



HAL
open science

Les approches agents pour la coordination d'activités dans les chaînes graphiques

Dhouha Anane, Suzanne Pinson, Samir Aknine

► **To cite this version:**

Dhouha Anane, Suzanne Pinson, Samir Aknine. Les approches agents pour la coordination d'activités dans les chaînes graphiques. 2009. hal-00877029

HAL Id: hal-00877029

<https://hal.science/hal-00877029>

Preprint submitted on 25 Oct 2013

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.

CAHIER DU LAMSADE

291

Decembre 2009

Les approches agents pour la coordination
d'activités dans les chaînes logistiques

D. ANANE, S. PINSON, S. AKNINE

LES APPROCHES AGENTS POUR LA COORDINATION D'ACTIVITÉS DANS LES CHAÎNES GRAPHIQUES

COORDINATION OF SUPPLY CHAINS ACTIVITIES: AN AGENT BASED APPROACH

Dhouha Anane* — Suzanne Pinson*— Samir Aknine**

* Université Paris-Dauphine, F-75016 Paris, France

LAMSADE, CNRS FRE 3234

dhouhaanane@yahoo.fr, suzanne.pinson@dauphine.fr

** Université Lyon 1, F-69622 Villeurbanne, France

LIESP

43, boulevard du 11 novembre 1918

Samir.aknine@bat710.univ-lyon1.fr

RÉSUMÉ. Les entreprises évoluent dans un environnement de plus en plus compétitif et exigeant en termes de flexibilité et de réactivité. L'utilisation des modèles issus de l'intelligence artificielle distribuée (IAD) et plus particulièrement des systèmes multi-agents (SMA) dans les outils de gestion des entreprises s'avère être efficace pour simuler et reproduire les comportements collaboratifs et adaptatifs tels qu'ils apparaissent actuellement dans les entreprises. Ce papier modélise la coordination des différentes parties collaboratives aussi bien internes qu'externes d'une chaîne logistique en utilisant les modèles de coordination par formation de coalitions proposés dans les SMA. Dans une première partie, nous faisons un tour d'horizon des travaux déjà effectués sur ce problème. Dans une deuxième partie, nous proposons notre approche agent, un algorithme de formation de coalitions ainsi qu'un protocole d'interaction entre agents indispensable pour mettre en œuvre cette coordination distribuée. Nous illustrons notre démarche par un exemple pris dans le domaine de l'industrie aéronautique.

MOTS-CLÉS: chaînes logistiques, systèmes multi-agents, formation de coalitions, coordination.

ABSTRACT. Companies operate in an environment increasingly demanding in terms of flexibility and reactivity. The introduction of the entities resulting from Distributed Artificial Intelligence (DAI) and Multi-Agent Systems (MAS) in the management of enterprises prove to be an interesting technology to simulate and reproduce the collaborative and adaptive behaviors of enterprises. This paper models the coordination of the various collaborative parties both inside and outside a supply chain using coordination methods of MAS mainly coalition formation mechanisms. In the first part, we give an overview of some existing approaches. In the second part, we present our agent modeling of supply chains, then we detail the coalition formation algorithm and we illustrate our approach with an example chosen in the industrial domain

KEYWORD : supply chains, multi-agent systems, coalitions formation, coordination.

1. Introduction

L'amélioration de l'efficacité et de l'efficience des processus de gestion de chaînes logistiques passe par une mobilisation des techniques de simulation. Or les modèles analytiques de la recherche opérationnelle, fréquemment utilisés dans ce type de problème, ne permettent pas de prendre en compte la réalité temporelle des processus ni les interactions entre entités. La simulation multi-agents (Multi-Agent Based Simulation MABS) est un outil puissant pour comprendre la dynamique des systèmes complexes et en étudier les évolutions possibles car elle permet de représenter de façon explicite les phénomènes d'interaction et de collaboration entre les entités intervenant dans le système. Lorsque les entités doivent coopérer pour réaliser une tâche commune, et ceci de façon décentralisée, les modèles de formation de coalitions proposés en intelligence artificielle distribuée semblent parfaitement adaptés. Ce document a deux objectifs : le premier est de présenter un panorama des principaux travaux multi-agents sur la modélisation de la gestion des chaînes logistiques ; le deuxième est de proposer un nouveau modèle d'interaction entre les différentes entités d'une chaîne logistique afin de coordonner au mieux les activités de cette dernière.

2. L'approche Agents pour la Gestion des Chaînes Logistiques

Le monde industriel doit affronter de fortes mutations telles que la mondialisation ou les changements rapides des besoins de la clientèle. Les systèmes de production *centralisés* traditionnels ne demeurent plus suffisants pour faire face à ces changements. C'est pour répondre à ces problèmes que le concept de *chaîne logistique* (*Supply Chain SC*) est apparu. En effet, l'architecture de chaque entreprise manufacturière s'ouvre peu à peu pour intégrer ses activités à celles de ses fournisseurs, de ses clients et de ses partenaires dans de larges chaînes logistiques [Shen et al. 1999]. Et par conséquent, une chaîne logistique implique la contribution de plusieurs agents autonomes nécessitant de nombreux processus de coordination pour atteindre leurs objectifs.

Les principes des systèmes multi-agents (SMA) ont montré leur grande capacité à aborder et gérer les problèmes de planification, d'optimisation et surtout de synchronisation tous domaines confondus. On peut citer comme exemples : les problèmes liés au domaine de la robotique, les enchères sur le web, les problèmes du diagnostic médical, les allocations de ressources, et même les problèmes de gestion et management aussi bien au sein de l'entreprise même qu'au niveau de ses relations avec ses partenaires.

Nous montrerons comment l'approche SMA peut être intéressante pour modéliser la gestion des chaînes logistiques, en se fondant sur les travaux présents dans la littérature.

2.1. Problématique des chaînes logistiques :

Selon [Shen et al.1999], les systèmes de production du XXI^e siècle devront développer un avantage concurrentiel durable et ce en satisfaisant aux nécessités suivantes :

- Intégrer les différents secteurs de l'entreprise, c'est à dire faire fonctionner ensemble les différentes parties de l'entreprise
- Organiser l'entreprise de façon distribuée en ne s'appuyant pas sur une hiérarchie pyramidale
- Evoluer dans des environnements hétérogènes en tenant compte des différents types de consommateurs
- Tenir compte de *l'interopérabilité*
- Avoir une structure d'entreprise ouverte et dynamique pour s'adapter aux changements dans l'environnement
- *Coopérer* avec les autres entreprises de l'environnement
- Intégrer les personnes aux matériels et aux logiciels
- Avoir un *comportement agile*, c'est à dire que l'organisation doit avoir la capacité de prospérer dans un environnement d'affaires constamment changeant et imprévisible
- Vivre avec une organisation à taille dynamique
- Etre tolérant aux pannes.

2.2. Définition d'une chaîne logistique :

D'après [Swaminathan *et al.* 1998], une chaîne logistique se définit comme un réseau d'entités d'affaires **autonomes** ou **semi-autonomes**, responsables **collectivement** des activités d'acquisition, de fabrication et de distribution associée d'un produit ou d'une famille de produits (Figure 1)

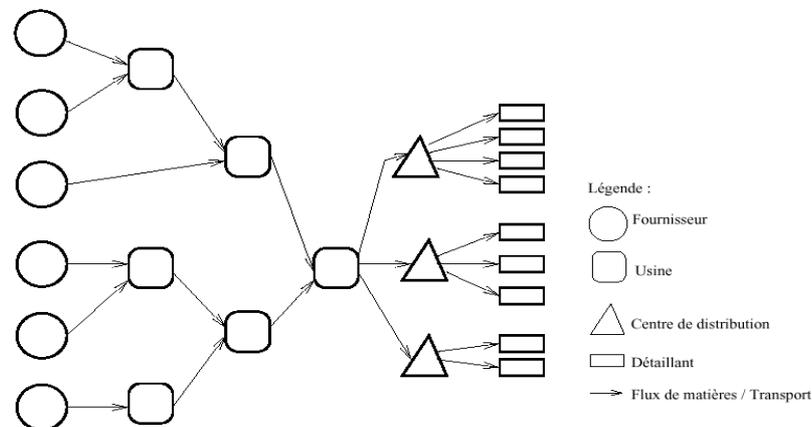


Figure 1. Exemple simplifié d'une chaîne logistique [Swaminathan *et al.* 1998]

Cette figure illustre une chaîne logistique simplifiée, mais en réalité elle peut être beaucoup plus complexe (Figure 2). De façon idéale, la chaîne devrait :

- S'étendre le plus possible entre le fournisseur de matières premières et le client final. Actuellement, l'entreprise cherche à optimiser les activités entre « *le client de son client et le fournisseur de son fournisseur* », mais à l'avenir elle cherchera à faire de même entre *l'acteur le plus en amont* (par exemple la mine qui extrait les matières premières) et *l'acteur le plus en aval* (le détaillant).

- Etre composée de petites unités spécialisées dans leur *métier de base* et *externalisant toutes leurs activités annexes*. Par exemple, une unité de production se focalisera sur la partie technique de la production et fera externaliser la conception des produits, la comptabilité, la gestion des approvisionnements, etc.

A cet effet, une chaîne logistique d'une entreprise va avoir la forme suivante:

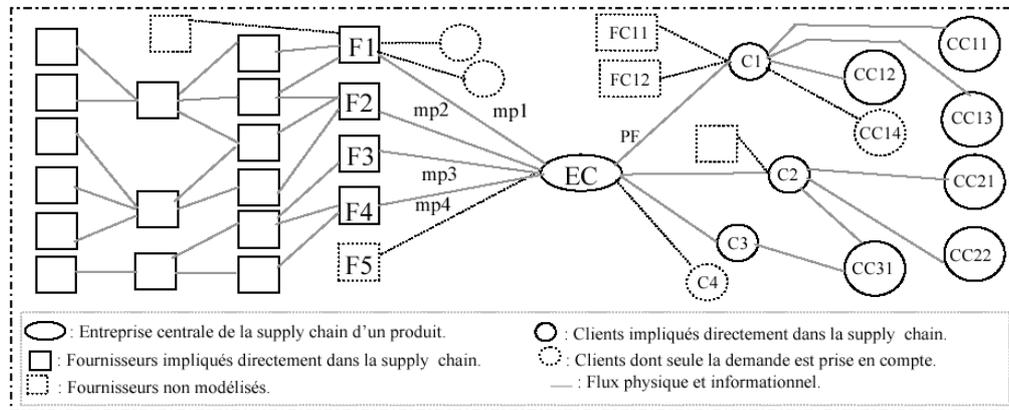


Figure 2. Exemple d'une chaîne logistique étendue [Nfaoui et al. 2006]

[Simchi-Levi et al. 2007] définissent le Supply Chain Management (SCM¹) comme « un jeu d'approches utilisées pour *intégrer* efficacement les fournisseurs, les producteurs, les entrepôts et les magasins, de manière à ce que les marchandises soient produites et distribuées au *bon endroit, au bon moment, en bonne quantité, en bonne qualité, et au meilleur coût possible*, afin de minimiser les coûts du système tout en satisfaisant les niveaux de service requis ».

Le SCM implique donc des entités à la fois autonomes mais aussi en forte interaction dans un environnement de plus en plus dynamique et donc leur gestion nécessitera une *coordination minutieuse* et une *synchronisation parfaite* si l'on veut réussir une optimisation des chaînes au sein d'une entreprise [Giard et Mendy 2006]. [Lehoux et al. 2008] ont proposé une approche collaborative gagnant-gagnant pour permettre l'optimisation des opérations dans une chaîne logistique.

Plusieurs technologies de l'information servent d'outils pour optimiser la SCM. Selon [Shapiro 2000], on peut distinguer deux grandes classes :

- les technologies de l'information *analytiques* impliquant l'implémentation et l'application de plusieurs modèles mathématiques (comme les statistiques, la recherche opérationnelle, les systèmes OLAP pour l'agrégation de données...)
- les technologies de l'information *transactionnelles* concernant l'acquisition et le traitement et la communication de données brutes à propos de la chaîne logistique d'une entreprise (comme les EDI², DSS³, ERP⁴, E-commerce...).

¹ SCM : Supply Chain Management (gestion de chaînes logistiques) se définit de manière très proche de la gestion de la logistique (logistics management) [Simchi-Levi et al. 2007].

² EDI : Electronic Data Interchange (échange électronique de documents)

Concernant les approches analytiques de modélisation du SCM, [Beamon B. M., 1998] a dressé un panorama de ces approches. Dans son étude, l'auteur distingue 4 grandes catégories de modèles, à savoir :

- *Les modèles analytiques déterministes* : à titre d'exemple, le modèle déterministe et non linéaire de programmation mathématique en nombre entiers qui a été étendu par un autre modèle d'optimisation sous contraintes [Cohen et Lee, 1989].
- *Les modèles analytiques stochastiques* : à titre d'exemple, les modèles stochastiques pour calculer la valeur des variables aléatoires introduites dans le modèle mathématique [Pyke et Cohen, 1993].
- *Les modèles économiques* : à titre d'exemple, les modèles de la théorie des jeux pour analyser les relations fournisseurs-clients [Christy et Grout, 1994].
- *Les modèles de simulation* : à titre d'exemple, les modèles de simulation pour évaluer les effets des stratégies sur l'amplification de la demande [Towill 1991], [Swaminathan et al. 1998], les modèles de simulation des processus de production de services [Giard et Balin 2007].

Plusieurs autres travaux de recherche peuvent être cités [Dupont et al. 2008], [Ghedira et al. 2008]. Ils traitent de la performance d'une chaîne logistique équitable dans laquelle la marge sur coûts directs est répartie équitablement entre les divers intervenants. Cette chaîne est constituée d'un donneur d'ordres et d'un sous-traitant dans un contexte contractuel particulier qui est le contrat réservation de capacité, les objectifs étant de maximiser la marge de la chaîne et de définir les modalités d'une répartition équitable. Après avoir présenté la modélisation mathématique, les auteurs ont constaté que les fonctions objectifs dépendaient de plusieurs paramètres. Pour résoudre le problème, ils ont utilisé une méthode exacte. Néanmoins, cette méthode s'est avérée très coûteuse en terme de temps. De ce fait, ils ont choisi d'adopter une méta-heuristique de type "recherche tabou".

