

ARCHITECTURE DISTRIBUÉE À BASE D'AGENTS POUR LA SIMULATION PROACTIVE ET L'AIDE À LA DECISION DANS LA CHAÎNE LOGISTIQUE

E.H. NFAOUI^{(a),(b)}, Y. OUZROUT^(a), A. BOURAS^(a), O. EI BEQQALI^(b)

^(a) LIESP – Université Lumière
Lyon 2, 160 Bd de l'université, Bron
CEDEX 69676. Lyon France

^(b) GRMS2I – FSDM, Université Sidi
Md Ben AbdEllah, B.P 1796
Fès-Atlas. Maroc

elhabib.nfaoui@univ-lyon2.fr, yacine.ouzrout@univ-lyon2.fr, abdelaiziz.bouras@univ-lyon2.fr
obekkali@fsdmfes.ac.ma

RESUME : *Au cours des dernières années, les industriels furent les principaux instigateurs de l'évolution de la chaîne logistique (Supply Chain) dans le but d'optimiser fabrication et distribution. Aujourd'hui, les clients sont de plus en plus exigeants et les industriels doivent tenter de répondre au mieux à leurs besoins en matière de personnalisation, de rapidité de traitement des commandes et de livraison. Dans cet article, nous proposons une architecture distribuée à base de système multi-agent pour la simulation proactive et l'aide à la décision collaborative dans les chaînes logistiques afin d'avoir la capacité à satisfaire au mieux les demandes des clients en termes de livraison, plus particulièrement, lors de la présence des commandes incertaines, des commandes imprécises et des exceptions (problème de production, problème de transport, erreur sur prévisions, retard de livraison, etc.). Nous proposons un ensemble de protocoles de négociation entre agents et nous les modélisons avec le langage Agent UML (Unified Modeling Language).*

MOTS-CLES : *Supply Chain Management, Incertitude des Informations, Systèmes Multi-Agent, Simulation Proactive, Protocoles de Négociation, Agent UML.*

1. INTRODUCTION

Les performances de nombreux systèmes sont réduites par les imperfections des flux qui les parcourent. En tant que système distribué composé d'entreprises et traversée par divers flux, la SC (Supply Chain) fait également face à ce problème. Ainsi, les flux d'informations et de produits circulant dans la SC doivent être bien maîtrisés et gérés. Plusieurs techniques et méthodes (CPFR - Collaborative Planning and Forecasting Replenishment-, Transshipment, etc.) ainsi que des outils informatiques (ERP -Enterprise Resource Planning-, APS -Advanced Planning Systems-, etc.) sont développées et mises en œuvre entre les acteurs d'une SC dans le cadre du supply chain management (SCM) en vue de coordonner leurs décisions et répondre ainsi à ce besoin. Cependant, la littérature montre que ces techniques n'apportent pas de solutions efficaces pour les situations d'urgence causées par la présence: des commandes incertaines, des commandes imprécises (une commande imprécise est une commande dont la quantité commandée n'est pas connue avec exactitude) et des exceptions (problème de production, problème de transport, erreur sur prévisions, retard de livraison, etc.).

Dans cet article, nous proposons une architecture distribuée à base de Système Multi-Agent (SMA) pour gérer ces *situations d'urgence*. Cette architecture

s'ajoute comme une couche complémentaire aux systèmes d'informations des acteurs de la SC. Elle est composée de différents agents d'interface intelligents qui soient capables d'aider à la prise de décisions collaboratives lors de la présence d'une situation d'urgence. La spécificité de notre outil est qu'il peut être utilisé aussi bien entre des acteurs collaborant et partageant les informations stratégiques que des acteurs concurrents qui pour eux le partage d'information présente un grand défi.

Nous présentons tout d'abord un état de l'art sur les méthodes et les outils SCM utilisés par les industriels ainsi que les récents travaux liés avec notre contribution. Ensuite nous montrons dans la section 3 l'intérêt de la simulation proactive et l'utilisation des systèmes multi-agent dans le domaine de la supply chain. La section 4 présente notre modèle à base d'agents et la section 5 est consacrée aux protocoles de négociation que nous proposons pour gérer les situations d'urgence. Nous présentons dans la section 6 le comportement des agents. La section 7 présente un cas d'étude qui teste les protocoles de négociation. Enfin, nous concluons.

2. ÉTAT DE L'ART

Différents niveaux de collaboration, coopération, coordination, communication et de partenariat entre les

entreprises ont été mis en place dans l'industrie ces derniers temps afin de résoudre les différentes problématiques de gestion de la supply chain, notamment celles dues aux flux d'informations et aux flux de produits. Ces problématiques couvrent les trois niveaux de planification dans la chaîne logistique (figure 1) ainsi que les différents horizons de la prise de décision (long, moyen et court termes).

Parmi les outils informatiques, on peut notamment citer le ERP et les APS. Les ERP (ERP, 2006) sont des outils transactionnels qui assurent la capture des données et leur stockage (commandes, gestion des stocks, etc.). Les logiciels de type APS sont des outils spécialisés structurant la décision pour la planification de la demande, de la production et de la distribution. Ils optimisent la planification et synchronisent les flux de la chaîne logistique en tenant compte simultanément d'un grand nombre de contraintes (ressources, capacités, délais, coûts, profits etc.) (CXP, 2001). La planification avancée (APS) permet aujourd'hui de recalculer rapidement un plan tactique sur la base d'une recherche d'optimum. Si cela garantit la faisabilité du plan, il n'en assure pas la robustesse (Lamouri, 2006). Le terme robustesse est généralement associé à celui de risque et de prise de décision (Kleijnen et Gaury, 2003).

