'organiser efficacement le parcours de l'ensemble des configurations à envisager. L'activité menée par le système valide/invalide certaines alternatives parmi un ensemble de combinaisons de grande taille, et permet de converger vers une solution. L'exploration aléatoire permet aux systèmes multi-agents de diversifier la recherche dans l'environnement pour découvrir de nouvelles pistes. L'exploitation permet de renforcer de façon auto-catalytique les pistes prometteuses découvertes (i.e. plus une piste est prometteuse, plus elle est exploitée).

Nous identifions les difficultés de résolution dans ce contexte à travers trois modèles agents. Pour le premier modèle que nous étudions, des agents participent à une négociation spatiale des espaces de productions disponibles sur un ensemble de machines qui représentent les ressources critiques du problème étudié. Nous sommes alors confrontés au problème de 'suractivité des agents' qui nuit à l'efficacité du processus de résolution. Ceci nous incite à superviser le contrôle de l'activité des agents par un contrôle hiérarchique entre deux groupes hétérarchiques de résolution. Les conclusions que nous tirons de l'étude de ces deux modèles nous pousse à envisager le défaut de 'suractivité' comme un problème de perception décentralisée, et à proposer une approche innovante : CESNA (*Complex Exchanges between Stigmergic Negotiating Agents*). Des agents négocient à travers un réseau la validation de contrats qui leur conviennent. Lorsque tous les agents sont satisfaits, on obtient une solution pour le problème. Certaines mesures montrent la diffusion d'informations de coordination complexes de local à global et vice et versa à travers le réseau de négociation. Nous mesurons par ailleurs des performances de résolution comparables à celles que l'on connaît pour les approches spécifiques de coloration de graphes. Ces résultats prometteurs soulignent les capacités de perception émergente et d'auto-organisation résultat d'interactions complexes entre les agents, ils mettent à jour la pertinence de cette nouvelle façon d'envisager les relations complexes entre agents pour le partage de ressources critiques.

Abstract

For this work, we study situated agent systems to address the resource sharing problem. We show that multi-agent systems manifest good characteristics for dynamic and open industrial applications.

The resource sharing problems are complex systems : the sub-parts of these problems are interrelated. Also, they are combinatorial optimization problems, i.e., the solution space of these instances of problem is unusually-large. In this context, we propose a generic approach to benefit from the complexity of decentralized relations between agents to fit the complex relations of this class of problem. Moreover, it is necessary to efficiently conduct the system state toward a satisfying configuration standing for a solution to the sharing problem, among a large space of possibilities.

The multi-agent systems address the resource sharing problems as a decentralized negotiation problem. While decentralizing the problem, we meet a correlation need between the local behaviours of agents with respect to the global characteristics for the problems. For the resource sharing problems, a defective coordination of the global agent activity leads to a dysfunction qualified as hyperactivity of agents, which one prohibits to reach the solution : the agents form and propagate disturbances through the resolution process. We propose an indirect communication mediated by the environment (stigmergy) as a mean to coordinate the global activity of agents. This one allows to express the influence of past behaviours of agents, remaining in the environnement, on futur behaviours.

Thus, a part of the study consists in finding a model to decentralize the resource sharing problem, and get an efficient support for social interactions between negotiating agents, providing a mean to broadcast complex informations for the global coordination.

The next step consists in exhibiting an emergent perception for the system in order to cover appropriately the set of configurations to estimate. The system moves along the configurations while exploring and exploiting in order to build a solution for the sharing problem. While exploring, it attempts to elaborate new possible patterns, while exploiting, the system autocatalytically reinforces the elaborated combinations it perceives as relevant (i.e. the more a pattern is relevant, the more it is exploited).

We underline some resolving difficulties through three agent models. For the first model, the agents participate to a spatial negotiation of available machine locations which constitute the critical resources. We meet an hyperactivity phenomena for agents which prevent to abord efficiently the problem. We propose an other model to control the agent activity with a hierarchical relation between too heterarchical resolution groups of agents. Our conclusion on the two models leads us to improve the capacity for the system to perceive the complexe relations for the sharing problems, and to innovate with a new approach : CESNA (Complex Exchange between Stigmergic Negotiating Agents). In order to represent the complex characteristics of the problem, the negotiation takes place through a negotiating network. The network contains contracts to validate/invalidate which represent the possible assignments of agents to resources and the constraints between agents allocations. In this context, agents negotiate to validate/invalidate contracts fitting their needs with respect to the needs of other agents. We measure some performances that can be confronted with some of specific optimisation methods for the graph coloring problem. These good results underline the emergent perception and self-organizing capacities resulting from complex interaction between agents, which encourage to establish a new way to address the complex resource sharing problems with multi-agents systems.