Ces différents travaux présentent plusieurs limites : ils ne modélisent pas les interactions entre les différentes entités, ils ne modélisent pas de façon explicite le comportement des entités, notamment leur caractère réactif, autonome et proactif. L'approche agent apparaît comme une technologie intéressante pour modéliser la complexité de la chaîne logistique et de ses comportements. Comme le montre la figure 3, la notion d'agent apparaît naturellement et le terme de « chaînes logistiques » va faire référence aux petits regroupements d'entreprises travaillant en partenariat.

³ DSS : Decision Support Systems (système d'aide à la décision)

⁴ ERP : Entreprise Resource planning (système d'intégration des activités d'une entreprise)

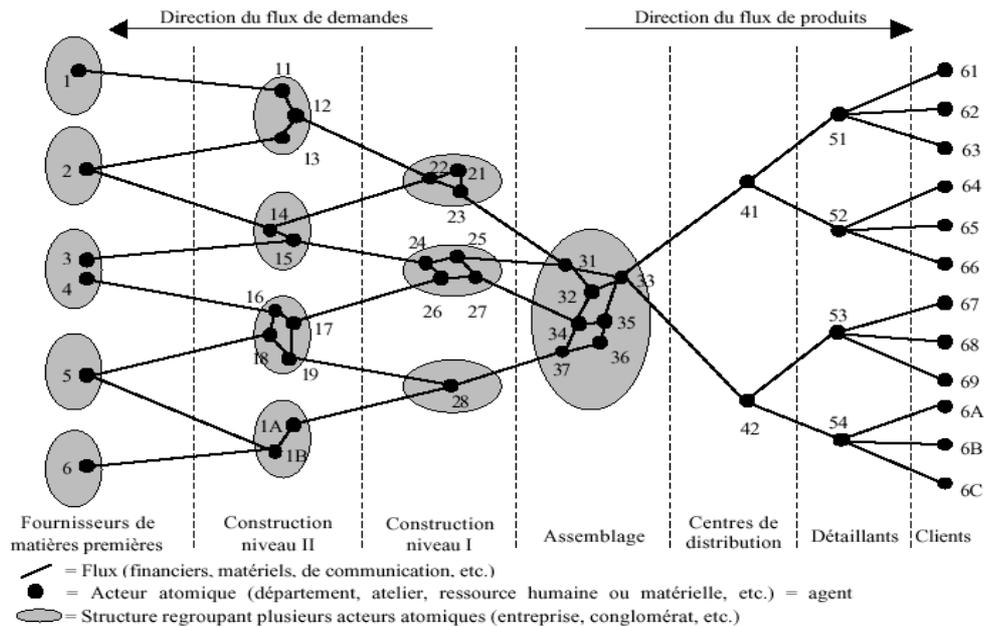


Figure 3. Exemple d'une chaîne logistique avec regroupement d'acteurs [D'Amours et al. 2006]

L'approche agent peut être vue comme une combinaison des deux classes proposées par Shapiro :

- la classe impliquant une approche analytique : à travers la notion d'agents intelligents qui sont des entités autonomes pouvant décider en temps réel.
- la classe impliquant une approche transactionnelle : à travers les protocoles d'interaction à la disposition des agents pour pouvoir coopérer.

2.3. Motivation :

Mark Fox a été l'un des premiers à proposer d'organiser la chaîne logistique comme un réseau d'agents intelligents [Dodd et al. 2001]. En effet, les chaînes logistiques sont composées de sous-systèmes de production *hétérogènes* qui se regroupent en vastes *coalitions dynamiques et virtuelles*. Des systèmes intelligents distribués tels que les systèmes multi-agents (SMA) permettent donc la représentation de l'autonomie de chaque membre de ce réseau d'entreprises manufacturières. Chaque partenaire (ou sous-système) de production poursuit ses buts individuels tandis qu'il satisfait à la fois à ses contraintes internes et externes.

En reprenant la définition d'un agent intelligent de [Wooldridge et al. 1995], un (ou plusieurs) agent(s) permet donc de représenter un acteur de la chaîne logistique (entreprise, atelier, etc.), et s'appuie sur quatre caractéristiques de base :

- Chaque acteur peut exécuter des tâches par lui-même sans intervention extérieure (**autonomie**).
- Chaque agent peut communiquer avec un autre agent pour, par exemple, lui transmettre une commande de produits ou de services (**capacité sociale**).

- Chaque agent est capable de modifier son comportement si le marché ou la concurrence évolue (**réactivité**).
- Chaque agent peut initier lui-même de nouvelles activités, comme par exemple décider de lancer un nouveau produit sur le marché (**pro-activité**).

De plus, les SMA ont l'avantage de permettre d'élaborer des systèmes de production *décentralisés* plutôt que centralisés, *émergents* plutôt que planifiés, et *concurrents* plutôt que séquentiels : de ce fait, ils permettent de diminuer les contraintes induites par un contrôle centralisé, planifié et séquentiel [Parunak 1996]. Sur ce sujet de production décentralisée, on peut citer les investigations menées par [Hahndel et al. 1994], où ils présentent une approche de planification distribuée afin de montrer qu'il est possible de résoudre dynamiquement certaines tâches principales d'un système de planification de production. Cette approche repose sur une *modélisation SMA* en utilisant un *protocole de négociation* qu'ils ont défini pour coordonner la planification des activités de production. Ce travail est très intéressant étant donné qu'il traite un environnement flexible c'est-à-dire que le SMA qui le modélise peut être étendu par l'introduction de nouveaux agents sans avoir à modifier la structure existante

2.4 Travaux existants :

Plusieurs travaux ont été menés sur ce sujet, faisant principalement intervenir différentes techniques issues des SMA dans le but de *modéliser la coordination* entre les acteurs d'une chaîne logistique. Cette modélisation est souvent finalisée par *une simulation* pour mettre en œuvre le modèle développé.

Par ailleurs, du point de vue opérationnel, plusieurs applications ont été développées et mises sur le marché ; nous en présenterons quelques unes à la fin.

2.4.1 Modélisation de la coordination de plusieurs agents dans la chaîne logistique :

Dans cette section, nous présentons trois grands types de coordination multi-agents : coordination par interaction structurées, coordination par planification distribuée, coordination par négociation

2.4.1.1 Coordination par interactions structurées :

[Fox et al. 1996] considèrent que les problèmes liés à la coordination sont dus essentiellement à deux caractéristiques :

- L'existence de plusieurs actions possibles que l'agent doit choisir. Chaque choix va affecter aussi bien l'environnement que l'agent lui-même, impliquant ainsi un changement de son état interne.
- L'ordre et le délai d'exécution des actions qui vont affecter aussi bien l'état de l'environnement que l'état de l'agent.

Le but de leur travail est de montrer que le problème de coordination peut être maîtrisé efficacement par *une représentation explicite des informations relatives aux processus d'interactions* entre les agents. Cependant, étant donné qu'un agent n'a qu'une information partielle sur son environnement, la coordination devient plus difficile. La solution proposée est alors la mise en place *de conversations structurées* qui rendent l'information facilement accessible par tous les agents.

Les auteurs ont développé un formalisme et un langage, le formalisme COOL, pour modéliser la coordination dans les SMA. Il comprend un composant de communication qui implémente une version étendue du langage KQML⁵. Il reprend le format de KQML pour les messages mais il laisse toute liberté aux développeurs tout en respectant les types d'actions communicatives. De plus, il n'impose aucun langage pour contenu du message.

Dans ce qui suit nous donnons un exemple illustrant le langage étendu de KQML :

```
(propose                ;; new
 :language KIF          ;; communicative
 :sender A              ;; action
 :receiver B
 :content (or (produce 200 widgets)
              (produce 400 widgets))
 :conversation C1      ;; new slot
 :intent (explore      ;; new slot
          fabrication possibility))
```

Dans cette conversation, 'propose' représente le type de message, ':sender' et ':receiver' sont les agents impliqués dans cette conversation. ':content' est le contenu du message dans lequel l'agent A demande à l'agent B s'il peut produire 200 ou 400 unités, 'conversation' est le nom par lequel la conversation est identifiée. ':intent' est utilisé lors de l'initialisation d'une conversation, dans cet exemple, il indique au receveur le sujet du message.

Dans COOL, un *agent* est une entité programmable qui échange des messages avec d'autres agents via des *conversations structurées*, change d'état, et exécute des actions. Il est identifié par un nom et se rattache à un interpréteur qui lui choisit et lui gère ses conversations.

Dans une conversation structurée, un agent échange des messages selon les conventions choisies. Les auteurs ont défini des *classes de conversation* et toute conversation réelle sera une instance de cette classe. La figure 4 donne un exemple d'une conversation schématisée par un automate à états finis pour une meilleure lisibilité.

⁵ Knowledge Query and Manipulation Language

```

(def-conversation-class 'customer-conversation
 :name 'customer-conversation
 :content-language 'list
 :speech-act-language 'kqml
 :initial-state 'start
 :final-states '(rejected failed satisfied)
 :control 'interactive-choice-control-ka
 :rules '((start cc-1)
          (proposed cc-13 cc-2)
          (working cc-5 cc-4 cc-3)
          (counterp cc-9 cc-8 cc-7 cc-6)
          (asked cc-10 )
          (accepted cc-12 cc-11)))

```

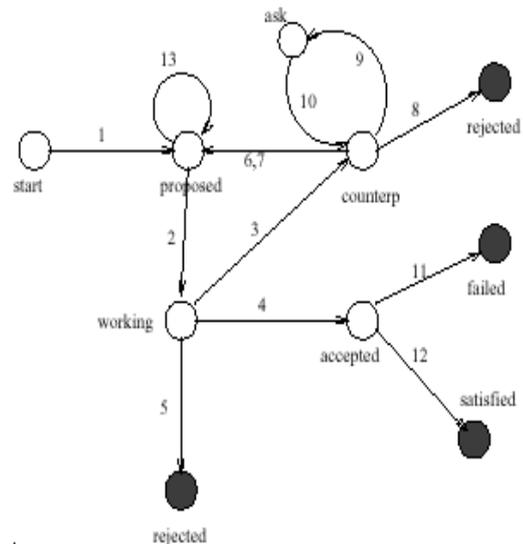


Figure 4. Définition d'une classe de conversation structurée et l'automate à états finis associé [Fox et al. 1996]

La partie gauche de la figure 4 représente une classe de conversation nommée 'customer-conversation'. Elle spécifie les états de début et de fin de la conversation, le langage utilisé et l'ensemble des règles que va suivre cette conversation. La partie droite schématise la conversation par un automate à état finis où les nœuds représentent les différents états de la conversation et les arcs représentent les règles de passage d'un état à un autre. Dans l'exemple, la conversation commence par l'état 'start'. Si la règle nommée 'cc-1' est appliquée, la conversation passe à l'état 'proposed' et reste dans cet état tant que la règle cc-13 est valide. La définition de celle-ci est donnée dans la figure 5. Elle implique l'agent 'logistics' qui envoie une proposition à un autre agent dont l'identifiant n'est pas encore défini '?agent'.

Les cc- # représentent les règles à considérer dans une conversation (# étant le numéro qui correspond à la règle en question).

Une classe de conversation spécifie :

- Les règles disponibles de la conversation, leurs *mécanismes de contrôle* et la *base de données locales* qui maintiennent la conversation. Ces règles sont indexées sur un ensemble fini de valeurs d'une variable donnée, *l'état courant*.
- *Le recouvrement des erreurs*. Ce composant spécifie comment on traite les incompatibilités relatives à un état.

Les règles d'une conversation décrivent les actions qui peuvent être exécutées quand la conversation est dans un état donné (cf. figure 5).

```

(def-conversation-rule 'cc-13
  :current-state 'proposed
  :received '(ask :sender logistics)
  :next-state 'proposed
  :transmit '(tell :receiver logistics
                  :sender ?agent
                  :conversation ?convn)
  :incomplete t)

```

Figure 5. Définition d'une règle dans une conversation structurée [Fox et al. 1996]

Notons que chaque classe de conversation décrit *une conversation du point de vue d'un seul agent*. Pour que plusieurs agents 'discutent', chaque conversation exécutée par un agent doit générer des séquences de messages que les autres classes de conversations peuvent traiter.

Selon les auteurs, une chaîne logistique ne fonctionne efficacement que si elle arrive à garantir une bonne coordination entre les différentes entités (agents) qui la composent. Mais malheureusement ce n'est pas toujours le cas étant donné que l'environnement est très souvent dynamique et imprévisible (matières premières pas encore livrées, salariés malades, équipements de production en panne,...). Dans ces cas de figures, de tels événements ne peuvent pas être traités par un seul agent car ils affectent parfois plusieurs autres agents de la chaîne logistique.

La figure 6 montre la forte dépendance entre les différents agents (classés selon deux niveaux) d'une chaîne logistique :

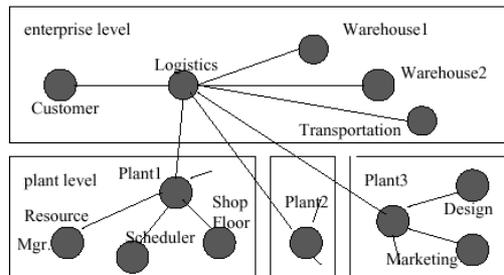


Figure 6. Chaîne logistique multi-niveaux [Fox et al. 1996]

Par exemple, l'agent 'Logistics' interagit avec l'agent 'Customer' à propos d'une commande. Une fois la commande reçue, l'agent 'Logistics' la décompose en différentes activités dont il va négocier l'exécution avec les différents agents ('Plants', 'Transportation Companies',...). Si un plan d'exécution est accordé, les agents participants vont se mettre au travail. Mais si l'un des agents échoue dans son engagement, il communique son échec à l'agent 'Logistics' qui va rechercher d'autres agents potentiels ou qui négocie à nouveau avec l'agent 'Customer'. D'autres événements peuvent se produire et donc cette information doit être communiquée aux autres agents pour pouvoir remédier à la situation. Cet exemple permet d'illustrer l'importance de la coordination entre agents pour pouvoir réagir en temps réel afin d'atteindre le but désiré.

