

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Ecole Nationale Polytechnique



Département Génie industriel
Laboratoire d'informatique de Paris 6

Mémoire de projet de fin d'études pour l'obtention du
diplôme d'ingénieur d'état en management de
l'innovation

Une approche multi-agents à base de négociation pour la gestion de la chaîne logistique

M. Mohamed SAADI

M. Mourad LAAZIZ

Sous la direction de :

M. Iskander ZOUAGHI

M. Sylvain DUCTOR

Mme. Zahia GUESSOUM

Présenté et soutenu publiquement le (20/06/2016)

Composition du Jury :

| | | | |
|-----------|----------------------------|------------------------|-----|
| Président | M. Wassim BENHASSINE | Maitre de conférence B | ENP |
| Examineur | M. Redouane EL HADJ KHALEF | Maitre de conférence A | ENP |
| Promoteur | M. Iskander ZOUAGHI | Maitre de conférence B | ENP |

ENP 2016

Important :

Suite à la réalisation de ce projet de fin d'étude, nous avons abouti à la conception d'un système multi-agents à base de négociation pour le management de la chaîne logistique.

Vue l'importance d'un tel système actuellement pour les entreprises ainsi que la forte demande par rapport à l'offre qui est en plein développement. Nous avons opté pour lancer les démarches dans le but de breveter notre produit afin de le protéger. De ce fait plusieurs informations propre au projet, tel que le code source de l'application, ne vont être visible que pour l'enseignant encadreur ainsi que le jury de soutenance, et ceci dans le but d'évaluer notre travail de fin d'études.

REMERCIEMENTS

Avant tout, nous remercions Allah.
Le tout puissant pour cette existence.

Toute notre gratitude s'adresse à nos parents, les membres de nos familles et nos amis.

Nos reconnaissances et vifs remerciements vont tout d'abord à notre encadreur Dr. Iskander ZOUAGHI, qui nous a épaulé tout au long de notre travail, et a répondu présent par sa bienveillance et ses conseils.

Nous tenons à remercier en tout premier lieu Mme. Zahia GUESSOUM pour nous avoir accepté au sein du LIP6. Ses remarques, orientations et conseils nous ont prêté main forte pour arriver au bout de ce travail. Aussi M. Sylvain DUNCON pour sa grande disponibilité, son soutien et sa considération respectueuse.

Enfin, nous remercions les membres du jury qui nous font l'honneur d'évaluer notre travail.

ملخص

تحت ظل بيئة معقدة سريعة التقلب وتتطلب الكثير من المرونة وسرعة الاستجابة. فإن تسيير السلسلة اللوجستية القائم على تسيير العلاقات الداخلية قد اثبت محدوديته 'بولويب إفاكت' *bullwhip effect* في هذا السياق أدركت المؤسسات أن عملية التنسيق قد تكون مصدرا تنافسيا وحلا لمثل هذه المشاكل. ورغم ذلك فإن عملية التنسيق في حد ذاتها لا تخلو من العديد من المشاكل المتعلقة بعاملين أساسيين هما المعلومات والجوانب السلوكية.

من أجل مواجهة هذه المشاكل، فإن الأنظمة متعددة العملاء تبدو الحل الأنسب. لذلك فإن هذه الدراسة هي تجسيد لهذه الأنظمة في سلسلة لوجستية ثنائية المستوى بين موزع ومورد. التنسيق بين الجانبين تم عن طريق المفاوضة بدمج عقدي تنسيق 'الباي باك' *Buyback* و 'الريفنيو شارينج' *Revenue Sharing*. الدراسة تتضمن التشكيل التدريجي للحل، انجاز التصميم، إضافة إلى إنجاز النظام على أرض الواقع.

الكلمات المفتاحية: تسيير السلسلة اللوجستية، تنسيق نظام متعدد العملاء، بولويب إفاكت (*Bullwhip effect*)، مفاوضة، عقود السلسلة اللوجيستية.

Abstract

In a dynamic, complex environment requiring both flexibility and reactivity. SCM approaches based on managing intra-organizational relations have proven their limits (*Bullwhip effect*, etc.). In this context, companies realized that coordination might be a source of competitiveness, and a solution for such limits. However, collaboration approaches also present many problems mainly related to two factors, information and behavioral aspects.

To deal with those problems, MAS appear to be the most adequate solution. This study, present a multi agent approach for a two stage supply chain consisting of one supplier and one retailer. The coordination is achieved through bargaining and combination of two contracts (*Buyback* and *Revenue sharing*). The study involves the solution formulation, system's design and implementation.

Key words: Supply Chain Management (SCM), Coordination, Multi-Agent Systems (MAS), *Bullwhip effect*, Bargaining, Supply chain contracts.

Résumé

Dans un environnement de plus en plus complexe, dynamique et exigeant en termes de flexibilité et de réactivité. Le SCM basée sur la gestion des relations intra-organisationnelles a prouvé ses limites (*Bullwhip effect*, etc.). Dans ce contexte les entreprises ont réalisé que la coordination peut être une source de compétitivité, aussi une solution à ces limites. Cependant les approches de collaborations présentent aussi plusieurs problèmes principalement liés à deux facteurs, l'information et les aspects comportementaux.

Pour faire face à ces problèmes, les SMAs apparaissent comme la solution la plus adéquate. Cette étude, présente une approche multi agents dans une chaîne logistique à deux niveaux avec un fournisseur et un distributeur. La coordination entre les deux parties est atteinte à travers le bargaining et une combinaison de deux contrats (*Buyback* and *Revenue sharing*). L'étude comporte la formulation de la solution, la réalisation du modèle ainsi que son implémentation.

Mots clés : Supply chain management, Coordination, Systèmes multi-agents, *Bullwhip effect*, Bargaining, Contrats de la chaîne logistique.

TABLE DES MATIERES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES ABREVIATIONS

| | |
|--|----|
| Introduction Générale..... | 11 |
| Chapitre 1 : Management et dynamique de la chaîne logistique..... | 20 |
| 1.1. Introduction | 20 |
| 1.2. Définitions..... | 21 |
| 1.2.1. Chaîne logistique..... | 21 |
| 1.2.2. Structure de la chaîne logistique | 23 |
| 1.2.3. Management de la chaîne logistique | 25 |
| 1.3. Dynamique et complexité du management de la chaîne logistique | 26 |
| 1.3.1. Evolution du management de la chaîne logistique | 26 |
| 1.3.2. Optimisation de la chaîne logistique | 27 |
| 1.3.3. Effet coup de fouet | 28 |
| 1.3.4. Coordination et collaboration de la chaîne logistique | 32 |
| 1.4. Les contrats de la chaîne logistique..... | 34 |
| 1.4.1. Introduction aux contrats de la chaîne logistique..... | 34 |
| 1.4.2. Présentation des différents contrats..... | 39 |
| 1.4.3. Problèmes liés aux contrats et alternatives de résolution..... | 43 |
| 1.5. Conclusion..... | 44 |
| Chapitre 2 : Les systèmes multi-agents..... | 46 |
| 2.1. Introduction | 46 |
| 2.2. Principes généraux | 47 |
| 2.2.1. Le paradigme agent | 47 |
| 2.2.2. Les systèmes multi agents | 49 |
| 2.3. La communication entre les agents | 54 |
| 2.4. L'interaction..... | 55 |
| 2.5. Méthodologies et environnements de développement..... | 57 |
| 2.5.1. Méthodologies de développement..... | 57 |
| 2.5.2. Environnements de développement..... | 60 |
| 2.6. Les systèmes multi agents pour la gestion de la chaîne logistique | 65 |
| 2.6.1. Choix des systèmes multi agents..... | 65 |

| | |
|--|-----|
| 2.6.2. Approche de coordination de la chaîne logistique à travers les SMAs | 68 |
| 2.7. Conclusion..... | 77 |
| Chapitre 3 : Présentation de la solution..... | 80 |
| 3.1. Introduction | 80 |
| 3.2. Description du problème | 81 |
| 3.2.1. Contexte d'étude | 81 |
| 3.2.2. Définition du problème | 82 |
| 3.3. Solution proposée..... | 85 |
| 3.3.1. Solutions existantes | 85 |
| 3.3.2. Description de la solution..... | 86 |
| 3.4. Conclusion..... | 104 |
| Chapitre 4 : Modélisation et implémentation | 106 |
| 4.1. Introduction | 106 |
| 4.2. Méthodologie de développement | 109 |
| 4.3. Développement des modèles de notre système | 111 |
| 4.4. Implémentation : | 129 |
| 4.4.1. Mécanisme de coordination..... | 131 |
| 4.4.2. Mécanisme de détection de fraude | 135 |
| 4.4.3. Le système respecte-il les concepts clés du paradigme agent ? | 141 |
| 4.5. Conclusion..... | 145 |
| Conclusion générale | 146 |
| BIBLIOGRAPHIE | 150 |
| LISTE DES ANNEXES..... | 162 |

LISTE DES TABLEAUX

| | |
|---|-----|
| Table 1 : Dilemme de prisonnier sans contrat..... | 35 |
| Table 2 : Dilemme de prisonnier avec contrat | 36 |
| Table 3 : Notation des contrats de la chaîne logistique | 37 |
| Table 4: Différence entre un agent et un objet | 49 |
| Table 5 : Les architectures d'agents (Briot et Demazeau. 2016). | 54 |
| Table 6 : les typologies des interactions (Ferber. 1995)..... | 57 |
| Table 7 : la forme standard matricielle..... | 72 |
| Table 8 : Les modèles de coordination de la chaîne logistique Li et Wang (2007, p-3)..... | 82 |
| Table 9 : Les résultats de complexité avec la contrainte NLS (Kedad-Sidhoum et al. 2015b). | 83 |
| Table 10 : Les scénarios possibles par rapport aux bids. | 98 |
| Table 11: Mediation request message. | 119 |
| Table 12: Bid message (from supplier to mediator) | 119 |
| Table 13: Reply regarding bargaining request message..... | 119 |
| Table 14: Bid message (from retailer to mediator). | 119 |
| Table 15: Posting Retailer reply message. | 120 |
| Table 16: Bid request (from mediator to supplier)..... | 120 |
| Table 17: Propose solution message. | 120 |
| Table 18: Bargaining request message..... | 120 |
| Table 19: Bid request (from mediator to retailer). | 121 |
| Table 20 : Propose solution message. | 121 |
| Table 21: Submit purchase order message..... | 121 |
| Table 22: Communicate expected sales message..... | 121 |
| Table 23: Agreement message. | 122 |
| Table 24: Counter propose message..... | 122 |
| Table 25: FIPA-ACL Performatifs..... | 183 |

LISTE DES FIGURES

| | |
|--|-----|
| Figure 1 : Les cycles de Kondratiev ou les vagues longues de la croissance (Dennis et al., 2010, p.6)..... | 12 |
| Figure 2 : Vente de Smartphones dans le monde (Samsung, Apple, Nokia, RIM, HTC)..... | 13 |
| Figure 3: Déroulement du projet | 17 |
| Figure 4 : Les types de la chaîne logistique (Mentzer et al, 2001-p5) | 22 |
| Figure 5 : Structure de la chaîne logistique (Lambert, 2009)..... | 24 |
| Figure 6 : L'effet de coup de fouet (Pino et al. 2003)..... | 29 |
| Figure 7 : Présentation de la chaîne logistique (Chopra et Meindl, 2012)..... | 34 |
| Figure 8 : Le modèle basic de la chaîne logistique à une période (Höhn 2010). | 37 |
| Figure 9: le profit en fonction du crédit (Snyder et Shen 2011, p-293) | 40 |
| Figure 10 : Les phases de réaction d'un agent (Mandiau et Strugeon. 2002) | 48 |
| Figure 11 : Différence entre un agent et un objet (Ferber 1995)..... | 48 |
| Figure 12: Agent avec simple reflex (Chaib-draa 2006)..... | 50 |
| Figure 13 : Agent avec reflex (Chaib-draa 2006)..... | 50 |
| Figure 14 : Agent à base de buts (Chaib-draa 2006)..... | 51 |
| Figure 15 : Agent à base d'utilité (Chaib-draa 2006)..... | 51 |
| Figure 16 : Approche voyelle (Briot et Demazeau. 2016) | 53 |
| Figure 17 : Les types de relation entre les agents dans un SMA..... | 56 |
| Figure 18 : Utilisation de CPU avec lancement du behavior | 61 |
| Figure 19 : Utilisation de CPU sans lancement du behavior..... | 61 |
| Figure 20: Filtrage des messages | 62 |
| Figure 21: Customizing a JFrame Window..... | 63 |
| Figure 22 : Making a custom agent out of jade.core.Agent class | 64 |
| Figure 23: Représentation d'un système multi agents (Wooldridge 2009)..... | 66 |
| Figure 24 : chaîne logistique avec regroupement des acteurs (D'amours et al. 2006). | 67 |
| Figure 25 : Contract net protocol (Wooldridge 2009)..... | 68 |
| Figure 26 : Arbre de jeu (Leyton-Brown et Shoham. 2010) | 72 |
| Figure 27 : Arbre de jeu avec information imparfaite (Leyton-Brown et Shoham. 2010)..... | 73 |
| Figure 28 : Arbre de jeu avec information incomplète (Leyton-Brown et Shoham. 2010) | 73 |
| Figure 29 : Séquences du bargaining (Florea 2012)..... | 75 |
| Figure 30 : Application de la négociation dans la chaîne logistique (D'amours et al. 2008) .. | 76 |
| Figure 31: Organisation du chapitre de la présentation de la solution. | 80 |
| Figure 32 : Problème du lot-sizing entre fournisseur et distributeur..... | 81 |
| Figure 33 : Les solutions proposées lors de l'étude de Kedad-Sidhoum et al (2015a)..... | 84 |
| Figure 34 : Les types d'optimisation d'une chaîne logistique à deux niveaux | 87 |
| Figure 35 : Diagramme de séquence | 88 |
| Figure 36 : Les étapes de la solution | 89 |
| Figure 37: Simulation graphique cas 1.1 | 94 |
| Figure 38: Simulation graphique cas 1.2..... | 94 |
| Figure 39: Simulation graphique cas 2.1 | 95 |
| Figure 40: Simulation graphique cas 2.2..... | 95 |
| Figure 41: Simulation graphique cas 3.1 | 96 |
| Figure 42: Simulation graphique cas 3.2..... | 96 |
| Figure 43: Simulation graphique cas 4.1 | 97 |
| Figure 44: Simulation graphique cas 4.2..... | 97 |
| Figure 45: Méthodologie de développement..... | 108 |
| Figure 46: Agent UML model (mediator)..... | 116 |
| Figure 47: Agent UML model (supplier) | 117 |

| | |
|--|-----|
| Figure 48: Agent UML model (Retailer) | 118 |
| Figure 49: Mediator activity model..... | 123 |
| Figure 50: Supplier activity model..... | 124 |
| Figure 51: Retailer activity model..... | 125 |
| Figure 52: Mediator environment model..... | 126 |
| Figure 53: Supplier environment model..... | 127 |
| Figure 54: Retailer environment mod | 128 |
| Figure 55: Project tree | 130 |
| Figure 56: Code Java pour le calcul de I et S..... | 132 |
| Figure 57: code Java pour calculer les profits du retailer | 132 |
| Figure 58: Code Java pour calculer le profit du fournisseur | 133 |
| Figure 59: Code Java pour calculer le profit du fournisseur..... | 133 |
| Figure 60: Détermination de la solution Pareto Optimal. | 134 |
| Figure 61: Vérification de la convergence du bargaining. | 135 |
| Figure 62: Les trois scénarios possibles | 136 |
| Figure 63: Approuve customer order | 137 |
| Figure 64: Retailer refuse to coordinate..... | 138 |
| Figure 65 : Initiate bargaining..... | 139 |
| Figure 66: Bidding process | 140 |
| Figure 67: Algorithme de Wagelmans - Initialisation (Wagelmans et al 1992) | 164 |
| Figure 68: Algorithme de Wagelmans - Itérations itérations. (Wagelmans et al 1992)..... | 166 |
| Figure 69: Algorithme de Wagelmans - Construction de la solution (Wagelmans et al 1992) | 166 |
| Figure 70: L'interface principale de JADE..... | 167 |
| Figure 71: Les messages de JADE..... | 168 |
| Figure 72: Architecture JADE (Bellifemine et al, 2004) | 169 |
| Figure 73: Relation entre les éléments de l'architecture (Bellifemine et al, 2004)..... | 169 |
| Figure 74: Les services de la page jaune (Bellifemine et al, 2004)..... | 170 |
| Figure 75: Le diagramme de class des outils de JADE (Bellifemine et al, 2004)..... | 172 |
| Figure 76: JADE RMA GUI | 173 |
| Figure 77: Démarrer un agent..... | 174 |
| Figure 78: Définir les attributs d'un agent | 175 |
| Figure 79: Création de l'agent dans le conteneur..... | 176 |
| Figure 80: Envoi de message | 177 |
| Figure 81: Interface message (ACL-FIPA)..... | 177 |
| Figure 82: Dummy agent..... | 178 |
| Figure 83 : Inspector agent..... | 179 |
| Figure 84: Sniffer Agent | 180 |
| Figure 85: Directory facilitator avec description. | 181 |
| Figure 86: Log manager GUI..... | 182 |
| Figure 87: Seller agent gui | 184 |
| Figure 88: Buyer agent gui..... | 185 |
| Figure 89: Le diagramme d'interaction..... | 186 |
| Figure 90: Le processus d'interaction par 'Sniffer Agent' | 186 |
| Figure 91: Dummy agent book trading | 187 |
| Figure 92 : Dummy agent Book trading two..... | 188 |
| Figure 93: Buyer agent information of deployment | 188 |
| Figure 94: Extended goal model | 189 |
| Figure 95: Complete protocol model | 190 |

LISTE DES ABREVIATIONS

ACL : Agent Communication Langage.

CDSSs: Collaborative Decision Support Systems.

CSCMP: Council of Supply Chain Management Professionals.

EOQ: Economic ordering quantity.

IA : Intelligence Artificielle.

IAD : Intelligence Artificielle Distribuée.

IB: Inventory Bounds.

JADE: Java Agent Development framework.

KQML: Knowledge Query and Manipulation Language.

NE: Nash Equilibrium.

NLS: No Lot Splitting.

SC: Supply Chain.

SCM: Supply Chain Management.

SMAs: Systèmes Multi Agents.

OLAP: Online Analytical Processing.

PO: Pareto Optimum.

RO : Recherche Opérationnelle.

ULS : Uncapacitated Lot Sizing.

Introduction Générale

Le contexte économique actuel est caractérisé par la globalisation des échanges, les transformations et les innovations dans tous les domaines (technologiques, organisationnels, etc.) de plus que l'intensification de la concurrence. Devant ces facteurs et depuis plusieurs années les entreprises opèrent dans un environnement très difficile et instable. Cela est dû selon Dennis et al. (2010) et reporté par Zouaghi (2013) aux successions des périodes de croissance, de récession et même de dépression (figure 1).

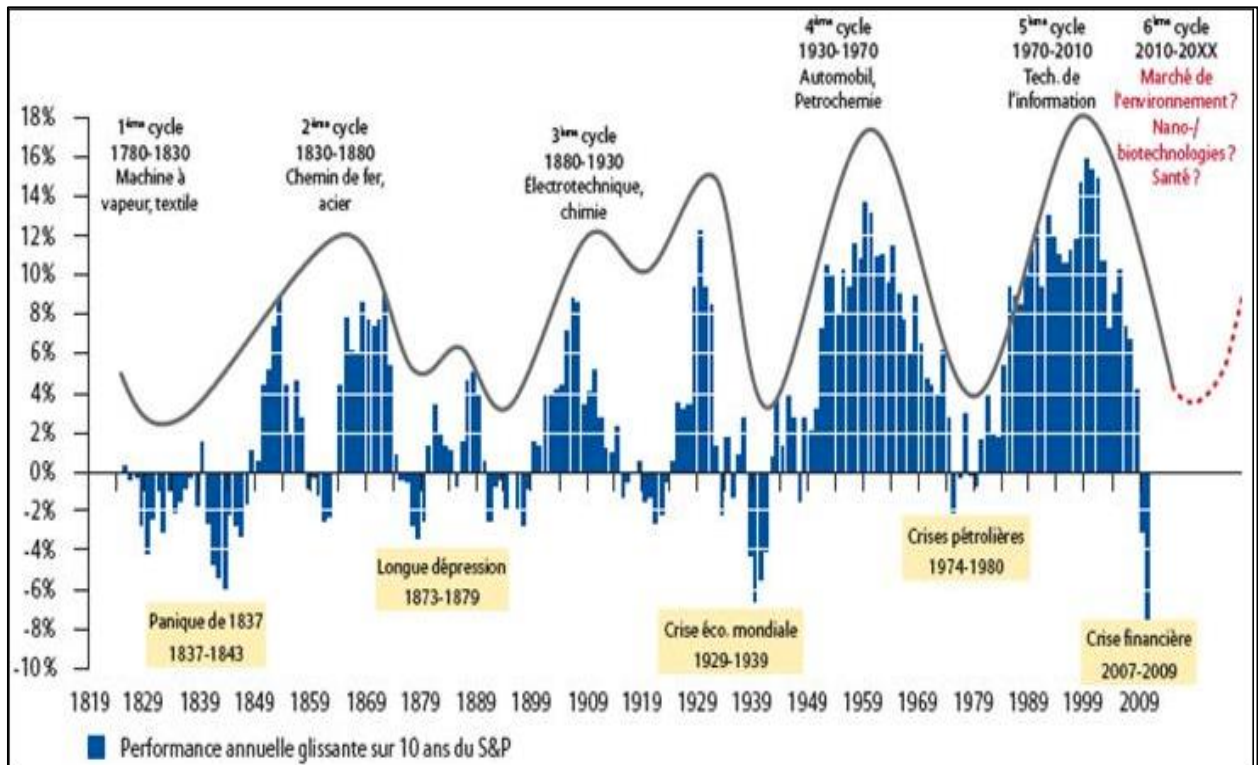


Figure 1 : Les cycles de Kondratiev ou les vagues longues de la croissance (Dennis et al., 2010, p.6)

Aussi, un environnement de plus en plus dynamique avec des transformations dans le marché et des exigences variables des clients. Le marché des smartphones illustre très bien cette notion. Ce secteur a connu dans la période entre 2009 et 2011 des grands changements de la configuration par rapport aux constructeurs, comme montre la Figure.

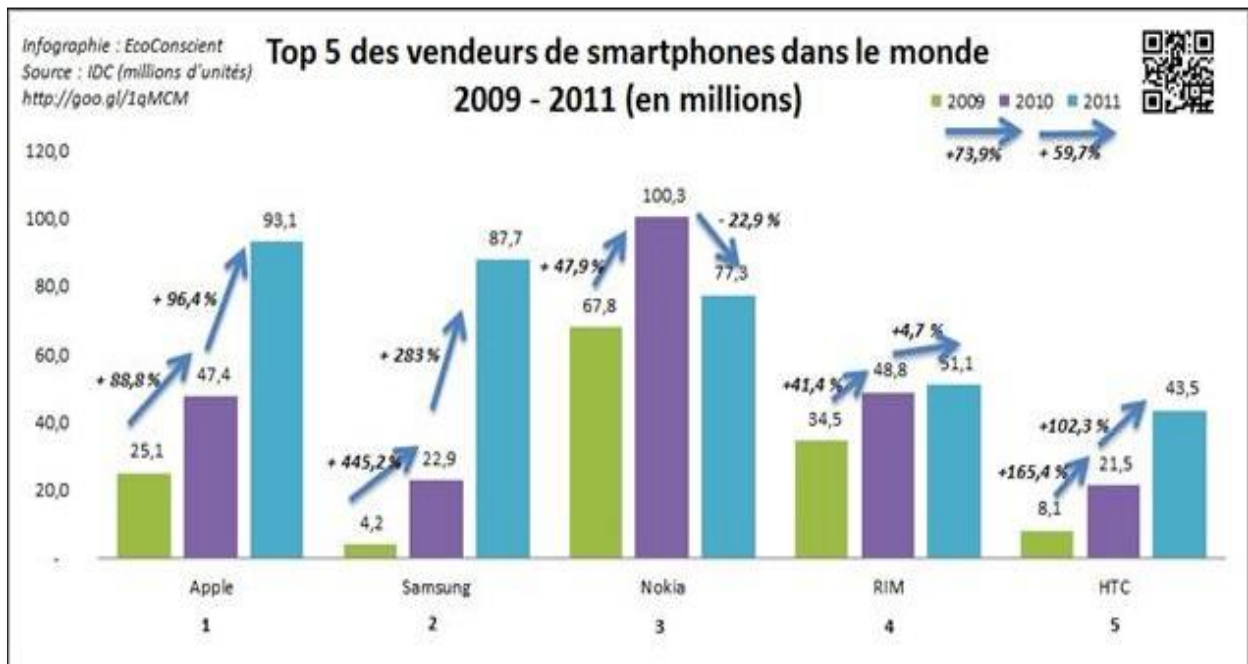


Figure 2 : Vente de Smartphones dans le monde (Samsung, Apple, Nokia, RIM, HTC)

De plus, la complexité et l'accroissement du risque de l'environnement liée principalement à la l'augmentation de l'incertitude et la structure compliquée du marché ainsi que l'organisation. Dans un tel contexte, les entreprises doivent non seulement satisfaire les besoins en termes de coût et de qualité mais aussi en termes de flexibilité et de réactivité. Pour se faire le SCM se présente comme l'approche la plus adéquate. Initialement le SCM avait comme priorité la réduction des coûts avec la résolution de problèmes d'ordre opérationnel (intra-organisationnels). Ces améliorations étaient invisibles pour des acteurs isolés. Aussi plusieurs difficultés sont apparus, incarnant un phénomène majeur qui est le 'Bullwhip Effect'.

Vers la fin de ce dernier siècle, le SCM est devenu une source d'avantage concurrentiel plutôt qu'une simple solution orientée vers la réduction des coûts. Depuis les entreprises se sont intéressées à la gestion des relations inter-organisationnelles à travers les approches de collaborations et d'intégrations de plus de la gestion des relations intra-organisationnelles.

Par ailleurs, ces approches et malgré leur apport considérable pour les entreprises, présentent plusieurs limites devant un environnement qui devient de plus en plus dynamique et complexe. D'abord la limitation de l'autonomie des acteurs à cause de la centralisation et le partage d'informations. Aussi qu'elles ne reflètent pas la réalité par rapport au comportement et

l'interaction des acteurs. Pour faire face à ces problèmes, aussi assurer une certaine autonomie des acteurs avec des comportements coopératif et compétitif à la fois. Les systèmes multi agents qui s'inscrivent dans le domaine de l'intelligence artificielle distribuée, se présentent comme un des choix le plus adéquat vue qu'ils se situent entre les systèmes distribués et l'intelligence artificielle.

Plusieurs solutions ont en effet été proposées dans la littérature mais la majorité des solutions qui utilisent les systèmes multi-agents présentent des limites par rapport à des points importants. Par exemples, ces solutions utilisent des agents mais ne proposent pas des solutions pour gérer des problèmes de coordination.

Donc les questions qui se posent sont, pourquoi les SMAs et comment les utiliser pour faire face aux problèmes de la chaîne logistique ? Et comment que les SMAs peuvent être utilisés pour coordonner la chaîne logistique ?

Pour répondre à ces questions et à travers ce projet, nous allons réaliser une solution multi-agents pour gérer la chaîne logistique. Cette solution sera fondée sur des mécanismes de négociations.

Notre travail peut être subdivisé en quatre chapitres repartis entre deux parties :

La première partie contient deux chapitres. Elle est consacrée à l'état de l'art où nous allons traiter les principales définitions et concepts relatifs au thème du projet, le cadre conceptuel et l'émergence de la problématique. La deuxième partie est composée de trois chapitres. Elle est consacrée à la solution, où nous allons d'abord définir le contexte du problème, le modéliser pour ensuite passer à sa résolution.

Chapitre 1 :

Lors du premier chapitre et dans le premier point, nous allons commencer par les définitions et les concepts clés relatives au management et à la dynamique de la chaîne logistique. D'abord nous allons décrire le lien entre la logistique et la chaîne logistique, ainsi que la complexité que peut atteindre cette dernière pour passer ensuite à la description de sa structure. Pour enfin introduire le concept du management de la chaîne logistique.

Le deuxième point, traitera l'évolution du SCM d'une direction coût à une source d'avantage concurrentielle. Après, ce passage sera détaillé en décrivant en premier lieu l'optimisation locale ainsi que ses limites. Une des conséquences majeures sera par la suite discutée, il s'agit du Bullwhip effect. Pour ensuite parler sur l'émergence de l'optimisation globale à travers les approches de collaborations et de coordination pour faire face à ces problèmes.

Le troisième point, présentera les contrats de la chaîne logistique comme un appui aux approches de collaborations. Pour enfin cités les limites des approches de collaborations, aussi la proposition de la solution multi agents et qui va être le sujet du chapitre 2.

Chapitre 2 :

Le premier point de ce chapitre traitera les définitions et les concepts clés autour des systèmes multi agents, en commençant par le paradigme des agents. Ainsi, vu que dans les SMAs l'interaction est un élément très important, le deuxième point va être consacré aux types de communication entre les agents, alors que le troisième illustre les différents types de relations entre les agents aussi bien que les situations d'interaction. Dans le quatrième point nous aborderons le volet développement, avec un aperçu sur les méthodologies et les environnements dédiés au développement des SMAs.

Après que nous avons eu une vision générale sur les SMAs. Le cinquième point présente une justification de leurs choix pour la gestion de la chaîne logistique. Nous détaillerons à la fin les mécanismes de négociation vue leurs importances pour la réalisation de notre système.

Chapitre 3 :

Après avoir bien cerné la problématique par la description du SCM et des SMA à travers les deux premiers chapitres et qui constituent la boîte à outils pour le troisième chapitre dans lequel nous allons d'abord définir le problème et en proposer une solution.

Dans le premier point qui est consacré à la description du problème, nous allons commencer par définir le contexte de notre étude (chaîne logistique à deux niveaux), pour ensuite passer à la définition du problème à travers l'analyse d'une étude qui a été faite récemment dans le même contexte (problème de dimensionnement de lots - *lot-sizing problem*).

Le deuxième point abordera la solution proposée. Nous allons d'abord voir les solutions existantes par une revue de littérature des modèles pertinents dans le même contexte pour bien situer notre solution. Par la suite et dans le même point, nous allons décrire la solution bargaining avec médiateur) par un diagramme de séquence qui sera détaillé par la suite étape par étape, avant de passer à la modélisation du système multi agent.

Chapitre 4 :

Une fois la solution élaborée, nous définissons le modèle à travers une méthodologie bien définie dans ce chapitre. Dans ce sens nous avons travaillé avec une combinaison entre Gaia et ROADMAP, suivant la méthodologie nous avons procédé à la modélisation qui comporte 21 modèles. Le chapitre présente les modèles essentiels (Extended goal, Complete rôle, Complete protocole, activity and agent UML) qui vont servir lors de l'implémentation par la suite.

Enfin et une fois que nous avons terminé la modélisation, nous pouvons passer à l'implémentation du modèle. Pour ce faire et lors de ce chapitre nous allons décrire l'implémentation de notre modèle. Avec une présentation du fonctionnement générale, aussi la réalisation des tests et des simulations dans le but de valider le système.

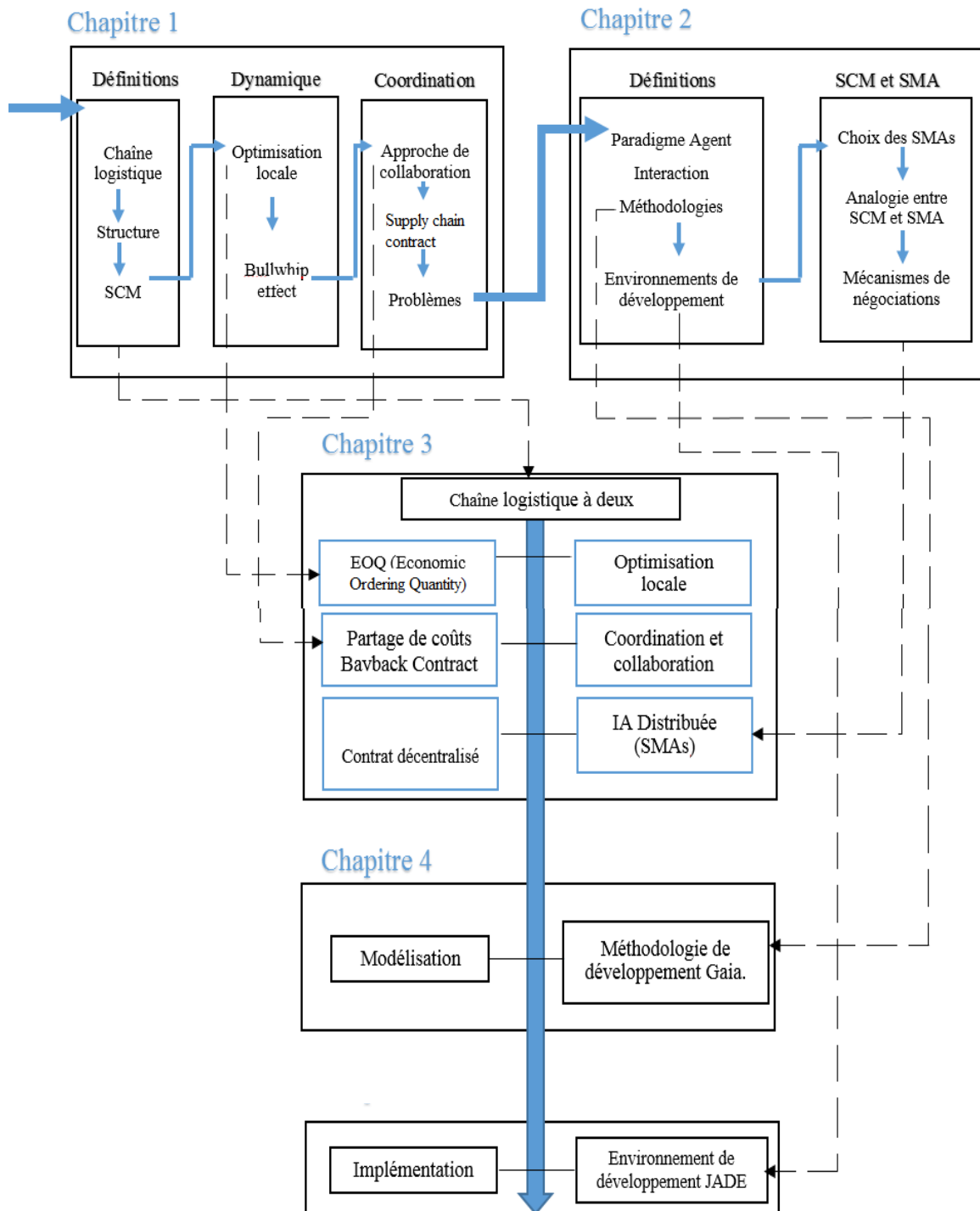


Figure 3: Déroulement du projet

Partie 1 : Etat de l'art

Chapitre 1 : Management et dynamique de la chaîne logistique

1.1. Introduction

Traditionnellement la majorité des entreprises se voyaient comme des entités qui existent indépendamment des autres et donc pour survivre il faut de la compétition dans une vision purement rationnelle (utilité individuelle). Cependant, avec une telle philosophie les entreprises suspendent toute sorte de coopération. Alors que cette dernière et avec la dynamique et la complexité de l'environnement actuel, se présente comme la solution la plus adéquate pour renforcer la compétitivité dans une logique SCM. Et ce pour faire face aux problèmes de la vision traditionnelle mais sans pour autant compromettre l'autonomie de ces acteurs.

A travers ce chapitre nous allons voir les notions et les concepts clés du SCM autour de notre thème, avec une structure bien définie :

Le premier point, regroupe les définitions relatives au management et à la dynamique de la chaîne logistique. Le deuxième point, traitera l'évolution du SCM de l'optimisation locale à l'optimisation globale à travers les approches de collaboration et de coordination. Le troisième point, présentera les contrats de la chaîne logistique comme un appui aux approches de collaborations.

1.2. Définitions

La logistique est une fonction dont la finalité est la satisfaction des besoins exprimés ou latents, aux meilleures conditions économiques pour l'entreprise et pour un niveau de service déterminé. Les besoins sont de nature interne (approvisionnement de biens et de services pour assurer le fonctionnement de l'entreprise) ou externe (satisfaction des clients) (Norme AFNOR X 50-600).