L'une des pratiques collaboratives qui permet à deux partenaires (fournisseurs et distributeurs) de collaborer pour identifier et résoudre les exceptions pour la prévision des ventes et des commandes est le processus CPFR (Simchi-Levi et al., 2000). C'est un standard qui améliore le VMI (Vendor Managed Inventory) et le CRP (Continuous Replenishment Program) en incorporant la prévision commune de la demande. La particularité du CPFR, et ce qui en fait son attrait, c'est le fait que cette méthode prend aussi bien en compte la gestion de l'offre que la gestion de la demande. Il faut donc, pour que ce processus puisse être exploité de la meilleure façon qu'il soit, que producteurs et distributeurs partagent et communiquent les informations qu'ils détiennent dans un environnement de transparence et dans la perspective d'un partenariat « gagnant-gagnant ». La section 5.4 présente les étapes de déroulement du CPFR sous forme d'un protocole de négociation entre deux agents représentant chacun un acteur de la SC.

Actuellement, la majorité des pratiques exigent un partage d'informations, ce qui n'est pas possible dans le cas d'une SC formée d'acteurs indépendants ou parfois concurrents. Par exemple, des distributeurs concurrents qui s'approvisionnent en produits finis chez le même producteur. En outre, elles ne donnent pas des solutions pour résoudre les situations d'urgences (présence des commandes incertaines, des commandes imprécises ou des exceptions, etc.) au moment de leur présence, et surtout pendant un intervalle de temps acceptable.

Le traitement des commandes incertaines, ou commandes imprécises, a été envisagé dans plusieurs travaux de recherche (Fargier et al., 2002), (Hapke et al.,

2000), (Geneste et al., 2000) et (Letouzey et al., 2001). Mais le travail qui a englobé la majorité des études et a analysé simultanément l'imprécision et l'incertitude des informations a été proposé par (Reynoso, 2004). Après avoir balayé l'apport des autres travaux, l'auteur propose une démarche similaire à MRP2 (Application de gestion des approvisionnements) qui succède au MRP (Material Requirements Planning) en lui ajoutant des fonctions de gestions des capacités et de calcul des coûts) avec une approche basée sur la logique floue (fuzzy logic): 'MRP flou'. Giannoccaro (Giannoccaro et al., 2003) a aussi utilisé la logique floue pour proposer une méthode de gestion de stock dans la supply chain. Il a modélisé l'incertitude de la demande et les coûts du stock (coût de possession et coût des commandes en retard).

Notre contribution s'inscrit principalement dans ce contexte, nous proposons une architecture distribuée composée de différents agents d'interface intelligents qui soient capables d'aider à la prise de décisions collaboratives en apportant des solutions instantanées et automatisées aux situations d'urgences. Dès qu'un événement survient dans la chaîne, c'est-à-dire lorsque l'état de la SC dévie de l'état planifié, les alertes sont générées conduisant les gestionnaires à re-planifier, tout en se basant sur les propositions automatisées des agents représentant chaque entité. Des contre-mesures sont lancées rapidement pour corriger et sauver la situation, synchronisant à nouveau l'ensemble des maillons de la chaîne avec les nouvelles conditions. Notre système prend en compte tous les acteurs potentiels de la SC si nécessaire, en commençant du fournisseur le plus en aval jusqu'au client le plus en amont. Notre angle de vue se résume comme le suivant: *Une situation d'urgence est survenue même si on a pris toutes les précautions et les mesures, alors que peut-on faire pour sauver la situation? Sachant que les acteurs peuvent être indépendants.*

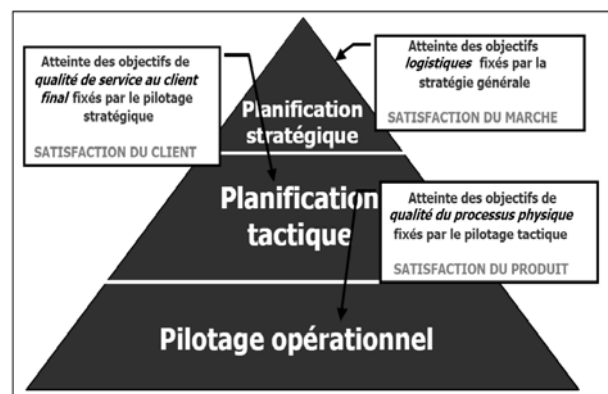


Figure 1. Les trois niveaux de planification dans la chaîne logistique (Lamouri, 2006)

3. LA SIMULATION PROACTIVE ET LES SYSTÈMES MULTI-AGENT DANS LE DOMAINE DE LA SUPPLY CHAIN

La simulation a longtemps été cantonnée à la phase de conception ou de re-conception des systèmes. La deuxième utilisation de la simulation est la prise de décision. Ce type de simulation est connu sous le terme de simulation proactive (Cardin et Castagna, 2006). Elle est largement utilisée pour le pilotage des systèmes industriels. Nous citons à titre d'exemple, la simulation pour l'aide au pilotage d'une unité organisée au moyen de KANBAN (Pujo et al., 2004) et le projet commun entre la société SIMPARTNERS et l'AIP PRIMECA qui avait comme objectif la réalisation d'un prototype de prise de décision et de simulation en ligne sur une unité d'assemblage réelle (Iassinovski et al., 2006).

Dans le cadre du SCM, les décisions sont prises de manière collaborative, puisque chaque acteur intègre dans sa prise de décision les préoccupations de ses partenaires. Or, devant une situation de planification opérationnelle (court terme, par exemple : commande urgente) ou bien une situation de négociation pour modifier une date de livraison en cas d'aléas (problème de production, problème de transport, etc.), le système de décision ne dispose en général que de peu de temps pour faire son choix, et il doit donc être capable d'évaluer les conséquences correspondant aux différents scénarios qui s'offrent à lui dans un délai plus court. En plus, vu la nature distribuée de la SC (Supply Chain), l'évaluation doit se dérouler elle aussi de manière distribuée, puisque chaque partenaire de la SC poursuit ses buts individuels tandis qu'il satisfait à ses contraintes locales et externes.