Dans la chaîne logistique modélisée en figure 6, le plan global des conversations est exprimé par l'automate à états finis de la figure 7.

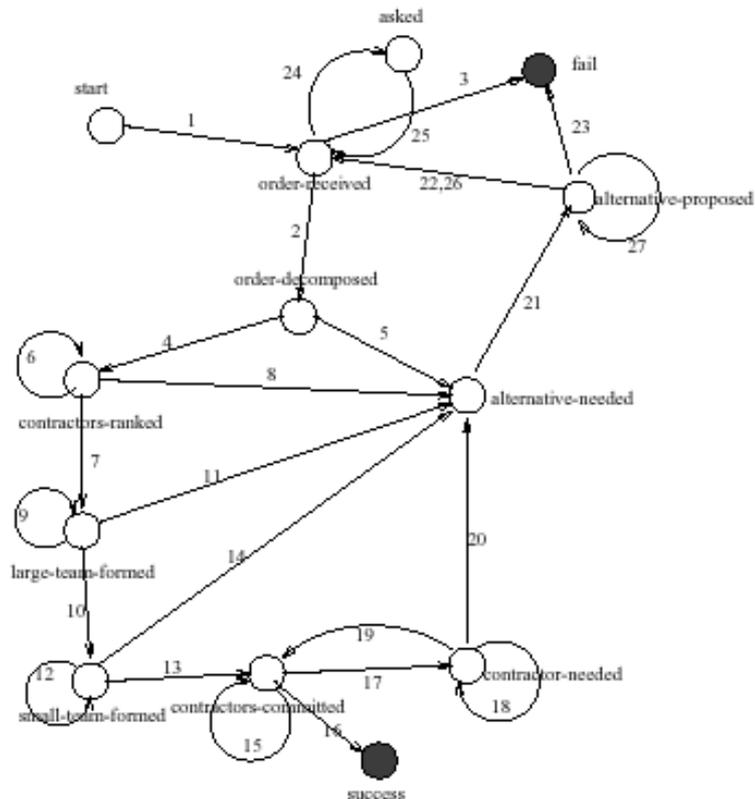


Figure 7. L'exécution du plan des conversations 'Logistics' [Fox et al. 1996]

Le processus commence par l'agent 'Customer' qui demande de passer une commande suivant la classe de conversation définie auparavant (cf. Figure 4), l'agent 'Logistics', ayant reçu cette demande, la décompose en différentes activités comme production, assemblage, transport etc.. Ceci est fait en exécutant l'une des règles associées à l'état 'order-received'. Si la décomposition ne peut pas être faite, alors le processus se termine et se met dans l'état 'fail'. En revanche, si la décomposition est faite avec succès, la conversation continue et se met dans l'état 'order-decomposed'. L'agent 'Logistics' cherchera ensuite les compétences nécessaires des agents pour accomplir ces activités, en dressant une liste ordonnée des contractants potentiels. Si l'agent 'Logistics' échoue dans sa recherche, il va relancer le processus avec un contrat légèrement modifié qui peut être exécuté par les agents disponibles (état 'Alternative-Needed'). Sinon si la recherche se termine avec succès, le reste du processus se fera selon deux étapes : la première étant de former une grande équipe de contractant (état 'Large Team Formed'). Cette équipe comprend tous les agents intéressés dans l'exécution des activités déterminées auparavant par l'agent 'Logistics'. Si cette équipe est formée avec succès, le processus passe ensuite à l'état 'Small Team Formed' qui va contenir exactement un contractant par activité. Une fois toutes les activités sont accomplies, le processus se termine avec succès.

Ce travail a été redéveloppé et détaillé dans [Fox et al. 2000] afin de fournir une architecture logicielle basée sur les agents en utilisant une structure de construction orientée agent (Agent Building Shell « ABS » [Barbuceanu et Fox, 1996]). Cette dernière fournit *des composants d'actes de communication génériques et réutilisables*. Les auteurs ont montré que l'utilisation de ces composants amènera à construire deux architectures qui correspondent d'une part à un travail coopératif complexe et d'autre part à la gestion des perturbations potentielles dues aux événements imprévus.

2.4.1.2 Coordination par planification multi-agents distribuée :

D'autres approches peuvent être intéressantes pour modéliser la coordination des activités d'une chaîne logistique comme les travaux qui se fondent sur la *planification multi-agents distribuée* [Balbo et al. 2002].

L'idée est de considérer la planification comme une spécialisation de la résolution distribuée des problèmes (RDP). Dans ce type de résolution, chaque agent élabore un sous-plan pour un sous-problème, ensuite tous les agents élaborent un *plan multi-agents* qui spécifie leurs actions futures et leurs interactions. La synthèse des plans exigera donc un raisonnement sur la manière dont les actions des différents agents peuvent interférer.

Selon [Balbo et al. 2002], cette synthèse de plans peut être faite de deux façons :

- Par une *planification multi-agents centralisée*, c'est-à-dire avec un agent coordinateur mais avec l'exécution des plans qui est distribuée. L'avantage majeur est la résolution des conflits et la convergence vers une solution globale ; en revanche elle présente des inconvénients comme le goulet d'étranglement du coordinateur au niveau de la communication et du traitement, ou encore la défaillance du système en cas de la défaillance du coordinateur.
- Par une *planification multi-agents distribuée* ; deux cas de figures sont possibles :
 - 1) la génération d'un plan centralisé après une synchronisation des sous plans,
 - 2) l'exécution directe des sous plans.Dans le premier cas, le coût de communication peut être important et sans garantie d'une convergence vers un plan global. En effet, une tâche est décomposée en plusieurs sous-tâches dont chacune est exécutée par un agent, chaque agent génère un plan ; après synchronisation, un plan global est construit. Dans le deuxième cas, on a ou bien une approche *orientée-tâche* (le problème est décomposé en sous-problèmes, chaque agent prend en charge un sous problème, génère un sous-plan et l'exécute), ou bien une approche *orientée-agent* (systèmes où plusieurs agents peuvent planifier et exécuter des plans de manière autonome et concurrente). Cette planification nécessite un formalisme (en général, les réseaux de Pétri) offrant une grande puissance d'expression du parallélisme, de la concurrence d'actions,...

Pour remédier à ces inconvénients, différentes approches ont été proposées en planification distribuée plus particulièrement en planification *incrémentale* où tout plan est inséré d'une

manière consistante dans un ensemble de plans déjà coordonné c'est-à-dire que chaque agent qui s'insère dans le système vient coordonner son plan avec le plan multi-agents courant [El Fallah *et al.* 1997].

Un processus de coordination par planification peut se résumer alors par le cycle suivant [Von Martial 1992] :

- Détection des situations d'interaction
- Résolution des situations d'interaction positives (détection des synergies entre situations) et négatives (conflits de ressources ou de buts).
- Prise en compte de la résolution par modification effective des plans.

En résumé, la cohérence d'un plan repose sur la détection de sous-buts en interactions positives et/ou négatives pour pouvoir anticiper la résolution des conflits potentiels.

Des modèles d'agents à comportement multiples, encore appelés agents hybrides, ont été proposés par [D'amours *et al.* 2007], [Forget *et al.* 2008] pour modéliser les processus de négociation et de planification de la chaîne logistique.

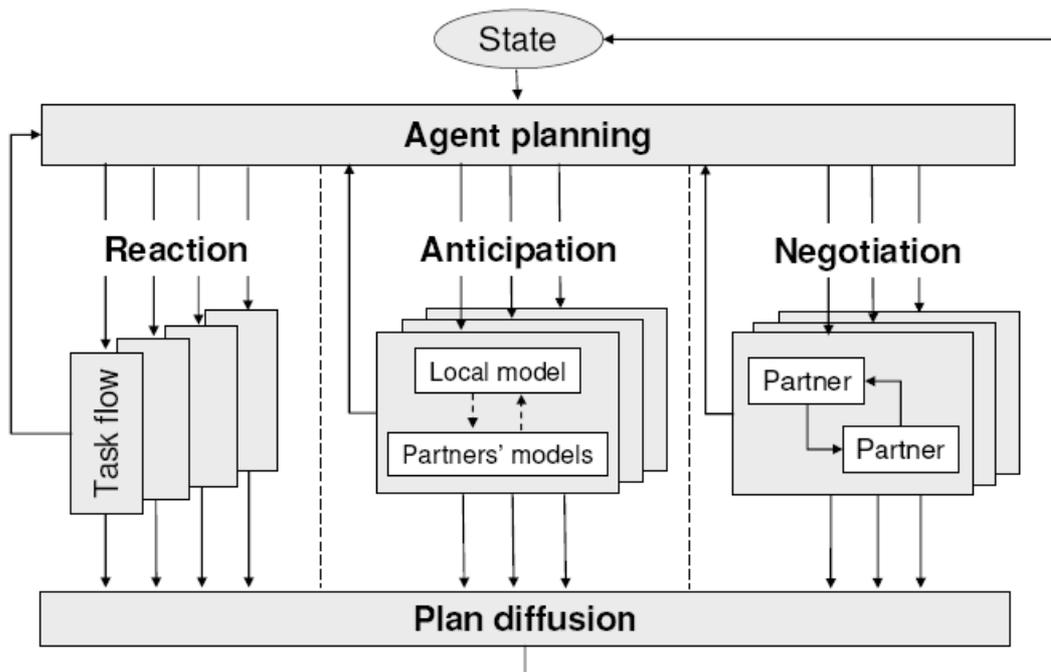


Figure 10 : *Modèle d'agent à comportements multiples* [D'amours *et al.* 2007]

Les auteurs définissent trois grandes classes de comportements qu'un agent peut avoir selon la situation étudiée :

- Le comportement à *réaction* : en utilisant des tâches où aucune nouvelle information n'a été détectée au cours du traitement. L'agent reconnaît un certain nombre de tâches qu'il peut utiliser pour répondre à une situation imprévue. Plusieurs algorithmes d'optimisation et fonctions objectifs peuvent être utilisés en fonction de la situation.

- Le comportement à *anticipation* : c'est une stratégie de planification qui utilise les modèles des partenaires en plus du propre modèle local de l'agent en question. Plus précisément, l'agent intègre les informations concernant ses partenaires dans son modèle d'optimisation.
- Le comportement à *négociation* : décrit les tâches d'envoi des propositions aux partenaires sous forme de plans alternatifs. Plus précisément, quand un agent n'est pas en mesure de répondre aux besoins de ses partenaires, il peut proposer des modifications de livraison ou encore fournir des produits substituants.

Ce modèle a été implémenté et testé sur une chaîne logistique forestière. Sa contribution réside essentiellement en deux points : le premier est la proposition d'un agent hybride conçu pour répondre spécialement aux problèmes de planification de production en utilisant les comportements multiples, le deuxième avantage est de pouvoir ajuster le comportement d'un agent selon le facteur externe constaté.

2.4.1.3 Coordination par négociation :

Selon [Jennings 2001], la négociation est la composante de base d'une interaction entre agents dans un SMA car les agents sont autonomes et intelligents c'est-à-dire qu'ils doivent chercher la solution *dynamiquement* pendant la résolution du problème (pas de solution prévue à l'avance) et selon l'évolution de leur environnement. A la fin de la négociation, ils doivent arriver à un accord commun sans conflits ou contradiction.

Plusieurs travaux sur les chaînes logistiques se sont servis des modèles de négociation proposés en SMA pour modéliser les interactions des acteurs d'une chaîne logistique et ce en utilisant le langage AUML.

AUML⁶ est une extension d'UML. [Huget 2002], [Huget *et al.* 2004] ont proposé une extension des diagrammes de classes afin de prendre en considération les caractéristiques des agents. AUML permet de représenter plusieurs niveaux d'abstraction mais deux niveaux sont couramment utilisés :

- Le niveau *conceptuel* représente une vue assez abstraite (attribut, rôles, ...) du système éliminant les détails de l'agent.
- Le niveau *implémentation* représente un détail sur le contenu d'un agent, son architecture.

[Nfaoui *et al.* 2006, 2008] se sont basés sur AUML pour proposer un modèle générique qui sert à délimiter les frontières de la chaîne logistique à modéliser (figure 2) et les relations entre ses différentes entités recensées. Ce modèle se résume en quatre étapes :

- Identifier le produit pour lequel est définie la chaîne logistique
- Identifier les fournisseurs qui y contribuent
- Identifier les clients destinataires

⁶ AUML : Agent Unified Modeling Language

- Enfin générer un modèle d'agent pour tous les acteurs identifiés

Pour passer de ce modèle générique au modèle basé sur les agents, ils commencent par modéliser les acteurs identifiés (l'entreprise centrale, les fournisseurs, les clients). Chaque acteur sera composé de 7 agents sauf les derniers fournisseurs et les derniers clients (détaillants) de la chaîne. Chacun de ces agents modélise une fonction ou un processus agissant sur la chaîne logistique.

Les 7 agents sont donc les suivants :

- *AgentPRC* : joue le rôle du processus client
- *AgentDis* : gère le stock de distribution
- *AgentPro* : joue le rôle du processus de production
- *AgentApp* : gère le stock de production
- *AgentAch* : gère le rôle du processus d'achat
- *AgentSCM* : joue le rôle des pratiques mises en place par les acteurs pour gérer leur chaîne logistique
- *AgentPer* : joue le rôle des perturbations permettant au modèle d'être ouvert et extensible.