La logistique met l'accent sur la coordination des opérations logistiques élémentaires afin d'obtenir un optimum satisfaisant. Alors que de nombreuses recherches ont montré que l'atteinte des objectifs dans un environnement multi acteurs doit passer par la coordination avec toutes les entités impliquées via leurs interdépendances. Selon Christopher (2011), de ce cadre vient le management de la chaîne logistique comme un concept plus large que la logistique afin d'assurer la coordination des processus des autres entités (fournisseurs, clients et l'organisation elle-même). Dans ce qui suit nous allons d'abord aborder quelques concepts clés autour du management de la chaîne logistique pour ensuite passer à sa définition.

1.2.1. Chaîne logistique

La notion de la chaîne logistique a été largement explorée dans la littérature, ce qui explique bien le grand nombre de définitions qui diffèrent d'un auteur à un autre. Parmi elles nous pouvons citer celle de Christopher (2011, p.13) qui la définit comme :

« Un réseau d'organisations impliquées par une liaison en amont et en aval dans des processus et des activités qui produisent de la valeur sous forme de produits et de services pour le consommateur final » Pour Mentzer et al. (2001, p.5), la chaîne logistique est un :

« Ensemble de trois entités (ou plus) directement impliquées par des flux, amonts et avals, de produits, de services, financiers, et/ou d'informations depuis la source jusqu'au consommateur » (Mentzer et al, 2001 page 5).

Suite à cette définition. Selon (Mentzer, et al. 2001) il est possible de déduire trois types de chaînes logistiques selon leur degré de complexité (figure 4) :

Le premier type est la chaîne logistique directe réduite à trois acteurs : une entreprise focale, un fournisseur et un client. Ces trois acteurs sont impliqués dans le transfert des flux (produits, services, financiers, d'informations) en amont et/ou en aval. Le deuxième type est la chaîne logistique élargie : la chaîne logistique est élargie à plusieurs clients directs et indirects et plusieurs fournisseurs directs et indirects. Enfin, le troisième type est la chaîne logistique globale : elle comprend tous les acteurs impliqués dans tous les types de flux amont et avals, du premier fournisseur au dernier client.

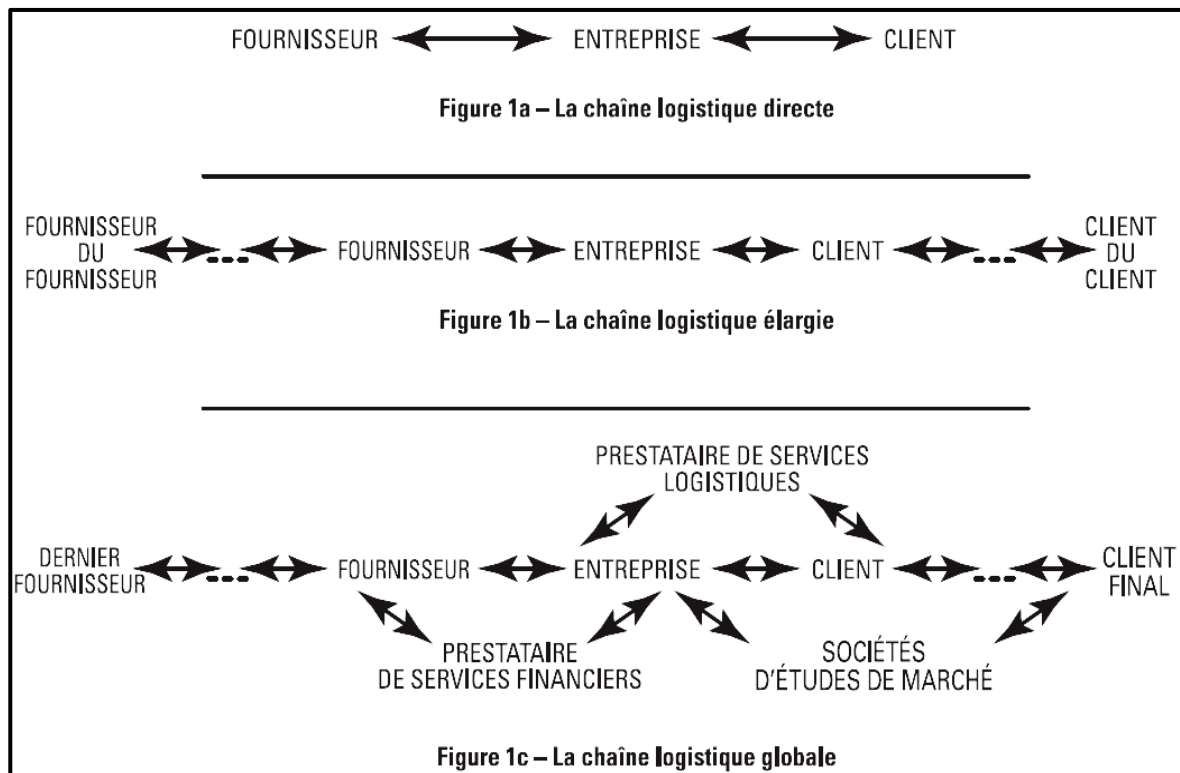


Figure 4 : Les types de la chaîne logistique (Mentzer et al, 2001-p5)

Actuellement, le concept de 'chaîne logistique' peut être remis en question par le fait que le terme 'chaîne' implique un système séquentiel comme celui de la figure 1a. Alors que de nos jours et avec la mondialisation et la croissance des interdépendances, cette dernière ressemble beaucoup plus à un réseau complexe comme celui de la figure 1c, qui montre les principales fonctions qu'une chaîne logistique complexe puisse comprendre, ainsi que les relations entre les éléments de cette chaîne. Dans ce qui suit nous allons voir la structure globale d'une chaîne logistique.

1.2.2. Structure de la chaîne logistique

La chaîne logistique est généralement représentée graphiquement comme un réseau qui illustre la relation entre ses membres comme le montre la figure 5, qui présente la structure générale d'une chaîne logistique. Selon Lambert et Martha (, 2000), les membres représentent les entreprises focales et les entreprises de soutien. L'entreprise focale correspond à l'entreprise autour de laquelle se construit la chaîne logistique. C'est l'entreprise qui génère la plus grande valeur, mais dans certain cas, l'entreprise focale n'existe pas. Les entreprises de soutien sont toutes les entreprises productives ou de prestations de services ayant des liens directs ou indirects avec les entreprises focales.

Par rapport aux dimensions structurelles. La structure horizontale concerne le nombre d'échelons. Par exemple, dans la chaîne logistique simple, il existe trois échelons : l'entreprise focale, le fournisseur et le client. La structure verticale concerne le nombre d'entreprises par échelon. Dans certains cas il y'a plusieurs entreprises focales, ou à titre d'exemple une chaîne qui contient trois fournisseurs ultimes.

Selon (Lambert et Martha, 2000), la chaîne logistique peut comporter plusieurs types de liens (figure 5) : Les liens des processus gérés qui sont relatifs aux processus que l'entreprise focale juge importants d'intégrer et gérer. Par exemple les liens avec le client immédiat et le fournisseur immédiat. Les liens des processus contrôlés qui ne sont pas aussi importants que les liens des processus gérés, mais l'entreprise focale doit s'assurer qu'ils sont intégrés et gérés d'une manière approprié. Par exemple : quelques liens entre les clients d'échelon 1 avec les clients d'échelon 2. Aussi les Liens des processus non-gérés qui bien qu'ils font partie de la chaîne logistique de l'entreprise focale mais ces liens sont peu importants. Par exemple quelques liens entre les clients d'échelon N et ceux d'échelon N-1. Enfin les Liens des processus non-membre qui ne font pas partie de la chaîne logistique de l'entreprise focale, même si celle-ci est consciente de leurs impacts indirects. Par exemple les liens entre ses clients avec les concurrents.

Principalement une chaîne logistique comporte trois types de flux à savoir. Les flux physiques qui se déplacent dans un seul sens : à partir du fournisseur ultime (matière première) vers le client final (produit à consommer), le challenge ici est d'assurer que ces flux se déplacent sans arrêt (par exemple l'existence des stocks dans des points divers dans la chaîne), pour

assurer une bonne circulation de la trésorerie. Les flux financiers qui se déplacent au sens contraire des flux physiques, le client paye le fabricant, et le fabricant paye le fournisseur. Enfin les flux d'information (facture, bon de commande, etc.) qui sont très importants du fait de la nécessité de partager l'information entre les différents acteurs de la chaîne logistique. Ces flux sont dans les deux sens. Par exemple les bon de commande, le planning mensuel, les réclamations de la qualité, etc.

Pour une chaîne logistique efficace et efficiente, il est très important que les trois flux soient managés correctement et avec un minimum d'effort.

La figure 5 donne une présentation générale de la structure d'une chaîne logistique.

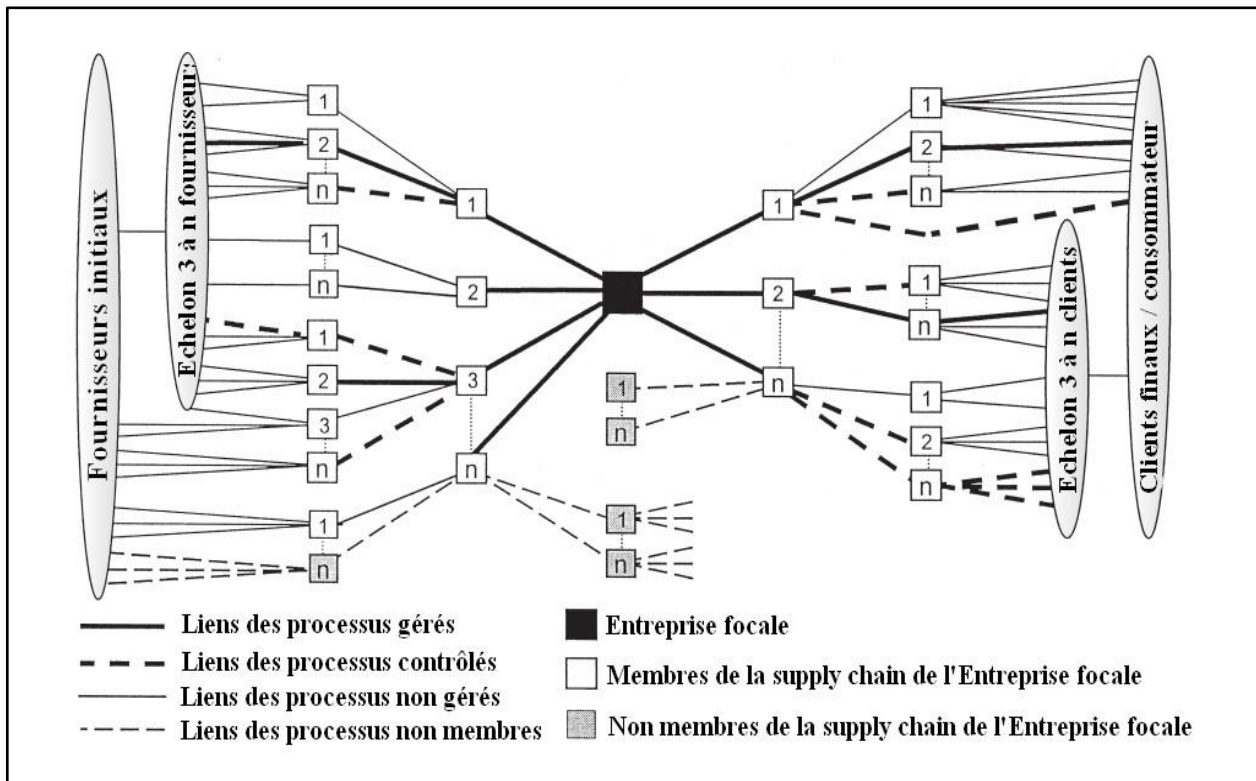


Figure 5 : Structure de la chaîne logistique (Lambert, 2009).

1.2.3. Management de la chaîne logistique

L'évolution continue des composantes de la chaîne logistique a fait que le terme 'management de la chaîne logistique' est difficile à définir et sa définition a eu beaucoup de changement avec le temps. Selon Snyder et Shen (2011), une des définitions les plus conventionnelles vient du CSCMP (Council of Supply Chain Management Professionals), qui la définit comme suit :

« SCM englobe la planification et la gestion de toutes les activités impliquées dans le sourcing et l'approvisionnement, la conversion, et toutes les activités de gestion logistique.

Surtout, elle comprend également la coordination et la collaboration avec les partenaires du canal, qui peuvent être les fournisseurs, les intermédiaires, les tiers fournisseurs de services et les clients.

Principalement, le SCM intègre la gestion de l'offre et de la demande au sein et entre les entreprises. Le SCM est une fonction d'intégration avec comme responsabilité principale relier des fonctions et des processus d'affaires importants au sein et entre les entreprises dans un modèle d'affaires cohérent et hautement performant. Il comprend toutes les activités de gestion logistique notées ci-dessus, ainsi que les opérations de fabrication, et il entraîne la coordination des processus et des activités avec et à travers le marketing, les ventes, la conception des produits, les finances et les technologies de l'information.» (CSCMP).

Après avoir vu les activités incluses dans le management de la chaîne logistique lors de la définition. Il est commode de penser sur les niveaux de décision et qui se présentent selon (Snyder et Shen. 2011) comme suit : Niveau Stratégique qui inclut les décisions qui ont des effets à long termes, typiquement des années ou des décennies. Comme exemples nous pouvons citer : La définition de l'offre de produits et services, le dimensionnement de l'outil industriel et logistique et la négociation de contrats et partenariat avec les fournisseurs et les clients. Niveau tactique qui concerne l'ensemble des décisions sur des horizons modérés de temps comme des mois. Ils peuvent être changés périodiquement mais avec une certaine difficulté. Les exemples incluent les révisions de vente, la planification, programmation et calcul des besoins et les règles d'approvisionnement et de gestion des stocks. Enfin le niveau opérationnel des décisions se produit sur des petites horizons de planification comme des jours ou des semaines au maximum. Les décisions opérationnelles doivent être exécutées mais ne peuvent pas être changées. Nous pouvons citer la gestion des commandes clients,

l'ordonnancement des opérations et gestion des aléas et l'exécution des commandes : production, distribution et facturation.

1.3. Dynamique et complexité du management de la chaîne logistique

1.3.1. Evolution du management de la chaîne logistique

Le terme 'chaîne logistique' est apparu dans les années 1980 pour désigner un ensemble d'activités coordonnées par un organisme pour gérer ses approvisionnements (Sharma, Garg et Agarwal. 2006). Le management de la chaîne logistique est né de la reconnaissance des managers que le fait de d'améliorer la façon que les activités de la chaîne logistique (production, transport ...) sont réalisées peut diminuer le coût (Snyder et Shen. 2011).

Depuis, les entreprises ont commencé à s'intéresser au SCM. Initialement, le SCM était relié à la gestion des stocks, ensuite il a été élargi pour inclure toutes les fonctions de l'entreprise au sein d'une chaîne logistique (Sharma et al. 2006). Le SCM avait comme priorité la réduction des coûts avec la résolution de problèmes d'ordre opérationnel comme la réduction des niveaux de stocks, le bénéfice des économies d'échelle. Ces améliorations étaient invisibles pour le consommateur.

Vers la fin de ce dernier siècle, la concurrence accrue des marchés mondiaux a fait que le SCM est devenu une source d'avantage concurrentiel plutôt que simplement orienté vers la réduction des coûts. Plusieurs entreprises confirment cette affirmation, nous pouvons citer l'exemple d'un des leaders mondiaux dans le secteur informatique, en l'occurrence 'Dell', qui arrive à délivrer des ordinateurs complètement personnalisés par rapport aux spécifications du consommateur en seulement quelques jours après la commande, et ce grâce à la configuration de sa chaîne logistique. Alors, Dell a changé le paradigme existant pour l'achat des ordinateurs, dans lequel les clients peuvent choisir seulement d'un nombre limité d'ordinateurs préconfigurés. Plusieurs autres exemples existent comme Walmart et Amazon. Dans ce qui suit nous allons voir les caractéristiques du passage du management dirigé coût vers une source d'avantage concurrentielle.

1.3.2. Optimisation de la chaîne logistique

Lors des 30 dernières années, le SCM est devenu l'un des domaines d'applications les plus convoité par la recherche opérationnelle. Celui-ci inclue des modèles mathématiques et informatiques qui permettent d'évaluer, d'analyser et d'optimiser la chaîne logistique dans le cadre de ce qui est appelé dans la littérature 'la théorie de la chaîne logistique' (*supply chain theory*).

Selon Snyder et Shen (2011), la théorie de la chaîne logistique est devenue non seulement un domaine d'application des outils de la RO mais aussi un domaine pour le développement des méthodologies et des outils propres à lui. Ses derniers et vu leurs performances sont eux-mêmes appliqués dans d'autres secteurs comme la finance, les services, etc.

La théorie de la chaîne logistique englobe des modèles de grande variété, à base de programmation mathématiques (linéaire, non linéaire, etc.), théorie de dualité, techniques d'optimisation (Lagrange, programmation dynamique, etc.), heuristiques, théorie des jeux, etc. Le but des modèles est généralement de trouver une solution avec un coût minimum ou un profit maximum sous des contraintes définies. Par exemple, choisir la location des entrepôts de façon à minimiser les coûts de transports, sous la contrainte que chaque client doit être servi. Le modèle idéal de la chaîne logistique optimise de façon globale chaque aspect de la chaîne, mais cependant un tel modèle présente une multitude de difficultés liées à sa complexité.

Généralement les modèles classiques se concentrent sur une optimisation locale d'un élément de la chaîne ou une intégration de plusieurs éléments mais avec moins de détails qui éloignent le modèle de plus en plus de la réalité. Ces mécanismes de fonctionnements traditionnels sont considérés comme étant centralisé dans le sens où les variables de décisions sont sous le control d'un seul décideur, comme l'optimisation des stocks, la localisation, etc. Mais cependant et par rapport à la globalité de la chaîne logistique, ils sont considérés comme étant décentralisés, puisque l'ensemble des acteurs de la chaîne ont des buts plus ou moins conflictuels. Dans ce qui suit, nous considérons la chaîne dans sa globalité et donc ces mécanismes sont considérés comme étant décentralisés.

Avec la forte concurrence, l'ouverture des marchés ainsi que la mondialisation, la chaîne logistique est devenue de plus en plus complexe et dynamique. Devant ces facteurs, assurer la

disponibilité des produits toute au long de la chaîne et à moindre coût devient très difficile pour les entreprises. Et là, les mécanismes traditionnels ont prouvé leurs limites :

- La sous-optimalité de la solution : dans les mécanismes classique chaque joueur agit d'une façon égoïste (dans le sens où il agit dans son intérêt et non pas d'une façon inapproprié) le profit totale de la chaîne est généralement inférieure que dans le cas où les acteurs peuvent d'une façon ou d'une autre aligner leurs actions (Höhn 2003)
- Devant ces facteurs les entreprises ne peuvent plus opérer de façon isolée des autres acteurs de la chaîne (Lumus et Vokurka 1999). La dynamique du marché et la forte concurrence assurent la disponibilité d'autres alternatives par rapport au produit et même des produits substituables et devant la constante évolution des habitudes d'achat des clients cela constitue un grand risque à l'entreprise.
- Plusieurs problèmes sont apparues, avec l'un des plus marquant l'effet de coup de fouet (Bullwhip effect), qui correspond à l'amplification de la variabilité de la demande dans la chaîne logistique d'aval en amont (Lee et al. 2004). Le premier qui a décrit ce phénomène dans le cadre des travaux sur la dynamique des systèmes est bien (Forrester 1958) à travers le célèbre jeu 'Beer game' au MIT.

1.3.3. Effet coup de fouet

Historiquement l'effet de coup (*Bullwhip Effect*) de fouet a été accepté comme un phénomène normal, ainsi qu'un résultat inévitable qui caractérise les systèmes de production et de distribution. Selon (Lee et al. 2004), Forrester (1958) a identifié que l'action, la réaction et l'interaction des flux entre les parties égoïstes dans la chaîne logistique mènent à l'oscillation, l'amplification et le retard dans les flux à travers la chaîne. Du point de vu de Sterman (1989), ce phénomène est relatif aux décideurs (managers) qui font des erreurs dans la perception de la dynamique de l'environnement dans lequel ils opèrent (comportement irrationnel).

Pour une longue période l'industrie cherche pourquoi un tel phénomène existe. Dans les débuts des années 90, des cadres à Procter and Gamble (P&G) ont remarqué une tendance particulière pour les commandes de Pampers (une marque de couche des bébés). La demande des couches au niveau des clients était très stable du fait que les bébés les utilisent à un taux constant. Mais

P&G a remarqué que les commandes des détaillants pour les distributeurs étaient un peu variables avec le temps. Les commandes des distributeurs à P&G étaient plus variables et ceux de P&G à leurs fournisseurs étaient encore plus variables (figure). Cet exemple est l'un des exemples les plus célèbres de l'effet de coup de fouet, il a été observé dans plusieurs autres industries comme Hewlett-Packard (HP), etc.

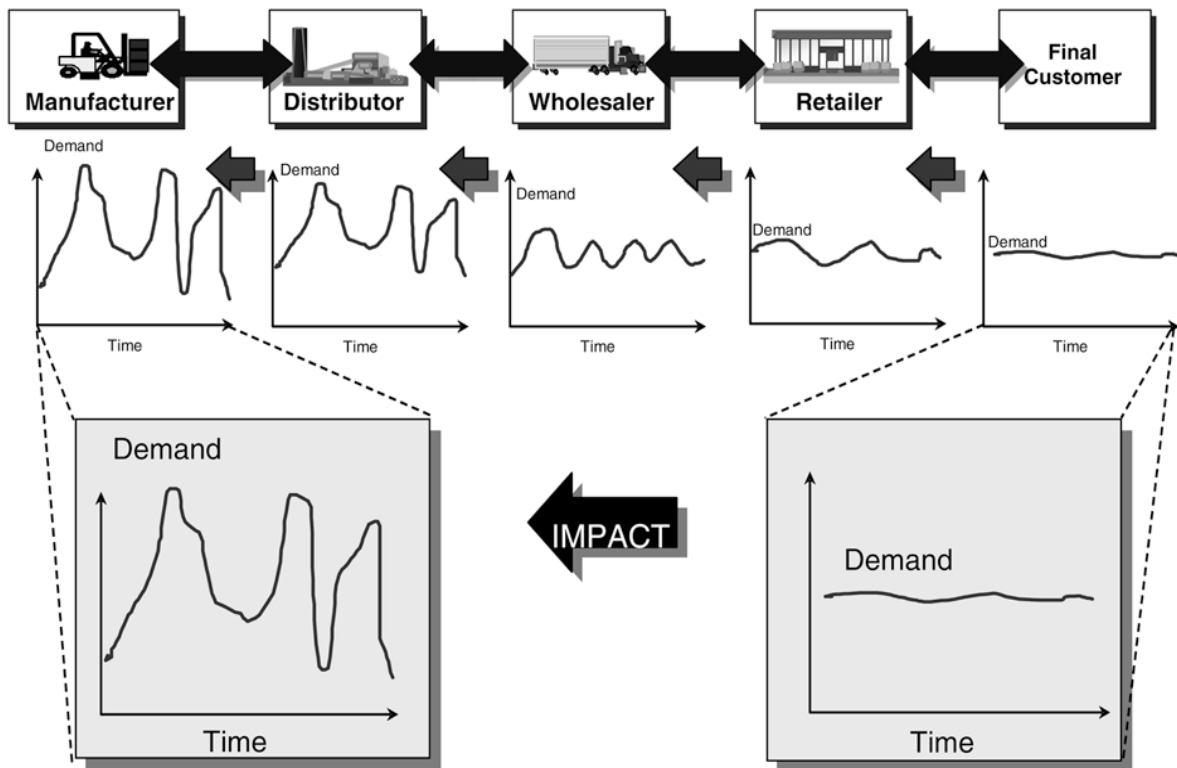


Figure 6 : L'effet de coup de fouet (Pino et al. 2003)

Lee et al (1997a, b), ont identifié quatre causes principales pour l'effet de coup de fouet :

Les prévisions de la demande :

Lee et al (1997a, b) a identifié que l'effet de coup de fouet est une conséquence naturelle des méthodes statistiques de prévision de la demande et qui dépendent de la multitude des niveaux de la chaîne logistique ainsi que le temps de passation de commande. Chaque fois que la demande est observée l'estimation est actualisée. Si la demande précédente était élevée la nouvelle estimation sera élevée et donc le niveau de stock augmente. Les commandes seront encore plus exagérées que les demandes. Ce phénomène peut être amplifié avec le temps. Hanssens (1998) est l'un des premiers à avoir fait une étude empirique sur le lien entre les méthodes de prévisions de la demande et l'effet de coup de fouet en utilisant les données sur les consommateurs de la haute technologie (*high-technology*).

Plusieurs travaux ont étudié le lien entre les prévisions de la demande et l'effet de coup de fouet, nous pouvons citer Graves (1999), Chen et al (1999), etc.

La taille des lots :

Un des paramètres clés qui mène à l'effet de coup de fouet est bien la fréquence des commandes et la taille des lots. Par rapport à la fréquence des commandes. La variabilité s'explique avec le fait qu'il y a un coût pour le lancement des commandes qui est différent d'un acteur à un autre, donc les acteurs lancent le minimum possible de commandes et donc avec une fréquence variable.

De plus que les commandes peuvent être élevées dans des périodes comme ils peuvent être basses dans d'autres, comme les acteurs peuvent utiliser leurs stocks cumulés. Sur ce qui concerne le cycle de commande, et puisque plusieurs entreprises utilisent le MRP (Materials requirements planning) pour déterminer le lancement des commandes. Les fournisseurs auront des demandes larges pendant les jours où les systèmes MRPs des clients lancent des commandes, et des petites demandes par ailleurs. Parfois ce phénomène est connu sous le nom du bâton de hockey (*hockey stick*). Tous ses facteurs contribuent énormément dans l'effet de coup de fouet.

Aussi et par rapport à la taille des lots, il est connu que la commande des acteurs de la chaîne logistique se fait avec des tailles variables d'un acteur à un autre. La multitude est liée à la grande variation des quantités transférer dans une même chaîne, pour des raisons logistiques (stockage, transport, etc.) la taille des lots elle est différente. Ceci peut impliquer une grande variabilité de la demande.

La variation du prix :

Généralement dans les chaînes logistiques où les matières brutes sont des produits (carburant, plastique, etc.), les prix changent tout le temps et les firmes tendent à stocker lorsque les prix sont bas et commander moins lorsqu'ils sont hauts. Aussi la littérature du marketing utilise le terme promotions (*trade promotion*) pour décrire le changement à court terme des prix des produits et qui a connu une grande croissance entre 1980-1990 (de 35% dans les débuts des années 80 à 53% vers la fin des années 90). Ces phénomènes mènent aux fluctuations des prix et donc la demande, et ils sont à la tête des causes de l'effet de coup de fouet.

L'anticipation des situations de pénurie :

Quand les distributeurs n'ont pas suffisamment de stock pour satisfaire la commande des détaillants, ils le partagent proportionnellement à la taille de la commande. Mais si les détaillants anticipent les situations de pénurie, ils lancent des commandes avec des inflations. Donc la variation des commandes des détaillants est plus grande que celle des clients. Cachon et Larivierre (1999) ont montré comment que le choix des mécanismes d'allocation, influence la performance de la chaîne logistique et que les proportions d'allocations linéaires et proportionnelles contribuent à la production de l'effet de coup de fouet.

Sterman a décrit les origines de l'instabilité de la chaîne logistique du point de vu de la complexité des systèmes. L'instabilité est souvent décrite par l'effet de coup de fouet. La dynamique des réseaux de la chaîne logistique croit avec la complexité de la structure qui inclue les éléments opérationnels et comportementaux. Dans ce sens, plusieurs recherches proposent deux catégories d'explication de l'effet de coup de fouet. La première se concentre sur les causes opérationnelles comme les quatre causes citées auparavant. La deuxième concerne les causes comportementales de l'instabilité qui inclut, la rationalité limitée des décideurs (Simon). Aussi plusieurs études ont montré que les gens souvent ont une visibilité réduite ainsi qu'ils ne peuvent pas prendre en compte tous l'ensemble des feedbacks et des retards (*time delays*). Selon Sterman, les modèles mentaux et les lois de décision des décideurs sont loin de l'optimalité. Alors ces derniers ont une grande influence sur l'amplification de la demande. Même sur la persistance de ce phénomène avec l'apprentissage lourd (*Slow-Learning*).

Le bulwhip effect engendre des conséquences fatales dans l'entreprise. D'abord il cause une augmentation significative dans les coûts de la chaîne logistique (Supply chain cost) à savoir les coûts de stockage par les niveaux de stock élevés, les coûts de production excessive et les coûts de transport par des capacités encore plus grandes et avec un surplus. Aussi il engendre la dégradation de la performance globale de la chaîne logistique avec les différentes variations et les fluctuations toute au long de la chaîne et le manque de coordination.

1.3.4. Coordination et collaboration de la chaîne logistique

Pour faire face aux problèmes présentés auparavant, en l'occurrence celui de l'effet de coup de fouet, plusieurs recherches se sont intéressées à la coordination. Les entreprises ont réalisé qu'à chaque fois qu'une entreprise coordonne avec une autre entreprise qui effectue la prochaine phase de la chaîne logistique, les deux parties sont gagnantes. Ceci est dû aux avantages des relations collaboratives intra et inter-organisationnelles (Lummus & Vokurka, 1999). Ce qui a poussé les entreprises à aller au-delà de leurs propres frontières et de s'insérer au SCM. D'après Zouaghi (2013), « nous pouvons distinguer deux principaux sous domaines, à savoir la logistique et la gestion des opérations dans son aspect opérationnel, tactique et stratégique, principalement intra-organisationnel, et la logistique globale et la SCM qui intègre les partenaires dans une logique systémique. Le premier sous-domaine est arrivé à maturation entre les années 1990 et 2000, alors que le deuxième est en pleine expansion et maturation » (Zouaghi 2013, p88).

Dans la littérature et dans le cadre inter-organisationnel (logistique globale) du SCM, plusieurs approches collaboratives et intégratives ont été proposées pour la coordination. Une des premières initiatives relève des années 90 aux Etats unies dans les secteurs du textile à travers la stratégie de réponse rapide (*Quick Response*) et plus tard l'ECR (*Efficient Consumer Response*) dans celui de l'agro-alimentaire.

En raison de la forte concurrence, les leaders du secteur de textile aux USA ont formé l'association « Crafted With Pride » en 1984 (Kurt Salmon Associates, Inc., 1993) qui a été chargée de conduire les analyses de la chaîne logistique. Les résultats de l'étude ont montré que beaucoup de pertes étaient liées à la chaîne logistique comme le temps de délivrance (66 semaines dont 40 semaines passées aux entrepôts ou transit). Comme solution ils ont développées, la stratégie de réponse rapide « the quick response strategy » (QR). Cette stratégie consiste en un partenariat entre les détaillants et les fournisseurs qui travaillent ensemble en partageant l'information pour répondre rapidement aux besoins des consommateurs.

En 1992 les leaders du secteur de l'agro-alimentaire ont créé un groupe de travail appelé 'The Efficient Consumer Response (ECR) working group'. Ce groupe a été chargé d'analyser la chaîne logistique de ce secteur pour identifier les opportunités d'augmentation de la compétitivité (Kurt Salmon Associates, Inc, 1993) d'après (Lummus & Vokurka, 1999).

Une étude a été menée par ‘Kurt Salmon Associates’ et a montré qu’en accélérant le flux d’informations dans la chaîne logistique, l’ECR permet aux distributeurs et fournisseurs d’anticiper la demande future d’une manière beaucoup plus précise que ce que le système permet actuellement (Lummus & Vokurka, 1999). Cette étude avait prévu une réduction globale de 37 % des stocks de la chaîne et des réductions globales de coûts de l’ordre de 24 à 30 milliards de dollars.

Le développement de l’ECR a donné naissance au concept de réapprovisionnement continu (VMI, CPFR). Ce concept permet d’abandonner la démarche des flux poussés basée sur la possession de stocks pour une démarche de flux tirés basée sur la demande des consommateurs. Les informations sur la demande sont transmises aux fabricants et permettant ainsi de garder le détaillant réapprovisionné en juste à temps. Le VMI (Vendor Managed Inventory) relève de Wal-Mart. Le leader de la distribution a pris l’initiative de travailler directement avec les producteurs clés. Ces producteurs sont responsables de la gestion du stock de Wal-Mart de leurs produits. En contrepartie Wal-Mart a estimé presque 100% de satisfaction de commande pour ces produits. Plusieurs autres exemples dans la littérature existent dans le cadre des approches collaboratives comme Hewlett-Packard (HP) avec le système DRP (*Distribution Requirements Planning*), etc. Les initiatives à la collaboration de la chaîne logistique sont devenues de plus en plus dominantes. Selon le SCC (Supply Chain Council, 1997), en une année les membres sont passés de 73 à plus de 300 dans le but de construire une relation assez forte entre les acteurs des chaînes logistiques des plus grands entreprises au monde.

Depuis 1995, et dans le même contexte, la coordination de la chaîne logistique s’est appuyée sur ce qui est appelé dans la littérature ‘les contrats de la chaîne logistique’ (*supply chain contracts*) pour couvrir les autres approches de collaborations à travers l’optimalité basée sur la théorie des jeux qui sera décrite dans ce qui suit.

A nos jours il y a un grand développement dans les systèmes d’aide à la décision. Notamment les systèmes collaboratifs de prise de décision (*Collaborative Decision Support Systems CDSSs*) qui sont caractérisés selon Zaraté (2013) par le passage d’un contexte avec un seul décideur à un autre avec une multitude de décideurs. Ces derniers agissent de façon séparée mais indépendamment les uns des autres, avec un besoin de coopération entre eux. Plusieurs outils étaient développés dans ce sens avec des modules développés récemment comme :

L'outil de communication interpersonnel (*Interpersonnel communication tool*) pour la communication entre les acteurs. L'outil de management des tâches (*Task management tool*) qui prend en charge la division des tâches en sous tâches ainsi que l'assignement des rôles. Etc.

1.4. Les contrats de la chaîne logistique

Après que nous avons citées les principaux travaux dans les approches de collaborations et d'intégration de la chaîne logistique. Nous allons dans ce qui suit discuté un des outils importants qui sont les contrats de la chaîne logistique.

1.4.1. Introduction aux contrats de la chaîne logistique.

La chaîne logistique englobe le management des flux physiques, monétaires et informatifs dans un réseau d'acteurs (Anupindi and Bassok 1999). L'échange des différents flux entre l'ensemble des acteurs peut être vu comme étant une routine de transaction entre chaque deux pair d'acteurs.

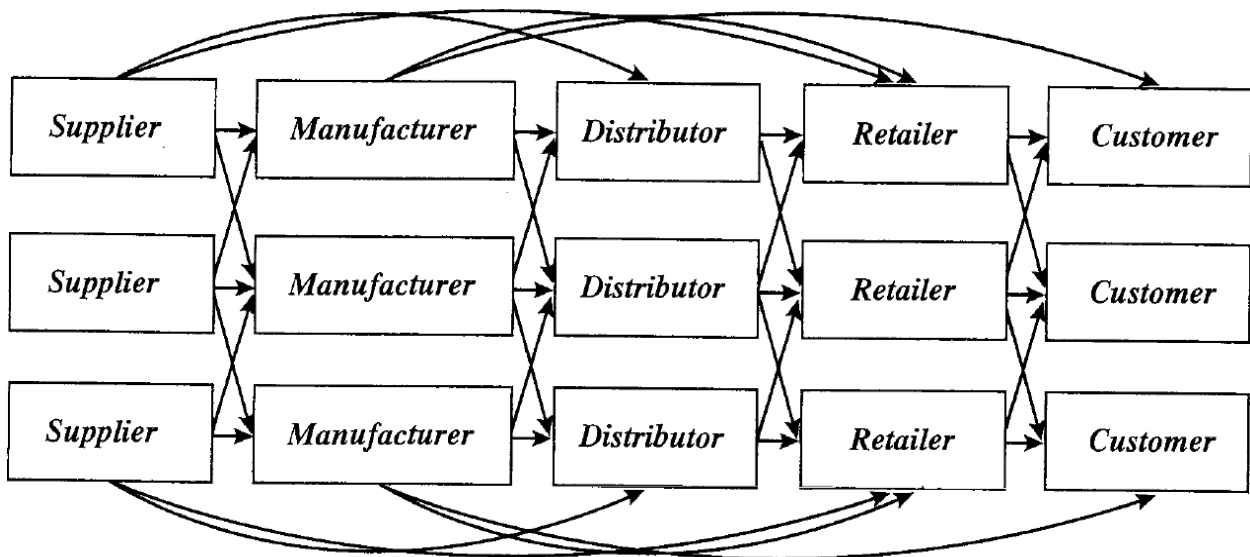


Figure 7 : Présentation de la chaîne logistique (Chopra et Meindl, 2012)

L'idée essentielle des contrats de la chaîne logistique est que les joueurs se mettent d'accord sur un contrat qui spécifie un paiement dit de transfert (T) transféré d'un acteur à un autre :

$$\pi^*_{Acteur\#1} \rightarrow \pi_{Acteur\#1} - T$$

$$\pi^*_{Acteur\#2} \rightarrow \pi_{Acteur\#2} + T$$

Avec ces contrats et bien que les joueurs sont égoïste, le profit de la chaîne est maximisé. Donc le rôle des contrats est non pas de forcer les acteurs à choisir une stratégie pour laquelle un acteur gagnera plus de profit au détriment d'un autre, mais de trouver une nouvelle solution qui est meilleur pour les deux acteurs à la fois par rapport à leur choix précédent sans contrat.