Nous proposons d'utiliser la simulation proactive en complément des technologies existantes, pour aider à la prise de décision collaborative et la résolution des situations d'urgence.

Dans le cas d'une supply chain, nous considérons que deux conditions sont indispensables à vérifier pour que la simulation proactive soit efficace et donne des résultats satisfaisants :

- Les informations initialisant notre simulation proactive doivent renseigner sur l'état exact des acteurs constituant la supply chain. C'est-à-dire que les agents représentant chaque acteur cherchent les informations qui constituent les états à partir desquels la simulation va démarrer.
- Etre capable d'évaluer les conséquences correspondant aux différents scénarios face à une situation donnée. Si nécessaire (en cas d'aléas), elle doit s'exécuter dans un délai plus court.

Afin d'avoir un modèle flexible, réactif et vérifiant les conditions énoncées ci-dessus, nous avons choisi l'approche multi-agent pour développer notre architecture.

Plusieurs projets concernant l'utilisation des systèmes multi-agent dans la chaîne logistique ont été réalisés. Ces projets se confrontent à des problèmes différents de la chaîne logistique dont la conception et la gestion sont

deux grandes catégories de problématiques. Cloutier (Cloutier et al., 2001), Maturana (Maturana et al., 1999) et Parunak (Parunak, 1996) ont montré l'intérêt d'utiliser cette approche dans le domaine des supply chain. En particulier, Cloutier (Cloutier et al., 2001) précise que le paradigme des agents est une métaphore naturelle avec les organisations en réseau depuis que les unités de production distribuées possèdent les mêmes caractéristiques que les agents (basé sur la définition d'un agent de (Wooldridge et Jennings, 1995)) :

- *Autonomie* : une entreprise exécute des tâches d'elle-même sans intervention extérieure ;
- *Capacités sociales* : un acteur de la chaîne logistique peut communiquer avec un autre acteur pour, par exemple, lui passer une commande de produits ou de services ;
- *Réactivité* : une firme modifie son comportement si le marché ou la concurrence évolue ;
- *Pro-activité* : une entreprise peut initier d'elle-même de nouvelles activités, comme par exemple décider de lancer un nouveau produit sur le marché.

4. MODÈLE MULTI-AGENT PROPOSÉ

Une difficulté majeure à laquelle sont confrontés les développeurs d'applications pour la SC réside au niveau du modèle de celle-ci. En effet, le processus de modélisation prend beaucoup de temps puisque actuellement il n'existe pas de méthodes génériques pour la modélisation des SC en utilisant les systèmes multi-agent (Govindu et Chinnam, 2007) à l'exception de la méthodologie proposée dans (Govindu et Chinnam, 2007) et qui se trouve encore en cours de développement. La littérature montre que la plupart des modèles sont liés à des problèmes particuliers.

De notre part, nous avons proposé un modèle basé sur les processus afin qu'il soit le plus générique possible et que notre architecture sera utilisé dans des secteurs variés (comme le textile et l'habillement, le sanitaire, etc.). Nous avons choisi de modéliser chaque acteur par sept agents. Chacun de ces agents modélise une fonction principale de l'entreprise ou un processus particulier agissant sur la SC. Le fait d'attribuer un agent à chaque fonction ou à chaque processus permet d'augmenter les performances de traitement, notamment en terme de rapidité qui joue un rôle important lors de la présence d'une perturbation. Nous présentons ci-dessous les noms et les rôles de ces agents :

AgentPRC : il joue le rôle du processus relatifs au client. Il reçoit les commandes des clients, établit avec l'agent « AgentDis » un programme de livraison avant de soumettre ce programme au client et de le confirmer.

AgentDis : il gère le stock de distribution (stock de produits finis). Plusieurs paramètres caractérisent cet agent: méthode utilisée pour la gestion du stock, capacité du stock, etc.

AgentPro : cet agent gère le processus de production.

AgentApp : il gère le stock de fabrication (stock de matières premières, de composants et de produits semi finis).

AgentAch : il joue le rôle du processus achat. En particulier, il approvisionne l'entreprise en matières premières nécessaires à la fabrication du produit fini.

AgentSCM : il modélise le rôle des pratiques et techniques mises en œuvre par les acteurs pour piloter et gérer leur supply chain. Elles sont connues sous le terme "Supply Chain Management".

AgentPer : il joue le rôle des perturbations. Il permet au modèle d'être ouvert et extensible pour modéliser une grande variété de perturbations et couvrir différents types de supply chain (taille mondiale, taille nationale, genre du secteur d'activité).

Dans le cas d'un acteur de type distributeur, les agents « AgentPro » et « AgentApp » n'existent pas.

Finalement, L'architecture complète (figure 2) est formée d'un ensemble d'agents représentant les acteurs de la SC. Ces agents communiquent entre eux et négocient à l'aide de protocoles. Ils cherchent les données qui les intéressent dans les systèmes d'informations des acteurs (comme les ERP).