Les modèles de classes aux deux niveaux , conceptuel et implémentation, proposés pour cette chaîne logistique sont illustrés respectivement dans figure 8.1 et figure 8.2 :

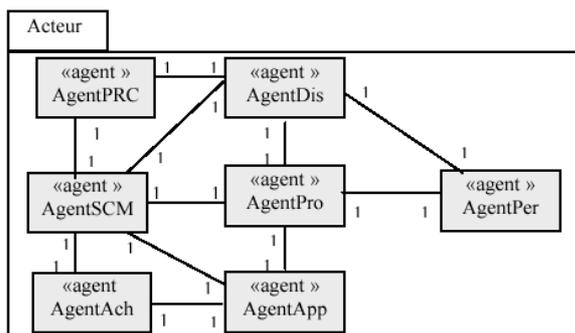


Figure 8.1 Diagramme de classes niveau conceptuel

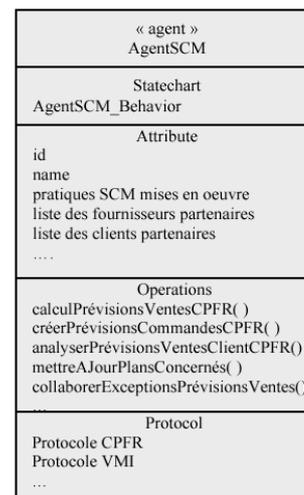


Figure 8.2 : Classe au niveau implémentation (pour 'AgentsSCM')

Une fois les agents identifiés, un processus de négociation doit alors être lancé afin d'optimiser le bon déroulement des activités de la chaîne logistique (la bonne quantité au bon endroit et avec le moindre coût).

Pour modéliser la négociation au sein d'une chaîne logistique, [Nfaoui *et al.* 2006] prennent en considération les aspects suivants :

- *L'objet de la négociation* : objet abstrait que les agents veulent négocier (délai de livraison, quantité et/ou prix...)
- *Le processus de décision* : le modèle que suit l'agent pour prendre une décision lors de la négociation.
- *Le langage de communication* : langage utilisé par les agents pour pouvoir échanger des messages. Ils utilisent le langage FIPA-ACL
- *Le protocole de négociation* : l'ensemble des règles qui régit la négociation, comme les participations légales ou encore la règle qui nous énonce la fin de la négociation

Dans le contexte de la gestion de chaîne logistique (SCM) les agents se montrent coopératifs et veulent atteindre un objectif commun qui n'est autre que l'agrégation des objectifs locaux, ils partagent et résolvent les problèmes. Pour cette raison, ils doivent fournir des réactions aux propositions qu'ils reçoivent.

Ces réactions peuvent être :

-*des critiques* : commentaires sur une partie de la proposition que l'agent accepte ou refuse.

-*des contre-propositions* : propositions possibles (refus/modification) en réponse de la proposition initiale.

Avec de telles réactions, l'agent pourra contribuer à trouver un accord. Pour ce faire, les agents doivent utiliser des protocoles qui dépendent de 3 paramètres :

- *Le secteur d'activité de la chaîne logistique à modéliser* : industrie textile, industrie de biens de consommation...
- *Méthodes et approches de la SCM utilisées pour la coordination*
- *Objets à négocier* : commande urgente, prévisions de vente...

[Nfaoui *et al.* 2006] ont défini un ensemble de protocoles de négociation propres à la SCM :

- Le protocole de négociation heuristique *ferme* : Dans certaines situations de négociation, les agents coopératifs sont obligés d'arriver à un accord. C'est le cas par exemple de deux agents « AgentSCM » d'acteurs différents collaborant sur les prévisions de ventes. Le nom « ferme » qualifie ce protocole puisqu'il mène toujours à un accord.
- Le protocole de négociation heuristique *récurive* : L'agent initiateur de la négociation envoie des messages (pas nécessairement identiques) de performative « PROPOSE » à tous les agents directs (amont et/ou aval) qu'il croit candidats à une négociation. Ainsi, il

lance plusieurs négociations indépendantes. Puisque les agents sont coopératifs, chacun d'eux recevant un message peut à son tour initier une négociation si nécessaire entre les agents se trouvant en amont et en aval afin de trouver une solution, d'où le nom « *récurive* » qualifiant cette heuristique.

- Les protocoles relatifs aux méthodes SCM comme le CPFR⁷, Transshipment : Le CPFR est un processus collaboratif mis en place entre les entreprises qui prend en compte toute la supply chain. Il s'agit d'une méthode de gestion globale de l'offre et de la demande.

Dans le *protocole de négociation heuristique* plusieurs propositions et contre-propositions sont échangées dans les différentes étapes afin d'arriver à un accord. Le nombre des contre-propositions est limité, une fois la limite est atteinte, les agents arrivent à un rejet.

Le fonctionnement est récapitulé dans le diagramme de séquence AUML (Figure 9) :

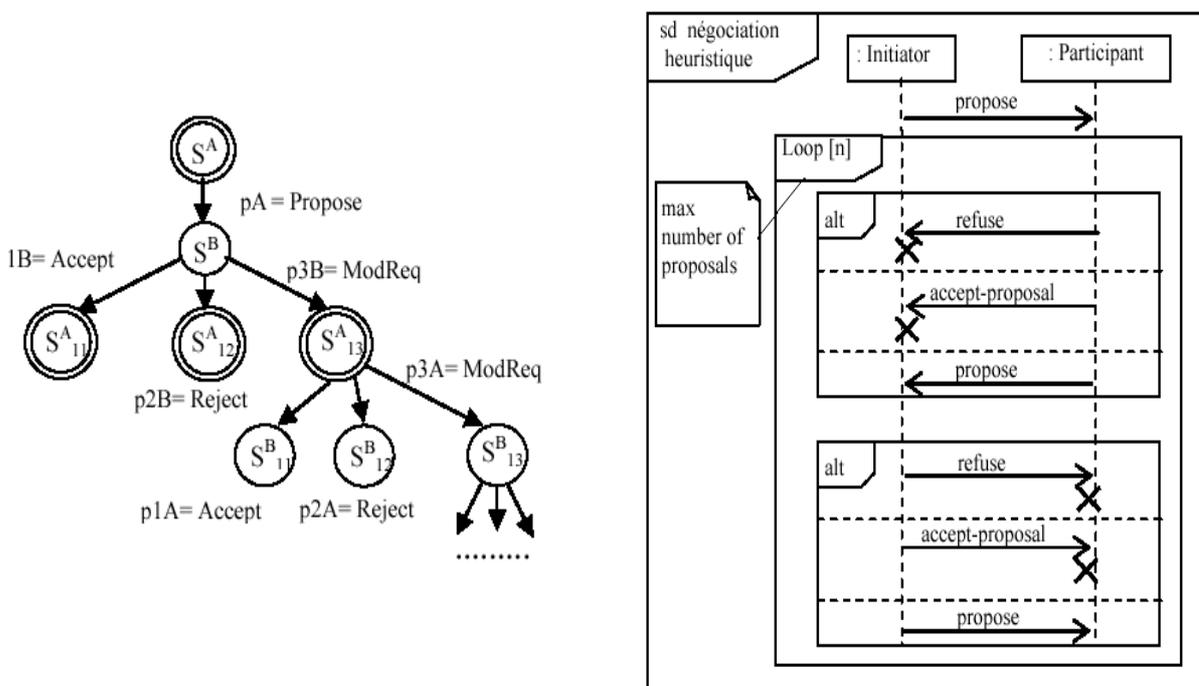


Figure 9 : Diagramme de séquences AUML : Protocole de négociation heuristique

L'agent commence la négociation avec la proposition pA, l'agent B peut répondre avec les contre-propositions p1B, p2B, p3B (en modification de la proposition). A son tour, l'agent A peut avoir le même type de réponse et ainsi de suite jusqu'à un certain niveau, paramètre du protocole de négociation.

⁷ CPFR : Collaborative Planning Forecasting and Replenishment

2.4.2 Les modèles de simulation multi-agents pour la chaîne logistique :

Selon [Chang *et al.* 2001], les modèles de simulation des chaînes logistiques présentent plusieurs avantages :

- Aider à comprendre la totalité des processus et les caractéristiques d'une chaîne logistique en utilisant les outils graphiques et les animations
- Rendre capable de tester la dynamique du système en utilisant les lois de probabilité ; l'utilisateur peut ainsi modéliser les événements imprévus et prévoir l'impact de ces événements sur la chaîne logistique.
- Diminuer considérablement les risques des changements qui peuvent surgir sur le processus de planification et ce par la simulation "what-if" (Si- alors), l'utilisateur peut essayer plusieurs alternatives avant d'effectuer tout changement dans la planification.

Les travaux de [Swaminathan *et al.* 1998] sont parmi les premiers sur ce sujet en modélisant la dynamique des chaînes logistiques par une approche multi-agents. L'apport de leur travail est la combinaison des méthodes analytiques et des méthodes de simulation afin de pouvoir modéliser respectivement l'aspect statique et l'aspect dynamique des chaînes logistiques.

[Nfaoui *et al.* 2008], ont étendu leur travaux [Nfaoui *et al.* 2006] en développant une architecture de simulation proactive à base d'agents qui est en cours de réalisation et qui, à terme, devrait aider à la prise de décision collaborative dans les chaînes logistiques. Selon les auteurs, les tests réalisés à ce jour sont très prometteurs et montrent qu'il est possible, en connectant les agents de cette architecture aux systèmes d'informations des acteurs (ERP – Entreprise Resource Planning à titre d'exemple) d'une chaîne logistique, d'apporter une aide importante dans la prise de décision collaborative en cas de situations d'urgence.

[Moyaux *et al.* 2004, 2008] utilisent l'approche multi-agents (figure 11) pour modéliser la variabilité de la demande des chaînes logistiques en fonction de la disponibilité de l'information (bullwhip effect). Plus précisément, ils modélisent un nouveau schéma pour passer les commandes (moins de stocks mais plus d'information) afin de réduire ce phénomène "coup de fouet" dans lequel la variabilité de la demande s'amplifie à chaque fois qu'on avance d'un niveau dans la chaîne logistique (vers l'amont) (figure 12) et induisant ainsi des coûts plus élevés dus au niveau élevé des stocks.

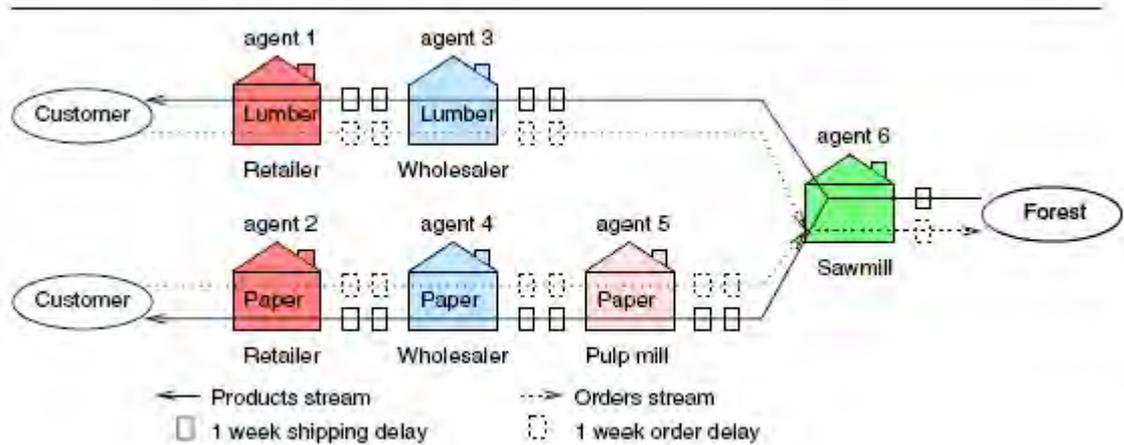


Figure 11 : modèle d'agents de la chaîne logistique forestière utilisé pour la simulation

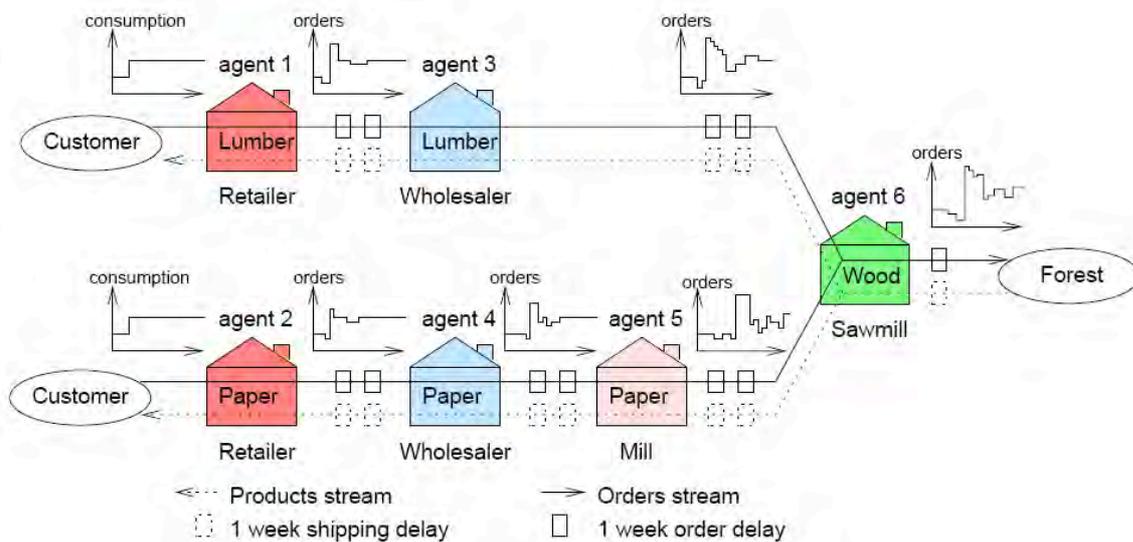


Figure 12 : modèle d'agents de la chaîne logistique forestière utilisé pour la simulation avec les courbes de variabilité de la demande

Ce processus a été simulé dans un jeu d'entreprise appelé QWSG (Quebec Wood Supply Game). La figure 12 montre le rôle pris par chacun des six "joueurs" (agents humains ou artificiels) du QWSG. Le jeu de simulation se déroule sur plusieurs étapes, chaque étape représente une semaine et est découpée en 5 jours joués parallèlement par tous les joueurs.