La littérature de la coordination de la chaîne logistique relève en partie de la théorie des jeux. Ceci est bien expliqué puisque cette dernière est définie comme étant l'étude mathématique des interactions entre des agents indépendants et égoïstes (Leyton-Brown et Shoham, 2010), qui correspond exactement à la relation entre les acteurs de la chaîne logistique qui sont principalement indépendant mais aussi égoïste dans le sens où il y a toujours des objectifs conflictuels entre eux ainsi qu'une séparation organisationnelle malgré le lien de la coordination.

Nous allons présenter les contrats à travers la théorie des jeux pour donner une vision assez claire et significative sur leurs apports. Dans ce sens nous allons d'abord introduire quelques notions clés concernant la théorie des jeux.

La base de ce domaine est bien la théorie d'utilité (Von Neumann and Morgenstern, 1944) qui modélise l'intérêt des acteurs. L'utilité quantifie le degré des préférences, leurs changements par rapports à l'incertitude et aussi la rationalité de la décision théorique. Dans le cas du SCM, un jeu consiste à un ensemble de joueurs (acteurs). Chaque joueur peut choisir entre un ensemble de stratégies possibles (choix), le choix correspond à un résultat (outcome) et pour chaque choix il aura un profit donné (payoff).

Les bénéfices de la collaboration entre les acteurs de la chaîne logistique en générale sont très bien illustrés par le dilemme du prisonnier (Christopher, 2011). Ce dernier est un scénario dans lequel deux personnes (A et B) ont été arrêtées et soupçonnées de vol de banque. Ces deux prisonniers sont séparés dans deux cellules différentes et ne peuvent pas communiquer (Palencia, 2013). Les détails du jeu sont donnés dans le tableau (dilemme de prisonnier avec Equilibre de Nash différent de L'optimum de Pareto).

Table 1 : Dilemme de prisonnier sans contrat

| | | Joueur B | |
|----------|---|----------|---------|
| | | 1 | 2 |
| Joueur A | 1 | (1,1) | (-4,2) |
| | 2 | (2,-4) | (-2,-2) |

Les colonnes présentes les stratégies du joueur B, les lignes correspondent aux stratégies du joueur A, donc si par exemple le joueur A choisi la stratégie 1 et le joueur B choisit la stratégie 2, le profit est -4 pour A et 2 pour B.

Un résultat est un optimum de Pareto (PO) s'il n'existe aucun résultat pour lequel l'ensemble des joueurs auront un profit plus élevé ou bien lorsqu'aucun des joueurs ne peut obtenir un profit plus élevé sans que le profit d'un des autres diminue. Dans notre exemple il correspond au choix de la stratégie 1 par les deux acteurs.

L'équilibre de Nash (NE) est un résultat pour lequel aucun joueur ne peut changer de stratégie de façon unilatérale et améliorer son profit. Il y a un seul équilibre de Nash dans le jeu : chaque joueur choisit la stratégie 2.

Si les joueurs agissent de façon égoïste, le jeu va converger vers l'équilibre de Nash. Dans le dilemme de prisonnier l'équilibre de Nash diffère de l'optimum de Pareto. Donc les joueurs vont toujours se retrouver dans une solution indésirable (NE) malgré la présence d'une solution meilleure (PO).

Maintenant supposant que les joueurs vont rentrer dans le contrat. Les profits résultants sont exprimés dans le tableau (Equilibre de Nash = Optimum de Pareto -> après le contrat).

Table 2 : Dilemme de prisonnier avec contrat

| | | Joueur B | |
|----------|---|----------|---------|
| | | 1 | 2 |
| Joueur A | 1 | (1,1) | (-1,-1) |
| | 2 | (-1,-1) | (-2,-2) |

Dans ce cas l'équilibre de Nash correspond à l'optimum de Pareto (choix de la stratégie 1 pour les deux joueurs). Avec l'introduction du contrat, les joueurs choisissent la meilleure solution bien qu'ils sont égoïstes. Nous pouvons noter que le contrat ne force pas les acteurs à choisir une solution mais simplement il restructure les profits de façon que les joueurs eux même veulent choisir.

Le but de la coordination de la chaîne logistique à travers les contrats est de changer la structure des profits des acteurs de la chaîne avec la fonction de transfert pour que l'équilibre de Nash devient optimum de Pareto.

Comme un point de départ pour l'analyse des contrats de la chaîne logistique nous allons utiliser un des modèles standards les plus étudiés dans ce sens qui correspond à un produit et une période avec une demande stochastique qui est le 'Newsvendor model' (Cachon 2003).

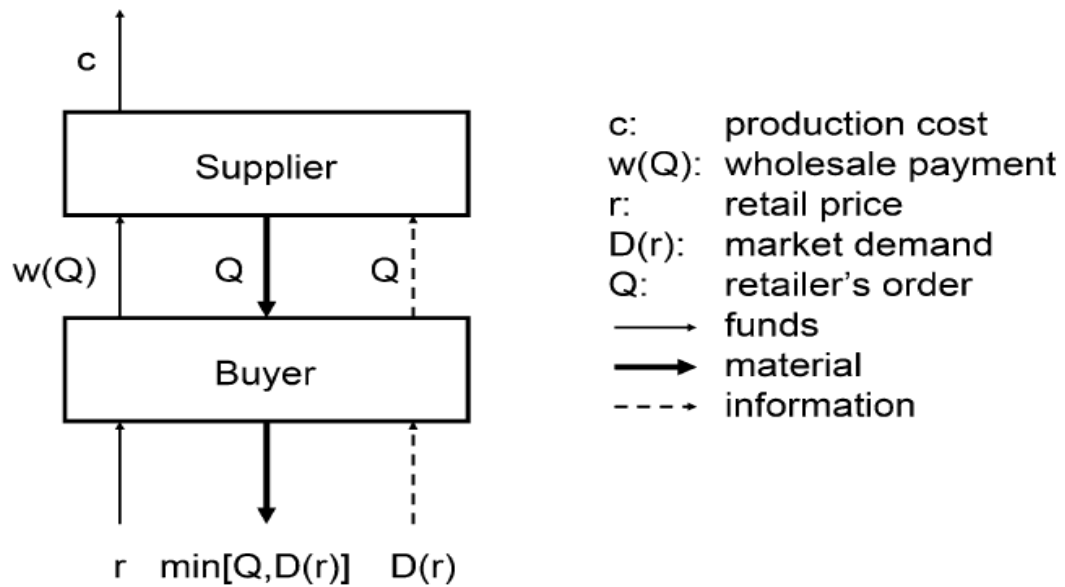


Figure 8 : Le modèle basic de la chaîne logistique à une période (Höhn 2010).

La notation des contrats est définie dans le tableau :

Table 3 : Notation des contrats de la chaîne logistique

| Variable | Signification |
|----------|---|
| D | La demande durant la période. |
| μ | La moyenne de la demande |
| F | La fonction de distribution de la demande suivant une loi normale |
| F | La fonction de densité |
| R | Le prix de vente du distributeur |
| c_s | Le coût de production du fournisseur |
| c_r | Le coût de production du distributeur |
| p_s | Pénalité de la demande non satisfaite par le fournisseur |
| p_r | Pénalité de la demande non satisfaite par le distributeur |
| V | Valeur des dépenses sauvées |
| Q | La quantité commandée par le distributeur |

La séquence des évènements est la suivante :

1. Le fournisseur choisit les paramètres du contrat (chaque contrat à ces propres paramètres).
2. Si le distributeur accepte le contrat, il choisit la quantité à commander et la lance au prêt du fournisseur.
3. Le fournisseur produit et délivre la demande au distributeur.
4. Finalement le transfert de paiement se fait entre les acteurs selon les paramètres du contrat.

Nous commençant par la définition des profits des deux acteurs :

- Soit $S(Q)$ l'estimation des ventes en fonction de Q avec : $S(Q) = E [\min \{Q, D\}]$.
- $I(Q)$ correspond à l'estimation du niveau de stock avec : $I(Q) = E [(Q-D)^+] = Q - S(Q)$.
- Et $L(Q)$ qui représente les ventes non réalisées avec : $L(Q) = E [(D-Q)^+] = \mu - S(Q)$.
- Posons : $c = c_r + c_s$ et $p = p_r + p_s$.
- $F(Q)$: Fonction de distribution de la demande.

Le profit estimé pour le distributeur est :

$$\pi_r(Q) = r \times S(Q) + v \times I(Q) - p_r \times L(Q) - c_r \times Q - T$$

Le profit estimé pour le fournisseur est :

$$\pi_s(Q) = -p_s \times L(Q) - c_s \times Q + T$$

Le profit estimé pour la chaîne logistique globale est :

$$\pi(Q) = \pi_r(Q) + \pi_s(Q) = (r - v + p) \times S(Q) - (c - v) \times Q - p \times \mu$$

Nous allons chercher la quantité Q^0 qui maximise le profit total $\pi(Q)$:

$$\pi'(Q^0) = 0 \xLeftrightarrow[\text{equivalent}] F(Q) = \frac{r - c + p}{r - v + p}$$

$$\text{Et } \pi''(Q^0) < 0$$

La chaîne logistique est dite coordonnée si et seulement si $Q_s^* = Q_r^* = Q^0$. Un contrat assure la coordination d'une chaîne logistique s'il existe une fonction de transfert avec des paramètres qui assure la relation précédente.

Le paiement de transfert (les paramètres) dépend des types de contrats que nous allons discuter par la suite.

1.4.2. Présentation des différents contrats

Après avoir vu le principe général des contrats de la chaîne logistique. Nous allons voir les paramètres de chacun de ces contrats. Pour ce faire nous allons se référer à (Cachon 2003), avec la notation introduite auparavant.

Le contrat Wholesale price

Le contrat le plus simple est Wholesale price, dans lequel le distributeur paye le fournisseur un coût w par unité. Les paramètres sont identiques à la notation précédente, sauf que maintenant le coût de vente est une variable de décision du fournisseur.

Le coût de transfert de ce contrat est le suivant :

$$T_w(Q, w) = w \times Q$$

Selon l'analyse de Lariviere et Porteus (2001), la valeur de w pour que la chaîne logistique est coordonnée est la suivante :

$$w = c_s - \frac{c - v}{r - v + p} \times p_s$$

Dans ce cas le fournisseur aura un profit non-positif, puisque $w \leq c_s$. Pour que $w > c_s$ il faut que $Q_r^* < Q^0$.

Donc ce contrat est considéré comme étant non-coordonateur puisque il n'y a aucune valeur de w pour laquelle $Q_s^* = Q_r^* = Q^0$ et les deux joueurs auront un profit positif. Cependant les autres contrats que nous allons discuter sont considérés comme étant coordonnateurs.

Le contrat Buyback

Dans le Buyback (Pasternack 1985), le distributeur achète le produit à w /unité et pour chaque perte (pénurie de stock ou excès non vendu) le fournisseur le rembourse à b /unité. Notons que pour ce type de contrat la fonction de transfert est la suivante :

$$T_w(Q, w, b) = w \times Q - b \times I(Q) = b \times S(Q) + (w - b) \times Q$$

Selon Pasternack (1985), nous assumons que $b \leq w + c_s - v$.

La valeur de w qui assure la coordination est la suivante :

$$w(b) = b + c_s - (c - v) \times \frac{b + p_s}{r - v + p}$$

L'utilité du fournisseur est croissante tandis que l'utilité du distributeur est décroissante en fonction de paramètre b (figure 9). Un intervalle $[w_1(b_1); w_2(b_2)] = I \cup J$ est y lié au-delà duquel les deux acteurs auront des profits négatifs.

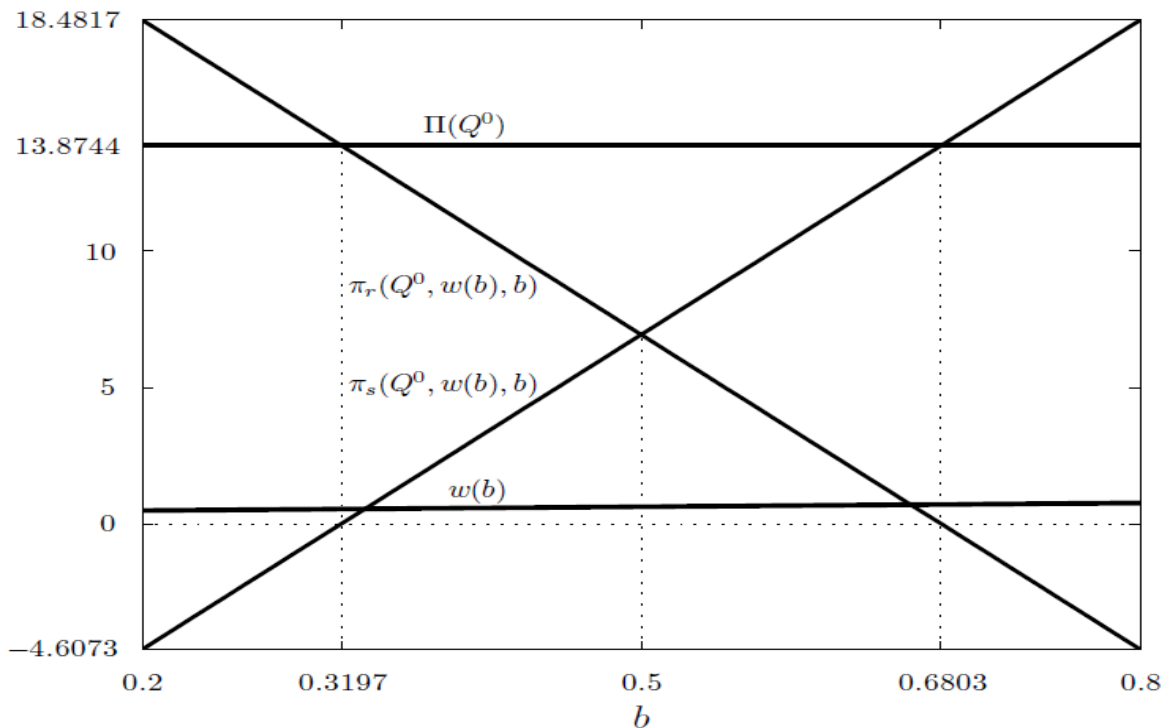


Figure 9: le profit en fonction du crédit (Snyder et Shen 2011, p-293)

Le contrat Revenue Sharing

Dans le contrat revenue sharing (Cachon et Lariviere 2005), le fournisseur charge le distributeur un prix de w /unité et le distributeur donne au fournisseur un pourcentage de son revenu. Tout le revenu est partagé (revenu des ventes et valeur sauvée (v)). Posons ϕ la fraction du revenu que le distributeur garde et $1 - \phi$ la fraction qu'il donne au fournisseur.

Le paiement de transfert est le suivant :

$$T_r(Q, w, \phi) = wQ + (1 - \phi) \times r \times S(Q) + (1 - \phi) \times v \times I(Q)$$

La valeur du paramètre w qui assure la coordination pour une valeur donnée de ϕ avec $0 \leq \phi \leq 1$ est la suivante : $w(\phi) = -c_r + \phi \times v + (c - v) \times \frac{\phi(r-v)+p_r}{r-v+p}$

Cachon et Lariviere (2005) a montré que le buyback est équivalent au revenue sharing si le prix de vente est fixé. Puisque le distributeur dans le revenue sharing paye $w_r + (1 - \phi) \times v$ par unité achetée et $(1 - \phi)(r - v)$ par unité vendue. Donc et par analogie avec le buyback :

$$w_b = w_r + (1 - \phi) \times v$$

$$b = (1 - \phi)(r - v)$$

Le contrat Quantity Flexibility

Le contrat quantity flexibility (Tsay 1999) est similaire au buyback dans le sens où le distributeur paye w au fournisseur pour chaque unité achetée et le fournisseur le rembourse pour les unités non vendues. La différence est que, dans le buyback le fournisseur rembourse partiellement chaque unité non vendue. Alors que dans le quantity flexibility il rembourse le totale d'une portion des unités non vendues.

Ce contrat a deux paramètres w et δ . Le distributeur paye le fournisseur w /unité, et le fournisseur paye $(w + c_r - v) \times \min\{I, \delta Q\}$ au distributeur.

La fonction de transfert est la suivante :

$$T_q(Q, w, \delta) = w \times Q - (w + c_r - v) \left[\int_{(1-\delta)Q}^Q F(d) dd \right]$$

Le paramètre w qui assure la coordination est le suivant :

$$w(\delta) = \frac{(r - v + p_r) \bar{F}(Q^0)}{\bar{F}(Q^0) + (1 - \delta) F((1 - \delta)Q^0)} - c_r + v$$

Après avoir présenté les différents contrats, nous allons faire une comparaison entre eux.

Pour évaluer les différents contrats, les critères suivants sont de grande importance (Cachon 2003) :

- La coordination de la chaîne logistique.
- Le profit de la chaîne logistique.
- Les coûts administratifs : Généralement les coûts sont liés au type et mesure des flux informationnel et physique.

Le modèle Newsvendor est utilisé pour faire une comparaison entre les contrats (Höhn 2010). En tenant compte des coûts administratifs, les contrats Wholesale price et Quantity discount sont à des coûts minimaux et égaux, puisque ils ne demandent qu'une seule transaction. Les autres sont plus coûteux, par le fait qu'ils exigent des flux matériel et informationnel supplémentaire. Mais cependant et par rapport aux deux premiers critères, Wholesale price et quantity discount, demeurent moins performantes que le deuxième ensemble.

De plus et par rapport au deuxième ensemble, il faut noter que le buyback et le Quantity discount, et dans le cas où ils sont bien implémentés, peuvent améliorer la performance au maximum. Pour cela ces derniers ont trouvé plusieurs applications dans l'industrie (Cachon 2003).

Maintenant et entre ces deux contrats, c'est difficile de faire la comparaison vu que chacun des deux présente des avantages dans des situations par rapport à l'autre. Mais reste à dire que l'avantage principal que peut marquer le buyback est que ces paramètres ne dépendent

pas de la distribution de la demande contrairement au Quantity discount. Puisque le fournisseur n'a pas besoin de connaître la distribution de la demande pour coordonner (Höhn 2010).

1.4.3. Problèmes liés aux contrats et alternatives de résolution

Les contrats de la chaîne logistique et bien qu'ils assurent la coordination de la chaîne logistique mais plusieurs travaux mettent l'accent sur le fait qu'ils présentent plusieurs problèmes. Un des plus importants est lié à l'information. Malgré la présence des technologies de l'information qui assure la disponibilité l'analyse et la transaction de l'information à nos jours au sein de l'entreprise, avec leurs deux grandes classes (Shapiro 2001) :

- les technologies de l'information analytiques impliquant l'implémentation et l'application de plusieurs modèles mathématiques (comme les statistiques, 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 DSS (Décision support system), ERP (Entreprise resource planning), ...).

Mais plusieurs limites sont présentes : Ils ne modélisent pas les interactions entre les différentes entités sachant qu'elles sont primordiales dans un tel système surtout par rapport à la dynamique. Aussi ils ne reflètent pas la réalité. Puisque à la base les approches de collaborations nécessitent le partage des informations privées alors que dans la réalité les informations sont asymétriques et implicites. Encore, il faut dire que les coûts liés aux technologies d'informations sont très élevés et cela peut poser problème lors de la collaboration entre les acteurs.

Un autre problème est lié au comportement des acteurs. La critique majeure est adressée à l'hypothèse posée lors des contrats sur le fait que les acteurs sont rationnels et honnêtes, dans le sens où il partage toujours des données fiables, alors que rien n'assure que les acteurs sont malhonnêtes. De plus que le comportement n'est pas représenté de façon explicite, notamment l'autonomie des acteurs et la proactivité globalement à cause de la centralisation.

A nos jours, il y a une forte émergence des techniques distribuées de résolution de problèmes. Ces derniers présentent une alternative excellente pour faire face aux limites citées auparavant,

puisque non seulement ils simulent le comportement collaboratif des entités, l'interaction et l'organisation (Parunak. 1996), mais aussi ils sont bien adaptés aux systèmes complexe et dynamique (Dodd et al. 2001).

Plus particulièrement les systèmes multi agents se présentent comme l'un des outils les plus favorables entre les techniques de résolution distribuées des problèmes.

1.5. Conclusion

Dans ce chapitre, nous avons vu que le management classique de la chaîne logistique à travers l'optimisation locale n'est plus suffisant pour affronter la dynamique, la forte concurrence et la mondialisation. C'est pour cela que les entreprises s'ouvrent peu à peu pour intégrer ses activités à celles de ses fournisseurs, de ses clients et de ses partenaires. Ceci à travers les approches de collaboration et d'intégration. Mais cependant nous avons montré que ces dernières présentent plusieurs problèmes. Pour répondre à ces problèmes nous avons proposé comme solution les systèmes multi agent que nous allons les décrire dans le chapitre suivant pour ensuite justifier leur choix.

Chapitre 2 : Les systèmes multi-agents

2.1. Introduction

Actuellement l'automatisation de l'intelligence est devenue nécessaire devant la complexité des systèmes. Une des solutions à ce problème est l'intelligence artificiel qui d'après (Bellman, 1978) c'est l'automatisation des activités que nous associons à la pensée humaine.

Cette discipline était accentuée sur la notion d'agent rationnel, en se focalisant sur une seule entité pour la résolution des problèmes. Vers la fin des années 1970 s'est devenue de plus en plus distribuée à travers la coordination d'un ensemble d'agents et qui forment un système multi agents, dans le cadre de l'intelligence artificielle distribuée (IAD).

D'après (Wooldridge 2009), les tendances qui ont menées aux SMAs sont les suivantes :

L'augmentation significative de la délégation des tâches aux systèmes informatiques, le développement de l'intelligence des dispositifs informatiques et la façon dont nous interagissons avec l'ordinateur s'est devenue semblable à la façon dont nous interagissons avec les personnes. Ce qui implique le besoin de concevoir des systèmes informatiques qui peuvent agir à notre place d'une manière efficace.

Aussi l'interconnexion et la distribution des systèmes informatiques par l'interaction et la communication impliquent que les systèmes informatiques doivent supporter l'interaction. Enfin et puisque à nos jours l'informatique est devenue ubiquitaire¹, elle investit les objets de la vie courante. Ces systèmes peuvent être intégrer dans notre actualité.

Le domaine des SMAs est très large dû au fait qu'il est situé entre plusieurs disciplines notamment : L'informatique répartie et génie logiciel, l'intelligence artificielle, la vie artificielle, la psychologie sociale, les sciences cognitives, etc.

Ce chapitre est organisé comme suit : Nous allons commencer par une description du paradigme agent, pour ensuite passer aux principaux concepts et définitions autour des systèmes multi agents. Enfin et après avoir eu une vision générale sur ce domaine nous allons justifier leurs applications à la chaîne logistique avec une argumentation du choix par rapport aux autres alternatives possibles, aussi bien qu'une explication des approches de coordination dédiées.

2.2. Principes généraux

2.2.1. Le paradigme agent

Selon Jennings, Sycara et Wooldridge (Jennings et al. 1998) un agent est un système informatique, situé dans un environnement, et qui agit d'une façon autonome et flexible pour atteindre les objectifs pour lesquels il a été conçu.

Les mots « situé », « autonomie » et « flexible » sont définis comme suit :

- . Situé : l'agent est capable de percevoir son environnement à partir des entrées sensorielles (Ex : capteurs) et d'en agir également par le biais de ses actuateurs.
- . Autonome : l'agent est capable d'agir sans l'intervention d'un tiers (humain ou agent) et contrôle ses propres actions ainsi que son état interne.
- . Flexible : l'agent dans ce cas est :
 - capable de répondre à temps (Réactif) : l'agent doit être capable de percevoir son environnement et d'élaborer une réponse dans les temps requis.
 - proactif : l'agent doit exhiber un comportement proactif et opportuniste, tout en étant capable de prendre l'initiative au bon moment.
 - social : l'agent doit être capable d'interagir avec les autres agents (logiciels et humains) quand la situation l'exige afin d'accomplir ses tâches ou aider ces agents à accomplir les leurs.

Les capacités de l'agent sont de trois ordres qui répondent aux trois phases générales de réalisation d'une tâche (Mandiau et Strugeon. 2002) :

- Une phase de perception : l'agent perçoit son environnement et met à jour ses représentations internes de l'environnement et des autres agents.
- Une phase de cognition : l'agent détermine ce qui est à faire (sa tâche) et décide quand et comment le faire.
- Une phase d'action : la réalisation effective des actions qui ont été décidées.

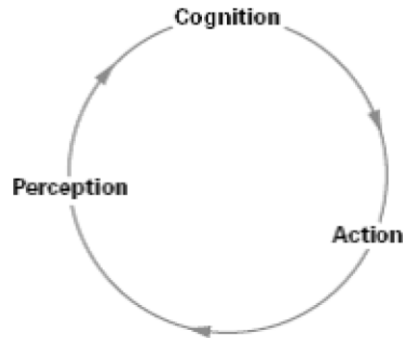


Figure 10 : Les phases de réaction d'un agent (Mandiau et Strugeon. 2002)

Selon (Jennings 2001), la technologie agent va représenter dans les prochaines années un nouveau paradigme de programmation pour le génie logiciel similaire à la programmation orientée objet et qui pourrait s'appeler la programmation orientée agent. Dans ce qui suit, nous allons voir la différence entre ses deux paradigmes.

Un objet encapsule² un ensemble de méthodes (ou services) publics qui sont accessible directement par les autres objets même s'il contredit les buts de l'objet ce qui s'oppose clairement au concept d'autonomie.

Si nous allons voir dans la progression de la programmation, nous trouvons au début le code machine (Code G), dernièrement la programmation orientée objet (Java) et plus récemment la programmation orientée agent.

Le tableau ainsi que la figure ci-dessous récapitulent les majeures différences présentes entre ces deux paradigmes.

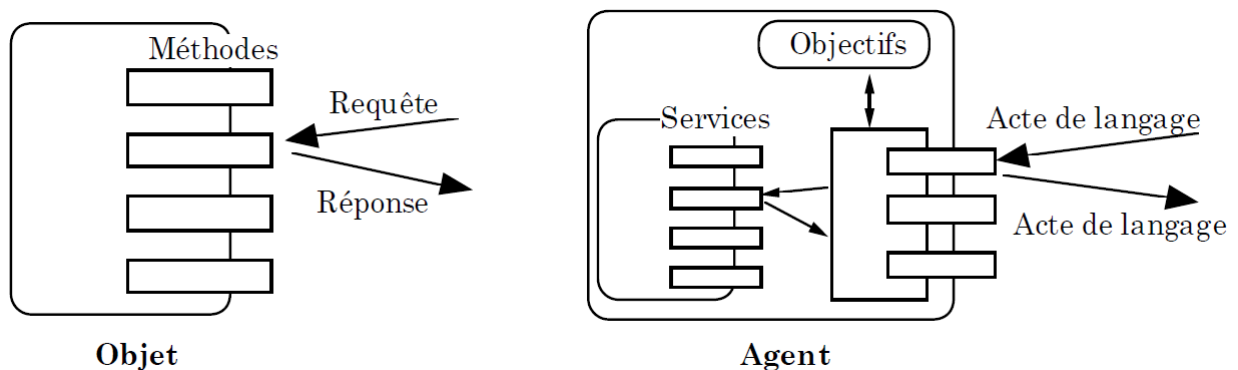


Figure 11 : Différence entre un agent et un objet (Ferber 1995)

Table 4: Différence entre un agent et un objet

| Objet | Agent |
|--|--|
| L'exécution de la méthode m1 encapsulé par l'objet O1 revient à la décision de l'objet O2. | L'agent reçoit une requête de la part d'un autre puis il décide de l'accepter ou la rejeter. |
| Les objets ne possèdent pas cette propriété. Pourtant, ils communiquent par échange de messages mais la sociabilité entre les agents va vers ce que nous appelons les actes de communications (informer, demander, accepter, rejeter...etc.) inspirés de la communication humaine. | Flexibilité (réactivité-proactivité-sociabilité) |
| Contrôle global. | Auto-contrôlable |

La citation courante dans la communauté multi-agents qui décrit la différence entre un objet et un agent est la suivante : « Les objets la fait gratuitement mais les agents la fait car ils la veulent».

2.2.2. Les systèmes multi agents

Un agent agit dans un environnement qui est à la fois physique et informatique. Il y a des situations où l'agent peut agir seul. Cependant cette situation est devenue rare principalement avec l'interconnexions des ordinateurs aujourd'hui. Donc usuellement les agents interagissent avec d'autres agents en formant un système multi agents.

Un système multi-agents (SMA) est un ensemble d'agents qui interagissent entre eux. Généralement, les agents agissent avec des objectifs et des motivations différentes. Pour interagir avec succès, ils auront besoin de coopérer, coordonner et négocier, autant que les gens font (Wooldridge 2009).

Un SMA est décentralisé, puisque chaque agent a juste une connaissance partielle sur son environnement. Aussi un SMA généralement comporte différents types d'agents. Selon (Norvig et Russel 2009), il existe quatre types d'agents intelligents, qui diffèrent entre eux selon la sélection des actions du plus réactif au plus cognitif.

Un agent avec simple reflex :

Ces agents choisissent leurs actions sur la base de la perception actuelle, en ignorant le reste de l'histoire de la perception. Ils réagissent selon les règles condition-actions.

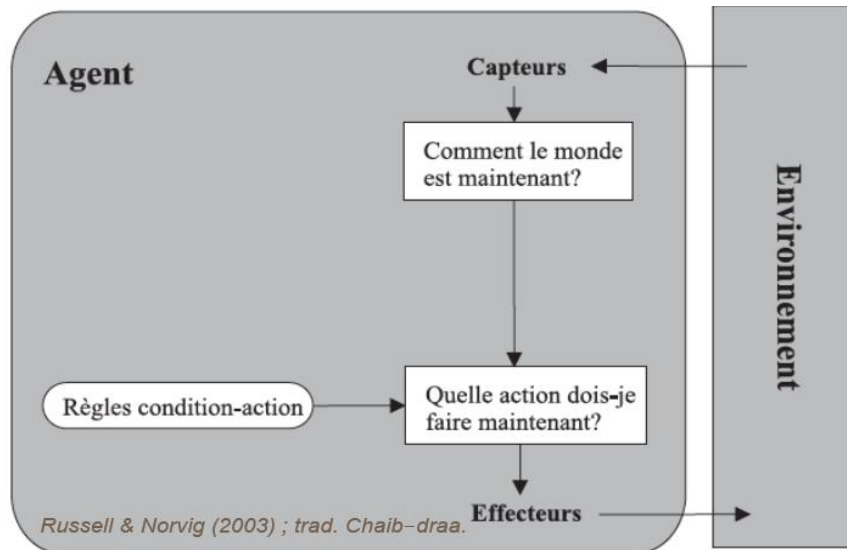


Figure 12: Agent avec simple reflex (Chaib-draa 2006).

Agent avec reflex à base de model :

L'agent cette fois-là, maintient une sorte d'état interne qui dépend de l'histoire de perception et reflète ainsi aux moins quelques aspects inobservés de l'état actuel. Du coup, l'agent a besoin de quelques informations sur la façon dont le monde évolue, aussi bien que sur la façon dont ses propres actions affectent le monde.

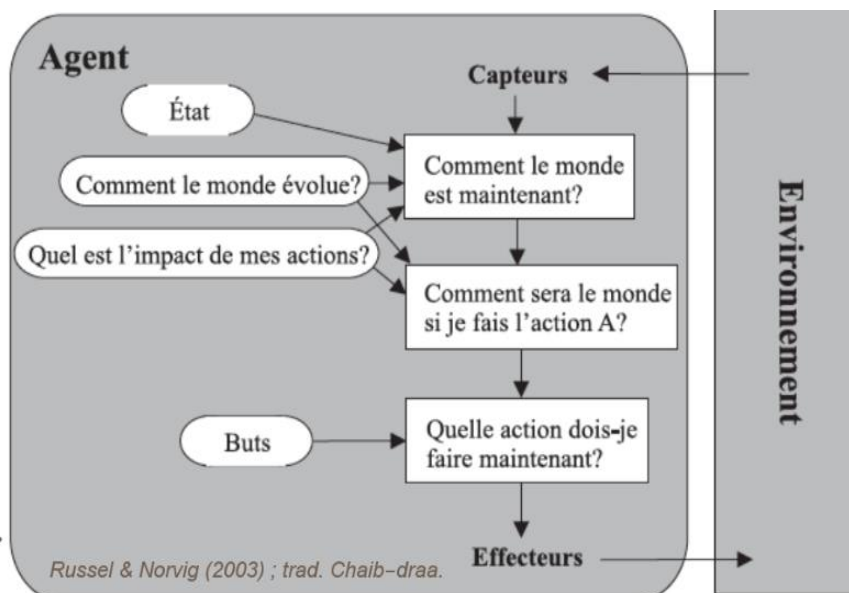


Figure 13 : Agent avec reflex (Chaib-draa 2006)

Agent à base de buts :

Etend l'agent à base de model par l'ajout de buts (états et situations désirables). L'agent choisit l'action qui se rapproche le plus des buts.

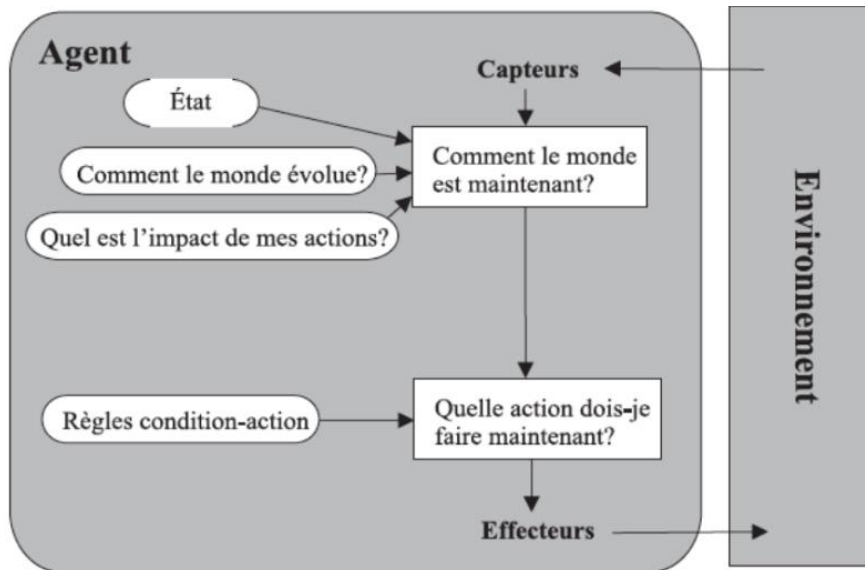


Figure 14 : Agent à base de buts (Chaib-draa 2006)

Agent à base d'utilité :

Etend l'agent à base de buts par l'utilité qui est une fonction permettant de mesurer l'attractivité d'une action, pour que dans une situation à plusieurs alternatives il choisit celle maximisant son utilité.

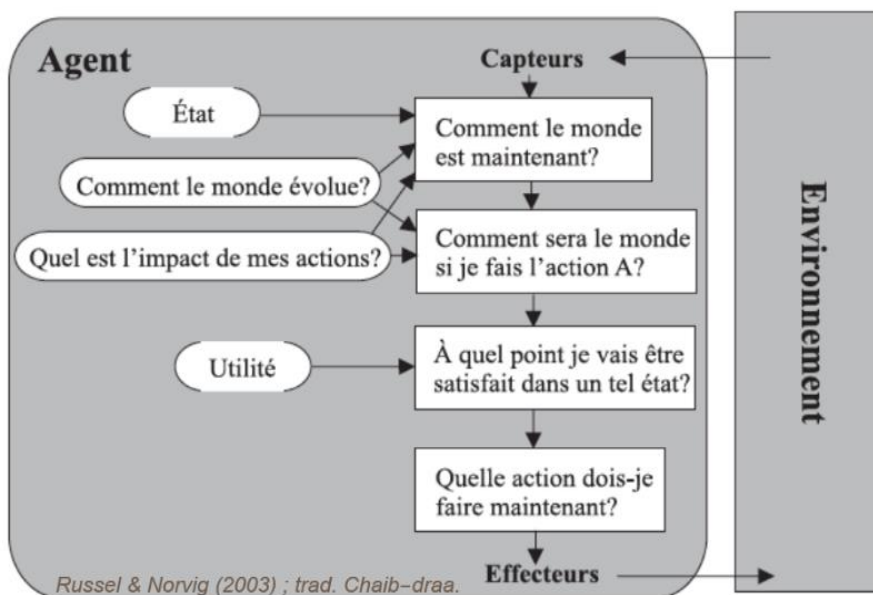


Figure 15 : Agent à base d'utilité (Chaib-draa 2006).