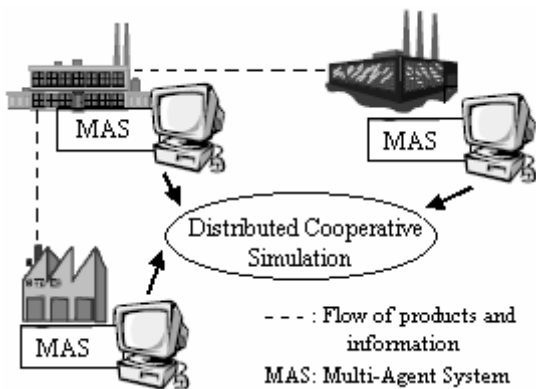


Figure 2. Architecture complète

5. PROTOCOLES DE NÉGOCIATION

La négociation est le mécanisme par lequel les agents peuvent arriver à un accord commun. Dans le cas des agents intelligents et des SMA, La négociation est une composante de base de l'interaction et cela parce que les agents sont autonomes (Jenning, 2001) ; il n'y a pas de solution imposée à l'avance et les agents doivent arriver à trouver des solutions dynamiquement, pendant qu'ils résolvent les problèmes. Pour modéliser la négociation entre les agents composant notre système, nous avons pris en compte les aspects suivants :

- *L'objet de négociation* : un objet abstrait qui comprend les attributs que les agents veulent négocier. Dans notre architecture, plusieurs objets sont sujets à négocier selon la situation. Nous trouvons entre autres, le scénario acceptable en cas d'aléas, la commande avec ses attributs quantité et date de livraison, les prévisions avec leurs attributs quantité, date et exception.
- *Le processus de décision* : le modèle que l'agent utilise pour prendre des décisions pendant la négociation. La

partie la plus importante de la prise des décisions dans ce cas est la stratégie de négociation qui permet de déterminer quelle primitive de négociation l'agent doit choisir à un certain moment.

- *Le langage de communication* : le langage utilisé par les agents pour échanger des informations pendant la négociation. Dans notre architecture de simulation, les agents communiquent à l'aide du langage FIPA-ACL (FIPA, 2002), il s'agit du langage standard spécifié par l'environnement JADE (Java Agent Development Framework) (Bellifemine et al., 1999) que nous avons utilisé pour développer notre architecture.
- *Le protocole de négociation* définissant l'ensemble des règles qui régit la négociation : Les participants possibles à la négociation, les propositions légales que les participants peuvent faire, les états de la négociation et enfin une règle pour déterminer quand on est arrivé à un accord ou quand il faut s'arrêter parce qu'aucun accord n'a pas pu être trouvé.

Dans le contexte du supply chain management, les agents sont coopératifs, ayant le même but (agrégation des objectifs locaux), partagent et résolvent ensemble des problèmes. Pour cette raison, ils doivent fournir des réactions plus utiles aux propositions qu'ils reçoivent. Ces réactions peuvent prendre la forme d'une critique ou d'une contre-proposition (proposition refusée ou modifiée). Une critique est un commentaire sur la partie de la proposition que l'agent accepte ou refuse. Une contre-proposition est une proposition alternative engendrée en réponse à une proposition. A partir de telles réactions, l'agent doit être capable d'engendrer une proposition qui est probablement plus apte à mener à un accord. En conséquence, les agents de notre système doivent utiliser des protocoles respectant les critères qui viennent d'être énoncés et dépendant essentiellement de trois paramètres :

- Le secteur d'activité de la supply chain (industrie de textile et habillement, industrie des biens de consommation, etc.).
- Méthodes et approches SCM utilisées pour la coopération et la coordination.
- Objet à négocier : commande urgente, commande normale, prévisions de ventes, prévisions de commandes, plan de livraison en cas d'aléas, etc.

Nous proposons un ensemble de protocoles de négociation entre agents pour gérer les situations d'urgence dans la supply chain : Le protocole de négociation heuristique *ferme*, le protocole de négociation heuristique *récurive* et les protocoles correspondant aux différentes méthodes et pratiques SCM (comme : le CPF, le Transshipment, etc.). Nous nous focaliserons dans les sections suivantes sur les deux protocoles de négociation heuristique *récurive* et *ferme* et le protocole de négociation de type CPF utilisé entre l'agent « AgentSCM » d'un distributeur et l'agent « AgentSCM » de son fournisseur. Nous modéliserons aussi ces protocoles à l'aide des diagrammes de séquence AUML (Huget et Odell, 2004).

5.1 Négociation heuristique

La négociation heuristique est montrée dans la figure 3 (Florea, 2002). Dans ce protocole plusieurs propositions et contre-propositions peuvent être échangées dans les différentes étapes afin d'arriver à un accord. L'agent A, avec la proposition pA, est l'initiateur de la négociation, alors que l'agent B (participant) peut répondre avec les réponses p1B, p2B, p3B (to Modify Request). Le nombre des contre-propositions est limité. Une fois que cette limite est atteinte, les agents arrivent à un rejet. Nous proposons de récapituler le fonctionnement du protocole de négociation heuristique à l'aide d'un diagramme de séquence AUML (figure 4).

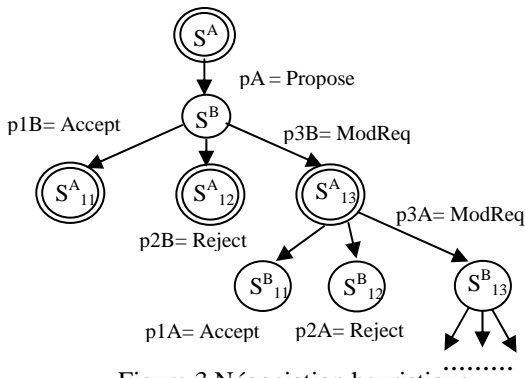


Figure 3. Négociation heuristique

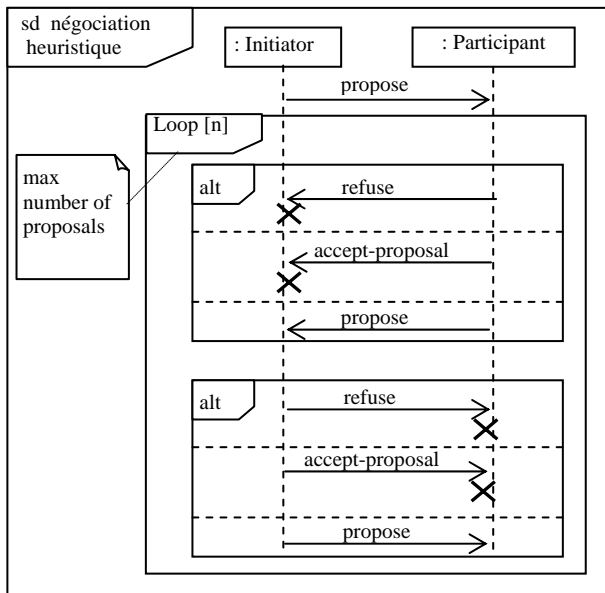


Figure 4. Négociation heuristique

5.2 Proposition d'une 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. Pour cela, la négociation heuristique étudiée dans la section précédente (cf. figure 4) doit comporter seulement les