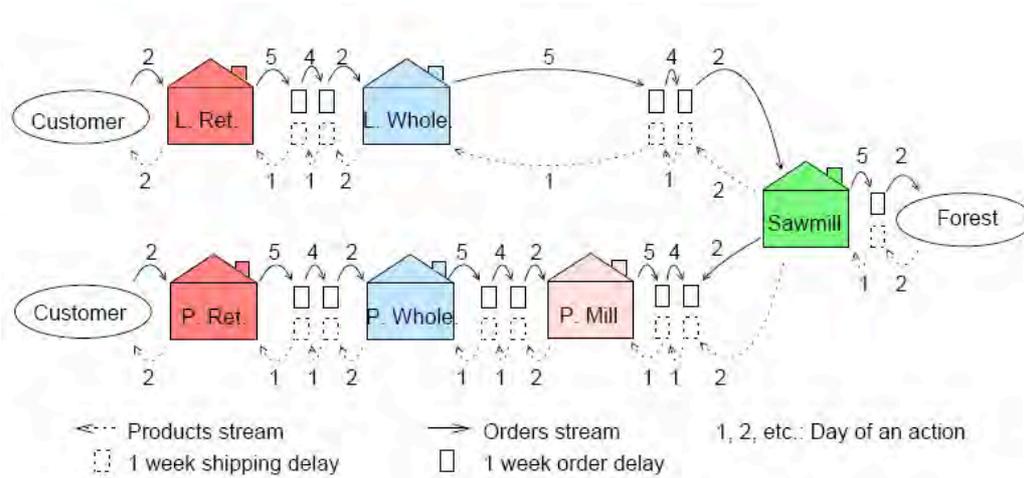


Figure 13 : L'exécution du jeu de simulation QWSG

La figure 13 décrit l'action menée parallèlement chaque jour par chaque joueur, i.e. au premier jour, les joueurs reçoivent les produits et les ajoutent à leurs stocks (ces produits ont été livrés par leurs fournisseurs deux semaines avant (puisque'il y'a eu un retard de livraison de deux semaines), ensuite les joueurs déplacent les produits entre les deux carrés appelés "shipping delay", à la fois entre eux même et entre leurs clients (ce déplacement est exprimé par les flèches dans la figure), et ainsi de suite pour les autres jours.

Ce jeu met en évidence la modélisation agent d'une chaîne logistique en proposant un modèle agent pour chaque entité.

[Montreuil et al. 2007, 2008] ont proposé un autre point de vue pour la simulation de la coordination dans les chaînes logistiques. L'étude se base sur le principe suivant : les agents identifiés dans une chaîne logistique sont classés en deux catégories : des agents *réactifs* et des agents *cognitifs* pour simuler respectivement *les activités opérationnelles* et *les activités à comportements complexes* adoptés dans le cadre de processus de prise de décision. La figure 14 montre un exemple de communication entre ces deux types d'agents dans le cadre d'une interaction entre un processus client et processus fournisseur.

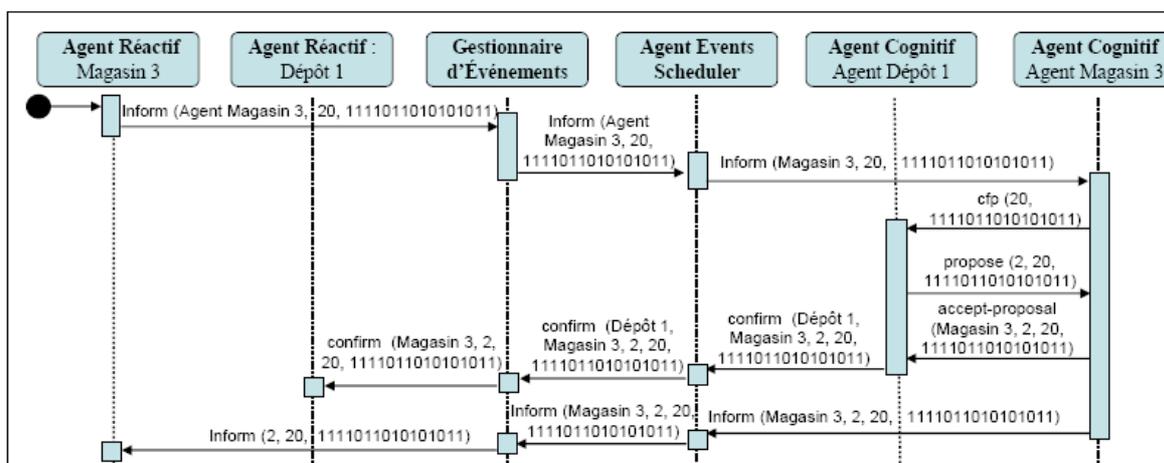


Figure 14 : Diagramme de séquence d'une interaction

Dans cet exemple, le message envoyé de l'agent réactif Magasin 3 vers l'agent cognitif Agent Magasin 3 contient des informations concernant la quantité ainsi que la référence de produits à commander (20 produits de référence 1111011010101011). Le message envoyé par l'agent cognitif Agent Dépôt 1 vers l'Agent Magasin 3 est composé de l'offre, de la quantité et de la référence produits (délai de 2 jours pour 20 produits de référence 1111011010101011). Après acceptation de l'offre, ces informations seront transmises aux agents réactifs correspondants pour la mise à jour des plans de livraison de l'agent réactif Dépôt 1 et de la liste des réceptions de l'agent réactif Magasin 3.

Notons que l'approche agent ne demeure pas le seul moyen pour développer des applications pour la simulation des chaînes logistiques comme le montrent les travaux de [Felix-texeira *et al.* 2003, 2004]. Ils ont développé une application pour simuler *la dynamique* d'une chaîne logistique, qui contrairement aux approches statiques classiques dans lesquelles les paramètres comme le délai de livraison et le stock moyen des matières premières sont utilisés en entrée, utilise ces données comme résultat d'un processus de simulation en rendant la chaîne logistique plus *flexible*. Ces travaux ont été complétés dans [Felix-texeira *et al.* 2007] par la proposition de deux applications complémentaires *scMod.exe* et *scSim.exe*, la première pour la modélisation des différents nœuds de la chaîne logistique et l'autre pour la simulation.

[D'Amours *et al.* 2006] ont établi une synthèse des applications qui ont été développées pour la modélisation de la coordination dans les chaînes logistiques, à savoir :

MASC : qui étudie les modes de coordination des acteurs de la chaîne logistique. Ces modes de coordination sont des appels d'offre à des soumissionnaires qui répondent selon leur capacité et leur charge de production respectifs. Les acteurs qui remportent ces appels d'offre forment ainsi la chaîne logistique permettant d'amener un produit au consommateur.

MASCOT : (Multi-Agent Supply Chain cOordination Tool) qui est une architecture reconfigurable, à niveaux et basée sur les agents pour la planification et la coordination en vue d'améliorer l'agilité d'un réseau logistique. A cet effet, ce système coordonne la production à travers les multiples installations (internes ou externes) de l'entreprise, et évalue les décisions de conception de nouveaux produits/sous-composants et les décisions d'affaires stratégiques.

OCEAN : (Organization and Control Emergence with an Agent Network) qui est un système de pilotage à architecture ouverte, décentralisée et basée sur des contraintes (distribution des ressources de production, distribution des données techniques, adaptabilité pour répondre à la dynamique de l'environnement). Le but de ce système est de montrer que des coopérations au niveau global peuvent apparaître à partir des compétitions au niveau individuel.

2.5 Conclusion :

Le panorama des travaux existants a montré que l'approche SMA offre des solutions prometteuses pour la modélisation et l'optimisation des chaînes logistiques. Cette approche permet de prendre en compte facilement les entités mêmes qui entrent en jeu dans la gestion

des chaînes logistiques, à savoir, les nombreux acteurs de la chaîne, leur comportement idiosyncratique, les interactions entre ces acteurs, les processus de négociation, l'environnement en perpétuel évolution, etc. Les principaux avantages de cette approche sont les suivants : 1) modélisation d'entités autonomes en interaction, qui doivent se coordonner pour atteindre un objectif global qui n'est autre que l'agrégation des objectifs individuels et ce via un processus de négociation, 2) modélisation des protocoles d'interaction et de coordination grâce à l'outil AUML et aux automates à états finis, 3) modélisation du comportement réactif des entités qui réagissent à un environnement en général dynamique.

3. Notre approche multi-agents :

Dans cette section, nous allons présenter une nouvelle méthode de coordination par formation de coalition que nous considérons comme mieux adaptée à la modélisation des chaînes logistiques dans leur globalité. Après avoir présenté la problématique, nous détaillerons le protocole de négociation mis en place et nous donnerons l'algorithme distribué de formation de coalitions.

Selon [Shehory et Kraus, 1998], une *coalition* se définit comme un groupe d'agents qui décident de coopérer afin de réaliser un but commun. A ce but est associée une valeur (utilité) partagée entre les différents agents formant la coalition. En nous appuyant sur les travaux de [Aknine *et al.* 2004], nous cherchons ainsi à modéliser les relations de partenariats entre les différentes entités tout en distinguant les coalitions selon deux niveaux d'abstraction:

- Coalitions internes : ce sont les coalitions qui se forment entre les différentes entités à l'intérieur de la même chaîne comme par exemple le regroupement de plusieurs usines de la même entreprise afin de faire face à une importante commande.
- Coalitions externes : ce sont les partenariats qui peuvent s'établir entre des entités de la chaîne et des entités externes à l'entreprise et ce dans le but de se procurer des ressources qui leur manquent ou des compétences dont elles ne disposent pas.

3.1 Exemple d'application :

Nous considérons une entreprise qui fabrique et commercialise des avions. Nous avons choisi ce domaine industriel en raison de la complexité de la chaîne logistique qui met en œuvre principalement la sous-traitance de plusieurs pièces et la multitude d'entités qui y contribuent. Cette entreprise est localisée principalement en France et possède plusieurs sites de production (usines) sur plusieurs continents. Elle possède également divers centres de distribution en Europe qui livrent différents marchés (Europe, Moyen-Orient ...). Par ailleurs, ses fournisseurs sont physiquement dispersés.

A l'intérieur de chaque usine (figure 15), nous pouvons identifier plusieurs autres unités qui travaillent en étroite collaboration les unes avec les autres. Ce qui complique encore une telle situation ce sont les liens qui existent entre ces entités et les entités des autres chaînes logistiques, d'où d'éventuels retards ou dysfonctionnements au niveau d'un point de la chaîne qui peuvent se propager au niveau de tous les points qui lui sont reliés.

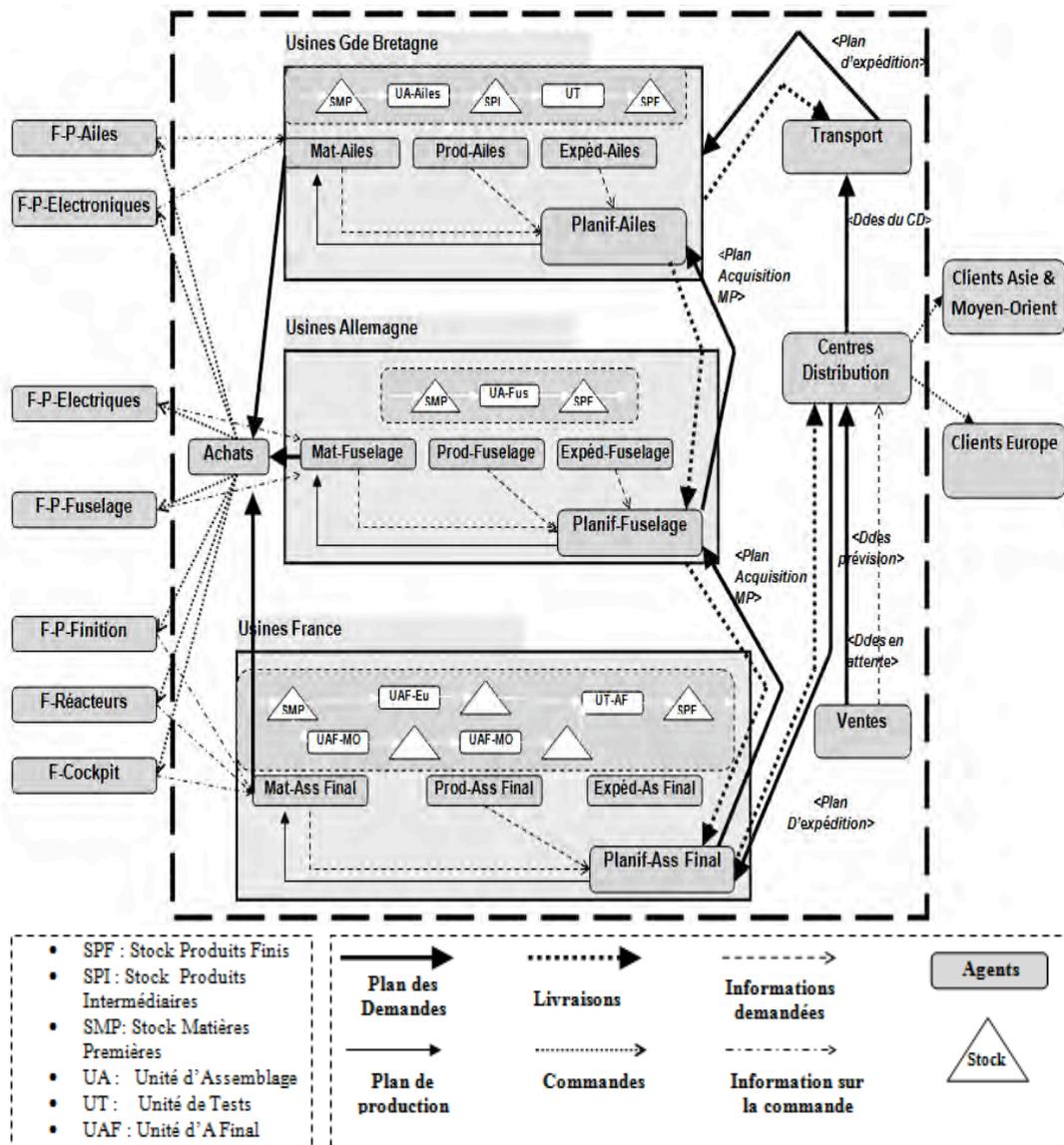


Figure 15 : Exemple d'une chaîne logistique et des interactions entre les entités

3.2 Architecture d'un agent :

Nous reprenons la structure classique d'un agent cognitif, (Figure 16), à savoir fonctions de perception, de décision et d'action, dans laquelle nous explicitons la base de connaissances, les structures de données objectifs et les états de l'agent. Un agent est défini selon la structure suivante :