Le rôle de l'intelligence artificielle est de concevoir un programme de l'agent qui décrit le fonctionnement entre la perception et l'action et qui diffère selon les types d'agents cités auparavant. Cependant là où sera exécuté le programme est appelée l'architecture :

$$\text{Agent} = \text{Architecture} + \text{programme (Norvig et Russel. 2009)}.$$

Les architectures peuvent être présentées selon deux visions d'après (Briot et Demazeau. 16), d'abord par rapport à la coordination, et pour affiner ils ont prêté en compte la manière dont l'agent prend des décisions (raisonnement).

-Vision raisonnement :

- Agent réactif.
- Agent Délibératif (proactif)
- Agent Hybrides (combinaison des deux)

-Vision Coordination :

Afin de préciser les capacités liées à la coordination, l'approche VOYELLES est utilisée (Demazeau 1995), elle modélise un SMA selon quatre dimensions (facettes) qui sont :

- L'agent (A) : désigne l'ensemble des fonctionnalités de raisonnement interne de l'agent
- L'environnement (E) : regroupe les fonctionnalités liées aux capacités de perception et d'action de l'agent.
- L'interaction (I) : fonctionnalités liées aux capacités d'interaction de l'agent avec les autres agents.
- L'organisation (O) : concerne les fonctions et les représentations liées aux capacités de structuration et de gestion des relations des agents entre eux.

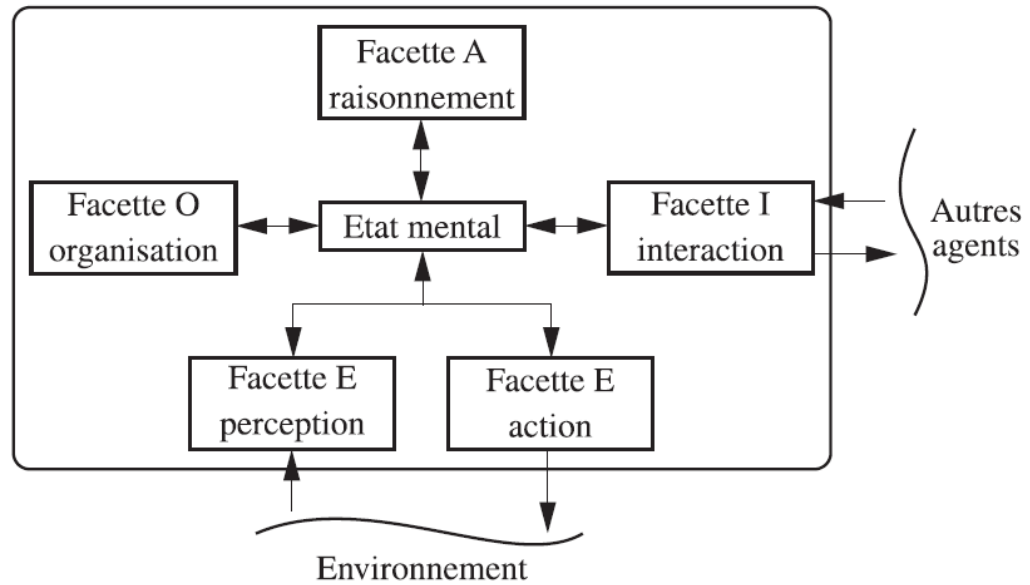


Figure 16 : Approche voyelle (Briot et Demazeau. 2016)

Donc les types de coordination peuvent être définis comme suit :

- Agent autonome (possédant les facettes A, E).
- Agent interagissant (possédant les facettes A, E, I)
- Agent social. (possédant les facettes A, E, I, O)

Les principales architectures développées par rapport au deux dimensions décrites auparavant sont présentées dans le tableau.

Table 5 : Les architectures d'agents (Briot et Demazeau. 2016).

| Raisonnement | Agent réactif | Agent hybride | Agent délibératif |
|---------------------|------------------------------|-----------------|--|
| Coordination | | | |
| Agent autonome | Subsumption PENGLI, MANTA | Touring Machine | <i>IRMA, PRS/dMARS</i> |
| Agent interagissant | PACO, SMARRPS | <i>InteRRap</i> | IMAGINE, ARCHON, AOP, COSY, GRATE* |
| Agent social | ECO, PACORG | | <i>ADEPT, ASIC, STARS, SAM, DIMA, CIA.</i> |

2.3. La communication entre les agents

Les agents peuvent interagir entre eux soit en accomplissant des actions linguistique avec la communication, soit en accomplissant des actions non linguistiques qui modifient leur environnement (Briot et Demazeau 16). La communication est la base de l'interaction puisque sans elle, l'agent est considéré isoler.

Dans les systèmes multi agents ils existent deux types de communication, direct par échange de message ou indirect par accès à une base de données partager. Un mécanisme à part est celui de Marchandise tacite (Tubbs. 1984) ou l'agent peut inférer les plans des autres agents sans communiquer avec eux. Dans ce qui suit nous allons voir la description ainsi que l'évolution des différents mécanismes selon (Briot et Demazeau 16) :

- Transfert de messages :

Cette approche consiste qu'un agent X communique son plan en totalité a un agent Y et Y communique aussi son plan en totalité a X.

- Le tableau noir :

Cette technique est très utilisée dans l'intelligence artificielle, Dans cette approche les agents peuvent écrire des messages, insérer des résultats et obtenir de l'information du tableau noir.

○ Acte de langage :

Le but de la communication et de plus du transfert de l'information est d'induire chez l'autre un comportement spécifique par l'affectation de son état mentale, ceci étant inspiré de la philosophie de langage, qui considère selon (Searle. 1969) que dire quelque chose c'est en quelques sortes agir. Donc dans les actes de langage, l'intérêt est de structurer les messages de façon à influencer les agents.

○ Les langages de communications :

Un des premiers langages les plus pertinent est bien le KQML (Knowledge Query and Manipulation Language), il a été proposé par DARPA-KSE en 1993, dont l'objectif est de définir un standard. Ce langage est fondé sur la théorie de l'acte de langage, il définit un ensemble de types de message (performatifs) et des règles qui définissent les comportements suggérés lors de la réception des messages.

Dernièrement un langage plus riche sémantiquement que le KQML est apparu, s'est bien le langage ACL (Agent communication language) de la FIPA (1999), également fondé sur la théorie de l'acte de langage, ce dernier a bénéficié grandement des résultats du KQML. La liste des performatifs de l'ACL est présentée dans l'annexe 4.

2.4. L'interaction

Dans les systèmes multi agents l'interaction joue un rôle primordial disant déterminant dans la dynamique du système, or les options mises pour modéliser leurs mécanismes sont pour ainsi dire infinies. Selon (Ferber. 1995), une interaction est une mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques.

Dans les systèmes multi agent, les agents interagissent en vue de réaliser conjointement une tâche ou d'atteindre un but particulier. Ceci distingue un SMA à un ensemble d'agents indépendants.

Chaque agent peut être caractérisé par trois dimensions :

- Ses buts.
- Ses capacités à réaliser certaines tâches.
- Les ressources dont il dispose.

Lorsque les agents réalisent leurs actions, certaines actions exécutées simultanément peuvent soit entraîner des conflits ou soit améliorer les performances. Selon (Von Martial.

1992) Types de relations entre les agents sont les suivantes :

Relations négatives (ou conflictuelles) : Gênent ou empêchent plusieurs actions de se réaliser simultanément. Elles comportent l'incompatibilité de buts aussi bien que la limitation de ressources (tous les éléments environnementaux et matériels utiles à la réalisation d'une tâche).

Relations positives (ou synergiques) : Favorisent les actions en les faisant bénéficier les unes des autres. Elles incluent les relations d'égalité où certaines actions ne sont pas liées à un agent particulier et peuvent être réalisées par un autre agent. Aussi les relations de subsumption où l'action 'a' d'un agent A fait partie des actions 'b' d'un agent B, et en réalisant 'b', B réalise du même coup 'a'. Et enfin les relations de faveur où l'action en se réalisant favorise la possibilité d'en réaliser une autre.

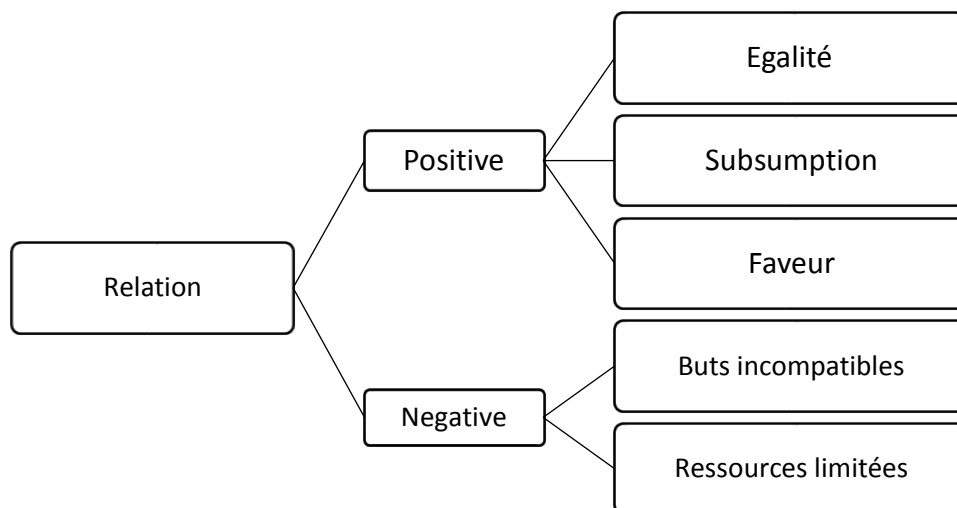


Figure 17 : Les types de relation entre les agents dans un SMA.

Les situations d'interactions entre les agents sont nombreuses et diverses, (Ferber 1995) a proposé une typologie pour avoir une vision claire par rapport aux trois dimensions de l'agent (buts, capacités et ressources).

Dans cette typologie les situations interactions des agents d'un SMA sont motivées par l'interdépendance des agents selon trois dimensions :

- Leurs buts ou les intentions des agents peuvent être compatibles ou non.
- Toutes agent à besoin de ressources pour réaliser ses actions. La limitation des ressources est à l'origine des conflits puisque les agents peuvent désirer des ressources que les autres possèdent.
- Un agent X peut disposer d'une capacité nécessaire à un agent Y pour l'accomplissement d'un des plans d'action de Y.

La typologie est présentée dans le tableau suivant

Table 6 : les typologies des interactions (Ferber. 1995).

| Buts | Ressources | Compétences | Types de situation | Remarques |
|---------------|-------------------|--------------------|---|----------------------------|
| compatibles | suffisantes | suffisantes | <i>indépendance</i> | situation d'indifférence |
| compatibles | suffisantes | insuffisantes | <i>collaboration simple</i> | situations de coopérations |
| compatibles | insuffisantes | suffisantes | <i>emcombrement</i> | |
| compatibles | insuffisantes | insuffisantes | <i>collaboration coordonnée</i> | |
| incompatibles | suffisantes | suffisantes | <i>compétition individuelle pure</i> | situations d'antagonismes |
| incompatibles | suffisantes | insuffisantes | <i>compétition collective pure</i> | |
| incompatibles | insuffisantes | suffisantes | <i>conflits individuels pour des ressources</i> | |
| incompatibles | insuffisantes | insuffisantes | <i>conflits collectifs pour des ressources</i> | |

2.5. Méthodologies et environnements de développement.

2.5.1. Méthodologies de développement

Le moment où le domaine des SMA est devenu bien reconnu, un effort croissant a été dédié pour développer des méthodologies d'analyse et de développement pouvant simplifier leur réalisation. Les premiers travaux étaient focalisés sur leurs classifications et ont permis de dégager 3 catégories principales selon (Wooldridge 2009) :

- Des Méthodologies Orientés Agent (MOAs), dédiées particulièrement au paradigme agent et personnalisé par rapport à ces concepts clés.
- Des extensions orientées objets ou MOAs orientées objets, qui s'inspirent des techniques orientées objets existantes renforcées par quelques concepts liés au paradigme agent.
- Des méthodologies qui adoptent le management de connaissance.

Mais selon notre modeste expérience dans ce domaine nous supportons ce que (Argente et al, 2006) a proposé, là, où les MOAs sont subdivisées en deux grandes classes :

- MOAs à base d'organisations qui considèrent les agents comme entités interagissantes au sein d'une organisation et jouant des rôles spécifiques (Ex : Gaia et ROADMAP)
- MOAs orientées agent qui focalisent sur les agents eux-mêmes et leurs comportements internes. Les SMAs dans ce cas sont conçus selon le paradigme BDI (Ex : AAIL, INGENIAS)

D'autres tentatives étaient consacrées à créer l'unique standard méthodologie mais ils ont tous échoués et malheureusement ce but reste le sujet de travail de plusieurs chercheurs.

Une grande partie de notre projet vers cette optique et nous a conduit à proposer une nouvelle méthodologie à base de ROADMAP et Gaia. La question qui se pose est pourquoi choisir Gaia et ROADMAP comme méthodologie de base et pourquoi proposer une nouvelle méthodologie ?

Gaia et ROADMAP se présentent comme :

Des méthodologies orientées processus (ou rôle) qui adopte alors une approche systémique plus adapté pour notre réflexion en tant que ingénieurs en génie industriel se retrouvant bien familiarisé avec la modélisation des processus industriels.

Dans leurs cœurs, ils suivent une approche Top-down en commençant par l'élaboration d'une série de modèles décrivant le système entier et ses buts et qui tendent à se concrétiser en descendant. Par contre, une méthodologie comme *Cassiopeia* suivant une approche Bottom-up commence par identifier les comportements élémentaires et leurs interdépendances puis passe à concevoir les comportements organisationnels du système en entier. Ce dernier est de plus fortement soutenu dans Gaia et ROADMAP ce qui peut énormément servir à limiter les interactions entre les agents pour les SMAs impliquants plusieurs agents. En effet, un SMA qui vient également simuler le comportement d'un système existant devra ainsi imiter les interactions ayant lieu dans le système réel. Ces dernières suivent un certain modèle dont l'identification préalable permet d'identifier la structure ou formation d'agents la plus adaptée permettant de réduire le flux de communication entre les agents.

En outre, Gaia propose lors de l'élaboration de « organisational structure model » d'associer un ensemble de contraintes et de politiques afin d'éviter les comportements indésirables et exclure

les agents tricheurs. C'est tout à fait analogue à notre société humaine là, où les lois sociales et les réglementations gouvernementales limitent les violations interdites et expulsent ou exilent les criminels.

ROADMAP a été lancé dans l'intention d'étendre GAIA avant de devenir une méthodologie indépendante. Par conséquent, elles partagent plusieurs similarités. Les deux méthodologies supportent l'utilisation des diagrammes UML et AUML jugées très pratiques lors de la description de certains aspects tels que : protocoles d'interactions entre agents, l'aspect environnementale, ou la structure d'agent lui-même en tant qu'objet dans JADE. Les modèles de sortie final issus des deux méthodologies ressemblaient très proche d'être traduit en code JADE que nous avons déjà choisi. De plus, d'autres méthodologies tels que : Tropos (Bresciani et al. 2004), PASSI (Cossentino et al. 2005) et Prometheus (Padgham et Winikoff, 2004) ignorent certains formations organisationnelles comme : les structures hiérarchiques, holarchiques et à base d'équipes. Ce qui peut sans doute limiter leurs champs d'applicabilité. Au contraire, Gaia et ROADMAP peuvent même s'élargir pour regrouper les structures coalitionnelles, à base de congrégations, les sociétés et base de marchés.

ROADMAP a assez évolué jusqu'à devenir une méthodologie entièrement indépendante et un peu loin de Gaia grâce à l'introduction de nouveaux modèles jugées complémentaires. Elle n'avait que des libellés mais aucun squelette explicite décrivant la logique de passage d'un modèle à un autre. Contre Gaia, qui a eu un squelette claire enveloppant tous se modèles et suivant une logique linéaire lisible et facile à suivre.

En partant de cette logique nous avons bien jugé que :

- Gaia et ROADMAP peuvent nous bien servir comme point de départ pour l'instant en tant que nouveaux entrants et que chacun d'eux possède ses propres défauts pouvant se couvrir par l'autre. Ce qui nous a conduit alors à proposer une nouvelle méthodologie -ou plutôt modestement- un prototype visant à capturer le meilleur des deux et pouvant même couvrir la phase d'implémentation ignorée également par la plupart des méthodologies.

- Il faudra aussi bien noter, que notre choix peut ne pas être assez rationnel du fait que nous pouvions pas couvrir tous les méthodologies existantes comme ASPECS la plus récente méthodologie introduite en 2010 pour viser les système à haut degré de complexité et qui demeure donc une boîte noire non encore assez explorée et pouvant également être plus attractive ou une bonne source d'inspiration. Mais d'emblée, après avoir jeté un coup d'œil

superficielle elle ressemblait *plus complète* dans certains aspects car basée sur l'expérience gagnée des méthodologies antérieures *mais trop complète* car visant le développement des systèmes complexes dépassant les exigences du système que nous avons l'intention de développer.

De plus, les travaux de recherche existants manquent une comparaison explicite entre les méthodologies ce qui nous a conduit à prendre une décision intuitive reflétant notre propre perception.

2.5.2. Environnements de développement

A nos jours le paradigme agent connaît un grand succès dans le milieu industriel. Cependant et pour assurer le succès il faut l'accompagner avec des outils de conception et d'implémentation dite les environnements de développement. Ces derniers permettent de réaliser les applications des systèmes multi agents pour une meilleure validation du modèle.

Plusieurs environnements de développement ont été développés. Pour notre projet nous avons choisis JADE comme environnement de développements.

JADE ou **J**ava **A**gent **D**evelopment framework est une plateforme de développement de Systèmes Multi Agents qui exploite ce nouveau paradigme émergent "Agent Oriented Software Engineering" mais par une approche orienté objet qui est bien familiarisé chez la communauté des développeurs afin de les permettre de réaliser leurs propres applications distribués facilement.

JADE exploite le paradigme "agent" dans la manière il implémente tous les concepts clés relatifs à cette abstraction. On parle bien de : 1-Autonomie, réactivité et proactivité, 2- Comportement social des agents, 3-La libre volonté des agents.

1. Autonomie, réactivité et proactivité :

Puisqu'un agent ne peut jamais avoir control sur un autre, en JADE un agent ne peut plus fournir son référence d'objet. Cependant, les agents dans une plateforme JADE sont identifiés par leurs GUID (Global Unique IDentifier) ou simplement leurs noms.

Chaque agent possède son propre fil d'exécution (ou de l'anglais thread) pour qu'il lui-même puisse contrôler l'exécution de son cycle de vie. Et Puisqu'être à jour pour un agent est extrêmement important pour ajuster son état interne continuellement, un agent doit contrôler

toute la plateforme puis sélectionner les évènements qui lui intéressent. Ce qui est non seulement une lourde tâche mais aussi qui va certainement bruler le micro-processeur. Alors que le mécanisme est beaucoup plus simplifié en JADE à travers un système dit « Event Notification System (ENS) » là, où l'agent n'a besoin que de s'inscrire pour certains évènements spécifiques pour être notifié plus tard lors de leur occurrence. Ceci, s'agit donc d'une juste réactivité au juste coût. La figure A et B ci-dessous montre respectivement le pourcentage d'utilisation de CPU pour un agent exécutant un CyclicBehaviour pour disons mettre à jour en temps réel la liste de fournisseurs disponibles dans le marché. Quant à la figure B montre le pourcentage d'utilisation après avoir arrêté le comportement (*behaviour*).

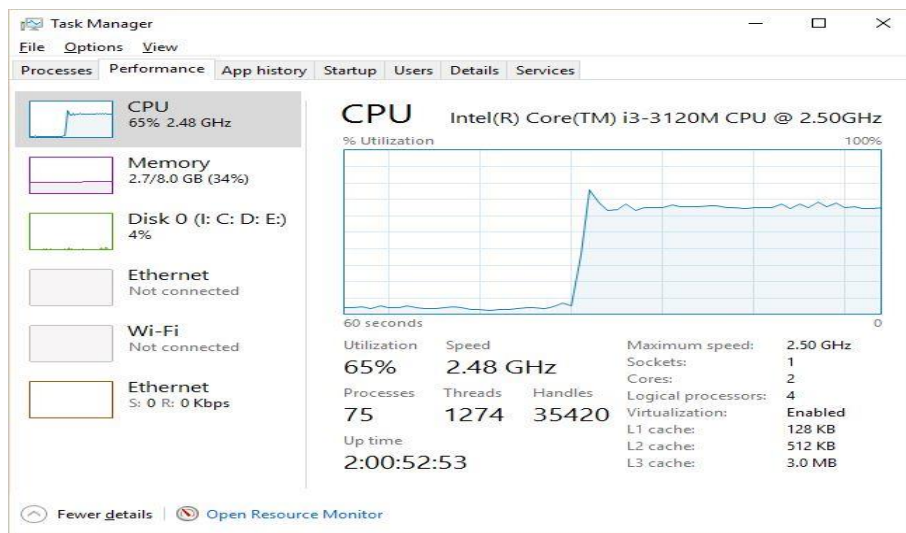


Figure 18 : Utilisation de CPU avec lancement du behavior

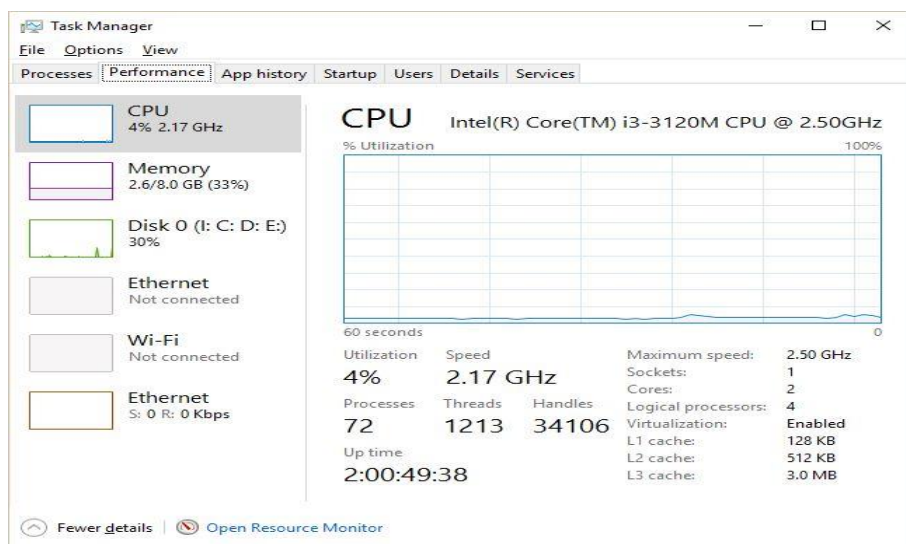


Figure 19 : Utilisation de CPU sans lancement du behavior

2. Comportement social des agents

La théorie des SMA énonce que les agents doivent interagir par passation de messages et non par accès direct à l'état interne de chacun d'autre. JADE implémente ce concept à travers un système de communication asynchrone avec comme son nom l'indique aucune *dépendance temporelle* entre les acteurs (expéditeur et récepteur). Ceci, permet l'expéditeur du message de continuer à contrôler son fil d'exécution et ne pas être bloqué jusqu'à recevoir une réponse.

3. La libre volonté des agents

En JADE chaque agent possède son propre queue de messages dans laquelle tous les messages reçus sont empilés. Mais c'est à l'agent propriétaire de sélectionner les messages à considérer et les messages à rejeter à travers un filtre de sélection personnalisé. Et là, pas mal de critères de filtrage sont disponibles.

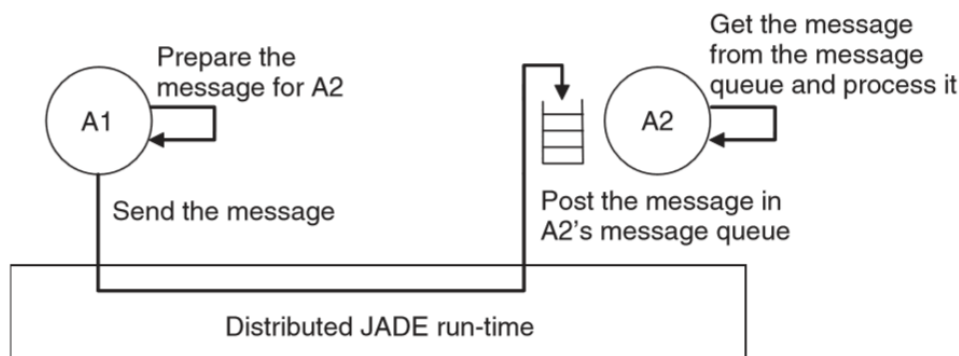


Figure 20: Filtrage des messages

D'autres fonctionnalités secondaires sont aussi là pour simplifier la tâche du développeur mais qui ne sont pas en relation avec les concepts cœurs décrits antérieurement tel que les services « pages jaunes », les services « pages blanches », outils de débogage et le support au concept de mobilité d'agent.

Le service de « pages blanches » est le registre de tous les agents présents dans la plateforme et est managé par un agent spécial dit «Agent Management System (AMS) ». D'autre part, le service « pages jaunes » est un mécanisme pour publier-découvrir les services

des agents et est supervisé par l'agent « Directory Facilitator (DF) ». À titre d'exemple, un agent « vendeur » peut publier son service « location de voitures touristiques » pour être découvert ensuite par un agent « acheteur ». Ceci est inspiré du monde réel là où les entreprises lancent leurs publicités par TV pour que le consommateur cible en soit conscient. Et c'est toute a fait une habitude bien reconnue dans l'intelligence artificielle qui vise à simuler la vie réelle.

Un kit de débogage graphique comprenant 4 agents pré-implémentés permettant au développeur de tester son agent personnalisé.

La plateforme comprenne aussi des classes supportant la mobilité d'agent pour permettre à ce dernier de migrer totalement avec son code source et son état interne d'une machine à une autre sans dégrader le processus d'interaction avec ses pairs agents. autrement dit, au lieu de ramener les données à l'agent on déplace l'agent vers les données exactement dans la même manière un employé doit se déplacer vers son bureau de travail pour traiter l'archive des données au lieu de ramener l'archive chez lui.

Ceci est bien la raison d'avoir choisi JADE comme plateforme de développement.

- **Analogie avec développement d'interfaces graphiques :**

Dans la programmation graphique (ou développement d'interfaces graphiques) un ensemble de classes implémentant les composants de l'interface sont fournies (JFrame, JButton, JCheckBox ...etc.) et c'est au programmeur de les utiliser directement ou les personnaliser via héritage.

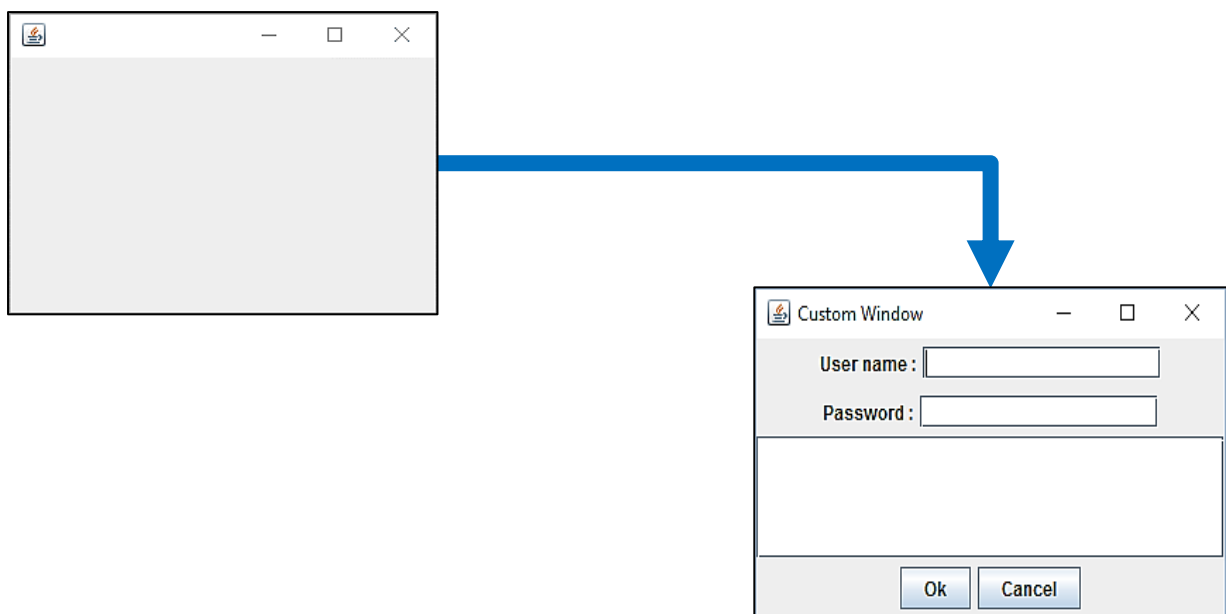


Figure 21: Customizing a JFrame Window

Suivant la même manière JADE n'est qu'un ensemble de classes que l'utilisateur doit les personnaliser selon le domaine d'application.

```
    Import jade.core.Agent
    Public class MyCustomAgent extends
        Agent
        {
            /*additional attributes*/
            /*additional methods*/
            /*class constructor*/
            /*overriding super class methods*/
            /*internal classes*/
            .
        }
```

Figure 22 : Making a custom agent out of jade.core.Agent class

2.6. Les systèmes multi agents pour la gestion de la chaîne logistique

2.6.1. Choix des systèmes multi agents

De nos jours, les systèmes multi agents ont connus une avancée considérable. Ils sont appliqués avec un grand succès dans de nombreux domaines comme le commerce électronique, la télécommunication, l'ingénierie des systèmes complexes, etc.

Un des domaines d'application est bien la gestion de la chaîne logistique. Selon (Anane et al. 2009) l'approche agent apparaît comme une technologie intéressante pour modéliser la complexité de la chaîne logistique et de ses comportements.

Parmi les outils de résolution de problèmes distribués, les systèmes multi agents se présente plus particulièrement approprié à la chaîne logistique. Selon (Briot et Demazeau 16), ces systèmes possèdent les avantages traditionnels de la résolution distribuée et concurrente de problèmes comme la modularité, la vitesse (grâce au parallélisme) et la fiabilité (grâce à la redondance).

Ils héritent aussi des bénéfices envisageables de l'intelligence artificielle comme le traitement symbolique (au niveau des connaissances), la facilité de maintenance, la réutilisation et la portabilité, mais surtout ils ont l'avantage de faire intervenir des schémas d'interaction sophistiqués. Les types courants d'interaction incluent la coopération (travailler ensemble à la résolution d'un but commun), la coordination (organiser la résolution d'un problème de telle sorte que les interactions nuisibles soient évitées ou que les interactions bénéfiques soient exploitées) et la négociation (parvenir à un accord acceptable pour toutes les parties concernées).

De plus, les SMA ont l'avantage de permettre d'élaborer des systèmes 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).

Dans ce qui suit nous allons faire le lien entre un acteur de la chaîne logistique et un agent pour bien voir l'adéquation des deux domaines.

D'abord, un système multi-agents contient un certain nombre d'agents qui peuvent être représenté selon (Wooldridge 2009) par la figure comme des agents :

- Qui interagissent entre eux.
- Sont en mesure d'agir dans un environnement.
- Ont différentes «sphères d'influence» (qui peuvent coïncider).
- Seront reliés par d'autres relations (organisation).

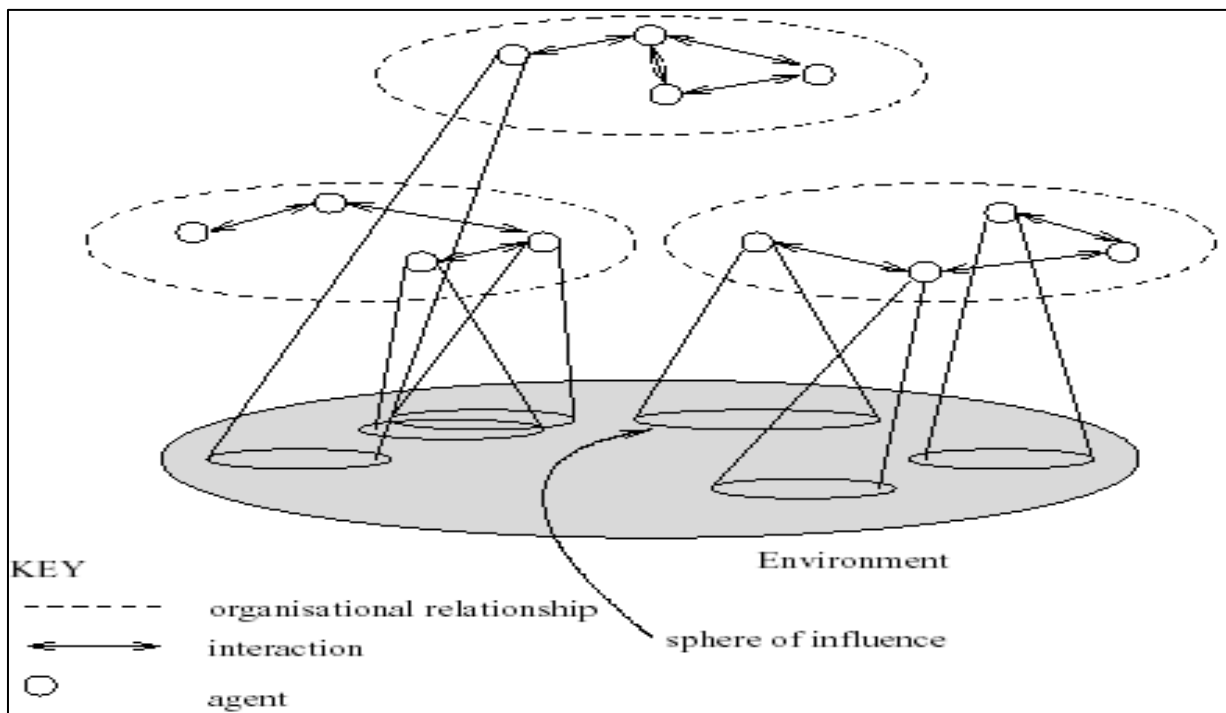


Figure 23: Représentation d'un système multi agents (Wooldridge 2009).

L'approche agent peut être vue comme une combinaison des deux classes proposées par (Shapiro 2001). Une première concerne l'approche analytique à travers la notion des agents intelligents qui sont des entités autonomes pouvant décider en temps réel. Deuxièmement l'approche transactionnelle, avec les protocoles d'interaction misent à la disposition des agents pour pouvoir coopérer.

En reprenant la définition d'un agent intelligent, un agent permet donc de représenter un acteur de la chaîne logistique (entreprise, atelier, etc.), avec :

- Chaque acteur peut exécuter des tâches par lui-même sans intervention d'autres acteurs de l'extérieur (autonomie).

- Chaque agent peut communiquer et interagir avec un autre agent pour, par exemple, passation de commande de produits ou demande de services (capacité sociale).
- Chaque agent est capable de modifier son comportement comme réactivité aux changements comme : l'évolution du marché (réactivité).
- Chaque agent peut initier lui-même de nouvelles actions, comme par exemple décider de lancer un nouveau produit ou intégrer un autre marché (proactivité).
- L'agent s'adapte automatiquement avec l'incertitude de l'environnement (apprentissage).

Donc la notion du système multi agents apparait naturellement comme la représentation la plus adéquate de la chaîne logistique. Dans un même système il existe un ou plusieurs organisations. Dans le cas de la chaîne logistique, une organisation peut être considérée comme une entreprise. Un agent peut désigner une partie de l'elle (département, etc.) ou sa totalité. L'ensemble des agents agit sur un environnement qui peut être représenté par le marché. Une illustration du niveau inter-organisationnel des entreprises peut être présentée avec la figure suivante.