primitives ACCEPT-PROPOSAL ou PROPOSE. Ainsi, nous proposons le protocole de négociation heuristique *ferme* qui est un cas particulier de la négociation heuristique. Le nom « ferme » qualifie ce protocole puisqu'il mène toujours à un accord. La figure 5 montre le diagramme de séquence AUML décrivant ce protocole.

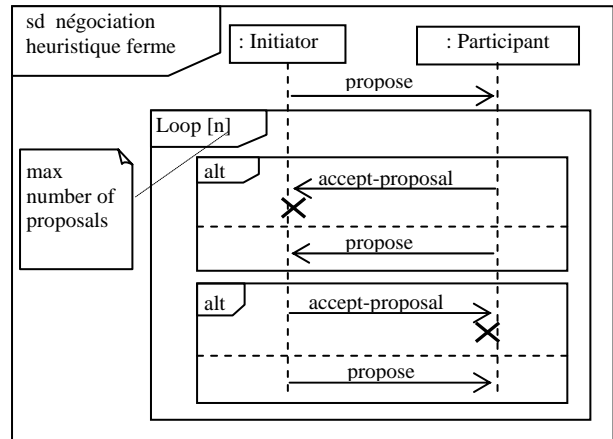


Figure 5. Négociation heuristique ferme

5.3 Proposition d'une négociation heuristique récursive

Le protocole de négociation heuristique *récursive* que l'on propose se déroule au moins entre trois agents, l'agent initiateur de la négociation, l'agent destinataire qui pourra être l'initiateur d'une nouvelle négociation heuristique avec le troisième agent. D'où le nom « récursive » qualifiant cette heuristique. La figure 6 montre le diagramme de séquence AUML correspondant.

Dans notre architecture de simulation proactive, les agents utilisent le protocole de négociation heuristique *récursive* afin de négocier une commande urgente ou le scénario à adopter en cas d'aléas. Ce protocole peut aussi être utilisé à l'intérieur d'un protocole correspondant à une pratique ou une démarche SCM. Dans le cas général, il se déroule de la manière suivante :

- 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. Il n'attend pas toutes les réponses pour prendre une décision. De plus, selon la situation et l'intervalle de temps dont il dispose, il peut chercher la meilleure solution en créant de nouvelles propositions déduites des réponses reçues ;
- Puisque les agents de notre architecture 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 « récursive » qualifiant cette heuristique.

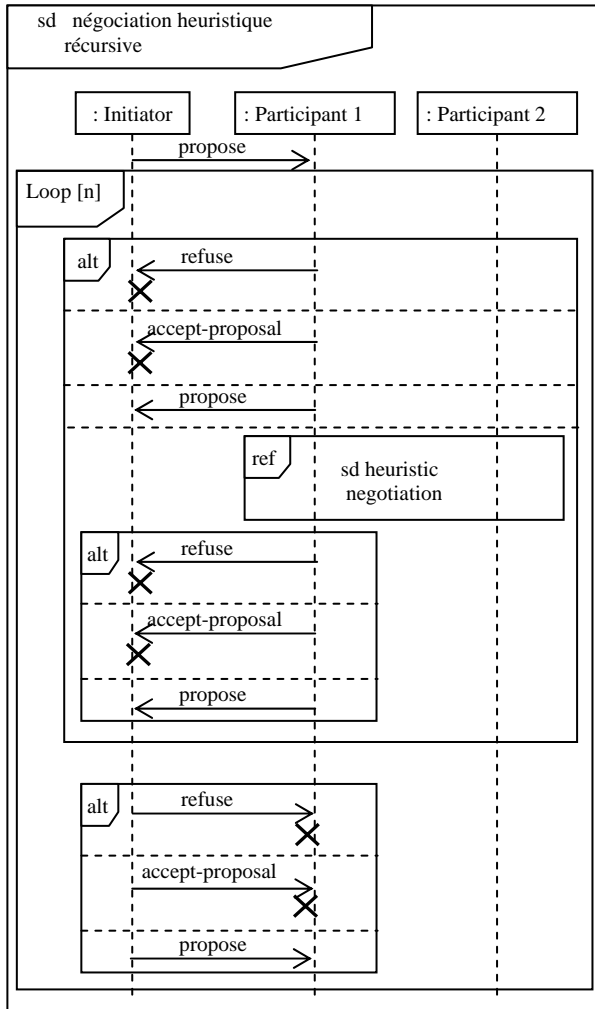


Figure 6. Négociation heuristique récursive

5.4 Protocole de négociation CPFR

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. Le protocole de négociation CPFR entre un fournisseur et un distributeur se déroule selon les étapes suivantes :

- La négociation débute entre les deux agents « AgentSCM » du distributeur et son fournisseur dès que l'un d'eux crée des prévisions de ventes et en informe l'autre.
- L'agent destinataire consulte et analyse ces prévisions, ensuite il envoie un message de confirmation à l'émetteur ou bien initie une négociation heuristique *ferme* avec lui s'il n'est pas d'accord afin de modifier ces prévisions. Dans tous les cas, l'émetteur et le destinataire doivent se mettre d'accord sur des prévisions de ventes puis les partager.
- Chacun des deux peut être l'auteur d'une négociation heuristique qui résout une exception (changements et mises à jour) sur les prévisions de ventes créées. Cette négociation heuristique entre les deux agents est sujette à devenir *récursive* pour impliquer d'autres agents pouvant aider à la résolution. La négociation se

termine lorsque toutes les exceptions sont résolues ou le délai fixé dynamiquement est expiré.