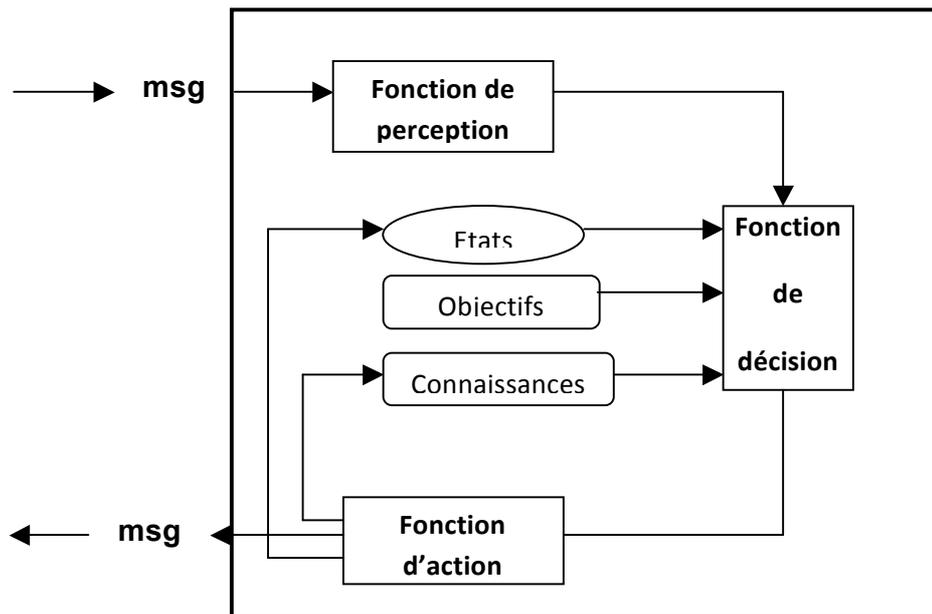


Figure 16 : Architecture d'un agent

3.3 Protocole d'interaction :

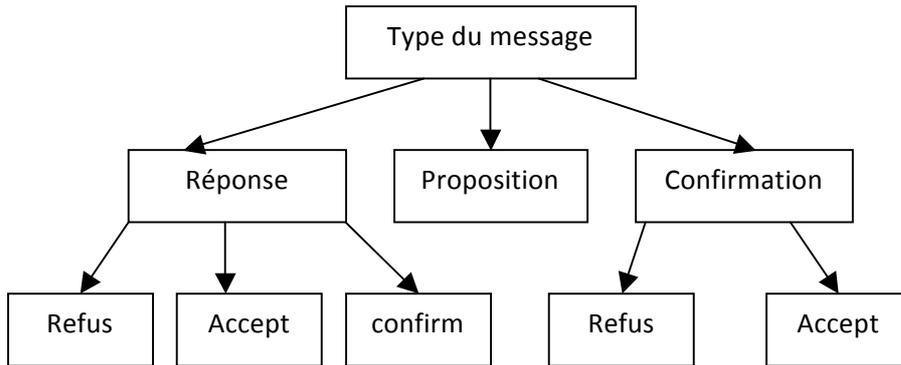
Tous les agents du système communiquent via des messages et suivent des règles prédéfinies dans leur protocole. Nous décrivons aussi bien la structure des messages ainsi que les règles du protocole. Nous nous appuyons sur la théorie des actes de langage [Austin, 1970], [Searle, 1972] pour formaliser les messages échangés. Un message est représenté par le vecteur suivant :

$$\text{Msg} = \langle \text{MsgType}, \text{SenderID}, \text{ReceiverID}, \text{Content} \rangle,$$

Le champ « Content » comprend toutes les informations sur la coalition comme :

- La ou les tâches pour lesquelles on souhaite former un groupe d'agents
- Les agents sollicités
- Les agents acceptés
- Les agents confirmés
- Les délais

Chaque message est d'un type spécifique (appelé performatif dans la théorie des actes de langage) : Réponse / Proposition / Confirmation comme schématisé par le diagramme suivant:



Chaque agent détient quatre listes de messages qui sont les suivantes :

- *ProposedCoalitions* : liste des coalitions proposées
- *AcceptedCoalitions* : liste des coalitions acceptées (temporairement)
- *RefusedCoalition* : liste des propositions qui ont été refusées
- *ConfirmedCoalition* : coalition confirmée (une seule)

Chaque agent ne peut initier qu'une seule coalition à la fois et, de la même façon, il ne peut en confirmer qu'une seule aussi. Un agent ayant confirmé une coalition peut accepter temporairement une autre coalition (car une coalition confirmée peut être annulée par un refus d'un autre agent). Nous supposons qu'il n'y a pas de conditions sur le nombre de coalitions acceptées.

Dans ce qui suit, nous présentons les règles d'interaction selon le type des messages échangés.

▪ ***Proposition :***

Les agents coordonnent les activités pour qu'il n'y ait qu'une seule proposition de formation de coalitions à la fois mais en revanche un agent peut proposer une formation de coalitions à n'importe quel moment. Il envoie un message aux agents potentiels avec tous les paramètres nécessaires (la tâche à exécuter, le délai...), pour que les agents sollicités puissent prendre une décision. Il ajoute la coalition proposée dans sa liste *ProposedCoalitions* et attend la réponse des agents sollicités.

▪ ***Réception de proposition :***

A la réception d'une proposition, l'agent sollicité l'analyse en fonction de ses capacités et des tâches qui peuvent précéder la tâche concernée ensuite il envoie sa réponse. Sa réponse peut être l'une de ces trois alternatives :

- Acceptation temporaire de proposition : l'agent indique à l'initiateur de la proposition qu'il accepte temporairement la coalition et l'ajoute à sa liste *AcceptedCoalitions*.
- Refus de proposition : l'agent indique à l'initiateur qu'il refuse de participer à cette coalition.
- Confirmation de proposition : l'agent accepte la coalition et l'ajoute à sa liste *ConfirmedCoalitions*.

▪ **Réception d'une réponse :**

Un agent initiateur peut recevoir à tout instant une réponse concernant une de ses propositions de coalitions stockée dans sa liste *ProposedCoalitions*.

- Dans le cas d'une acceptation de proposition, l'agent initiateur ajoute l'agent qui a accepté la coalition dans sa liste *AcceptedAgents* (encapsulée, par exemple, dans le contenu du message). Si le nombre d'agents nécessaires à la coalition est atteint ou que l'objectif de l'agent initiateur est satisfait, il envoie une demande de confirmation aux agents ayant temporairement accepté.
- Lorsqu'un agent sollicité refuse la coalition et que l'agent initiateur estime que ce refus compromet la formation de la coalition, il ajoute la proposition à la liste *RefusedCoalitions*. Il envoie un message d'annulation de la formation à tous les agents concernés. Il pourrait peut être la reprendre ultérieurement en substituant ses objectifs mais ce cas ne sera pas discuté durant cette étude.
- Dans le cas d'une confirmation, l'agent initiateur l'ajoute à sa liste *ConfirmedAgents*.

▪ **Demande de confirmation :**

Après avoir reçu toutes les réponses temporaires, l'agent initiateur demande aux agents sollicités de confirmer définitivement leurs décisions c'est-à-dire qu'ils peuvent :

- Refuser d'intégrer la coalition et dans ce cas l'initiateur doit relayer l'annulation aux autres membres de la coalition.
- Accepter de participer et s'engager dans la coalition.

▪ **Confirmation définitive :**

Après réception de toutes les confirmations définitives, l'initiateur envoie un message à tous les agents pour les informer qu'une coalition est formée.

En considérant les primitives suivantes (cf. tableau 1), le protocole d'interaction est représenté par deux diagrammes d'états : le premier décrivant le comportement de l'agent initiateur (I) (figure 17) et le deuxième décrivant le comportement des agents sollicités (S) (figure 18).

Primitive	De → Vers	Description
(1,0)	I→S	Proposition
(2,1)	S→I	Acceptation Proposition
(2,2)	S→I	Refus Proposition
(2,3)	S→I	Confirmation Proposition
(3,1)	I→S	Demande Confirmation
(3,2)	I→S	Annulation Proposition
(4,1)	S→I	Confirmation
(4,2)	S→I	Annulation
(5,1)	I→S	Confirmation Coalition
(5,2)	I→S	Annulation Coalition

Tableau 1. *Les primitives du protocole d'interaction*

Le premier diagramme explicite les échanges et les transitions d'états possibles de l'initiateur avec les agents sollicités. Il commence par l'état (*S*) et envoie un message annonçant une proposition pour former une coalition (*Primitive (1,0)*), et se met dans l'état d'attente d'une réponse (*waitForResponses*). Tant qu'il n'a pas reçu la totalité des réponses des agents sollicités (*not (everybodyAccept)*) il reste dans cet état en recevant les acceptations et les confirmations (*Primitive (2,1)* et *Primitive (2,3)*) et les met dans les listes associées précédemment décrites (*AcceptedCoalitions*) et (*ConfirmedCoalitions*). S'il reçoit une confirmation de tous les agents sollicités (*Primitive (2,3) & (everybodyConfirm)*), il passe directement à l'état (*coalitionFormed*) et envoie des messages de confirmation de la coalition aux agents sollicités (*Primitive (5,1)*), Sinon il se met dans l'état de (*waitFor Confirmations*). Il peut se mettre aussi dans l'état (*propositionCancelled*) s'il ne reçoit que des acceptations temporaires ou des annulations. Il peut aussi recevoir des annulations lorsque certains agents ne confirment pas et dans ce cas il passe à l'état (*confirmationCancelled*).

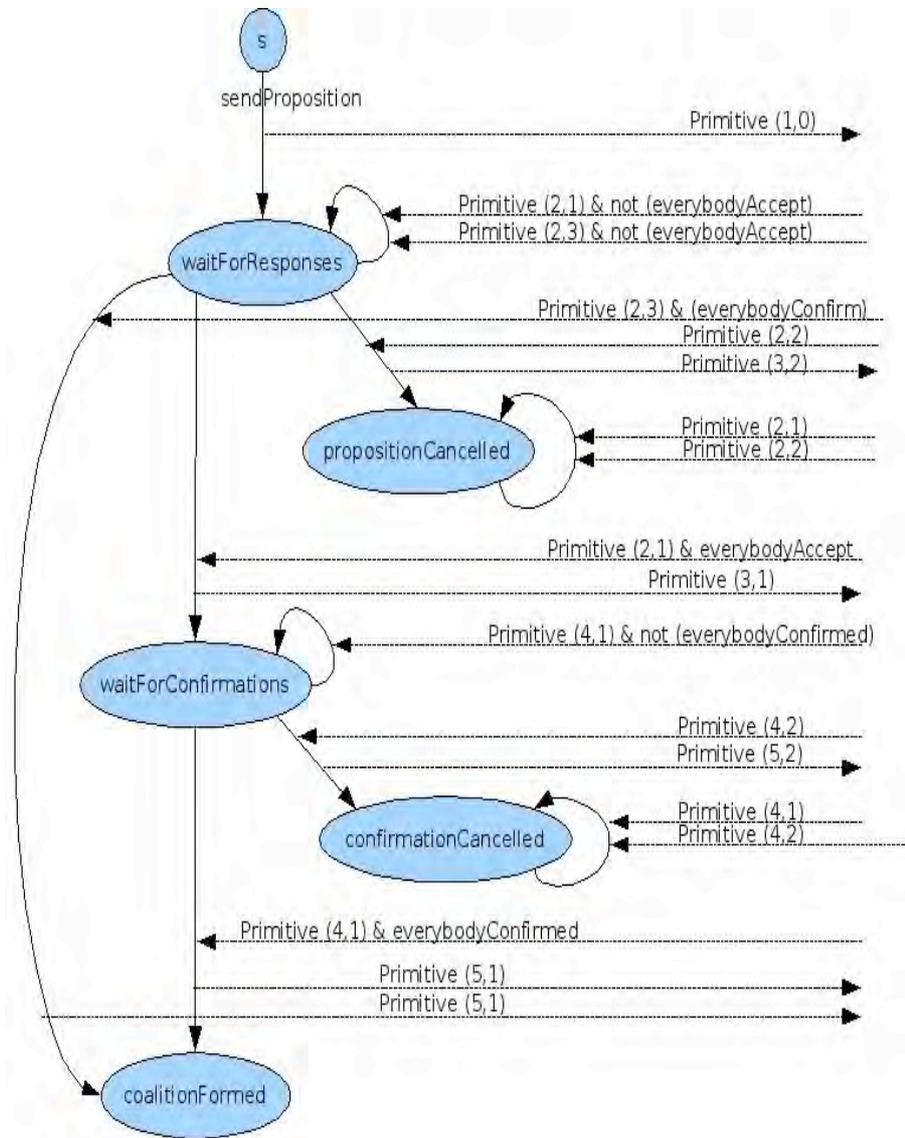


Figure 17 : *Comportement de l'initiateur de la formation de coalitions*

Le deuxième diagramme (figure 18) décrit les mêmes échanges mais du côté agent sollicité en utilisant les mêmes primitives.