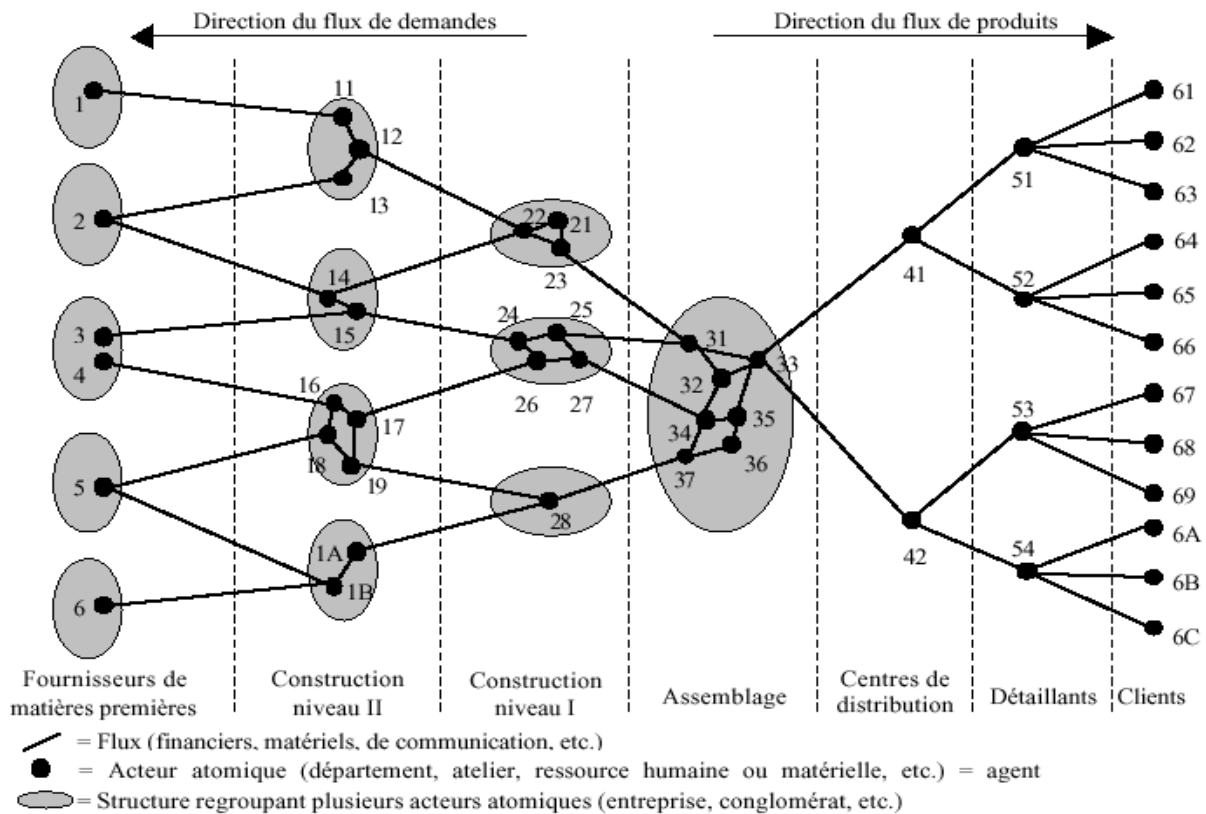


Figure 24 : chaîne logistique avec regroupement des acteurs (D'amours et al. 2006).

2.6.2. Approche de coordination de la chaîne logistique à travers les SMAs

Dans la chaîne logistique plusieurs solutions ont été proposées mais la majorité des solutions utilisent de façon naïve les systèmes multi-agents. Par exemples, ces solutions utilisent des agents mais ne proposent pas des solutions pour gérer des problèmes de coordination. Un des problèmes les plus étudiés dans les systèmes multi agents est bien les approches de coordination. Ceci est dû à la grande importance de l'interaction. Ces approches peuvent être divisées en deux grandes catégories.

La première concerne les situations coopératives qui inclus les cas où les préférences individuelles des agents ne sont pas prisent en compte. Donc les agents collaborent pour atteindre un but global. Selon (Leyton-Brown et Shoham, 2010) le but peut soit être partagé entre les agents, comme dans le cas du modèle de planification globale partielle (PGP) proposé par (Durfee, 1991), et qui a été utilisé pour le contrôle de la circulation des voitures dans un espace divisé en zones (DVMT : Distributer vehicule monitoring testbed). Le but peut être aussi designer par un agent centrale dans le cadre des protocoles de coopérations comme le CNP (contract protocole net) proposé par (Smith, 1980) (figure). Dans ce protocole un agent manager décompose un problème en sous problèmes. Il annonce ensuite chaque sous-problème à un ensemble d'agents contractants. Les agents contractants évaluent l'annonce et proposent des offres. L'agent manager évalue les soumissions et accorde le contrat à l'agent contractant qui a fait la meilleure offre.

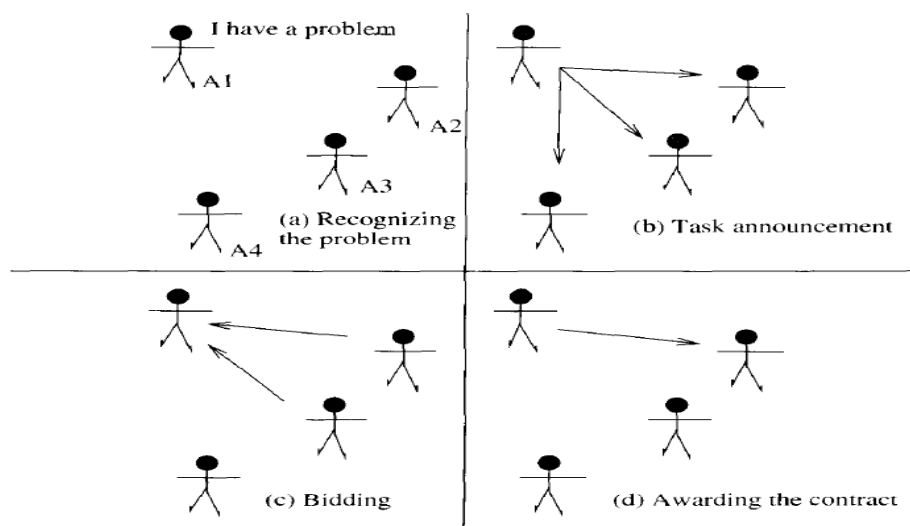


Figure 25 : Contract net protocol (Wooldridge 2009).

La deuxième inclut les situations compétitives où les préférences individuelles des agents sont prises en compte d'où la génération des conflits. Principalement la résolution des conflits passe par la négociation, d'autres alternatives peuvent avoir lieu dans certains cas comme la décision autoritaire. Par exemple la réglementation dans la société et malgré la présence des buts incompatibles mais la relation entre les membres est soumise aux lois. La négociation dans un SMA est un processus par lequel deux ou plusieurs agents interagissent dans le but d'atteindre un arrangement mutuellement acceptable (Crary 1992). Dans la littérature plusieurs modèles à base de négociation ont été proposés. Généralement dans la Chaîne logistique ((D'amours et al. 1999), (Esmaeili et al. 1999), etc.) et l'e-Commerce ((Dayong et al. 2005), (Papazouglo. 2001), etc.)

Par rapport à la chaîne logistique, la négociation demeure comme la solution la plus adéquate pour gérer les interactions entre les acteurs. Puisque, il existe toujours des préférences individuelles entre les acteurs ce qu'il engendre nécessairement des situations conflictuelles. De plus et non seulement elle permet de représenter la conformité des processus modéliser mais aussi simuler les conditions réelles à travers les dimensions de la négociation comme le comportement, l'émotion et la confiance ((Jiang et al. 2005), (Haghpanah. 20011), (Denko et al. 2011), etc.)

Les mécanismes décisionnel de négociation décrivent la façon dont les agents partagent les informations et prennent leurs décisions. Dans la littérature plusieurs typologies ont été proposées, principalement toutes ces dernières incluent la théorie des jeux (coopérative et non-coopérative), les enchères, l'argumentation (persuasion) et le bargaining.

Les enchères :

Une des solutions les plus efficaces pour le problème d'allocation de ressources entre les agents égoïstes est bien les enchères. Selon (Leyton-Brown et Shoham. 2010), les enchères sont des protocoles qui permettent aux agents d'indiquer leurs intérêts sur un ou plusieurs ressources et qui utilisent ces indications pour déterminer l'allocation des ressources et le paiement relatif. Le type le plus familiarisé d'enchère est celui à un seul bien à vendre avec un vendeur et multiple acheteurs, le vendeur à sa propre évaluation et les acheteurs veulent acheter au minimum prix possible. Les enchères canoniques (Leyton-Brown et Shoham. 2010), présente un ensemble d'enchères les plus célèbres : Anglaise, Japonaise, Allemande et Vickrey.

Par exemple dans les enchères Anglaises le vendeur annonce un prix pour un bien, les acheteurs ont le droit d'annoncer des offres successives mais avec des valeurs strictement croissantes. Il y a différentes lois pour fermer l'enchère, des fois dans un temps fixé, d'autres fois dans une période fixé ou il n'y a pas de nouvelle offre, etc. le participant avec la plus grande soumission gagne l'objet. Par ailleurs, il y a d'autres types d'enchères avec des biens multiples avec deux cas. Le premier cas concerne plusieurs biens de même genre (enchères à multi unités) et le deuxième cas qui présente la forme la plus complexe des enchères avec une variété de biens qui ne sont pas interchangeables entre eux (Les enchères combinatoires).

L'argumentation :

Selon (Amgoud et al. 2007) l'argumentation est une activité verbale et sociale de raisonnement visant à augmenter (ou diminuer) l'acceptabilité d'un point de vue controversé pour un auditeur ou lecteur, en proposant des arguments prévus pour justifier (ou réfuter) le point de vue avant un jugement raisonnable. L'argumentation à base de négociation permet à un agent de préciser des informations supplémentaires aux partenaires pour justifier le refus d'une offre ou bien la partie problématique de lui. Donc elle permet d'améliorer le potentiel du processus de négociation avec des contre-propositions informatives. Généralement un agent utilise des arguments pour convaincre l'autre d'accepter la proposition faite, il y a plusieurs types d'arguments et chaque type d'argument définit des préconditions pour son utilisation, si elles sont remplies, alors l'agent peut utiliser l'argument. La forme basic de l'argumentation est bien la critique qui est une forme d'une nouvelle information sur la rejection d'une proposition (D'amours et al 2008).

Un des exemples de base de l'argumentation est le système d'argumentation ANA (Kraus et al. 1998). Les arguments utilisés dans ANA sont les suivantes :

- Appels à une promesse passée où le négociateur rappelle son partenaire d'une promesse concernant l'objet de négociation mais il doit d'abord vérifier si une promesse a été reçue au passé dans une négociation conclue avec succès.
- Promesse d'une récompense future dans laquelle le négociateur promet quelque chose pour son partenaire à un moment dans le futur. La précondition est que le négociateur doit trouver un désir du partenaire pour un moment dans le futur qui peut le satisfaire par une action alors que le partenaire ne peut pas.

- Appels au propre intérêt où l'agent négociateur croit que arriver à un accord est dans l'intérêt du partenaire. Donc il essaye de le convaincre par rapport à ça. La précondition est de trouver un des désirs qui sera satisfait avec l'accomplissement de l'action.
- Le négociateur menace le partenaire de refuser de faire ou d'offrir quelque chose avec quelque chose qui contredit les désirs du partenaire. Dans ce cas la précondition est que le négociateur doit trouver un des désirs du partenaire directement influencé par l'action.

La théorie des jeux :

Selon (Leyton-Brown et Shoham. 2010), il n'y a pas question que la théorie des jeux est l'outil clé de la négociation. Elle est très utilisée dans différents domaines surtout en économie pour étudier par exemple le comportement des oligopoles (théorie de la firme). Quand le marché est contrôlé par un petit nombre d'entreprises, chaque firme analyse les comportements des autres et établit sa stratégie en fonction de celles des autres. Ainsi, ces comportements interdépendants peuvent apparaître comme des jeux, où chaque firme jouerait un coup, puis recevrait la réponse d'une autre, etc. C'est pourquoi la théorie des jeux est souvent utilisée en économie pour modéliser les comportements oligopolistiques. Elle est aussi utilisée dans d'autres domaines de l'économie industrielle, en économie du travail, des échanges internationaux, etc. A nos jours elle est de plus en plus utilisée dans la chaîne logistique. Elle est cependant plus normative que descriptive, car rien ne prouve que les acteurs raisonnent exactement comme la théorie des jeux le prescrit. C'est plus une abstraction de comportement qu'un véritable modèle.

Après que nous avons vu dans le premier chapitre de la première partie lors de la présentation des contrats les notions fondamentales de la théorie des jeux. Nous allons aborder dans ce qui suit des notions un peu avancées. D'abord la théorie des jeux inclut deux sous-domaines : la théorie des jeux coopérative (formation des coalitions) et non-coopérative. Selon (Leyton-Brown et Shoham 2010), la différence essentielle entre les deux est que, dans la théorie des jeux non-coopérative l'unité de base de modélisation est l'individualité, alors que dans la formation des coalitions nous parlons de group. Nous allons nous concentrer sur l'approche non-coopérative qui est devenue l'approche dominante de la théorie des jeux. Dans cette approche il existe plusieurs formes de jeux, la différence est liée principalement à deux critères majeurs : la séquence de jeu et l'information. Dans le cas où la séquence est simultanée, nous parlons de la forme normale représentée par la matrice standard de la théorie des jeux dans le tableau suivant.

Table 7 : la forme standard matricielle.

| | | |
|----------|-------------|-------------|
| | <i>C</i> | <i>D</i> |
| <i>C</i> | <i>a, a</i> | <i>b, c</i> |
| <i>D</i> | <i>c, b</i> | <i>d, d</i> |

Mais cependant et si la séquence du mouvement est considérée. Majoritairement dans les situations où les agents prennent des décisions à tour de rôle qui peuvent être décrites commodément à l'aide d'un arbre de jeu (une arborescence) présenté dans la figure suivante.

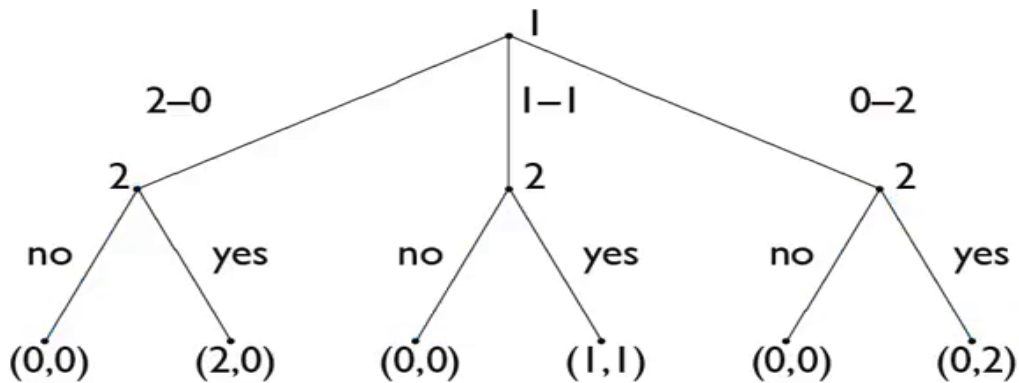


Figure 26 : Arbre de jeu (Leyton-Brown et Shoham. 2010)

Maintenant est par rapport au deuxième critère qui est l'information, nous pouvons distinguer d'autres formes de jeux. Premièrement par rapport à la perfection nous parlons de jeu à information imparfaite lorsque certains des joueurs ne savent pas toujours exactement à quel sommet de l'arbre de décision ils se trouvent.

La figure précédente présente une forme extensive à information parfaite puisque les joueurs savent toujours leurs positions par contre le cas d'information imparfaite peut être présenté dans la figure suivante :

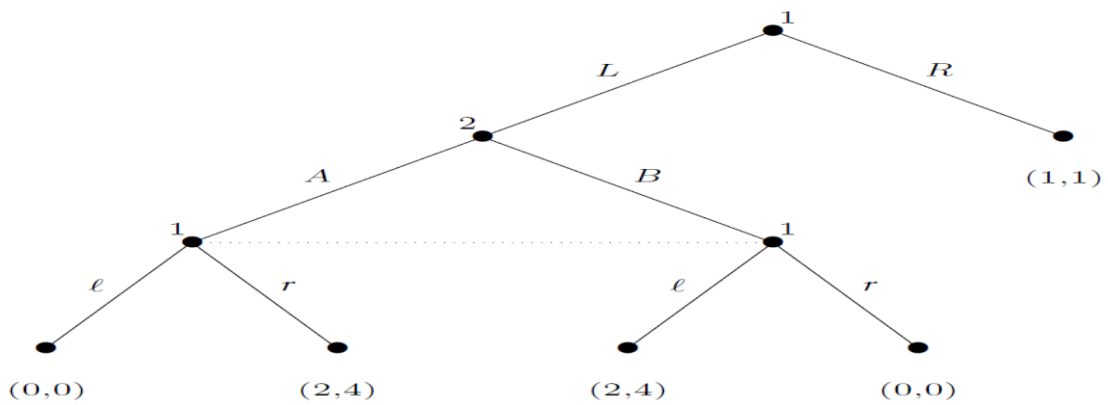


Figure 27 : Arbre de jeu avec information imparfaite (Leyton-Brown et Shoham. 2010)

Toujours par rapport à l'information, il se peut aussi que l'information soit incomplète lorsque certains joueurs ne savent pas exactement quels sont les paiements de la matrice de jeu (ils ne savent pas à quel jeu ils jouent). Cette situation peut être présentée par la figure suivante.

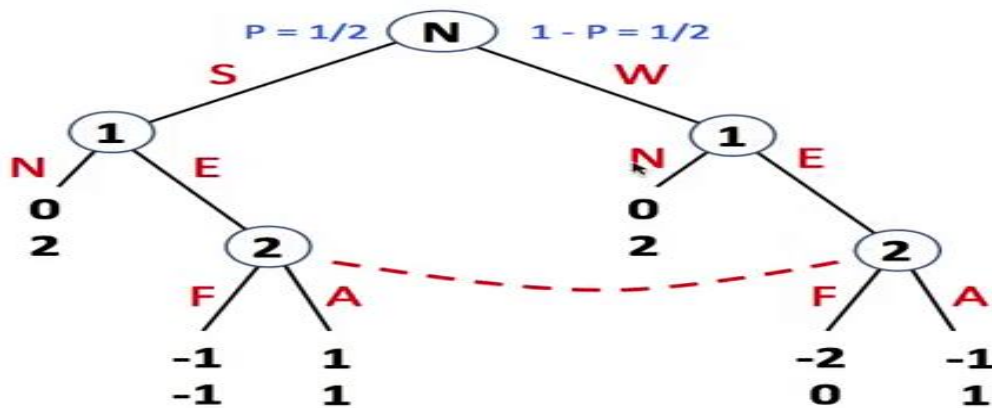


Figure 28 : Arbre de jeu avec information incomplète (Leyton-Brown et Shoham. 2010)

Dans les termes de la théorie des jeux. Nous sommes passés de l'optimalité à l'équilibre. Dans la décision d'un agent unique nous pouvons parler de l'optimalité qui maximise le profit d'un agent dans un environnement donné, mais la situation peut être très difficile devant les caractéristiques de l'environnement comme l'incertitude, etc. Même beaucoup plus complexe

dans un environnement avec plusieurs agents qui essaient de maximiser leurs profits dans le cadre d'un système multi-agents. La théorie des jeux essaie dans un tel problème d'identifier un ensemble de choix appelé les concepts de solution. A travers ces derniers l'agent peut trouver une solution convenable par rapport à n'importe quelle forme de jeu. Dépendamment des autres agents cette dernière peut être représentée par deux concepts qui sont l'optimum de Pareto et l'équilibre de Nash. L'optimum de Pareto reflète la rationalité collective alors que l'équilibre de Nash exprime bien la rationalité individuelle, cependant le comportement des agents doit être rationnel aussi bien qu'il doit est avoir un minimum d'informations sur les préférences des autres agents.

Le bargaining :

Les agents adoptent toujours dans la théorie des jeux un comportement rationnel. Ce type de comportement peut parfois les amener à des solutions non satisfaisantes. Aussi elle nécessite un minimum de partage d'informations entre les agents alors que dans la réalité et dans plusieurs cas les informations sont complètement asymétrique et privé. Une des solutions pour faire face au limite de la théorie des jeux est bien le bargaining dans lequel les agents ne sont pas obligé à être complètement rationnel aussi bien que les informations peuvent être privées (système décentralisé).

En effet, le bargaining ne conduit pas nécessairement à une solution optimale, selon (Klabi 2012) il utilise des stratégies approximatives et il n'examine pas tout l'espace possible des résultats. Cette approche ne peut pas prévoir le comportement exact du système et ceux de ses agents et elle suppose que l'agent possède une connaissance complète de ses désirs et de ses préférences.

Il existe deux types de bargaining selon (Weiss 1999) :

Bargaining axiomatique :

Par contre à la théorie des jeux, le bargaining axiomatique n'utilise pas l'idée de concept de solution mais il utilise les propriétés désirables de la solution appelées axiomes, et après la solution qui satisfait les axiomes est present. La solution Nash-bargaining se présente comme l'une des premières solutions dans ce sens.

Bargaining stratégique :

Contrairement au bargaining axiomatique le bargaining stratégique n'utilise pas les axiomes pour déterminer les solutions mais il modélise la solution sous forme d'un jeu basé sur les axiomes définis par les agents eux-mêmes et donc il est plus juste envers eux et il explique mieux le comportement de la rationalité des agents. Usuellement il analyse la séquence de bargaining où les agents alternent leurs offres les uns aux autres dans un ordre bien défini.

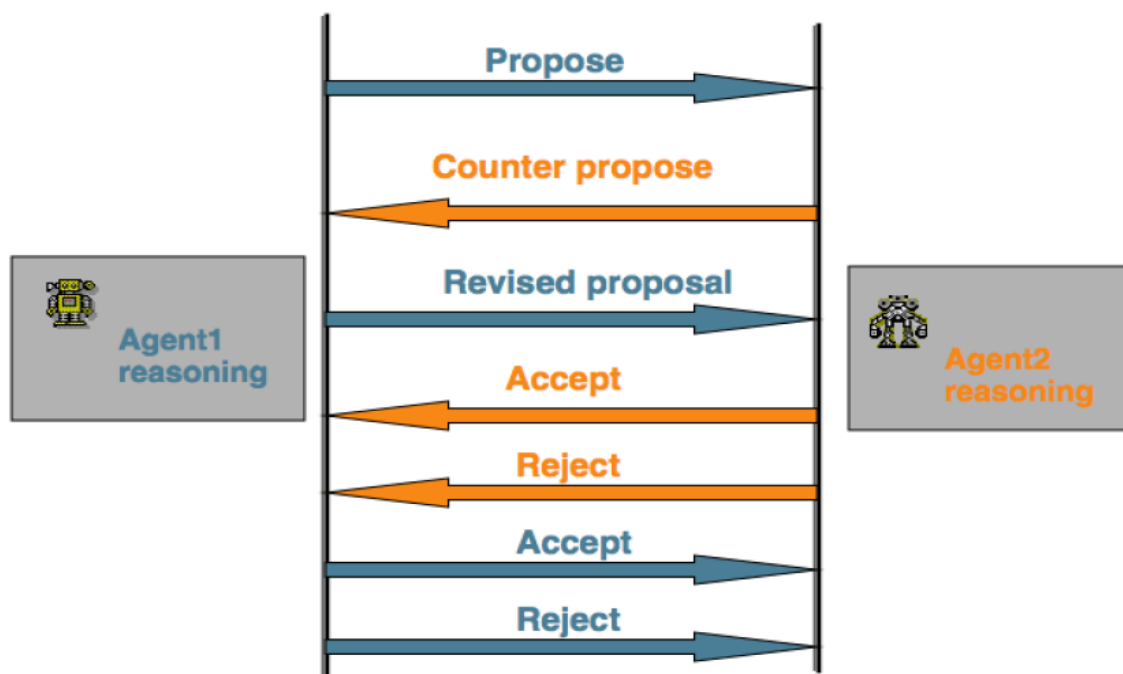


Figure 29 : Séquences du bargaining (Florea 2012)

Bargaining avec médiateur :

Toutefois et dans certaines situations, une partie de la coordination est déléguée à un médiateur, dans le cadre de ce qui est appelé le bargaining avec médiateur. Comme dans le protocole single text mediated (Raiffa, 1982) par exemple décrit par (Amgoud et al. 2007), dans ce protocole le médiateur a simplement pour rôle de faire une proposition initiale, sur laquelle tous les agents doivent se prononcer (accepter ou refuser). Si tous les agents acceptent l'offre courante, l'offre est marquée comme acceptée (mais le protocole ne s'arrête pas pour autant), sinon elle est marquée comme refusée. Puis le médiateur cherche à proposer une nouvelle offre, et le processus se répète pendant un nombre prédéfini de tours. Selon (Amgoud et al. 2007), un des problèmes majeurs de cette approche est le risque de l'injustice entre les agents. Plusieurs travaux cherchent à éliminer les limites de cette approche comme (Aydogan et al. 2014).

En se basant sur les mécanismes de négociation présentés auparavant plusieurs travaux ont couvert la coordination de la chaîne logistique, Selon (D'amours et al 2008) les différentes applications peuvent être positionnées selon les caractéristiques de négociation à savoir, le niveau de coopération et le nombre de participants dans la figure suivante.

| | | Pure adversarial - | | Pure collaborative + |
|------------------------|--------------|--|--|--|
| | | Cooperation level | | |
| General traits | | <ul style="list-style-type: none"> • Only local goals • Short-term relationships • Self-interested strategies | <ul style="list-style-type: none"> • Varied importance of local vs. collective goals • Long-term relationships of independent partners | <ul style="list-style-type: none"> • Only collective goals • Same organization members |
| Number of participants | One-to-one | <ul style="list-style-type: none"> • Opportunistic production scheduling (Fink, 2004) | <ul style="list-style-type: none"> • Collaborative negotiation to determine the order quantity (Homburg et al., 2000) • Collaborative negotiation-based planning (Dudek and Stadler, 2005; Chen et al. (1999) • One-to-one profit sharing (Nagarajan and Sosc, 2008) | <ul style="list-style-type: none"> • Intra-organization production planning |
| | One-to-many | <ul style="list-style-type: none"> • MAGNET (Collins et al. 2002) • Blackboard-based negotiation using open tender (Ito and Salleh, 2000) • Distributed wood procurement – non-collaborative approach (Beaudoin et al. 2007) • Multi-contract selection (Jiao et al. 2006) | <ul style="list-style-type: none"> • Win-win supplier selection model (Khouider et al., 2008) • Negotiation mechanisms for profit sharing (Nagarajan and Sosc, 2008) • ADEPT (Sierra et al., 1997) • Distributed wood procurement – collaborative approach (Beaudoin et al. 2007) • Auction-based approach for production planning (Ertogral and Wu, 2000) • Price negotiation (Homburg and Schneeweiss, 2000) | <ul style="list-style-type: none"> • Intra-organization marketplace |
| | Many-to-many | | <ul style="list-style-type: none"> • Coalition formation (Oliveira and Rocha, 2001; Nagarajan and Bassok, 2002) • Cooperative planning negotiation in construction supply chain (Xue et al. 2007) | |

Figure 30 : Application de la négociation dans la chaîne logistique (D'amours et al. 2008)

2.7. Conclusion

A travers ce chapitre, et après une présentation d'une vue générale sur les systèmes multi-agent, nous avons justifié leur choix pour faire face aux limites du management de la chaîne logistique principalement par rapport au niveau inter-organisationnel. L'accent était mis après sur les mécanismes de négociation à cause de leur importance significative dans un SMA mais aussi pour assurer la coordination de la chaîne logistique.

Dans ce qui suit nous allons passer à la deuxième partie qui concerne le développement de la solution multi agents dans un contexte bien défini de la chaîne logistique. Où nous allons s'appuyer sur les notions développées lors de la première partie.

Partie 2 : Conception du SMA

Chapitre 3 : Présentation de la solution

3.1. Introduction

Après que nous avons vu la pertinence des systèmes multi-agents pour régler les problèmes liés à la chaîne logistique à nos jours, reste maintenant à préciser le problème et élaborer la solution.

L'objet de ce chapitre est de présenter de manière claire et détailler la solution que nous avons construite lors de notre projet. Le chapitre est organisé comme suit :

Dans le premier point qui est la description du problème, nous allons commencer par définir le contexte de notre étude, pour ensuite passer à la définition du problème à travers l'analyse d'une étude intéressante qui a été faite récemment dans le même contexte.

Après et dans la partie solution proposée, nous allons d'abord voir les solutions existantes par une revue de littérature des modèles pertinents dans le même contexte pour bien situer notre solution. Par la suite, nous allons décrire la solution par un diagramme de séquence qui sera détaillé par la suite étape par étape, avant de passer à la modélisation et l'implémentation du système multi agent.

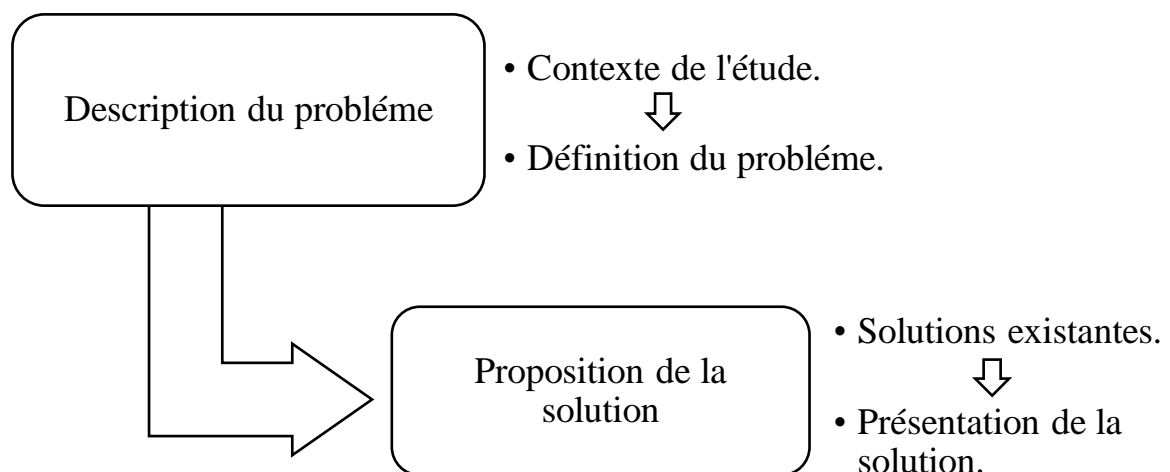


Figure 31: Organisation du chapitre de la présentation de la solution.

3.2. Description du problème

3.2.1. Contexte d'étude

Notre étude se concentre sur la chaîne logistique à deux niveaux, un niveau fournisseur et un niveau distributeur (annexe 1), du point de vue de la gestion d'approvisionnement et de production (figure). Le fournisseur peut être un grossiste qui obtient sa commande d'une source externe ou bien d'un producteur interne avec une capacité de production limitée. Le distributeur reçoit la demande des clients.

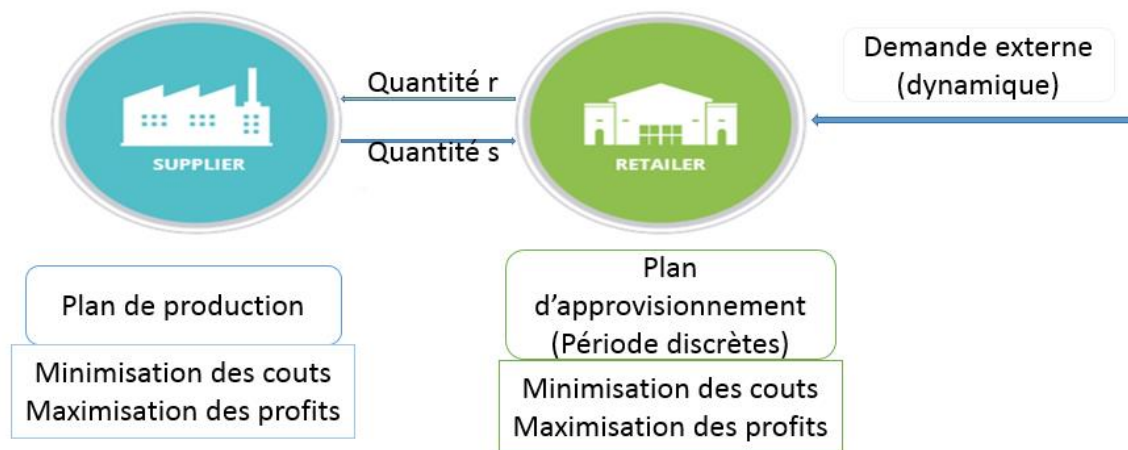


Figure 32 : Problème du lot-sizing entre fournisseur et distributeur

Ce sujet est très étudié dans la littérature et dans les deux cas, centralisé et décentralisé. Centralisé où les acteurs alignent leurs objectifs et les informations sont explicites. Deuxièmement, décentralisé où les objectifs individuels sont pris en compte et l'information est asymétrique (implicite). La majorité des travaux connus de la littérature reposent sur des problèmes d'inventaire (Economic Ordering Quantity EOQ). La gestion du stock dans une seule location consiste en deux décisions : Combien et quand commander. Selon Li et Wang (2007), il y a trois dimensions pour coordonner la chaîne logistique dans ce contexte. Premièrement par rapport à la quantité commandée n'est pas capable d'optimiser la globalité de la chaîne. Deuxièmement, la synchronisation des commandes pour réduire les niveaux de stock. Troisièmement, la qualité de l'information qui peut améliorer les décisions. Le tableau suivant présente des exemples réalisés selon la nature de la demande Li et Wang (2007).

Table 8 : Les modèles de coordination de la chaîne logistique Li et Wang (2007, p-3).

Supply chain coordination models

| Decision Structure | Nature of demand | |
|--------------------|---|--|
| | Deterministic | Stochastic |
| Centralized | <ol style="list-style-type: none"> 1. The classic lot-sizing problem for multi-echelon inventory systems (e.g., Wagner and Whitin, 1958; Zangwill, 1969; Schwarz, 1973) 2. Power-of-two and integer-ratio coordination policies (e.g., Roundy, 1985, 1986) 3. The classic joint replenishment problem (e.g., Goyal, 1973; Silver, 1976; Chakravarty, 1985; Federgruen and Zheng, 1992) | <ol style="list-style-type: none"> 1. Installation policies (e.g., Clark and Scarf, 1962; Deuermeyer and Schwarz, 1981; Svoronos and Zipkin, 1988) 2. Echelon stock policies (e.g., Clark and Scarf, 1960; Chen and Zheng, 1994a,b, 1998) 3. Information sharing models (e.g., Zheng and Zipkin, 1990; Cachon and Fisher, 2000; Moynadeh, 2002) |
| Decentralized | <ol style="list-style-type: none"> 1. Quantity discount models (e.g., Dolan, 1978; Monahan, 1984; Lal and Staelin, 1984; Parlar and Wang, 1994; Wang, 2002) 2. Profit-sharing models (e.g., Goyal, 1976; Banerjee, 1986b; Joglekar and Tharthare, 1990) | <ol style="list-style-type: none"> 1. Supply chain game models (Porteus and Whang, 1991; Cachon and Zipkin, 1999; Caldentey and Wein, 2003) |

3.2.2. Définition du problème

Lors de l'identification du problème, notre point de départ était l'étude récente de Kedad-Sidhoum et al (2015a, b). Trois approches ont été analysées pour minimiser le coût du fournisseur lors de l'étude : une première approche basée sur le partage de coût total, une deuxième approche avec partage des coûts de stockage et enfin une dernière approche sans partage de coût. Les analyses expérimentales ont montré que la deuxième approche est suffisante. Le deuxième volet de l'étude a porté sur la prise en compte de contraintes de capacité de stockage dans une chaîne à deux acteurs, puisque le transfert de coût de stockage implique dans la majorité des cas le stockage des quantités produites selon Kedad-Sidhoum et al (2015b).

Différents cas ont été considérés : le cas où seul le fournisseur possède une capacité limitée (IB_S : inventory bounds in supplier side), le cas où seul le distributeur a une capacité limitée (IB_R), et enfin le cas où les deux acteurs ont des contraintes capacitaires sur les quantités stockées (IB_{SR}). Aussi en tenant compte de la décomposition des lots (NLS : no lot splitting). Des analyses de complexité ont été menées pour ces différents cas de figures présentés dans le tableau.