- L'agent « AgentSCM » du distributeur envoie un message de type performative « INFORM » à l'agent « AgentSCM » de son fournisseur contenant des informations sur le niveau de ses stocks (quantités en commandes, en stock...).
- L'agent « AgentSCM » du fournisseur répond aussi par un message de type « INFORM » indiquant sa capacité de production, l'historique de sa production, de ses lead times...
- A ce stade, la négociation concernant les prévisions de ventes est terminée. Une nouvelle négociation débute entre les deux agents dès que l'un d'eux crée des prévisions de commandes et en informe l'autre.
- Chacun des deux peut être l'auteur d'une négociation heuristique qui résout une exception sur les prévisions de commandes créées. Cette négociation heuristique est sujette à devenir *récursive* pour impliquer d'autres agents pouvant aider à la résolution. La négociation se termine lorsque toutes les exceptions sont résolues ou le délai fixé dynamiquement est expiré.
- Enfin, selon la situation, l'un d'eux transforme la prévision d'une commande en un engagement d'une commande ferme, puis en informe immédiatement l'autre.

La figure 7 illustre le diagramme de séquence AUMI correspondant lorsque le fournisseur s'occupe de la création des prévisions de ventes et les commandes certaines.

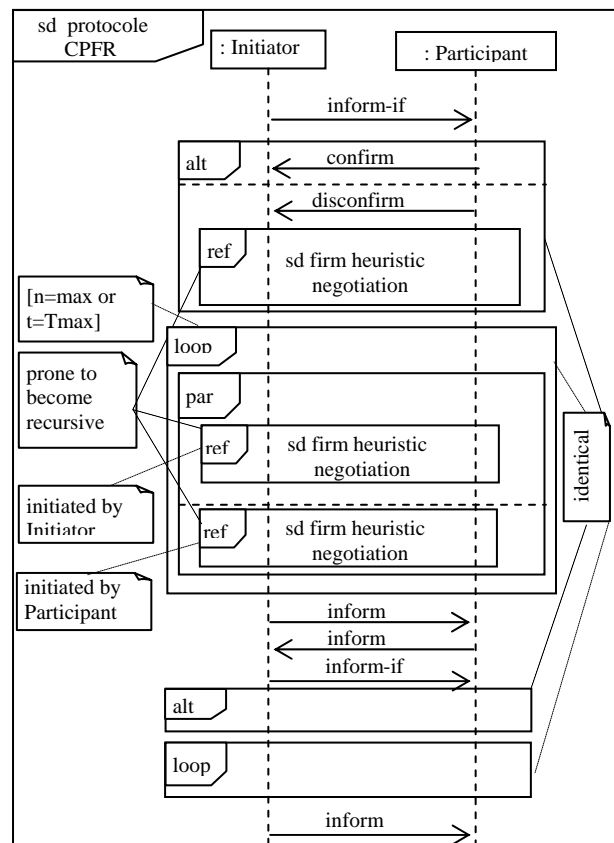


Figure 7. Protocole CPFR

6. COMPORTEMENTS DES AGENTS

Les agents passent d'un état donné à un autre en fonction des actions se produisant dans l'environnement ou en fonction des messages reçus. Les diagrammes d'états (statechart diagram) AUML semblent être une solution efficace pour décrire leurs comportements (Huguet, 2002). Le facteur essentiel que nous avons pris en considération lors de l'élaboration des diagrammes d'états est l'aspect parallèle (concurrentiel) des activités d'un acteur. Par exemple, l'agent « AgentPRC » peut traiter plusieurs commandes en même temps. Nous nous intéressons au comportement de l'agent « AgentDis ». Son comportement dépend du type de l'acteur (Distributeur ou Usine). Le diagramme d'états de la figure 8 illustre ce comportement pour une usine.

7. CAS D'ÉTUDE

Dans le but de tester les protocoles proposés dans cet article, nous avons considéré un cas particulier de supply chain des industries du textile et de l'habillement comprenant une entreprise centrale (maison-mère) et deux filiales F1 et F2 situées en trois pays différents. Les fonctions d'approvisionnement en matières premières, de fabrication et de distribution vers les stocks sont réalisées par l'entreprise centrale. Les fonctions de vente et de distribution aux grossistes sont à la charge des filiales. La filiale F1 distribue les produits à deux grossistes concurrents, tandis que la filiale F2 distribue les produits à deux grossistes non concurrents situés dans deux zones géographiques différentes. Chacun de ces deux derniers utilise le protocole CPFR pour collaborer avec son distributeur (filiale F2), alors que les deux grossistes concurrents utilisent une collaboration simple sans partage de données avec leur distributeur (même filiale F1) puisqu'ils craignent la diffusion de leurs informations stratégiques.

Nous avons représenté les clients finaux par un agent qui passe aux grossistes deux types de commandes :

- Commandes pseudo-cycliques concernant un produit P1.
- Commandes aléatoires uniformes liées au stade de commercialisation d'un nouveau produit de mode noté P2.

Au cours de la simulation, nous avons calculé l'indicateur de performance : *nombre des commandes livrées en retard* par les deux filiales, qui traduit une rupture en linéaire chez les grossistes. Le tableau 1 montre les résultats obtenus pendant la durée de simulation.