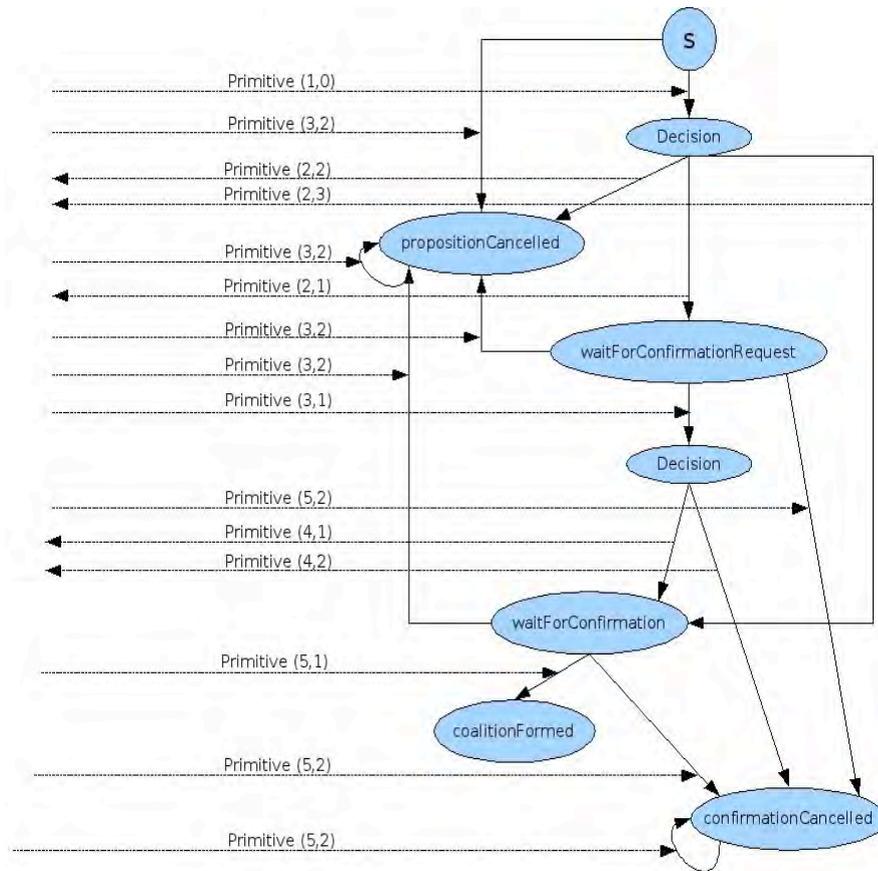


Figure 18 : *Comportement de l'agent sollicité pour la formation de coalitions*

Ces protocoles de communication entre agents ont l'avantage de représenter explicitement les échanges entre les agents. S'appuyant sur la théorie des actes de langage, ils permettent de définir une sémantique opérationnelle des messages échangés facilitant ainsi leur implémentation. Ces protocoles indiquent clairement la possibilité de former plusieurs coalitions temporaires. L'avantage réside dans le fait que, si une coalition temporaire est annulée, le processus de formation de coalitions n'a pas à reprendre à partir du début mais repartira à partir de coalitions temporaires partiellement formées.

3.4 Principe de l'algorithme proposé :

Une chaîne de tâches est représentée par un graphe orienté acyclique $\langle TC, E \rangle$, tel que $TC = \{t_1 \dots t_m\}$ représente l'ensemble des tâches complexes pour lesquelles un agent peut éventuellement lancer un processus de formation de coalitions afin de les réaliser. $E = \{(t_i, t_j)\}$ est l'ensemble des arcs reliant la tâche t_i qui précède la tâche t_j avec $(\emptyset, t_1) \in E$ comme tâche de début, et $(t_m, \emptyset) \in E$ comme tâche de fin. Nous définissons, à présent, les concepts suivants :

- soit $A = \{a_1 \dots a_n\}$ l'ensemble de tous les agents du système. Cet ensemble est formé de deux sous ensembles $A = A^I \cup A^E$ comprenant respectivement les agents internes $A^I = \{a_1 \dots a_k\}$ et les agents externes $A^E = \{a_{k+1} \dots a_n\}$
- soit $T = \{t_1 \dots t_c\}$ l'ensemble des tâches de la chaîne tel que $TC \subseteq T$;
- pour la formation de coalitions, un agent détermine, selon un modèle de préférences, un ensemble d'agents potentiels noté AP incluant lui-même ;
- après la formation de coalitions, on note par AR l'ensemble des agents de la chaîne ne participant pas à des coalitions mais contribuant à l'une des activités de cette chaîne ;
- chaque agent a_i a un vecteur de capacités noté $Q^i = \langle q_1^i \dots q_p^i \rangle$;
- chaque activité ou tâche complexe t_s a un coût d'exécution noté C_{t_s} qui n'est autre que l'agrégation des différents coûts des agents contribuant à son exécution. Ce coût représente la fonction d'utilité de l'agent. Nous supposons que cette fonction d'utilité est définie par une fonction linéaire ;
- nous définissons aussi, $CP_{a_i, a_j}^{t_s}$, le coût de passation du résultat de l'activité t_s de l'agent a_i à l'agent a_j . Ce paramètre sert à optimiser le coût total de la chaîne.

L'objectif de tout agent est de minimiser son coût individuel et par conséquent de minimiser le coût global de la chaîne. Dans le cas d'une coalition de plusieurs agents et afin de déterminer le coût d'une activité, une variable binaire est utilisée :

$$Z_{t_s}^{a_i} = \begin{cases} 1 & \text{si l'agent } a_i \text{ est sélectionné pour contribuer à la tâche } t_s \\ 0 & \text{Sinon} \end{cases}$$

Avec $a_i \in AP$ et $t_s \in TC$

Le but du système revient à réduire le coût total $C.T$ en résolvant le problème suivant :

$$\text{Min } C.T = \sum_{a_i \in AP} \sum_{t_s \in TC} z_{t_s}^{a_i} c_{t_s}^{a_i} + \sum_{(t_s, t_l) \in E} \sum_{a_i, a_j \in AP} z_{t_s}^{a_i} z_{t_l}^{a_j} cp_{a_i, a_j}^{t_s} + \sum_{a_i \in AR} \sum_{t_s \in T} c_{t_s}^{a_i} \quad [1]$$

$$\text{s/c : } \quad \forall t_m, t_l \in TC, (t_m, t_l) \in E \quad (a)$$

$$\sum_{t_s \in T} q_{t_s}^{a_i} z_{t_s}^{a_i} \leq Q^i \quad (b)$$

(a) pour satisfaire la contrainte d'ordre entre les tâches.

(b) La somme des capacités de l'agent a_i investies dans toutes les tâches ne doit pas dépasser sa capacité totale.

Les deux premières parties de [1] représentent le coût des activités pour lesquelles des coalitions sont formées. La troisième partie décrit le coût du reste des activités de la chaîne.

Nous définissons la préférence d'un agent par rapport à un autre par la matrice de préférences suivante :

$$\prod_{a_i}^p = \begin{pmatrix} y_{i1}^1 & y_{i2}^1 & \dots & y_{in}^1 \\ y_{i1}^2 & y_{i2}^2 & \dots & y_{in}^2 \\ \dots & \dots & \dots & \dots \\ y_{i1}^p & y_{i2}^p & \dots & y_{in}^p \end{pmatrix} \text{ avec } \begin{cases} i \text{ l'indice des agents } \in N \\ p \text{ l'indice des critères} \end{cases}$$

Chaque élément y_{ij}^k de la matrice représente la préférence d'un agent a_i pour l'agent a_j selon le critère k . Pour obtenir la préférence multicritère d'un agent pour un autre, cette matrice est réduite à un vecteur à l'aide d'un opérateur d'agrégation qui peut être *la somme pondérée* ou *l'intégrale de Choquet* [Grabisch, 1996]. Nous avons choisi d'utiliser le deuxième opérateur qui a l'avantage, contrairement à la somme pondérée, de ne pas biaiser le résultat de l'agrégation lorsque les critères ne sont pas indépendants. Il se base sur le principe suivant : l'agent a_i classe par ordre décroissant la colonne de ses préférences pour un agent a_j selon des critères spécifiques y_{ij}^k , étant donné les poids *individuels* et *collectifs* des différents critères k notés $\mu(E_k)$ définis par le concepteur du système (avec E_k un ensemble composé d'un seul ou de plusieurs critères). Il calcule ensuite le vecteur de ses préférences $\prod_{a_i} = (x_{ij})$ comme suit :

$$x_{ij} = \sum_{0 \leq k \leq p} (y_{ij}^k - y_{ij}^{k+1}) \mu(E_k) \quad \text{Avec } y_{ij}^{p+1} = 0 \text{ et } y_{ii}^k = 0 \quad [2]$$

Nous définissons un autre concept : *l'indice d'attraction* qui peut être *unilatérale* (U-Att) ou *bilatérale* (B-Att).

- **L'attraction unilatérale** représente une mesure de la force qui permet à un agent de convaincre les autres agents, pas encore membres, de rejoindre la coalition. Pour pouvoir la déterminer, l'agent classe les différentes valeurs du vecteur

$\prod_{a_i} = (x_{ij})$ selon un ordre décroissant. Formellement, cette valeur est définie comme suit :

$$U\text{-Att}(a_i) = \sum_{j=1}^n (x_{i,j} - x_{i,j+1}) \mu(A_j) \text{ avec } i \neq j \quad [3]$$

Dans [3], $\mu(A_j)$ étant un poids prédéfini pour l'agent a_i ou pour l'ensemble d'agent

$$A_j \in A = \{a_1 \dots a_n\}$$

- **L'attraction bilatérale** représente à la fois le désir d'un agent de rejoindre une coalition et le désir de cette coalition d'intégrer cet agent. Elle est définie comme suit :

$$B\text{-Att}(a_i, C_k) = \prod_{k_i} \times \mathfrak{R}(a_i, C_k) \quad [4]$$

Dans [4], $\mathfrak{R}(a_i, C_k)$ représente la préférence d'un agent a_i pour la coalition C_k

et \prod_{k_i} représente la préférence d'une coalition C_k pour l'agent a_i .

Un agent a_i reçoit un nouveau message provenant de son environnement, il l'analyse à travers sa *fonction de perception* qui le transmet à la fonction de décision (cf. figure 16). Sa *fonction de décision*, en consultant les objectifs de l'agent, l'état actuel de l'agent et de l'environnement ainsi que ses connaissances, choisit quelle action entreprendre. La *fonction d'action* est alors activée pour lancer le processus de coordination en envoyant des messages aux autres agents, elle met à jour l'état et les connaissances de l'agent. S'il s'agit d'un processus de formation de coalitions afin d'exécuter une tâche complexe l'agent a_i se met dans le rôle d'*initiateur* et entre en communication avec les autres agents selon l'algorithme suivant dont les deux étapes principales sont décrites ci-avant. Le protocole que nous proposons repose sur le principe d'appel d'offre des protocoles classiques comme le « CNP-Contract Net Protocol » (Davis et Smith, 1988). En revanche, son objectif est différent puisque nous recherchons l'élaboration d'accords multilatéraux entre plusieurs agents d'où une spécification différente de ce protocole.

- *Etape 1*

L'agent, après avoir construit son modèle de préférence, contacte les agents internes et externes qu'il préfère c'est-à-dire ayant les valeurs maximales dans son modèle de préférences (cf. figure 19). Il est à noter que nous nous limitons au modèle de préférence d'un agent a_i pour un agent a_j selon différents critères prédéfinis auparavant par le concepteur du système comme par exemple le délai, la distance, la qualité...

- *Etape 2*

Les agents sollicités consultent leurs capacités pour déterminer s'ils peuvent contribuer à cette activité en examinant la liste des activités qui la précèdent comme par exemple les ressources appropriées, la quantité de matières premières nécessaires... ; ensuite ils répondent à l'agent initiateur par une acceptation temporaire ou un refus. Si l'agent initiateur reçoit un refus lui seul peut décider s'il continue le processus ou l'annule (cf. Figure 20). Dans le premier cas, il envoie un message aux agents pour qu'ils confirment leur participation. S'ils acceptent définitivement, l'agent initiateur les rajoute à la liste de coalitions confirmées et détermine le coût total.

L'algorithme de formation de coalitions

Phase 1 : Identification des agents à contacter par calcul des modèles de préférence:

Chaque agent a_i voulant lancer une formation de coalitions (FC) :

- Identifie, selon ses connaissances sur l'environnement, les agents potentiels simultanément en interne $\forall a_j \in A^I$ et en externe $\exists a_j \in A^E$ et les ajouter dans l'ensemble AP

- Construit son modèle de préférence $\prod_{a_i}^p$ selon des critères prédéfinis (1...p)

- Construit son vecteur de préférence agrégé $\prod_{a_i} = (x_{i1}, \dots, x_{ii}, \dots, x_{i|AP|})$ avec $x_{ii} = 0$.

- Envoie son modèle aux agents et attend leurs modèles. En même temps il ajoute ces propositions à sa liste de coalitions proposées.

- A la réception des autres modèles : Calculer l'indice d'attraction unilatérale

$$U\text{-Att}(a_i) = \sum_{j=1}^{|AP|} (x_{i,j} - x_{i,j+1}) \mu(A_j) \text{ avec } \begin{cases} i \neq j \\ A_j \in AP \end{cases}$$

- Si le $U\text{-Att}(a_i \neq \text{initiateur})$ est le plus élevé
(*) Bloquer le Processus de FC //pour qu'il y' ait qu'un processus de FC à la fois.

Calcule l'indice d'attraction bilatérale

$$B\text{-Att}(a_i, C_k) = \prod_{ki} \times \mathfrak{R}(a_i, C_k)$$

Sinon

L'agent ayant l' $U\text{-Att}$ le plus élevé devient l'initiateur et revient à (*)

$AP = AP \setminus \{a_i\}$ // cas d'une sous-traitance d'une tâche

Finsi

Figure 19 : Phase 1 de l'algorithme de formation de coalitions

L'algorithme de formation de coalitions (suite)

Phase 2 : Formation de la coalition et calcul du coût total de la chaîne

Tant que l'objectif de la coalition n'est pas atteint **et** $\{AP\} \neq \text{vide}$ **faire**

- (**) Contacter dans l'ordre l'agent a , ayant le **B-Att** le plus élevé

Pour toutes les t_i tel que $(t_i, t_j) \in E$ **faire**

Si $\sum_{t_i \in T} q_{t_i}^j < Q^j$ **alors**

// Le coût d'exécution d'une tâche est la somme des coûts des tâches qui la précèdent.

$$c_{t_j} = \sum_{(t_i, t_j) \in E} c_{t_i}$$

// La capacité de l'agent est la somme de la capacité actuelle plus les capacités investies dans toutes les tâches qui la précèdent.

$$q_{t_i}^j = \sum_{(t_i, t_j) \in E} q_{t_i}^j$$

• Enlever (t_i, t_j) de E

$$Z_{t_i}^{a_j} = 1 \quad // \text{L'agent } a_j \text{ veut participer à la coalition}$$

$$C_k = C_k \cup \{a_j\}$$

Sinon

$$Z_{t_i}^{a_j} = 0 \quad // \text{L'agent ne peut pas participer à la coalition}$$

Revenir à (**)

Fin si

Fin Pour

- Envoyer un message à l'agent concerné par la coalition
- Recevoir la réponse (confirmation ou déclination).