Table 9 : Les résultats de complexité avec la contrainte NLS (Kedad-Sidhoum et al. 2015b).

| Problem | Complexity |
|---|------------------|
| ULS-NLS | strongly NP-hard |
| 2ULS-IB _R -NLS | strongly NP-hard |
| 2ULS-IB _S -NLS with demand time window | NP-hard [4] |
| 2ULS-IB _S -NLS | strongly NP-hard |
| 2ULS-IB _{SR} -NLS | strongly NP-hard |
| 2ULS-IB _{SR} -NLS with shared inventory | strongly NP-hard |

Deux solutions ont été proposées dans l'étude, une sans transfert d'argent et une autre avec transfert d'argent (Figure).

La première solution proposée par l'étude (sans transfert d'argent) s'inclue dans les modèles d'optimisation locale de la chaîne logistique qui sont à la base des problèmes que nous avons cités dans la première partie comme l'effet de coup de fouet (chapitre 1). La résolution du problème dans ce cas se fait à travers un algorithme de programmation dynamique en $O(T \log T)$ proposé par Wagelmans et al (1992) (Annexe 1).

La deuxième solution concernant le transfert d'argent à travers un algorithme de programmation dynamique en $O(T^3)$ proposé par Zangwill (1969) peut être incluse dans les approches de collaboration et d'intégration de la chaîne logistique dans le cas centralisé. Ce dernier présente plusieurs limites (chapitre 1), mais nous retenons principalement le partage d'information et les aspects comportementaux.

Lors du transfert d'argent entre les acteurs, les contraintes rajoutées (la capacité des lots, les limites des capacités de stockage et la décomposition des lots) rapprochent le modèle de la réalité mais les problèmes deviennent de plus en plus complexes à résoudre avec des algorithmes à base de recherche opérationnelle. Seul le problème de quantité de lot avec transfert d'argent et capacité limitée du côté du distributeur a été résolu dans l'étude. De même que ces cas de figures restent centralisés par le fait que les acteurs agissent pour un objectif

global sans prendre en considération les objectifs individuels principalement conflictuels. Aussi et dans la majorité des cas ces modèles ne représentent pas le comportement des acteurs dans la réalité surtout par rapport à l'autonomie et la proactivité lors de la prise de décision. Enfin l'interaction entre l'ensemble des acteurs n'est pas modélisée, bien qu'elle a une grande importance émanant de la dynamique et l'incertitude de l'environnement.

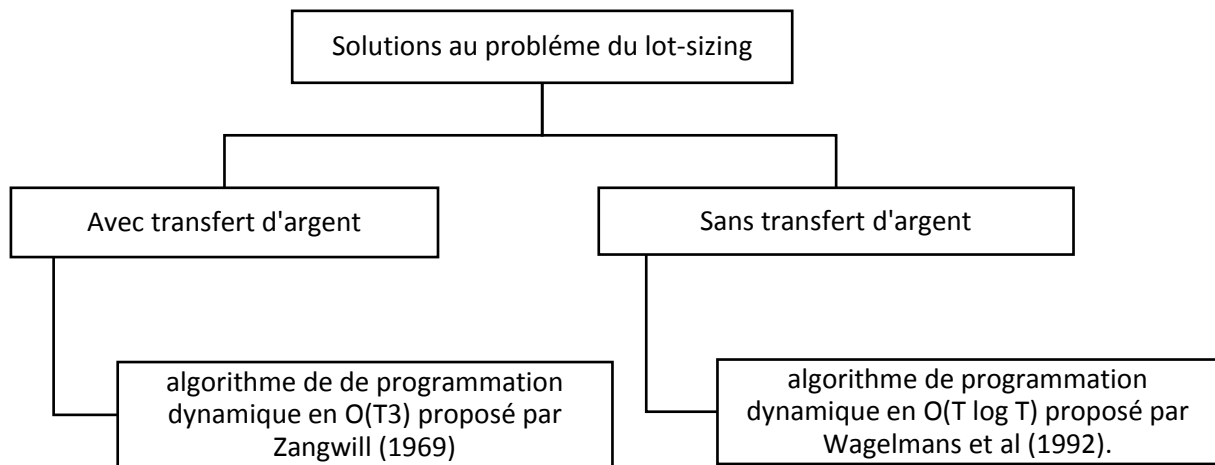


Figure 33 : Les solutions proposées lors de l'étude de Kedad-Sidhoum et al (2015a).

3.3. Solution proposée

3.3.1. Solutions existantes

Notre modèle représente une extension de l'étude précédant pour faire face aux limites de cette dernière. Ceci avec une approche multi agents à travers le bargaining avec médiateur.

Dans ce contexte et pour mettre en évidence nos contributions, nous allons voir seulement une revue de littérature qui est représentative et particulièrement pertinente pour notre étude.

D'abord il faut dire que l'idée d'optimisation conjointe pour l'acheteur (distributeur) et le vendeur (fournisseur) a été initiée par Goyal (1977) et plus tard renforcée par Bannerjee (1986). Ensuite, Weng (1999) a démontré que lorsque les deux parties coordonnent leurs actions, le profit commun augmente et le prix de vente diminue. Aviv (2001) a également étudié l'effet de la planification collaborative sur la performance de la chaîne d'approvisionnement. Schneeweiss et Zimmer (2004) ont présenté un mécanisme de coordination au sein d'une chaîne d'approvisionnement avec une information locale privée. Cette étude a donnée des recommandations pour concevoir les contrats de la chaîne logistique.

Mais cependant, les contrats de la chaîne logistique présentent plusieurs limites comme nous avons cité lors du premier chapitre, surtout par rapport à la centralisation lié principalement au partage d'informations privées. Bien que dans la majorité des cas l'implémentation des contrats relèvent de la théorie des jeux comme Aryanezhad et al (2009), Nagarajan et Susic (2006)... ; mais ces modèles et malgré la pertinence de leur résultats, limitent toujours l'autonomie des acteurs, comme par exemple lors du transfert d'utilité entre les acteurs.

Pour cela pas mal de travaux à nos jours essaient d'assurer la décentralisation lors de la coordination, nous pouvons citer Chu et Leon (2015) qui a fait en sorte de garder la fonction d'utilité ainsi que tous les paramètres privés à l'acteur. Une bonne partie des solutions qui existent dans la littérature dans ce sens relèvent des SMAs et surtout pour la chaîne logistique électronique avec des systèmes de négociations automatiques basées souvent sur le bargaining (Mohebbi et Shafaei (2010), Sucky (2003)...).

Parmi ces travaux, ils existent certains à base des contrats de chaîne logistique comme le revenue sharing (Hou et al (2008)), wholesile price (Banerjee et al (2013)) et le buyback contract (Guangxing et al (2013), Lin et al (2013), etc.). Le même problème que nous allons traiter a été étudié par Guangxing et al (2013) à travers le Nash-bargain, donc avec formulation des axiomes.

Notre solution consiste à décentraliser l'un des contrats les plus étudiés dans la littérature selon Snyder et Shen (2012) qui est le buyback contract et dans le but d'assurer la coordination nous s'appuyons sur le contrat revenue sharing (contrat hybride entre le buyback et le revenue sharing), ceci à travers le bargaining avec médiateur par les systèmes multi agents.

3.3.2. Description de la solution

De nos jours l'environnement dans lequel opèrent les différents acteurs de la chaîne logistique devient de plus en plus dynamique, aussi bien qu'il est doté d'une grande incertitude. Devant ces facteurs les solutions rigides ont prouvé leurs limites, alors la flexibilité est devenue primordiale. Donc nous avons proposé un modèle qui peut se rajouter à l'étude précédente comme une autre alternative pour faire face aux limites de cette dernière dans le cadre de la coordination de la chaîne logistique. Donc le modèle général regroupe plusieurs solutions pour assurer la proactivité du système multi-agents qui sera mené à choisir la meilleure solution par rapport à chaque situation.

Puisque le cas de l'optimisation locale ainsi que celui avec partage de coût de stockage ont été présenté. Notre solution consiste d'abord à introduire le buyback contract (centralisé) comme un moyen de coordination. Une alternative et qui sera implémentée par la suite, correspond à la décentralisation totale du buyback contract ainsi que le revenue sharing, à travers le bargaining avec médiateur de façon à trouver une solution sans partage des coûts mais aussi des utilités.

Dans ce cas le modèle peut satisfaire les différentes situations qui peuvent exister (figure). Dans le cas où les acteurs ne veulent pas coordonner, l'optimisation est locale et convient à résoudre un problème de lot sizing (Kedad-Sidhoum et al 2015a) (Annexe 1). Sinon si les acteurs veulent coordonner, deux alternatives sont possibles. Soit solution centralisé à travers le partage des coûts de stockage (Kedad-Sidhoum et al 2015a, b) ou bien avec un des contrats de la chaîne

logistique qui est le buyback contract. Ou soit décentralisé à travers le bargaining avec médiateur.

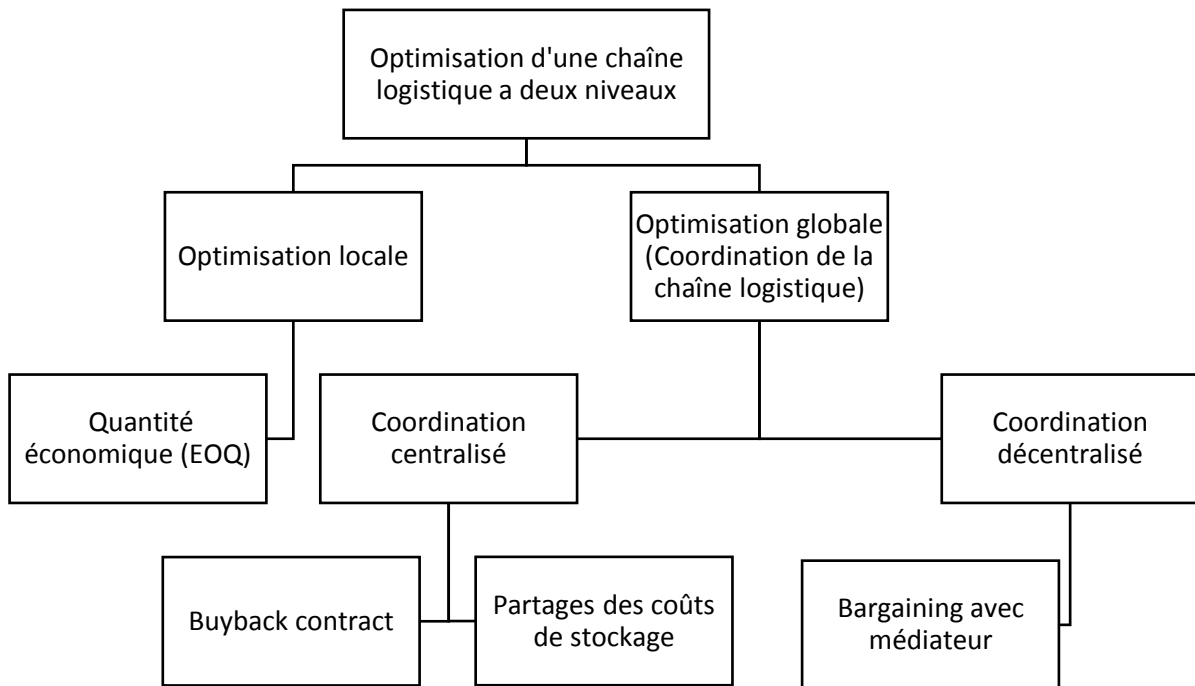


Figure 34 : Les types d'optimisation d'une chaîne logistique à deux niveaux

Le système multi-agents que nous allons introduire par la suite inclut une solution à base d'optimisation locale qui correspond à la détermination de la quantité économique, aussi bien que la solution d'optimisation globale avec la coordination décentralisée de la chaîne logistique à travers le bargaining avec médiateur.

Donc la solution multi-agent implique trois acteurs : "un fournisseur", "un détaillant" et "un médiateur". Le diagramme de séquence ci-dessous illustre la séquence d'exécution.

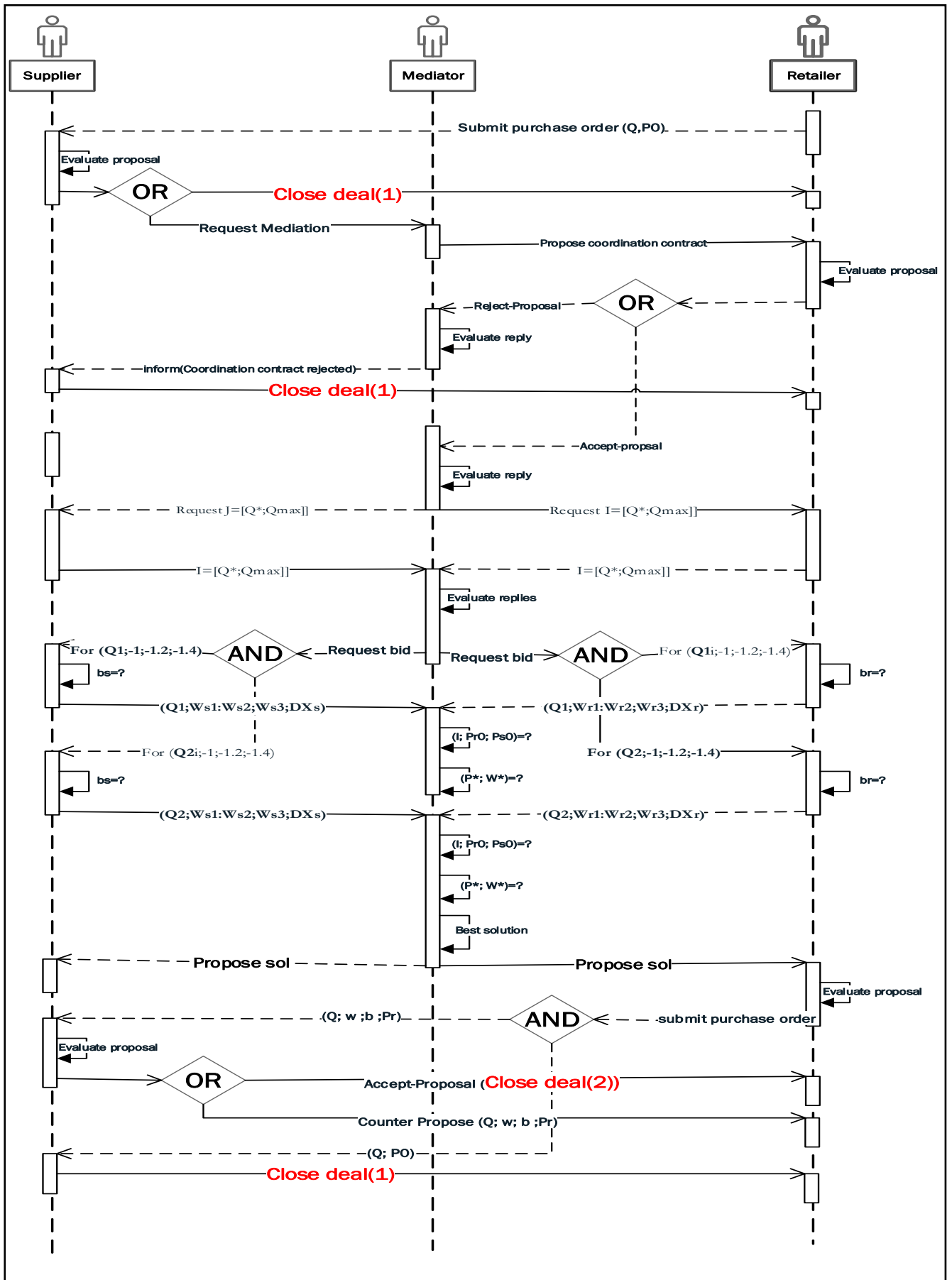


Figure 35 : Diagramme de séquence

Description du diagramme de séquence :

Nous allons dans ce qui suit décrire le diagramme de séquence. Puisque il y a un grand nombre de séquences nous allons les répartir en quatre étapes essentielles (Figure). La première étape concerne la solution sans bargaining donc l'optimisation locale où chaque acteur optimise indépendamment de l'autre (décentralisé). La deuxième est le début du bargaining dans laquelle les paramètres du contrat sont annoncés aux acteurs. La troisième correspond au cœur du bargaining, là où la solution est construite par le médiateur à travers la séquence des propositions des deux acteurs (supplier, retailer). Enfin la quatrième étape présente le choix des acteurs après l'évaluation de la solution du médiateur.

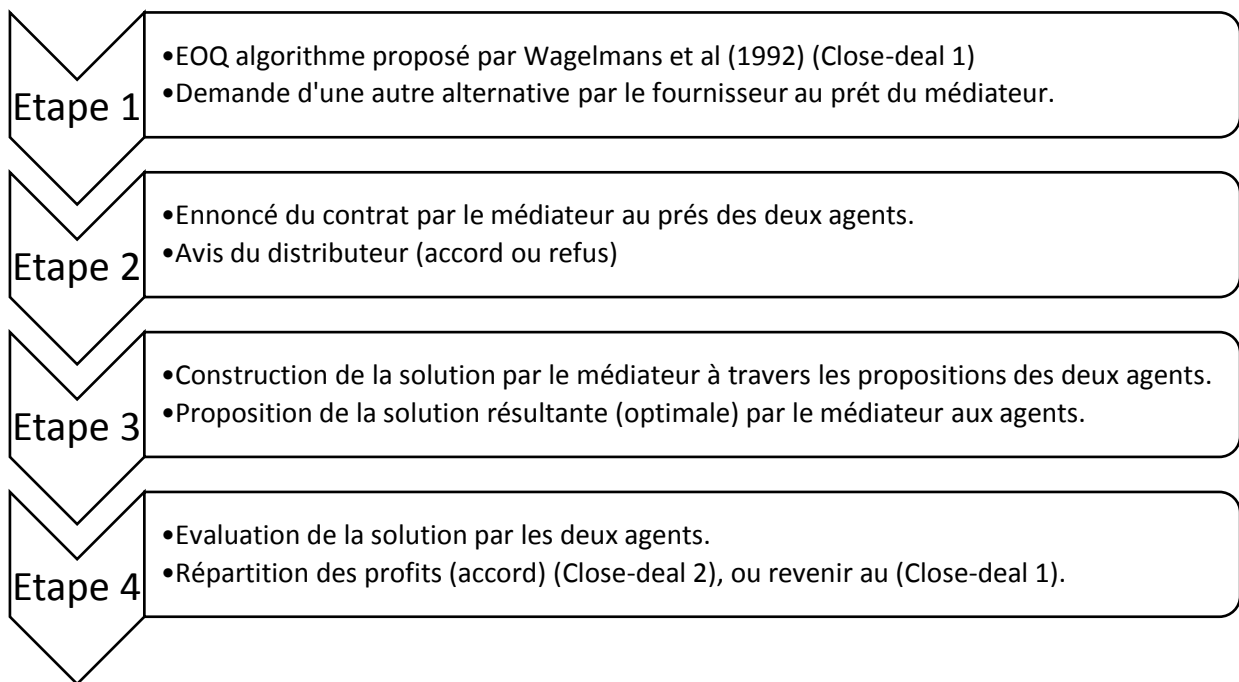


Figure 36 : Les étapes de la solution

Etape 1 : Solution sans bargaining :

a) Soumission de l'ordre d'achat :

Notre solution commence là où le distributeur (retailer) soumet un ordre d'achat au près du fournisseur (supplier) d'une quantité Q , ceci avec la résolution d'un problème de lot-sizing. Les problèmes de lot-sizing considérés dans notre étude sont à capacité infinie. Il existe plusieurs solutions dans la littérature, un des algorithmes de programmation dynamique les plus performants est celui de Wagelmans et Hoesel (1992) avec une complexité $O(T \log T)$ pour résoudre le problème du lot-sizing à capacité infinie.

b) Evaluation de la proposition :

Après et une fois que le fournisseur reçoit la proposition du distributeur, il l'évalue par rapport à la quantité ainsi que son profit. Puisque dans la majorité des cas la quantité commandée par le distributeur est inférieure à celle souhaitée par le fournisseur, mais il y a des cas où le fournisseur reçoit plusieurs propositions de plusieurs distributeur d'un même produit par la formation des coalitions par exemple, dans une même période, et donc la quantité peut être acceptable.

c) Acceptation de la proposition :

Si le fournisseur accepte la proposition, close-deal (1) : résolution d'un problème de lot-sizing par rapport à la quantité Q et définition du prix de vente P_{sup} .

d) Refus et recours au médiateur :

Sinon, le fournisseur fait recours au prêt du médiateur pour qu'il propose une autre alternative au distributeur dans le but d'inciter le distributeur d'augmenter la quantité, et nous passons à l'étape 2.

Etape 2 : Le début du bargaining :

a) Proposition du médiateur :

Le médiateur propose le contrat de négociation (le buy-back décentralisé) au distributeur :

Buyback décentralisé :

Dans ce contrat, « le distributeur » achète le produit à w par unité et pour chaque perte (pénurie de stock ou excès non vendu) « le fournisseur » le rembourse à b par unité. Notons que pour ce type de contrat, l'utilité du « fournisseur » est croissante tant que l'utilité du « distributeur » est décroissante en fonction du paramètre w .

Le contrat buy-back proposé par Pasternack (1985) est centralisé, puisque les étapes du contrat exigent :

Le partage des paramètres entre les acteurs : lors du calcul des fonctions de transfert par les acteurs, ils doivent connaître les paramètres du contrat qui sont exprimé en fonctions de leurs paramètres et donc ils doivent partager leurs paramètres, du coup ils ne sont plus privés.

$$w(b) = b + c_s - (c - v) \frac{b + p_s}{r - v + p}$$

Le partage des utilités : aussi et pour que le médiateur ressort l'optimum, les utilités doivent être partagées, donc aussi les utilités ne sont plus privés.

D'abord l'intérêt d'introduire le buyback est pour que le médiateur peut deviner les utilités des deux acteurs afin qu'il puisse déterminer la solution optimal de la chaine logistique. Nous avons fait des modifications de telle sorte que le contrat soit décentralisé. Nous avons proposé une fonction de couplage entre b et w , qui respecte les exigences liés à la relation entre les deux paramètres, pour jouer sur un seul paramètre lors du bargaining et pour assurer la décentralisation du contrat : $w = f(b)$.

La fonction de couplage malgré qu'elle ne permette pas d'assurer la coordination entre les deux acteurs, Elle doit obéir aux critères qui suivent :

$$C0 : w(b) > 0 \forall b > 0$$

Ceci est bien évident, pour éviter une fonction défini sur un intervalle comprenant une partie négatif ou ayant des images négatifs non acceptables car une valeur négative pour le ' b ' à titre d'exemple signifie que c'est le distributeur qui va payer pour les ventes perdues et non le fournisseur.

$$C1 : \frac{d\pi_r}{db} \times \frac{d\pi_s}{db} < 0$$

Ceci peut poser la question pourquoi ? Alors qu'il fallait opter pour une fonction assurant une croissance simultanée des deux fonctions d'utilités. Mais, c'est mathématiquement impossible d'avoir les deux fonctions ayant le même sens de variation sous un contrat de type Buyback.

En effet,

$$(C1) \quad \Leftrightarrow \left(\frac{dT}{db}\right) \times \left(-\frac{dT}{db}\right) < 0 \quad \Leftrightarrow \quad -\left(\frac{dT}{db}\right)^2 < 0 \quad \Leftrightarrow \quad \left(\frac{dT}{db}\right)^2 > 0$$

$$\frac{dT}{db} > 0 \quad \Leftrightarrow \quad -Q \times \frac{dw}{db} + I > 0 \quad \Leftrightarrow \quad \frac{dw}{db} < \frac{I}{Q} \quad (1)$$

$$\frac{dT}{db} < 0 \quad \Leftrightarrow \quad \frac{dw}{db} > \frac{I}{Q} \quad (2)$$

Donc on a le choix entre une fonction croissante vérifiant de plus $\frac{dw}{db} > \frac{I}{Q}$ ou plus simplement une fonction décroissante. Optons alors pour une fonction décroissante ce qui correspond à π_r croissante et π_s décroissante.

$$C2 : \exists (b_r, b_s) \in \mathbb{R}^2 > 0 \text{ tel que: } \pi_r(b_r) = \pi_s(b_s) = 0$$

C'est-à-dire l'existence d'un seuil critique de profitabilité qui est également le « bid » que devra proposer chaque acteur au médiateur lors du bargaining.

Ce raisonnement fait émerger intuitivement deux fonctions candidates :

Exponentielle de la forme : $w(b) = \beta e^{\gamma b}$ avec : $\beta > 0$ et $\gamma < 0$

Ou bien inverse de la forme : $w(b) = \frac{\beta}{\gamma + b}$ avec : $\beta > 0$ et $\gamma > 0$.

Les deux fonctions vérifient le critère C2. En effet :

Cas où $w(b) = \frac{\beta}{\gamma + b}$:

$$\begin{cases} \pi_s(b_s) = 0 \\ \pi_r(b_r) = 0 \end{cases}$$

$$\Leftrightarrow \left\{ \begin{array}{l} -p(Q) - b \times I + Q \times \frac{\beta}{\gamma + b} = 0 \quad \text{*** (1)} \\ -h(Q) - o(Q) + r \times S + |Q - S|\vartheta + b \times I - Q \times \frac{\beta}{\gamma + b} = 0 \quad \text{*** (2)} \end{array} \right.$$

$$\lim_{b \rightarrow +\infty} \pi_s(b) \times \lim_{b \rightarrow -\infty} \pi_s(b) = -\infty \times (+\infty) = -\infty < 0$$

$$\lim_{b \rightarrow +\infty} \pi_r(b) \times \lim_{b \rightarrow -\infty} \pi_r(b) = +\infty \times (-\infty) = -\infty < 0$$

Cas où $w(b) = \beta e^{\gamma b}$:

$$\begin{cases} \pi_s(b_s) = 0 \\ \pi_r(b_r) = 0 \end{cases}$$

$$\leftrightarrow \begin{cases} -p(Q) - b \times I + Q \times \beta e^{\gamma b} = 0 & \text{*** (3)} \\ -h(Q) - o(Q) + r \times S + |Q - S|\vartheta + b \times I - Q \times \beta e^{\gamma b} = 0 & \text{*** (4)} \end{cases}$$

$$\lim_{b \rightarrow +\infty} \pi_s(b) \times \lim_{b \rightarrow -\infty} \pi_s(b) = -\infty \times (+\infty) = -\infty < 0$$

$$\lim_{b \rightarrow +\infty} \pi_r(b) \times \lim_{b \rightarrow -\infty} \pi_r(b) = +\infty \times (-\infty) = -\infty < 0$$

Donc d'après le théorème de valeurs intermédiaires les équations (1), (2), (3) et (4) admettent tous une solution dans R . Dès ce moment on peut voir que si on optait pour une fonction croissante on ne pourra pas valider ce critère. Car dans ce cas-là :

$$\lim_{b \rightarrow \pm\infty} w(b) = \pm \infty \neq l \in R.$$

Les deux fonctions ont également survie le test, donc choisissons l'exponentielle et posons $\beta = 2P_0$ pour que $w \in]0; 2P_0]$

Convergence du bargaining :

On dit qu'un bargaining est convergent si est seulement s'il permet de tendre vers une solution valide, c à d il vérifie le critère C3 suivant :

$$C3 : \exists b^{opt} > 0 \text{ tel que: } \pi_r(b^{opt}) = \pi_s(b^{opt}) > 0$$

$$\begin{aligned} & \left[\lim_{b \rightarrow 0} [\pi_r(b) - \pi_s(b)] \right] \times \left[\lim_{b \rightarrow +\infty} [\pi_r(b) - \pi_s(b)] \right] \\ & = [-h(Q) - o(Q) + r \times S + |Q - S|\vartheta - Q\beta - (-p(Q) - Q\beta)] \times [+ \infty] \end{aligned}$$

On peut déduire donc, Le bargaining n'est pas convergent si : $\pi_r(b = 0) > \pi_s(b = 0)$.

Par contre, si $\pi_r(b = 0) > \pi_s(b = 0)$ cela signifie l'existence de b^{opt} uniquement et puisqu'elle ne peut pas être calculé que par une résolution numérique. On ne peut pas démontrer si $\pi_r(b^{opt}) > 0$

Mais, pour répondre à cette question il fallait passer par une simulation graphique générant huit scénarios possibles. Voici dans ce qui suit les graphiques illustrant tous les combinaisons de b_s et b_r possibles dans un point Q donné du bargaining.