	Filiale F1		Filiale F2	
	P1	P2	P1	P2
Nombre des commandes livrées en retard	4	11	1	4

Tableau 1. Indicateurs de performance

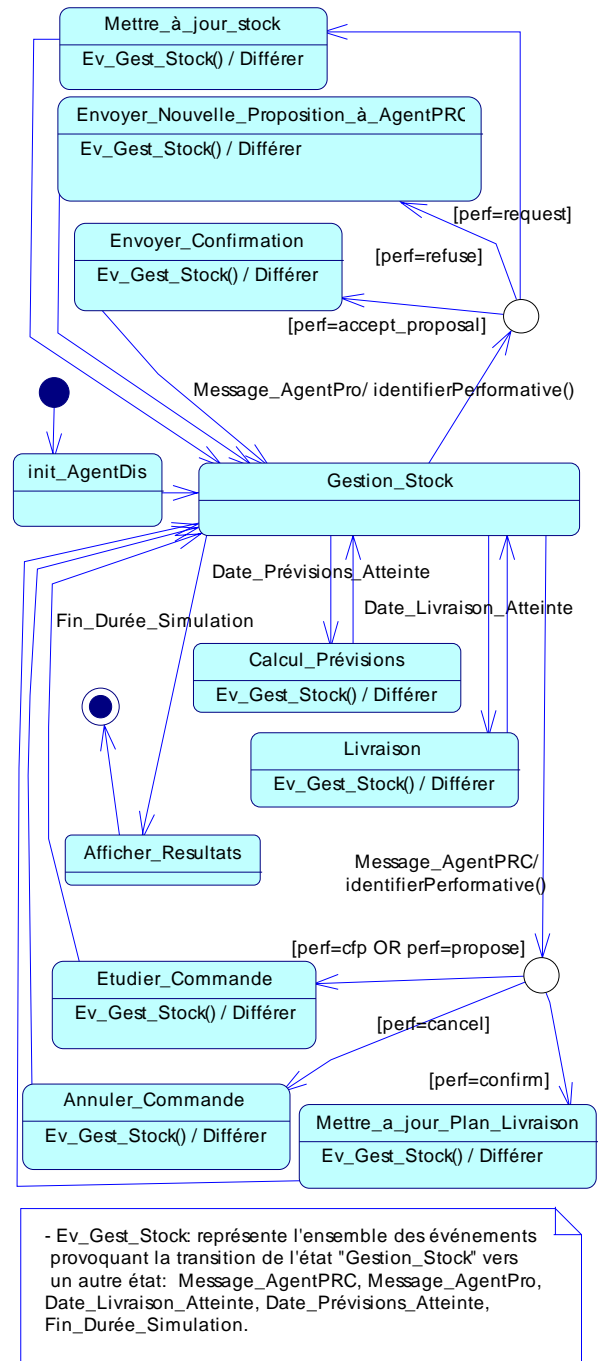


Figure 8. Diagramme d'états-transitions de l'agent "AgentDis" d'un acteur de type "usine".

Analyse des résultats :

- Pour le produit P2, nous remarquons que la filiale F2 a livré seulement 4 commandes en retard devant 11 livrées par F1. La différence entre les deux nombres est relativement importante. Ceci montre très bien que la filiale F2 et ses grossistes ont bénéficié de prévisions de ventes et de commandes plus précises puisqu'ils mettent en œuvre le processus CPFR, surtout que la nature des commandes du produit P2 est aléatoire uniforme. Ce type de commandes est relativement difficile à maîtriser,

car les acteurs (maison-mère, filiales et grossistes) n'ont pas l'idée du comportement du produit sur le marché. C'est le cas de plusieurs produits de mode dans le domaine du textile et d'habillement.

- Pour le produit P1, la filiale F2 a livré une seule commande en retard devant 4 livrées par F1. La différence n'est pas relativement grande, car les commandes concernant le produit P1 et passées par le client final sont pseudo-cycliques. C'est-à-dire qu'au niveau des acteurs, les variations de ces commandes sont connues. C'est le cas de plusieurs produits de textile et d'habillement dont la demande varie selon les saisons de l'année (tee-shirt pour la saison d'été, Manteau pour la saison d'hiver).

Nous déduisons donc que le processus CPFPR mis en place entre la filiale F2 et ses grossistes a réduit considérablement le nombre des commandes livrées en retard. En conséquence, le nombre de rupture en linéaire chez les grossistes sera réduit, ceci se traduira par un bon niveau de service dans l'optique de la fidélisation du client final.

Cette étude de cas qui n'a pour objectif que d'expérimenter les protocoles proposés, nous montre que lorsque le niveau de collaboration augmente, le nombre des commandes livrées en retard diminue. Le niveau de collaboration peut augmenter en choisissant des pratiques et méthodes de collaboration reposant sur le partage des données (comme le CPFPR) et en intégrant plusieurs acteurs dans la prise de décision.

8. CONCLUSION ET PERSPECTIVES

Dans cet article nous avons présenté une architecture de simulation proactive à base d'agents en cours de réalisation qui à terme devrait aider à la prise de décision collaborative dans les chaînes logistiques, en particulier lors de la présence des situations d'urgence dues aux commandes incertaines, commandes imprécises et aux exceptions (problème de production, problème de transport, erreur sur prévisions, retard de livraison, etc.). Le modèle développé à ce stade permet d'expérimenter un ensemble de protocoles de négociation que nous avons proposés et formalisés avec AUML.

Vu la diversité des plate-formes utilisées par les acteurs industriels et la nature distribuée de la supply chain, nous avons choisi l'environnement de développement JADE basé sur le langage JAVA pour implémenter les agents.

Les tests réalisés à ce jour sont très prometteurs et montrent qu'il est possible, en connectant les agents de notre architecture aux systèmes d'informations des acteurs (ERP à titre d'exemple) d'une supply chain, d'apporter une aide importante dans la prise de décision collaborative en cas de situations d'urgence.