Fin tant que

Calculer le coût total de la chaîne :

$$\text{Min } C.T = \sum_{a_i \in AP} \sum_{t_i \in TC} z_{t_i}^{a_i} c_{t_i}^{a_i} + \sum_{(t_i, t_j) \in E} \sum_{a_i, a_j \in AP} z_{t_i}^{a_i} z_{t_j}^{a_j} c p_{a_i a_j}^{t_j} + \sum_{a_i \in AR} \sum_{t_i \in T} c_{t_i}^{a_i}$$

Figure 20 : Phase 2 de l'algorithme de formation de coalitions

4. Conclusion

Dans cet article, nous avons présenté un panorama des techniques de modélisation des chaînes logistiques. Nous avons montré que l'approche multi-agents est bien adaptée et offre des solutions prometteuses pour la modélisation et l'optimisation d'une chaîne logistique.

Cette approche permet de prendre en compte facilement les entités qui entrent en jeu dans la gestion des chaînes logistiques, à savoir, les nombreux acteurs de la chaîne, leur comportement idiosyncratique, les interactions entre ces acteurs, les processus de négociation, l'environnement en perpétuel évolution, etc. Les principaux avantages de cette approche sont les suivants : 1) la modélisation d'entités autonomes en interaction, qui doivent se coordonner pour atteindre un objectif global qui n'est autre que l'agrégation des objectifs individuels et ce via un processus de négociation, 2) la modélisation des protocoles d'interaction et de coordination grâce à l'outil AUML et aux automates à états finis, 3) la modélisation du comportement réactif des entités qui réagissent à un environnement en général dynamique.

Nous avons ensuite proposé une méthode de coordination distribuée par formation de coalitions entre agents que nous avons empruntée au domaine des systèmes multi-agents.

La formation de coalitions permet d'atteindre des accords multilatéraux contrairement aux protocoles d'interaction classiques comme le « Contract Net Protocol », qui repose sur l'élaboration d'accords bilatéraux.

Nous avons aussi présenté la structure des messages échangés et le protocole d'interaction entre les agents sur lequel s'appuie l'algorithme de formation de coalitions. Un prototype multi-agents mettant en œuvre notre algorithme de coordination et le protocole d'interaction est en cours de développement et de tests.

Comme perspectives de recherche, plusieurs hypothèses faites au cours de la modélisation peuvent être étudiées : (1) le choix du protocole de formation de coalitions est en réalité dépendant du type de problème étudié c'est-à-dire des paramètres à prendre en considération comme par exemple le fait que les agents ont ou non le même objectif voire les mêmes fonctions d'utilité, ou le fait qu'ils se font ou non confiance pour pouvoir s'échanger leurs connaissances... tous ces paramètres peuvent générer des protocoles différents qui doivent être testés ; (2) le protocole proposé repose sur plusieurs principes qui peuvent ne pas être adaptés à la réalité industrielle en particulier dans le cas d'une concurrence entre les différents partenaires d'une chaîne logistique. Ils peuvent aussi avoir des objectifs radicalement opposés comme c'est le cas dans la relation fournisseur-producteur.

Bibliographie

- Aknine S., Pinson S. et Shakun M., (2004) "*A Multi-Agent Coalition Formation Based on Preference Models*", *International Journal Group and Decision and Negotiation*, Kluwer, vol 13, n° 6, 2004, p. 513-538.
- Austin J.L. (1970), "*Quand dire c'est faire*", Éditions du Seuil, Paris. (Traduction de "How to do things with Words" The William James Lectures delivered at Harvard University in 1955, Ed. Urmson, Oxford, 1962)
- Balbo F., El Fallah A., Pinson S. (2002) "*Coordination d'actions par planification multi-agents*". Dans *Organisation et application des Systèmes Multi-agents*, Mandiau R., Grislin- Le Strugeon E., Péninou A., Hermès- Lavoisier p93-108
- Barbuceanu M., Fox M.S., "*The Architecture of an Agent Building Shell*", *Intelligent Agents II*, Wooldridge M., Muller J.P. and Tambe M.(Eds.) *Lecture Notes in Artificial Intelligence*, Springer-Verlag, Berlin, vol 1037, 1996, p. 235-250.
- Beamon B. M. (1998), "*Supply Chain design and analysis: Models and Methods*", *The International Journal of Production Economics*, N ° 55 p 281- 294.
- Chang Y., Makatsoris H. (2001), "*Supply Chain Modeling using Simulation*", *International Journal of Simulation. Systems, Science & Technology*, Vol. 2 No.1, p. 24-30.
- Christy D. P. et Grout J.R. (1994), "*Safeguarding supply chain relationships*", *International Journal of Production Economics* N° 36 p 233-242.
- Cohen M. A. et Lee H.L. (1989), "*Resource deployment analysis of global manufacturing and distribution networks*", *Journal of Manufacturing and Operations Management* N°2 p 81- 104.
- Davis R. et Smith R.G. (1988), "*Negotiation as a Metaphor for Distributed Problem Solving*", *Artificial Intelligence* (20) 1, p 63-109.
- D'Amours S., Chaib-Draa B., Moyaux T. (2006), "*Supply Chain Management and Multi Agent System: an Overview*". In *MultiAgent-Based Supply Chain Management*, B. Chaib-draa and J. P. Müller, Editors, ISBN: 3-540-33875-6 (Springer).
- D'Amours S., Forget P., Frayret J-M. (2007), "*Multi-Behavior Agent Model for Planning in Supply Chains: An Application to the Lumber Industry*", *Robotics and Computer-Integrated Manufacturing Journal*, 2007
- Dodd C., Kumara S. R. T. (2001), "*A distributed multi-agent model for value nets*". IEA/AIE 2001, p 718-727.
- Dupont L., Ghedira K., Kallel O., Benjaafar I. (2008) "*Résolution d'un problème de contrat réservation de capacité dans une chaîne logistique équitable*", 7e Conférence Internationale de MODélisation et SIMulation - MOSIM'08 - Paris-France,2008.
- El Fallah A., Boussetta S., Haddad S., Moraitis P., Taghelit M. : "*Coordination d'agents rationnels par planification distribuée* ", *Revue d'Intelligence Artificielle*, (numéro spécial sur les SMA), Hermès, 1997.
- Felix-Teixeira J.M., Brito A. E. S. C. (2003), "*An Approach for Dynamic Supply Chain Modelling*". At the proceedings of The 2003 European Simulation and Modelling Conference, Naples, Italy, October 27-29.
- Felix-Teixeira J.M., Brito A. E. S. C. (2004), "*On Measuring the Supply Chain Flexibility*". At the proceedings of The European Simulation and Modelling Conference, UNESCO, Paris, France, October 25-27.
- Felix-Teixeira J.M., Brito A. E. S. C. (2007), "*scMod/Sim.exe For Supply Chain Modelling and Simulation*". Tutorial at the Industrial Simulation Conference 2007 (ISC'2007), Delft University of Technology, Holland, June 11-13.
- Forget P., D'Amours S., Frayret J-M., Gaudreault J. (2008), "*Motivations for Multi-Behavior Agent Model in Supply Chains planning*", 7ème Conférence de Modélisation, Optimisation et Simulation des Systèmes (MOSIM), Paris, France, Mars, 2008.
- Fox M.S., Barbuceanu M. (1996), "*Coordinating multiple agents in the Supply Chain*", *Proceedings of Fifth Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, Stanford, CA, IEEE Computer Society Press, p 134-142.
- Fox M.S., Barbuceanu M. et Teigen, R. (2000) "*Agent-Oriented Supply-Chain Management*", *The International Journal of Flexible Manufacturing Systems*, n° 12, 2000, p. 165- 188.
- Ghedira K., Dupont L., Kallel O., Benjaafar I. (2008) "*Multi-agent negotiation in a supply chain: case of the wholesale price contract*", 10th International Conference on Enterprise Information Systems, Barcelona, Spain,2008.

- Giard V., Balin S. (2007) "*Problèmes méthodologiques posés par la simulation de processus de production de services*", Journal Européen des Systèmes Automatisés (APII-JESA), vol. 41, n° 9-10, 2007.
- Giard V., Mendy G. (2006) "*Amélioration de la synchronisation de la production sur une chaîne logistique*", Revue Française de Gestion Industrielle, vol 25, n°1, p.63-82, 2006.
- Grabisch M., (1996), "*The Application of Fuzzy Integrals in Multi-criteria Decision Making*". *European Journal of Operational Research*, vol. 89, 1996.
- Hahndel S., Fuchs F., Levi P. (1994) "*Distributed Negotiation-Based Task Planning for a Flexible Manufacturing Environment*". In Proceedings of the 6th European Workshop on Modeling Autonomous Agents in Multi-Agent World, p 147- 158. Odense, Denmark.
- Huguet M. P. (2002) "*An application of Agent UML to Supply Chain Management*", In Proceedings of Agent Oriented Information System (AOIS-02), Paolo Giorgini and Yves Lespérance and Gerd Wagner and Eric Yu (eds.)
- Huguet M. P., Odell J. (2004). "*Representing Agent Interaction Protocols with Agent UML*". In Proceedings of the Fifth International Workshop on Agent-Oriented Software Engineering (AOSE 2004), Paolo Giorgini, Joerg Mueller and James Odell (eds.), New York.
- Jennings, N. (2001) "*Automated negotiation: Prospects, methods and challenges*". *Group Decision and Negotiation Journal*, 10(2), 2001, p.199-215.
- Kok A.G., Graves S.C. (2003) "*Supply Chain Management: Design, Coordination and Operation*", Handbooks in Operation Research and Management Science, Elsevier vol. 11, 2003, p. 1-16.
- Lehoux N., D'amours S., Langevin A. (2008) "*Vers une Approche Collaborative Gagnant-Gagnant au sein d'un réseau Bi-Echelon : Une Application à l'industrie des pâtes et papiers*", 7ème Conférence Internationale de MODélisation et SIMulation - MOSIM'08 - du 31 mars au 2 avril 2008 – Paris- France. Editions Tec&Doc – Lavoisier ISBN : 978-2-7430-1057-7
- Montreuil B., Labarthe O., Ferrarini A., Espinasse B. (2007) "*Coordination de chaînes logistiques centrées consommateurs : modélisations et simulations orientées agents. Démarche et résultats*", 7e Congrès international de génie industriel (CIGI 2007), Trois-Rivières, Canada, 5-8 juin., 2007.
- Montreuil B., Labarthe O., D'amours S., Roy D., Ferrarini A., Espinasse B., Monteiro T., Anciaux D. (2008) "*Simulation à base d'agents des systèmes de coordination et de planification des réseaux d'entreprises*", dans LAVOISIER - HERMES (Ed.), ``La simulation pour la gestion des chaînes logistiques'', Traité IC2, série systèmes automatisés, Caroline Thierry André Thomas et Gérard Bel, ch. 7, pp. 227 - 260, août 2008
- Moyaux T. Chaib-draa B. D'Amours S. (2004), "*Multi-Agent Simulation of Collaborative Strategies in a Supply Chain*", AAMAS 2004 - 3rd International Joint Conference on Autonomous Agents and Multiagent Systems, Columbia University, New-York City, USA, 19-23 juillet.
- Moyaux T. Chaib-draa B. D'Amours S. (2008), "*Spreadsheet vs. Multiagent based simulations : The case of supply chains*", *International Journal of Simulation and Process Modelling (IJSPM)*; Vol 4 n°2, p 89-105.
- Nfaoui H., Ozrout Y. (2006) "*SCM, AUML : un modèle d'agents pour la simulation proactive et l'aide à la décision dans le Supply Chain*", Conférence MajecSTIC 2006 MANifestation des JEunes Chercheurs en Sciences et Technologies de l'Information et de la Communication. 22-24 Novembre 2006, Lorient France
- Nfaoui E.H., Ozrout Y., El Beqqali O., Bouras A. (2008) "*Architecture Distribuée à base d'Agents pour la Simulation Proactive et l'Aide à la Décision dans la Chaîne Logistique*", 7ème Conférence Internationale de MODélisation et SIMulation - MOSIM'08 - du 31 mars au 2 avril 2008 – Paris- France. Pp : 1476-1484 (Volume 2), Editions Tec&Doc – Lavoisier ISBN : 978-2-7430-1057-7
- Parunak H. V. D. (1996), "*Applications of distributed artificial intelligence in industry*". In O'Hare G. M. P. and Jennings N. R., Editors, *Foundations of Distributed Artificial Intelligence*, John Wiley & Sons, p 71-76.
- Pyke D. F. et Cohen M. A. (1993) "*Performance characteristics of stochastic integrated production-distribution systems*", *European Journal of Operation Research* N°68 (1) 23-48.
- Shapiro J. F. (2006), "*Modeling the Supply Chain*", (Duxbury applied). South-Western College Pub; 2 edition ISBN-10: 0495126098 chapter 1, p 1-27
- Searle J.R. (1972), "*Les actes de langage*". Essai de philosophie du langage, Hermann.

- Shen W., Norrie D. H. (1999), "*Agent-based systems for intelligent manufacturing : A state-of-the-art survey*". Knowledge and Information Systems, 1(2), p 129-156.
- Shehory O., Kraus S. (1998), "*Methods for Task Allocation via Agent Coalition Formation*"; Artificial Intelligence Journal, Elsevier Science Vol. 101 (1-2) p 165-200.
- Simchi-Levi D., Kaminsky P., Simchi-Levi E. (2007), "*Designing and managing the supply chain*". McGraw-Hill edition, ISBN: 0-07-284553-8.
- Swaminathan J.M., Smith S.F., Sadeh N.M. (1998), "*Modeling supply chain dynamics: A multiagent approach*". Decision Sciences 29, p 607-632
- Towill D. R. (1991), "*Supply Chain dynamics*". International Journal of Computer Integrated Manufacturing N° 4 (4) p 197-208.
- Von Martial F. (1992), "*Coordinating Plans of Autonomous Agents*". Springer-Verlag, Berlin, 1992.
- Wooldridge M., Jennings N. (1995), "*Intelligent agents: Theory and Practice*". The knowledge Engineering Review, p 26-37