Cas 1.1 : $0 < b_r < b_s$

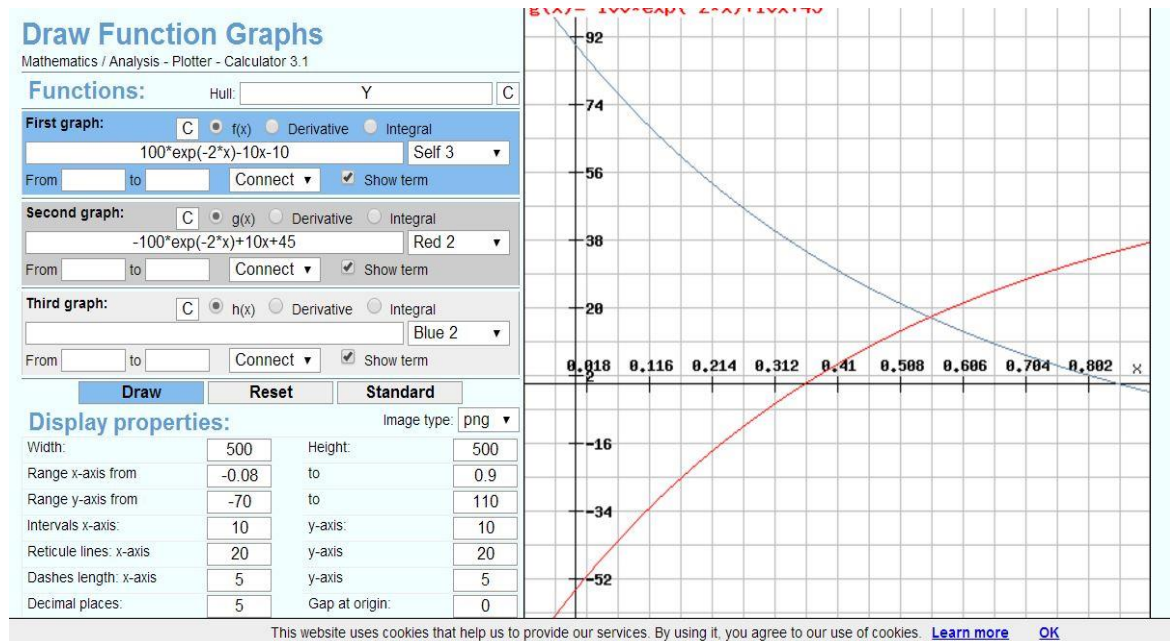


Figure 37: Simulation graphique cas 1.1

Cas 1.2 : $0 < b_s < b_r$

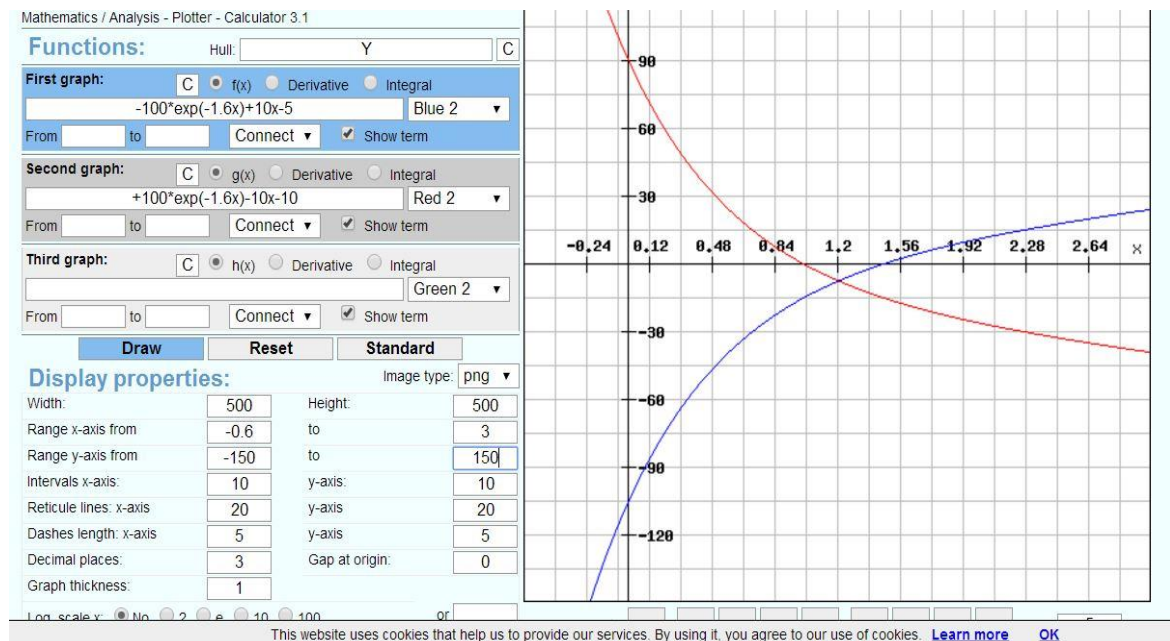


Figure 38: Simulation graphique cas 1.2

Cas 2.1 : $b_s > 0$ et $b_r < 0$

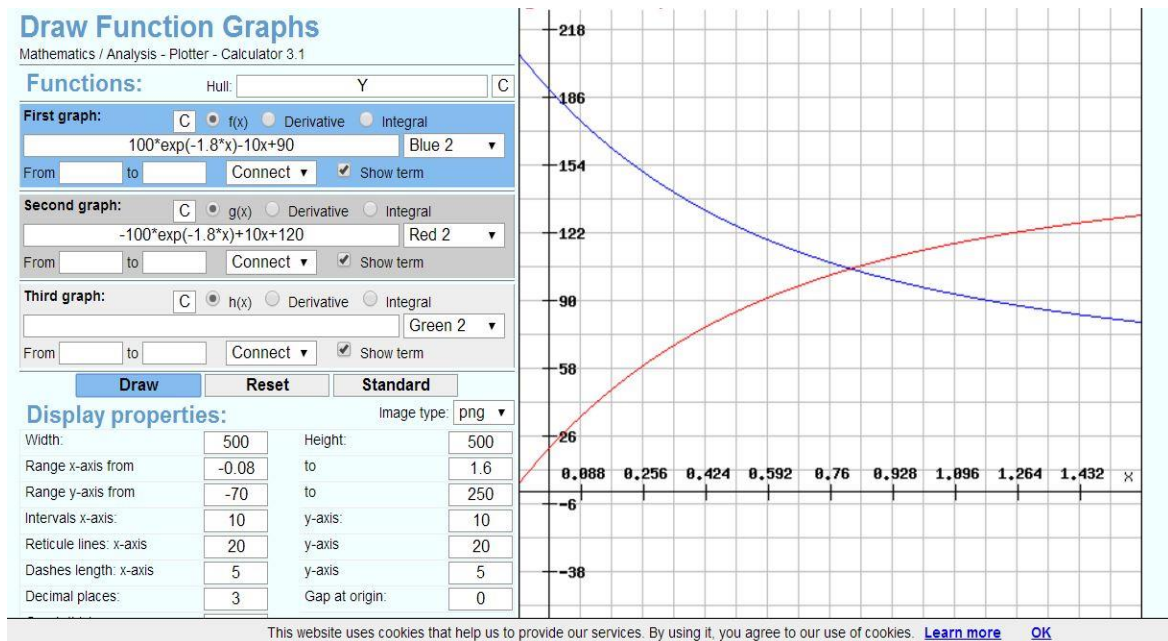


Figure 39: Simulation graphique cas 2.1

Cas 2.2 : $b_s < 0$ et $b_r > 0$

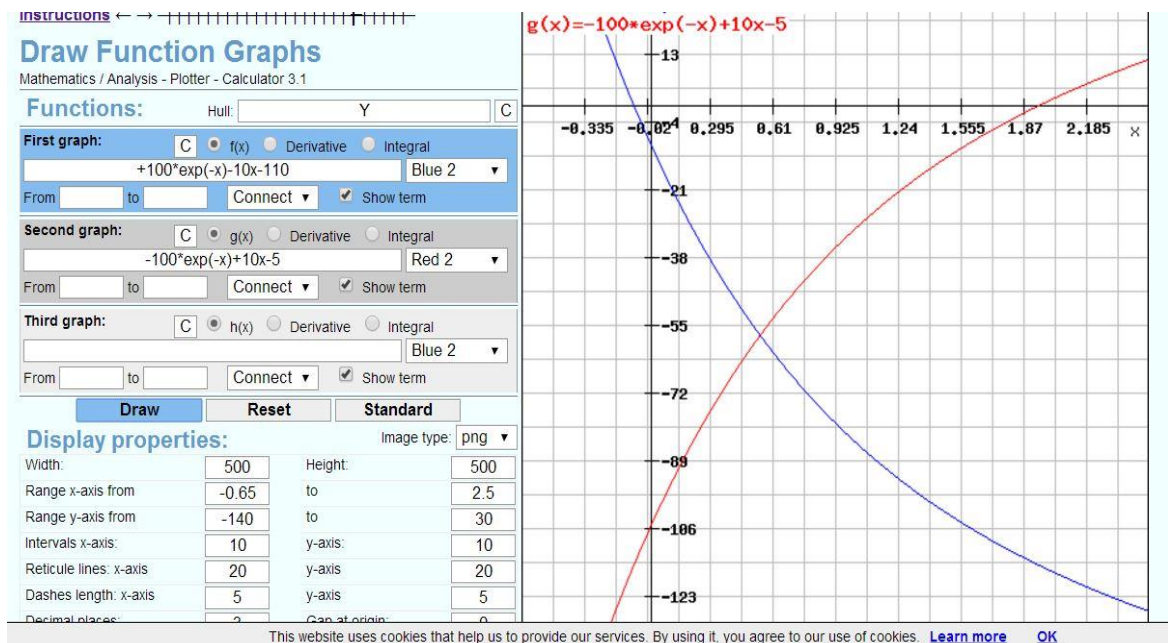


Figure 40: Simulation graphique cas 2.2

Cas 3.1 : $b_s > 0$ et $b_r < 0$

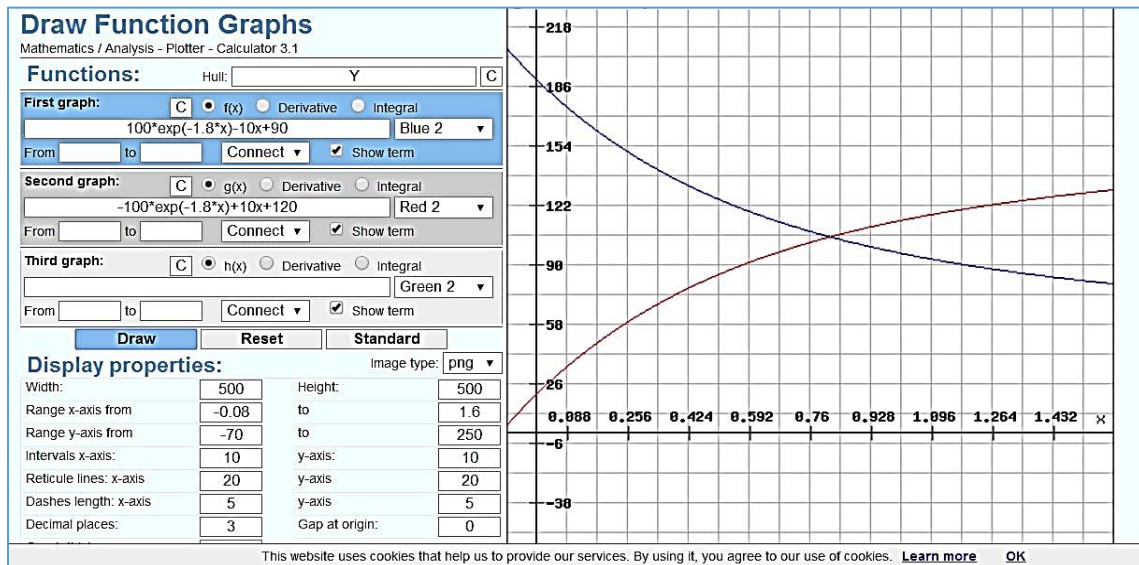


Figure 41: Simulation graphique cas 3.1

Cas 3.2 : $b_s < 0$ et $b_r > 0$

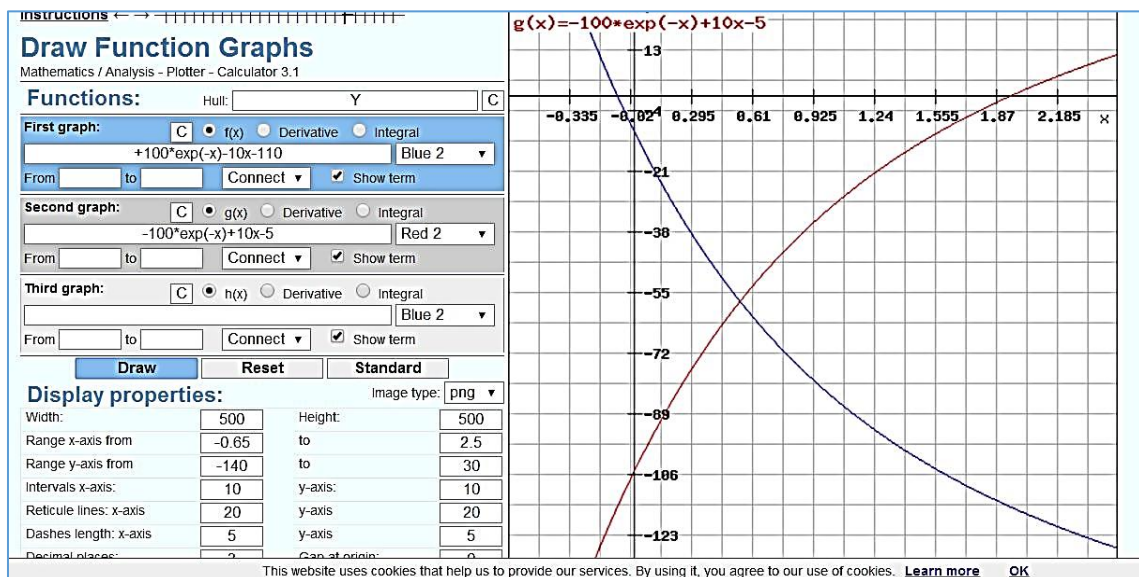


Figure 42: Simulation graphique cas 3.2

Cas 4.1: $b_r < b_s < 0$

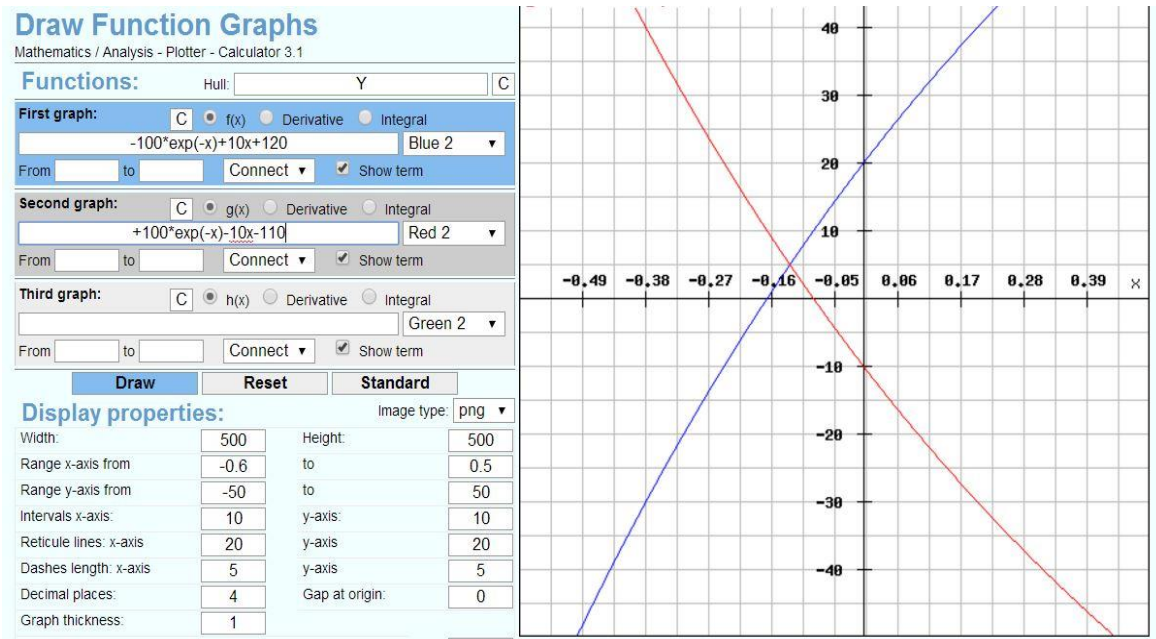


Figure 43: Simulation graphique cas 4.1

Cas #4.2 : $b_s < b_r < 0$

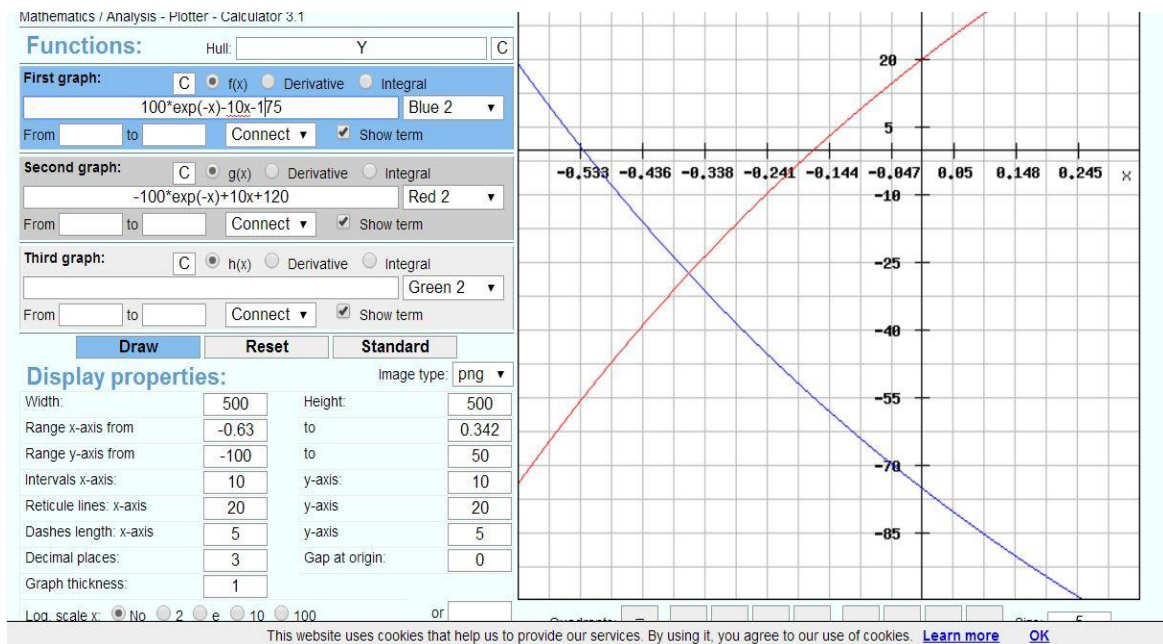


Figure 44: Simulation graphique cas 4.2

D'après cette illustration on peut résumer les huit cas en quatre scénarios en exprimant la situation terme de $\pi_s(b = 0)$ et $\pi_r(b = 0)$ plutôt qu'en fonction de b_s et de b_r :

Calcul de $\pi_s(b = 0)$:

$\pi_s(b = 0) = -p(Q) + \beta Q = -p(Q) + P_0Q + (\beta - P_0)Q$ Où $-p(Q) + P_0Q$ désigne le profit du fournisseur avant de rentrer dans le contrat. Et puisqu'il est positif, alors,

$$\pi_s(0) > 0 \forall Q \in N^* \dots (R1)$$

Calcul de $\pi_r(b = 0)$:

$$\begin{aligned} \pi_r(b = 0) &= -h(Q) - o(Q) + r \times S + |Q - S|\vartheta - \beta Q \\ &= -h(Q) - o(Q) + r \times S + |Q - S|\vartheta - P_0Q + (P_0 - \beta)Q \end{aligned}$$

Le signe de $\pi_r(b = 0)$ par contre, peut être positif ou négatif car $P_0 - \beta < 0 \dots (R2)$

L'interprétation de ce calcul est que le fournisseur est déjà rentable lorsqu'il vend à un prix P_0 . donc il sera plus rentable pour un prix $w(b = 0) = 2P_0$.

D'autre côté, le distributeur est rentable lorsqu'il approvisionne à un prix P_0 mais il se peut qu'il perd lorsqu'il achète à $w(b = 0) = 2P_0$.

Grace à R1 et R2 on pourra restreindre l'ensemble de scénarios possible à 4.

Table 10 : Les scénarios possibles par rapport aux bids.

| Cas N° | Retailer | Supplier | Scénario est il possible D'après (R1 et R2)? | C3 est-il vérifié? |
|------------|----------------|----------------|--|------------------------------|
| 1.1 et 1.2 | $\pi_r(0) < 0$ | $\pi_s(0) > 0$ | Oui | Oui pour 1.1 et Non pour 1.2 |
| 2.1 et 3.1 | $\pi_r(0) > 0$ | $\pi_s(0) > 0$ | Oui | Oui pour 2.1 et 3.1 |
| 2.2 et 3.2 | $\pi_r(0) < 0$ | $\pi_s(0) < 0$ | Non | / |
| 4.1 et 4.2 | $\pi_r(0) > 0$ | $\pi_s(0) < 0$ | Non | / |

De plus, le seul cas où le bargaining est impossible est le cas du scénario 1.2 là où $b_s < b_r$ contrairement à son dual 1.1 vérifiant $b_r < b_s$. Cette dernière est la ligne manquante dans le critère C3 qui devient donc :

$$\exists b^{opt} > 0 \text{ tel que: } \pi_r(b^{opt}) = \pi_s(b^{opt}) > 0 \text{ et } b_r < b_s$$

Enfin, il faut avouer que nous pouvons jamais aboutir à ces résultats sans passer pas la simulation graphique c'est pour cette raison nous voulions plutôt que d'énoncer directement la preuve de ré-exécuter la séquence du raisonnement avec le lecteur tel qu'elle est suivie.

Après avoir développé les critères précédents et mener les démonstrations nécessaires la fonction $W = 2 \times P_{sup} \times e^{\lambda b}$ satisfait parallèlement les 4 critères et est donc la fonction choisie avec $b \in]0, P^{sup}]$.

b) Evaluation de la proposition du médiateur par le distributeur :

Une fois la proposition est soumise par le médiateur, le distributeur fait une évaluation du contrat :

La fonction d'utilité du distributeur avant le contrat est la suivante :

$$\pi_r^0(Q, p^{sup}) = \pi_r(Q, p^{sup}, w = 0, b = 0) = \pi_r(P^{sup}, Q) - Q \times 0 + 0I(b)$$

La nouvelle fonction d'utilité du distributeur lors du contrat est :

$$\pi_r(Q, w_{min}^{ret}, b_{max}^{ret}) = \pi_r(P^{sup}, Q) - Q \times (w - P^{sup}) + bI(Q)$$

Le paramètre de réserve correspond à la valeur de b (resp w) pour laquelle la fonction de l'acteur après ça rentré au contrat est supérieure à celle d'avant. Donc l'évaluation consiste à prouver l'existence du paramètre de réserve.

$$\text{Nous avons : } \pi_r(Q, w_{min}^{ret}, b_{max}^{ret}) - \pi_r^0(Q, p^{sup}) = -Q \times (w - P^{sup}) + bI(Q)$$

Donc $F_r(b) = -Q \times (w - P^{sup}) + bI(Q)$ qui est la fonction de transfert du distributeur doit être supérieur ou égale à zéro avec un $b < P_{sup}$.

Alors, il elle doit vérifier les hypothèses suivantes, pour que le 'b' existe (resp W) :

- F est continu sur $[b_1, b_2]$.
- F est strictement monotone sur $[b_1, b_2]$.
- $F(b_1) \cdot F(b_2) < 0$. (théorème de la valeur intermédiaire).
Pour $b_1 = 0, b_2 = P_{sup}$

c) Refus de la proposition par le distributeur :

Après l'évaluation et dans le cas où il n'existe pas un b^* qui satisfait le distributeur, ce dernier rejette la proposition, après le médiateur informe le fournisseur que la coordination est impossible, les deux acteurs reviennent au Close-deal (1).

d) Accord du distributeur :

Dans l'autre cas, le distributeur envoie son accord au médiateur, qui lui de son côté demande au près des deux acteurs l'intervalle des quantités possibles.

Etape 3 : Le bargaining :

a) Elaboration des intervalles des quantités :

Le distributeur annonce la quantité qui peut commander et qui correspond au maximum qui peut stocker dans la période. Le fournisseur de son côté annonce la quantité commander par le distributeur au début.

b) Demande des offres des deux acteurs par le médiateur :

Si la quantité du distributeur est supérieure à celle du fournisseur, le médiateur segmente l'intervalle de l'intersection à n quantités discrètes.

Pour chaque quantité il demande les paramètres pour deux valeurs différentes de λ .

c) Recherche des paramètres par les deux acteurs :

D'abord il faut définir la nouvelle fonction d'utilité du fournisseur puisque celle du distributeur a été déjà définie à l'étape 2.

La fonction d'utilité du fournisseur avant le contrat est la suivante :

$$\pi_s^0(Q, p^{sup}) = \pi_s(Q, p^{sup}, w = 0, b = 0) = \pi_s(p^{sup}, Q) + Q \times 0 - 0I(b)$$

La nouvelle fonction d'utilité du fournisseur lors du contrat est :

$$\pi_s(Q, w_{max}^{sup}, b_{min}^{sup}) = \pi_s(p^{sup}, Q) + Q \times (w - p^{sup}) - bI(Q)$$

Donc $F_s(b) = Q \times (w - p^{sup}) - bI(Q)$ est la fonction de transfert du fournisseur.

Puisque les deux fonctions sont de la forme, $F(x) = a \times (x) + b \times e^{c \times (x)}$

avec $a, b, et c$ des constantes $\in R$.

La résolution des équations $F_s(b) = 0$ et $F_r(b) = 0$, pour La détermination des paramètres pour les deux acteurs pour une quantité donnée et pour chaque lambda, convient à résoudre une équation par une approximation des racines avec la méthode de dichotomie. Puisque dans notre cas, nous pouvons seulement localiser les solutions, et en calculer des valeurs numériques approchées.

Cette méthode consiste en une succession de divisions par deux de l'intervalle pour approcher de plus en plus de la racine de l'équation $f(x)=0$, jusqu'à ce qu'une précision ϵ soit atteinte.

La méthode de dichotomie a l'avantage d'exiger peu hypothèses sur la fonction.

Elle sert parfois de moyen de calcul d'une initialisation pour les algorithmes des autres méthodes.

Pour cela elle est très utilisée pour les solutions informatisées malgré son inconvénient majeur qui est la lenteur de convergence de son algorithme.

Méthode de la dichotomie :

Début

Saisir $b1, b2, e // e$: degré d'approximation.

Tant que $b2 - b1 \geq e$

m prend la valeur $(b1+b2)/2$

Si $f(b1) \times f(m) \leq 0$

Alors $b2$ prend la valeur m

Sinon b1 prend la valeur m

Fin Si

Fin Tant que

$b^*=b_2$ (distributeur), b_1 (fournisseur).

Afficher b^*

Fin

d) Détermination de la solution optimale par le médiateur :

Une fois que le médiateur reçoit une offre des deux acteurs (W_{r1} ; W_{r2}) and (W_{s1} ; W_{s2}) pour une quantité donnée Q_n , il détermine la meilleure solution (w^* , P^*) à travers un mécanisme par lequel le médiateur et avec les deux propositions de chaque acteur pour chaque λ_1 et λ_2 , d'où l'intérêt de demander deux offres pour chaque quantité.

De plus, nous avons trouvé que pour une quantité donnée : $P_r(w, b) + P_s(w, b) = \text{constante}$.
 Pour toute valeur de b (ou W). Ainsi, le médiateur proposera directement W_r ou W_s avec
 $P^* = (P_r + P_s) / 2$ (à savoir qu'ils doivent partager les bénéfices).

Donc le médiateur enregistre la meilleure solution pour chaque quantité, pour choisir la solution optimale entre eux qui correspond à une quantité Q^* pour laquelle : $P^{\text{opt}} \geq P^* \forall Q_n$.

Pour la solution qui sera proposée, nous incluons le deuxième contrat qui est le revenue sharing pour former le contrat hybride qui va assurer la coordination. Ceci avec la détermination du profit qui sera partagé entre les deux acteurs et qui s'exprime de la façon suivante :

$$\pi_{\text{transf}} = \pi_r(Q^*, w^*, b^*) - P^*$$

Enfin il propose la solution au distributeur contenant la quantité optimale Q^* , le prix optimale w^* , le profit optimale P^* et le profit a envoyé ou a demandé au fournisseur (revenue sharing) π_{transf} .

Etape 3 : la fin du bargaining :

a) Evaluation de la solution par le distributeur :

Une fois que le distributeur reçoit la solution proposée par le médiateur, il évalue encore une fois la proposition. Pour ce faire il compare le P^* à son profit avant le contrat $\pi_r^0(Q, p^{\text{sup}})$:
 Si, $P^* \leq \pi_r^0(Q, p^{\text{sup}})$, ce dernier rejette la nouvelle proposition, et revient au Close-deal (1).
 Sinon, il calcule son nouveau profit $\pi_r(Q^*, w^*, b^*)$, et soumet son offre au fournisseur.

b) Evaluation de la solution par le fournisseur :

Lors de la réception de l'offre du distributeur, le fournisseur évalue de son côté la proposition par :

Si, $P^* \leq \pi_s^0(Q, p^{\text{sup}})$, ce dernier rejette la nouvelle proposition, et revient au Close-deal (1).
 Sinon, il accepte la proposition et donc les acteurs passent au Close-deal (2).

3.4. Conclusion

Ce chapitre nous a permis de bien voir notre apport par rapport au contexte de l'étude. Nous avons pu voir une description détaillée de la solution que nous avons proposée, avec le développement et les formules mathématiques correspondantes. Reste maintenant dans ce qui suit l'élaboration du modèle suivant une méthodologie ainsi que l'implémentation que nous allons les voir dans les chapitres suivants.

Chapitre 4 : Modélisation et implémentation

4.1. Introduction

La modélisation est l'étape la plus importante pour tout projet de développement de systèmes complexes rendant la tâche du codeur une simple tâche d'exécution.

La modélisation de notre système est faite suivant la méthodologie proposée dont le squelette est présenté dans la figure ci-dessous. Mais pour des raisons de simplicité et afin de ne pas alourdir le travail et se focaliser sur son cœur cette section ne va pas décrire cette méthodologie en détail qui a pris une partie aussi importante de notre projet et sera consacrée à présenter uniquement les modèles essentiels de manière brève.

Après et une fois le modèle préparé, la tâche de l'implémentation devient quasi-automatique. Le modèle est donc le guide principale du développeur lors de cette étape déterminante.

Le développement de l'application a été réalisé avec la plateforme JADE en utilisant l'environnement de développement NETBEANS IDE v7.4.

D'abord et avant de commencer la modélisation. Par rapport à la méthodologie de développement. Nous avons bien jugé que Gaia et ROADMAP peuvent nous bien servir comme point de départ pour l'instant en tant que nouveaux entrants comme nous avons vu dans le deuxième chapitre. Ce qui nous a conduit alors à proposer une nouvelle méthodologie -ou plutôt modestement- un prototype visant à capturer le meilleur des deux et pouvant même couvrir la phase d'implémentation ignorée également par la plupart des méthodologies de plus de la phase d'analyse et de conception. Notre choix peut ne pas être assez rationnel du fait que nous ne pouvions pas couvrir tous les méthodologies existantes.

Mais d'emblée, après avoir jeté un coup d'œil superficielle elle ressemblait *plus complète* dans certains aspects car basée sur l'expérience gagné des méthodologies antécédentes *mais trop complète* car visant le développement des systèmes complexes et donc elle demeure adéquate par rapport à notre solution qui se place dans le contexte du management de la chaîne logistique et qui est caractérisé par la complexité et la dynamique.

La figure suivante présente une vue générale sur notre méthodologie utilisée lors de la modélisation et l'implémentation, depuis l'analyse et la collecte des exigences en passant par les trois phases jusqu'au code JADE.

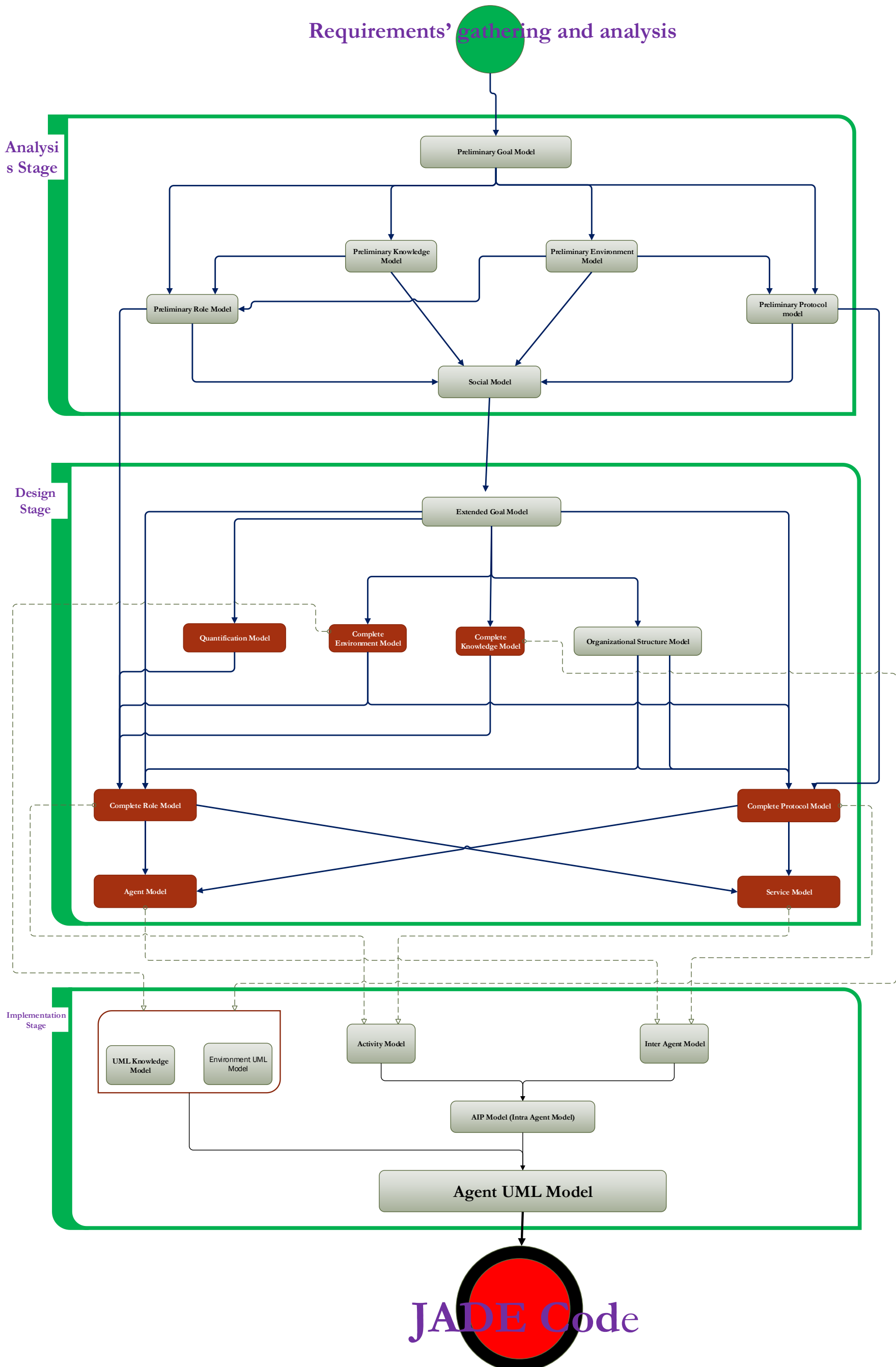


Figure 45: Méthodologie de développement

4.2. Méthodologie de développement

La méthodologie dans sa globalité comprend 3 étapes, une phase d'analyse, une phase de conception et une phase de pré-implémentation. Elle comporte un total de 21 modèles. Comme sa mère Gaia, elle est linéaire dans son approche facilitant le passage dans modèle à un autre et adopte une approche descendante commençant par le plus abstrait jusqu'à sortir par le plus concret qui s'implémente directement sur JADE.

Nous allons dans ce qui suit décrire brièvement les modèles jugés essentiels pour construire notre modèle générale. Et qui présentent les éléments importants pour comprendre par la suite l'implémentation.

Complete Rôle Model :

Un rôle est une description abstraite d'une fonction donnée (Ex : assistant client, premier ministre, contrôleur qualité ... etc.) les rôles correspondent tous simplement aux individus agissant au sein d'une organisation.

De toute façon, ce modèle a pour objectif de construire un ensemble de "rôle schemas". Un rôle schema est une représentation abstraite exprimée en termes de dix attributs.

- Rôle Name : étiquette d'identification du rôle.
- Rôle Description : une description courte du rôle (la fonction à accomplir)
- Goals : liste les buts visés par le rôle.
- Liveness Responsibilities: tous les situations à réaliser par le rôle étant donné certains conditions d'environnement.(Ex: soumettre un ordre d'achat)
- Safety Responsibilities: tous les situations à éviter par le rôle en maintenant un état acceptable (Ex: maintenir la température du four à 50°C pour éviter de bruler la Pizza)
- Knowledge Components : Des morceaux de code ou algorithmes représentant le cerveau du rôle et son Coeur de raisonnement.
- Permissions : ce sont les ressources d'environnement sur lesquels agit le rôle pour accomplir ses buts.
- Sub-Rôles: pour éviter d'alourdir le "rôle schema" et garder une bonne lisibilité, un rôle peut être exprimé en terme de sous-rôles dont leurs noms sont cités dans ce champs et qui possèdent de leurs tours leurs propres "rôle-schemas".

Complete Protocol Model : (Annexe 7)

Il génère comme sortie un diagramme de séquence AUML pour visualiser le protocole d'interaction entre les différents rôles du système.

Environment UML Model :

C'est un diagramme de classe schématisant les ressources environnementales et leur hiérarchie. Ce sont les structures de données sur lesquelles le rôle agit lors de son exécution (ou permissions au sens de « rôle schema »

Activity Model :

Correspond à un diagramme d'état de machine schématisant la séquence d'exécution des comportements de l'agent et décrivant la logique du passage d'un état (comportement) à un autre sur lequel on associe le « service model »

Agent UML Model :

Il constitue la sortie finale de notre méthodologie qui s'agit d'un diagramme de classe de l'agent, ses attributs (ses ressources environnementales et ses interfaces) et ses comportements.

Extended Goal Model : (Annexe 7)

Il s'agit d'un diagramme de cas d'utilisation schématisant les buts du système et ses qualités.

4.3. Développement des modèles de notre système

Dans ce qui suit nous allons décrire l'ensemble des modèles appliqués sur notre système :

Complete rôle model

Médiator :

1. **Rôle Name :** *Mediator*

2. **Rôle Description :**

Lance et supervise le processus du bargaining une fois requis par le fournisseur

3. **Goals :** proposer une meilleure alternative s'il est possible.

4. **Liveness responsibilities :**

- **StartUp=**

(**InstantiateAgentResources.PublishMediationService.LaunchMainInterface.WelcomeMessage**)

- **InstantiateAgentResources=** (Messages.LocalParams. AgentLog)

- **AwaitForMediationRequest=**

(ReceiveMediationRequest.AskForRetailerPermission(InitiateBargaining.Ab

- **Reload=**(InitializeInternalState.CheckInternalState.StandByMode)

5. **Safety responsibilities :**

- **SR#0:** : lié à la ressource MedLocalParams **during** StartUp //assurer que les attributs possèdent des valeurs par défauts appropriées.

- **SR#1:** vérifier la présence du distributeur dont le GUID est fourni dans la requête du fournisseur **during** WaitForMediatonRequest.

- **SR#2:** $W > 2P_0$ AND $W = 2P_0 e^{\gamma b}$ Assurant la validité du bid

$\exists b^{opt} > 0$ tel que: $\pi_r(b^{opt}) = \pi_s(b^{opt}) > 0$ et $b_r < b_s$ Vérifiant l'existence de la solution optimale

during AwaitForBids

6. **Knowledge Components :**

- Algorithme de la bisection pour calculer approximativement b^{opt}

- Différents algorithmes de validation y compris l'algorithme de validation des bids

- Algorithme de calcul de I , π_s et π_r

- Algorithme de détection de fraude

7. Permissions :

- **Read** supplierRequest, retailerReply, retailerBid, supplierBid
- **Create, Read** Messages
- **Create, Read and Update** localParams

8. Sub-Rôles: Rôle atomique

Supplier :

1. Rôle Name: *Supplier*

2. Rôle Description: Répond au commandes de clients recus et initiant un bargaining si nécessaire

3. Goals: Maximiser son profit,

Comparer(beforeCoordOffer ; afterCoordOffer) **During** AwaitForMediatorSolution

Comparer ($Q_{ret}; Q_{sup}^*$) **During** AwaitForCustomerOrder

4. Liveness responsibilities :

- **Startup**=(**InstantiateAgentResources.PublishSellingService.LaunchMainInterface.ReportToUser**)
- **InstantiateAgentResources**= (Messages.LocalParams.MedDetails.TransactionLog.AgentLog)
- **ReportToUser**=(WelcomeMessage.RequestUserToEnterParameters)
- **AwaitForCustomerOffer** = (ReceiveCustomerOffer.SubmitOffer)
- **AwaitForCustomerOrder**= (ReceiveAndEvaluateOrder.
(CPropose|Approve|**RequestMediation**))
- **RequestMediation** =
SendBargainingRequest.AwaitForMediatorConfirmation.(CloseDeal|**InitiateBargaining**)
- **InitiateBargaining**=(AwaitForBargainingRange.AwaitForRetailerExpectedSales.(AwaitForBidRequest).AwaitForMediatorSolution.**AwaitForCustomerOrder**)
- **Reload**=(**InitializeInternalState**.CheckInternalState.NotifyUser.DisplayTransactionDetails)
- **InitializeInternalState**=(InitInputParams.InitProductionCostFuncParams)

5. Safety responsibilities :

- **SR#0:** lié à la ressource SupLocalParams **during** StartUp //assurer que les attributs possèdent des valeurs par défauts appropriées.

$$\mathbf{SR\#1: } p^{order} = p^{supplier} \text{ and } Q^{max} \geq Q^{ret} > 0$$

Ou bien $W^{order} = W^{med}$, $b^{order} = b^{med}$, $Q = Q^{mediator}$ **during**

AwaitForCustomerOrder

- **SR#2:** $Q \in K$ and $\gamma < 0$ **during** AwaitForBidRequest (K étant l'intervalle du bargaining)
- **SR#3:** $\max\{\text{beforeCoordOffer}; \text{afterCoordOffer}\}$ and $Q \in K, W > 2P_0$ AND $W = 2P_0 e^{\gamma b}$ **during** AwaitForMediatorProposal

6. Knowledge Components:

- Algorithme de la bisection pour calculer approximativement le seuil de rentabilité b_s
- Fonction Plotter pour visualiser les variations de $p(Q)$ et $\pi_s(w)$
- Calcul automatique de Q^*
- Différents algorithmes de validation (ordre d'achat, solution proposée par le médiateur ...etc.)

7. Permissions:

- **Read** mediatorProposal, retailerOrder
- **Create, Read** Messages //une structure de données enveloppant tous les messages pouvant être envoyés ainsi que les filtres de réception.
- **Create, Read and Update** mediatorDetails, retailerDetails, localParams, supplierLog, transactionLog
- **Sub-Rôles:** *Rôle atomique*

Retailer :

1. **Rôle Name :** *Retailer*

2. **Rôle Description :** Initier des ordres d'achats tout en répondant aux suggestions des médiateurs.