Comme perspectives de notre travail nous nous proposons de développer une bibliothèque de classes pour assurer la connexion des agents avec un grand nombre de systèmes d'informations utilisés par les industriels. Nous envisageons pour cela de travailler parallèlement avec des éditeurs de systèmes

d'informations et des entreprises pour lesquelles la problématique de la gestion des commandes imprécises, des commandes incertaines et des exceptions est importante.

REFERENCES

- Bellifemine, F., Poggi, A., Rimassa, G., 1999. A FIPA-compliant agent framework. *CSELT internal technical report, et dans Actes de PAAM'99*, London, Avril 1999, pp.97-108.
- Cardin, O., Castagna, P., 2006. Utilisation de la simulation proactive: Une aide au pilotage des systèmes de production. *In Proceedings of the MOSIM'06*, Rabat, Morocco.
- Cloutier, L., Frayret, J.-M., D'Amours, S., Espinasse, B., Montreuil, B., 2001. A commitment-oriented framework for networked manufacturing coordination. *International journal of computer integrated manufacturing*, 14(6): 522-534.
- CXP, 2001. www.cxp.fr
- ERP, 2006. <http://prisma.insa-lyon.fr/GroupeERP/>. (last access : April 2006).
- Fargier, H. Thierry, C., 2002. The use of possibilistic decision theory in manufacturing planning and control; Recent Results in Fuzzy Master Production Scheduling. *Dans le livre de Scheduling under fuzziness*, R. Slowinski and M. Hapke, Ed. Springer-Verlag, pp. 45-59, 2002.
- FIPA, 2002. FIPA ACL Message Structure Specification. *Foundation for Intelligent Physical Agents*. (<http://www.fipa.org/spcs/fipa00061/SC00061G.pdf>)
- Florea, A., 2002. Using Utility Values in Argument-based Negotiation. *In Proceedings of IC-AI'02, the 2002 International Conference on Artificial Intelligence*, Las Vegas, Nevada, USA, p. 1021-1026.
- Geneste, L., Grabot, B., Moutarlier, P., 2000. Scheduling of heterogeneous data using fuzzy logic in a customer-subcontractor context. Scheduling under fuzziness, dans le livre *Studies in Fuzziness and Soft Computing serie*, Ed. R. Slowinski et M. Hapke, ISBN 1434-9922, Springer-Verlag.
- Giannoccaro, I., Pontrandolfo, P., Scozzi, B., 2003. A fuzzy echelon approach for inventory management in supply chains. *European Journal of Operational Research* 149 (2003) 185-196.
- Govindu, R., Chinnam, R.B., 2007. MASCF: A generic process-centered methodological framework for analysis and design of multi-agent supply chain systems. *Computers & Industrial Engineering*,
- Hapke, M. and Slowinski, R., 2000. Fuzzy set approach to multi-objective and multi-mode project scheduling under uncertainty, *Scheduling under fuzziness, Studies in Fuzziness and Soft Computing serie*, Ed. R. Slowinski and M. Hapke, Springer-Verlag, 197-221.
- Huget, 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.), Bologne, Italie.
- Huget, 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.
- Iassinovski, S., Kuiss, K., Fagnart, C., 2006. Prise de décision et simulation en ligne : Application sur la ligne d'assemblage du CTT de l'AIP AUVERGNE. *In Proceedings of MOSIM'06*, Rabat, Morocco.
- Jennings, N. R., e.a., 2001. Automated negotiation: Prospects, methods and challenges. *Group Decision and Negotiation Journal*, 10(2), 2001, p.199-215.
- Kleijnen, J.P.C., Gaury, E., 2003. Short-term robustness of production management systems: A case study. *European Journal of Operational Research*, 148, p. 452-465.
- Lamouri, S., 2006. Synchronisation des prises de décisions dans une chaîne logistique: robustesse et stabilité. *Mémoire d'Habilitation à Diriger des Recherches. École Doctorale d'Informatique et Électronique de Paris. Institut Supérieur de Mécanique de Paris (Supméca Paris)*.
- Letouzey, A., Geneste, L., Grabot, B., 2001. Production planning forecasting under uncertainty using possibility theory. *CASYS 2001, 5th International Conference on Computing Anticipatory Systems*, Liege Belgique, 13-18 Aout 2001.
- Maturana, F., Shen, W. and Norrie, DH., 1999. MetaMorph: an adaptive agent-based architecture for intelligent manufacturing. *International Journal of Production Research*, 37(10), pp. 2159-2174.
- 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*, pages 71-76. John Wiley et Sons.
- Pujo, P. Pedetti, M., Ounnar, F., 2004. Pilotage proactif des lignes de production kanban par modélisation DEVS et simulation temps réel. *In Proceedings of the 4e Conférence Francophone de MOdélisation et SIMulation. MOSIM'04*, Nantes, France, pp. 593-600.
- Reynoso, C., 2004. Prise En Compte De Commandes Incertaines Et Imprecises Dans Une Logique MRP2. *Dans les actes des Rencontres Francophones sur la Logique Floue et ses Applications (LFA'02)*, Montpellier – France, 21-22 Octobre 2002.
- Rota, K., 1998. Coordination Temporelle de Centres Gérant de Façon Autonomes des Ressources:Application aux Chaînes Logistiques Intégrées en Aéronautique. *Thèse de doctorat. Université de Toulouse, France*.
- Simchi-Levi, D. Kaminsky, P. Simchi-Levi, E., 2000. Designing and managing the supply chain: Concepts, Strategies and Case Studies. Editions, McGraw Hill.
- Wooldridge, M. and Jennings, N. R., 1995. Intelligent agents: Theory and practice. *The Knowledge Engineering review*, 10(2) : 115-152.