3. **Goals:** Maximiser son profit, *Comparer (BeforeCoordProfit; AfterCoordProfit)*

During AwaitForMediatorProposal

4. Liveness responsibilities :

- **StartUp=**
(**InstantiateAgentRessources.SearchForSuppliers.LaunchMainInterface.ReportToUser**)
- **InstantiateAgentRessources=**
(PurchaseOrder.LocalParams.SupDetails.MedDetails.Messages.TransactionLog.AgentLog)
- **SearchForSuppliers=(ContactDF.GetAvailableSuppliersList.AskForOffer)**
- **ReportToUser=(WelcomeMessage.DisplayAvailableSuppliersDetails.RequestUserToEnterParams)**
- **updateSupDetails = (searchForSuppliers)**
- **Purchase= (AwaitForUserRequest.
InitializeInternalState.SubmitPurchaseOrder.AwaitForReply)**
- **InitializeInternalState=(InitInputParams.InitOrderCostFuncParams.InitHoldingCostFuncParams)**
- **AwaitForReply=(MedBargainingRequest/SupConfirmation/SupCounterProposal)**
- **MedBargRequest=AskUserPermission.(Reject.ExpectSupApproval|AcceptAndInitiateBargaining).Reload**
- **SupConfirmation=(CloseOperation.NotifyUser)**
- **AcceptAndInitiateBargaining=(AwaitForBargainingRange.(AwaitForBidRequest).AwaitForMediatorSolution.SubPurchOrder.AwaitForReply)**
- **Reload=(InitializeInternalState.CheckInternalState.NotifyUser.DisplayTransactionDetails)**

5. Safety responsibilities :

- **SR#0:** lié à la ressource LocalParams **during** StartUp //assurer que les attributs possèdent des valeurs par défauts appropriées
 $Q^{max} \geq Q > 0$ and $P_0 > 0$ **during** StartUp (lors de la reception de l'offre du fournisseur)

- **SR#1:**
 $p^{order} < \dots < p^{supplier}$

Ou bien $W^{order} < \dots < (W^{mediator}), b^{order} < \dots < (b^{mediator}), Q < \dots < (Q^{mediator})$ **during** SubmitPurchaseOrder

- **SR#2:** $Q \in K$ and $\gamma < 0$ **during** AwaitForBidRequest (K étant l'intervall du bargaining)
- **SR#3:** $\max\{\text{beforeCoordOffer} ; \text{afterCoordOffer}\}$ and $Q \in K, W > 2P_0$ **AND** $W = 2P_0 e^{\gamma b}$
during AwaitForMediatorProposal

6. Knowledge Components:

- Algorithme de la bisection pour calculer approximativement le seuil profitabilité b_r
- Fonction Plotter pour visualiser les variations de $h(Q), o(Q)$ et $\pi_r(w)$
- Calcul automatique de Q^*
- Différents algorithmes de validation (d'offre, de la solution proposée par le médiateur ...etc.)

7. Permissions:

- **Read** mediatorProposal, supplierOffer, supplierCounterProposal, KInterval, bidParams
- **Create, Read** Messages //une structure de données enveloppant tous les messages pouvant etre envoyés ainsi que les filtres de reception.
- **Create, Read and Update** purchaseOrder, localParams, supplierDetails, mediatorDetails, agentLog, transactionDetails.

8. Sub-Roles: *Rôle atomique*

Agent UML model

Mediator:

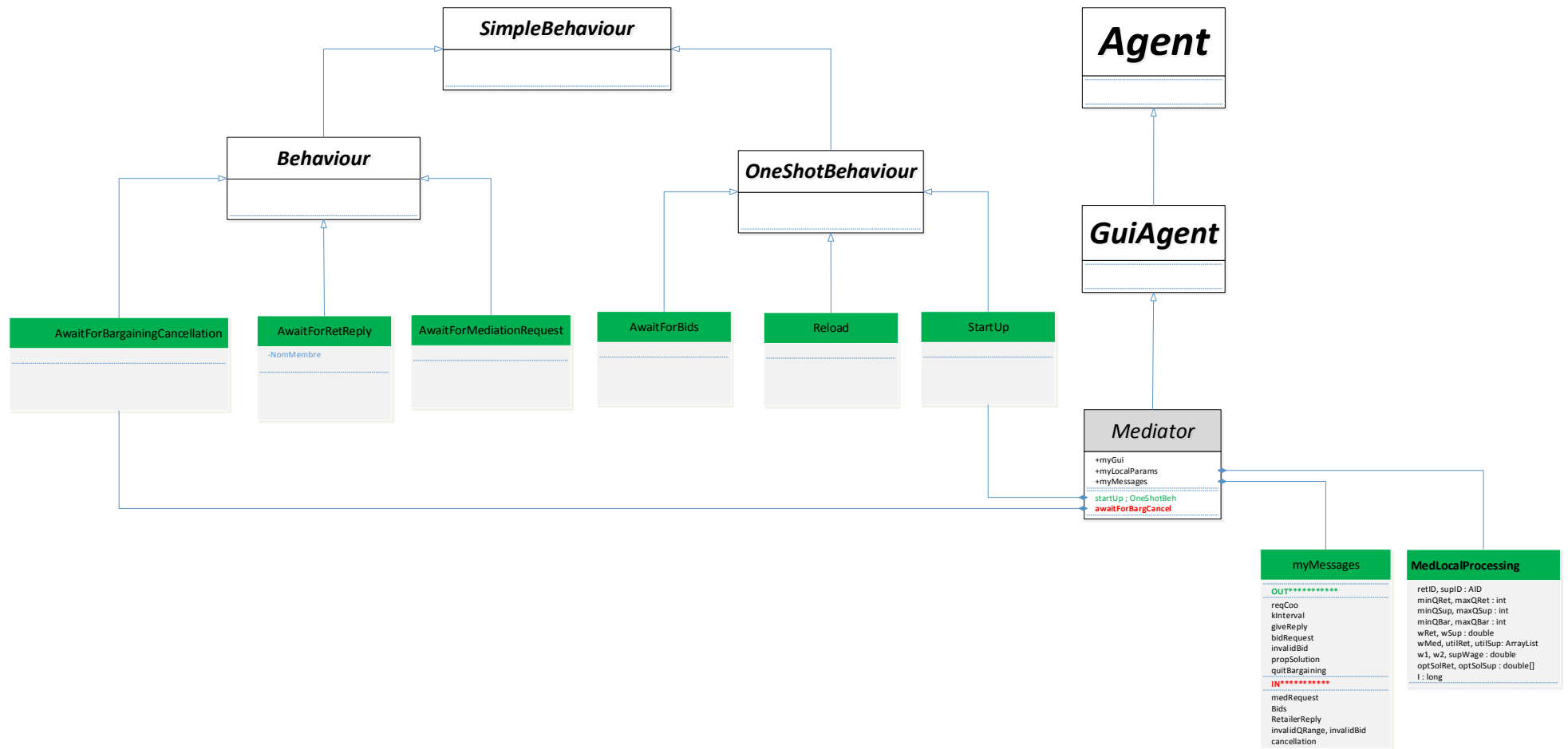


Figure 46: Agent UML model (mediator)

Supplier:

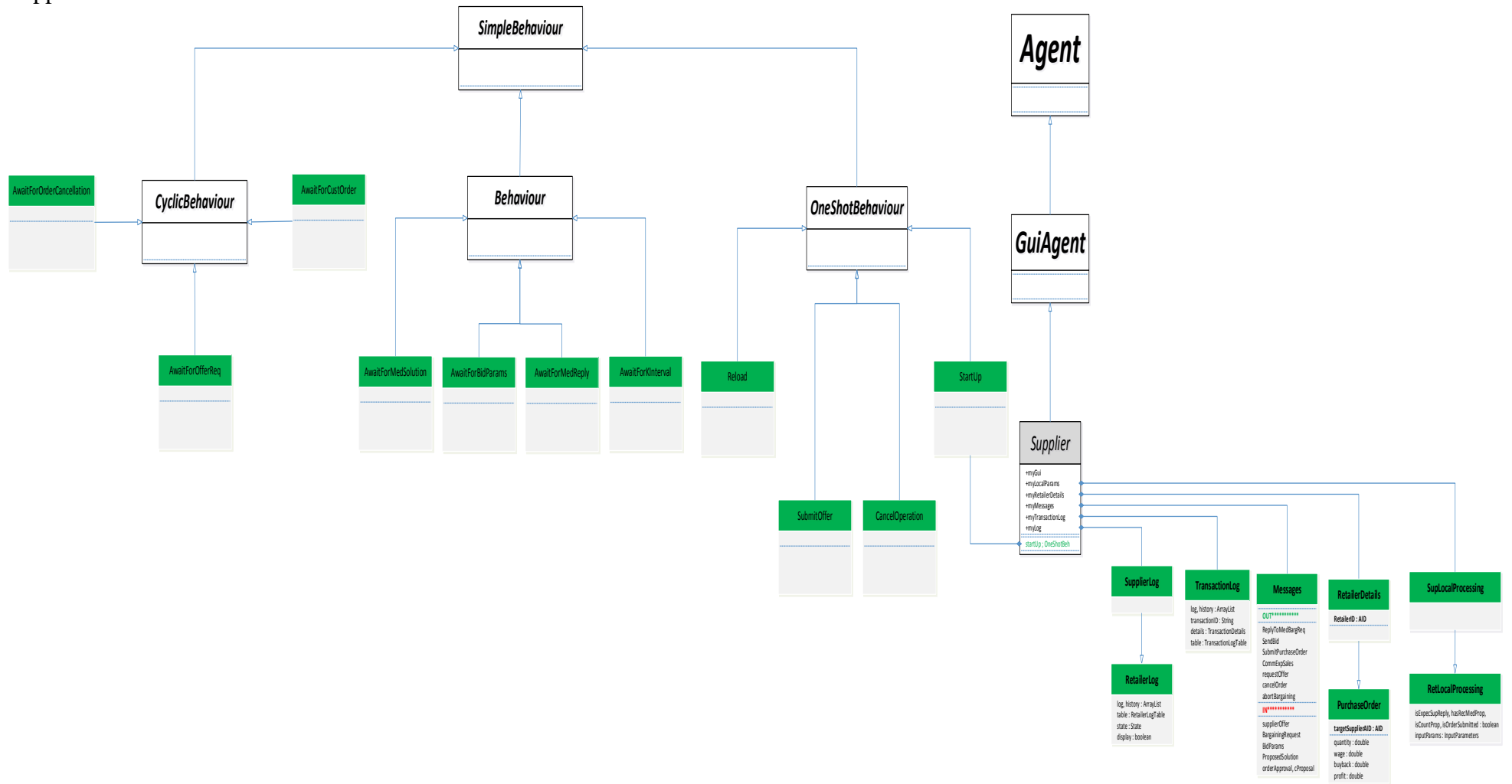


Figure 47: Agent UML model (supplier)

Retailer :

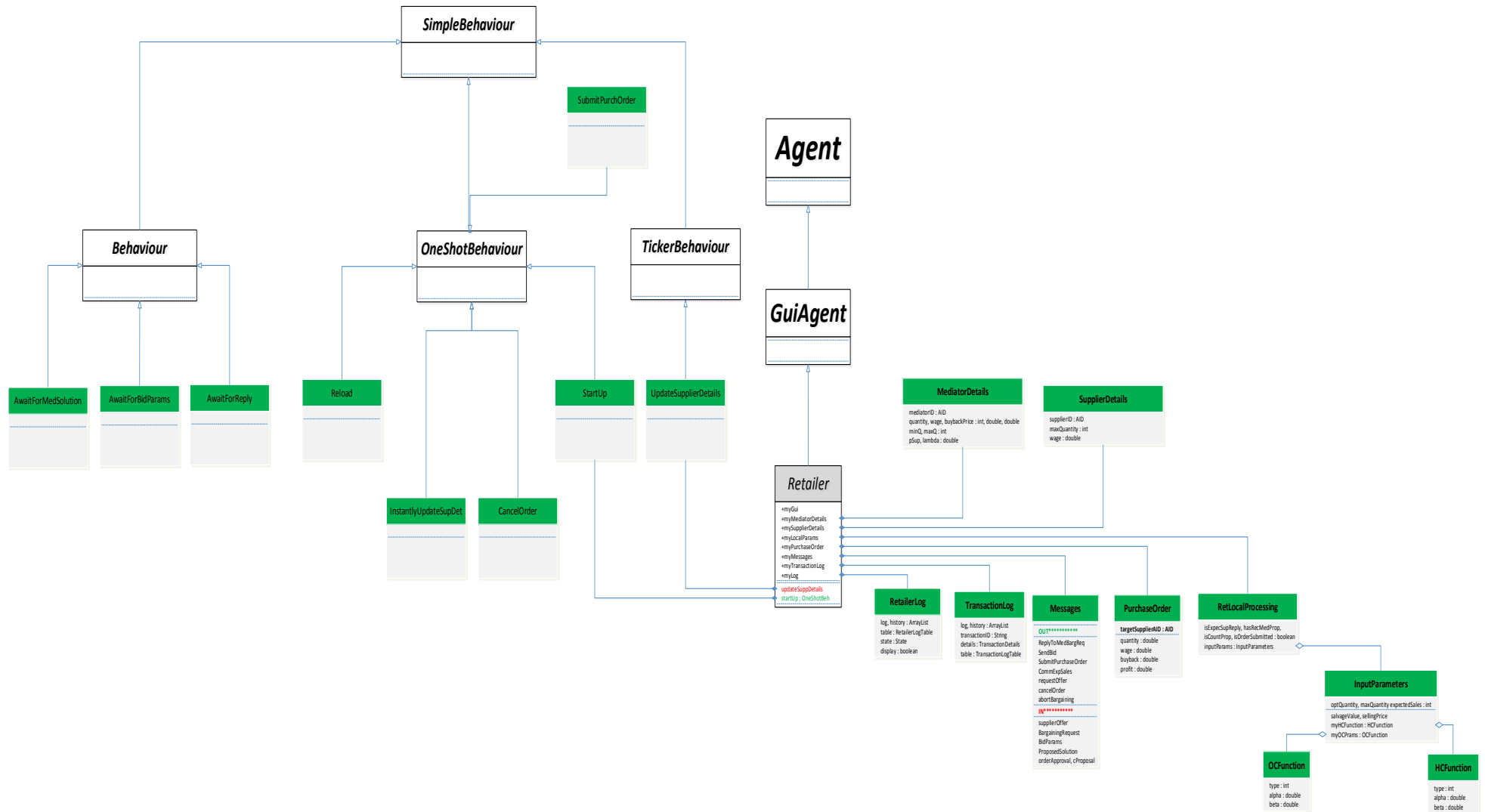


Figure 48: Agent UML model (Retailer)

Messages

- Supplier→Mediator

Table 11: Mediation request message.

| MediationRequest | |
|-------------------------|---|
| Performative | Request |
| In-Reply-To | SenderMessage.getReplyWith()//Bargaining ID |
| Content | MediationRequest |
| Properties | (ContType=BuyBack, CoupFunc($W(b) = 2Pe^{yb}$), J=[Qmin;Qmax], RetailerLocalName, RetailerPlatformID) |

Table 12: Bid message (from supplier to mediator) .

| Bid | |
|--------------|--------------------------------------|
| Performative | Inform |
| In-Reply-To | SenderMessage.getReplyWith()//Bid ID |
| Content | Bid |
| Properties | W^{sup} |

- Retailer→Mediator

Table 13: Reply regarding bargaining request message.

| Reply Regarding Bargaining Request | |
|---|--------------------------------|
| Performative | Agree (or Refuse) |
| In-Reply-To | SenderMessage.getReplyWith() |
| Content | BargainingProposalReply |
| Properties | MinQ ; MaxQ (or Null) |

Table 14: Bid message (from retailer to mediator).

| Bid | |
|--------------|--------------------------------------|
| Performative | Inform |
| In-Reply-To | SenderMessage.getReplyWith()//Bid ID |
| Content | Bid |
| Properties | W^{ret} |

- Mediator → Supplier

Table 15: Posting Retailer reply message.

| PostingRetailerReply | |
|-----------------------------|--|
| Performative | Agree (or Refuse) |
| Reply-with | BargainingID (The time he received the mediation request from the supplier) |
| Content | <i>RetailerReply</i> |
| Properties | Null |

Table 16: Bid request (from mediator to supplier).

| BidRequest | |
|-------------------|--|
| Performative | Request |
| Reply-with | $Q ; \gamma$ (or sending time associated to $Q ; \gamma$ in the mediator internal state)//Bid ID |
| Content | <i>BidRequest</i> |
| Properties | $(Q ; \gamma)$ |

Table 17: Propose solution message.

| ProposeSolution | |
|------------------------|-------------------------|
| Performative | Inform |
| Content | <i>ProposedSolution</i> |
| Properties | $(Q ; w ; b)$ |

- Mediator → Retailer

Table 18: Bargaining request message.

| BargainingRequest | |
|--------------------------|--|
| Performative | CFP |
| Reply-with | BargainingID (The time he received the mediation request from the supplier) |
| Content | <i>BargainingRequest</i> |
| Properties | (ContractType=BuyBack, CouplingFunctionType($W(b) = 2Pe^{\gamma b}$)) |

Table 19: Bid request (from mediator to retailer).

| BidRequest | |
|-------------------|--|
| Performative | Request |
| Reply-with | $Q ; \gamma$ (or sending time associated to $Q ; \gamma$ in the mediator internal state) |
| Content | <i>BidRequest</i> |
| Properties | $(Q ; \gamma)$ |

Table 20 : Propose solution message.

| ProposeSolution | |
|------------------------|-------------------------|
| Performative | inform |
| Content | <i>ProposedSolution</i> |
| Properties | $(Q ; w ; b)$ |

- Retailer→Supplier

Table 21: Submit purchase order message.

| SubmitPurchaseOrder | |
|----------------------------|--------------------------------|
| Performative | Request |
| Reply-with | SendingTime |
| Content | <i>PurchaseOrder</i> |
| Properties | OrderParameters($Q ; W ; B$) |

Table 22: Communicate expected sales message.

| CommunicateExpectedSales | |
|---------------------------------|----------------------|
| Performative | inform |
| In-Reply-To | BargainingID |
| Content | ExpectedSales |
| Properties | S |

- Supplier → Retailer

Table 23: Agreement message.

| Agree | |
|--------------|-------------------------------------|
| Performative | Agree |
| Content | OrderApproval |
| In-Reply-To | PurchaseOrderRequest.getReplyWith() |
| Properties | Null |

Table 24: Counter propose message

| CounterPropose | |
|-----------------------|-------------------------------------|
| Performative | Counter-propose |
| Content | CounterProposing |
| In-Reply-To | PurchaseOrderRequest.getReplyWith() |
| Properties | $(Q_{max}; P_0)$ |

Activity model

Mediator :

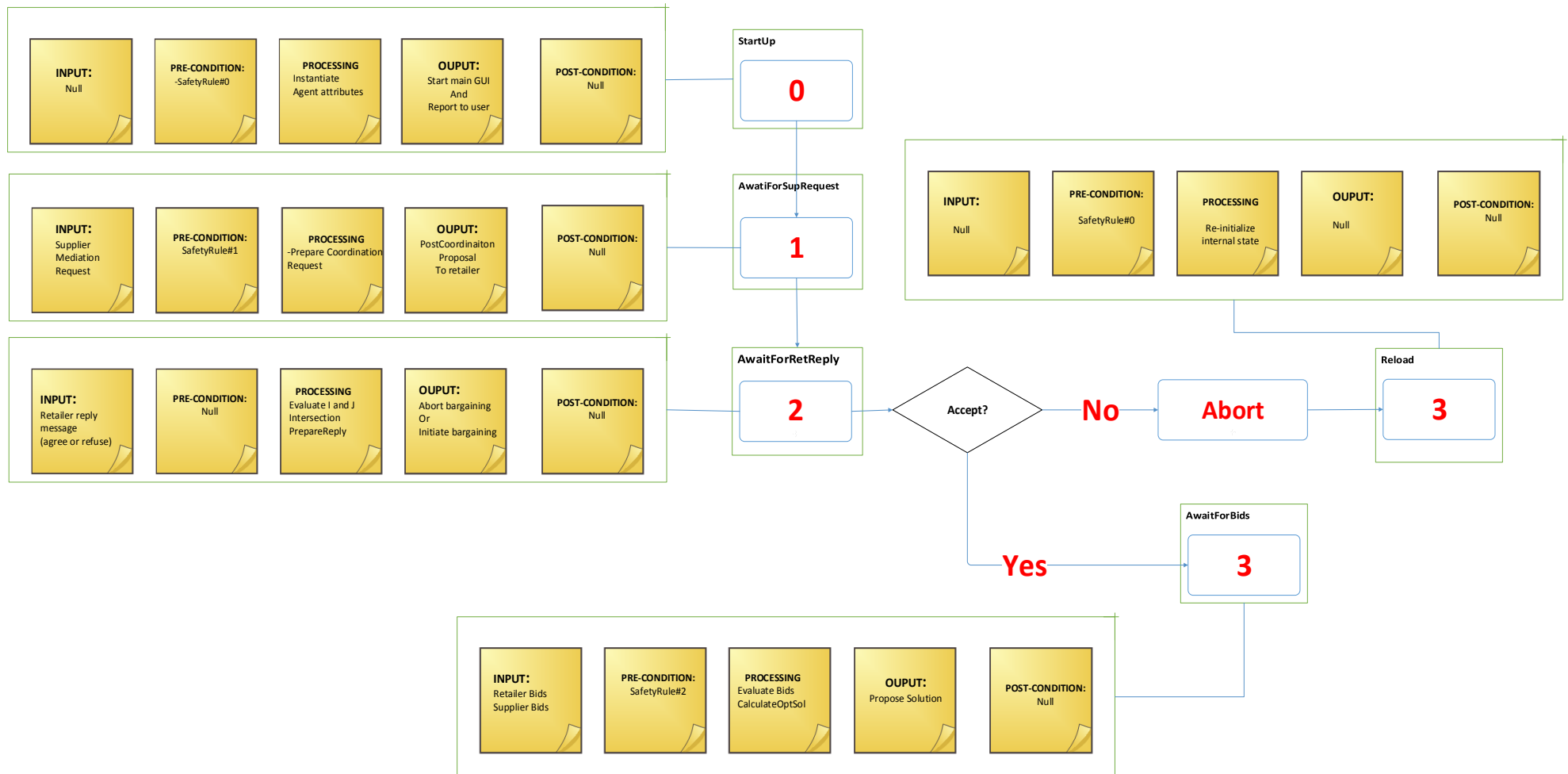


Figure 49: Mediator activity model

Supplier:

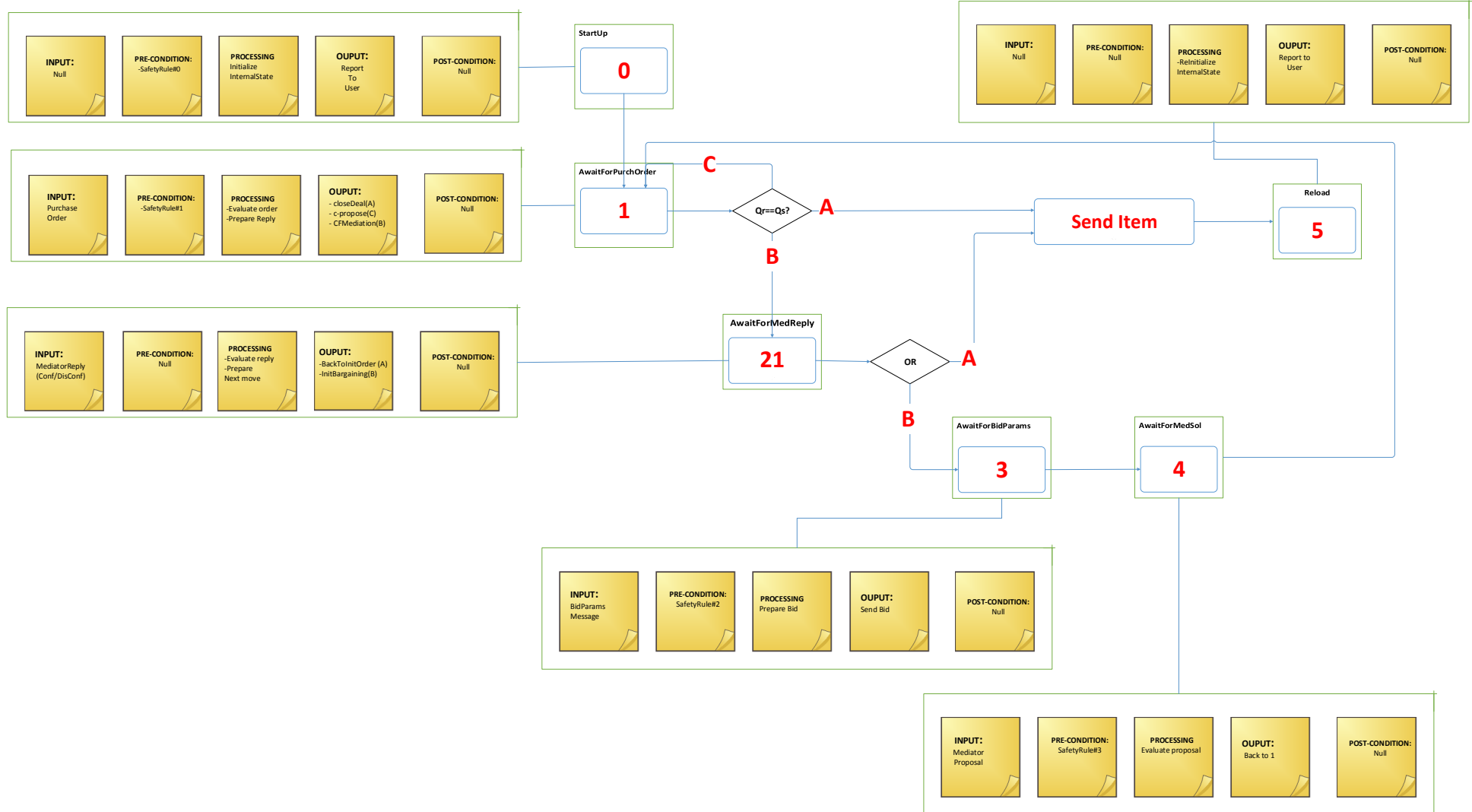


Figure 50: Supplier activity model

Retailer:

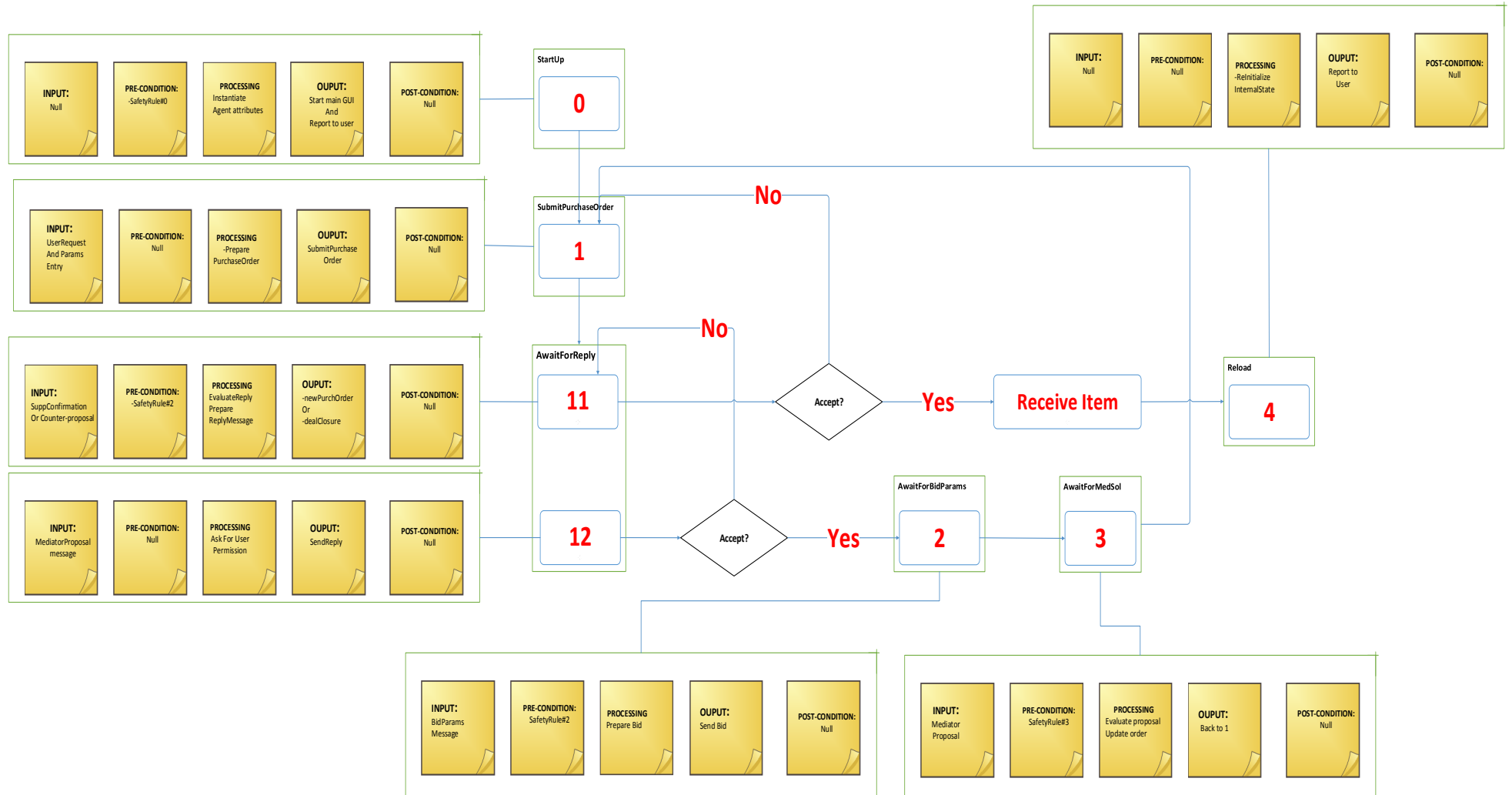


Figure 51: Retailer activity model.

Environment model

Mediator:

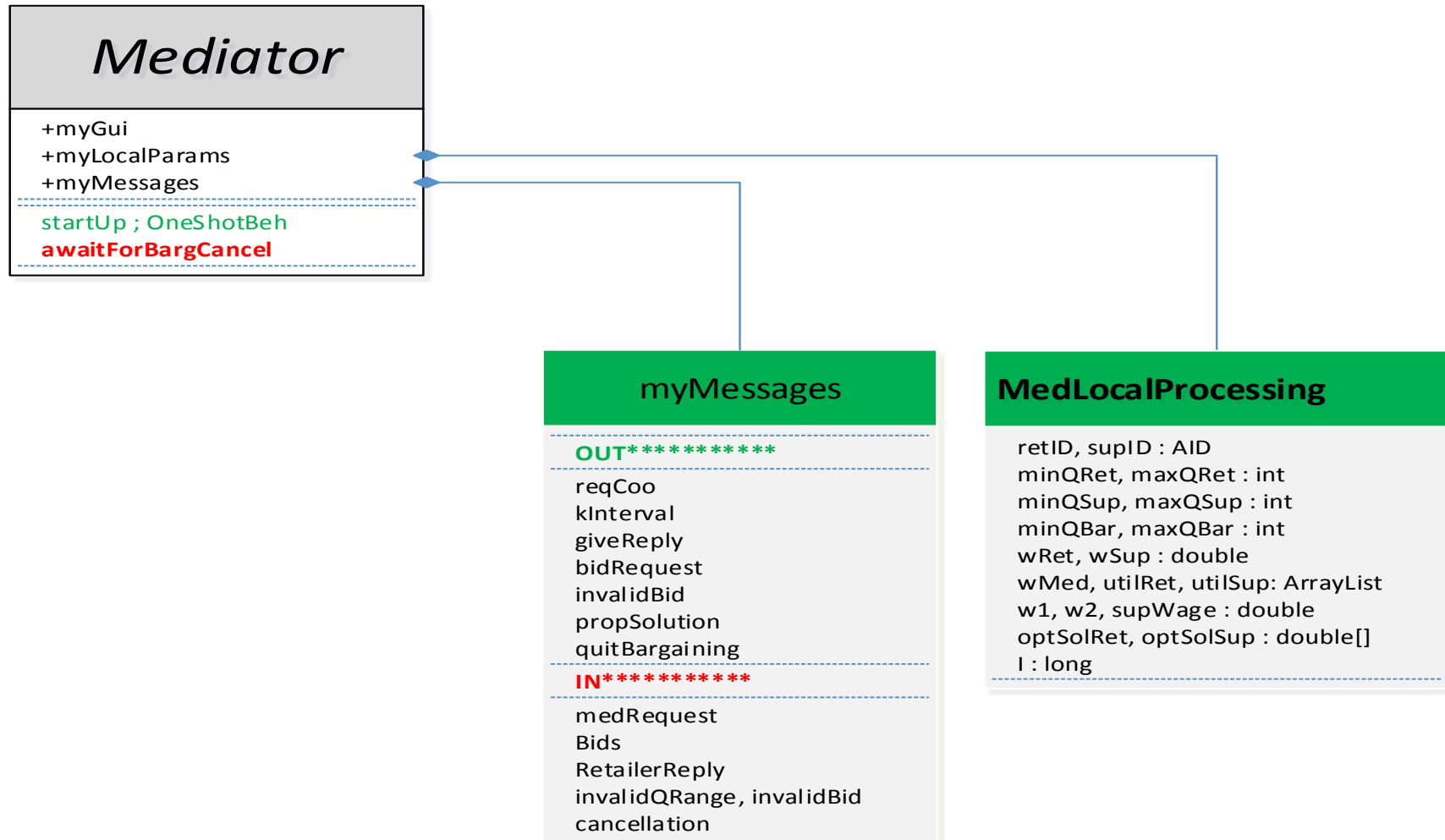


Figure 52: Mediator environment model.

Supplier:

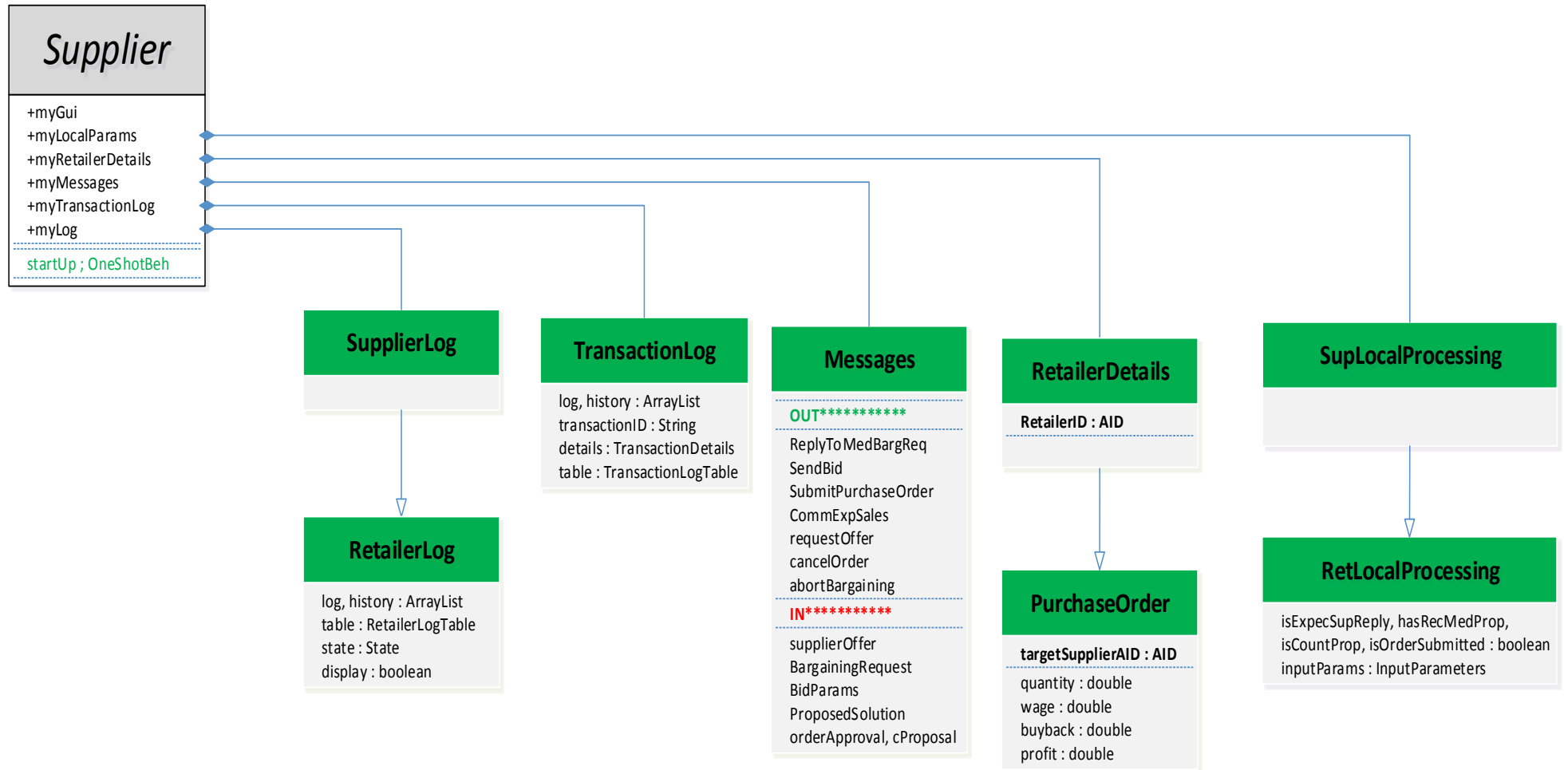


Figure 53: Supplier environment model

Retailer:

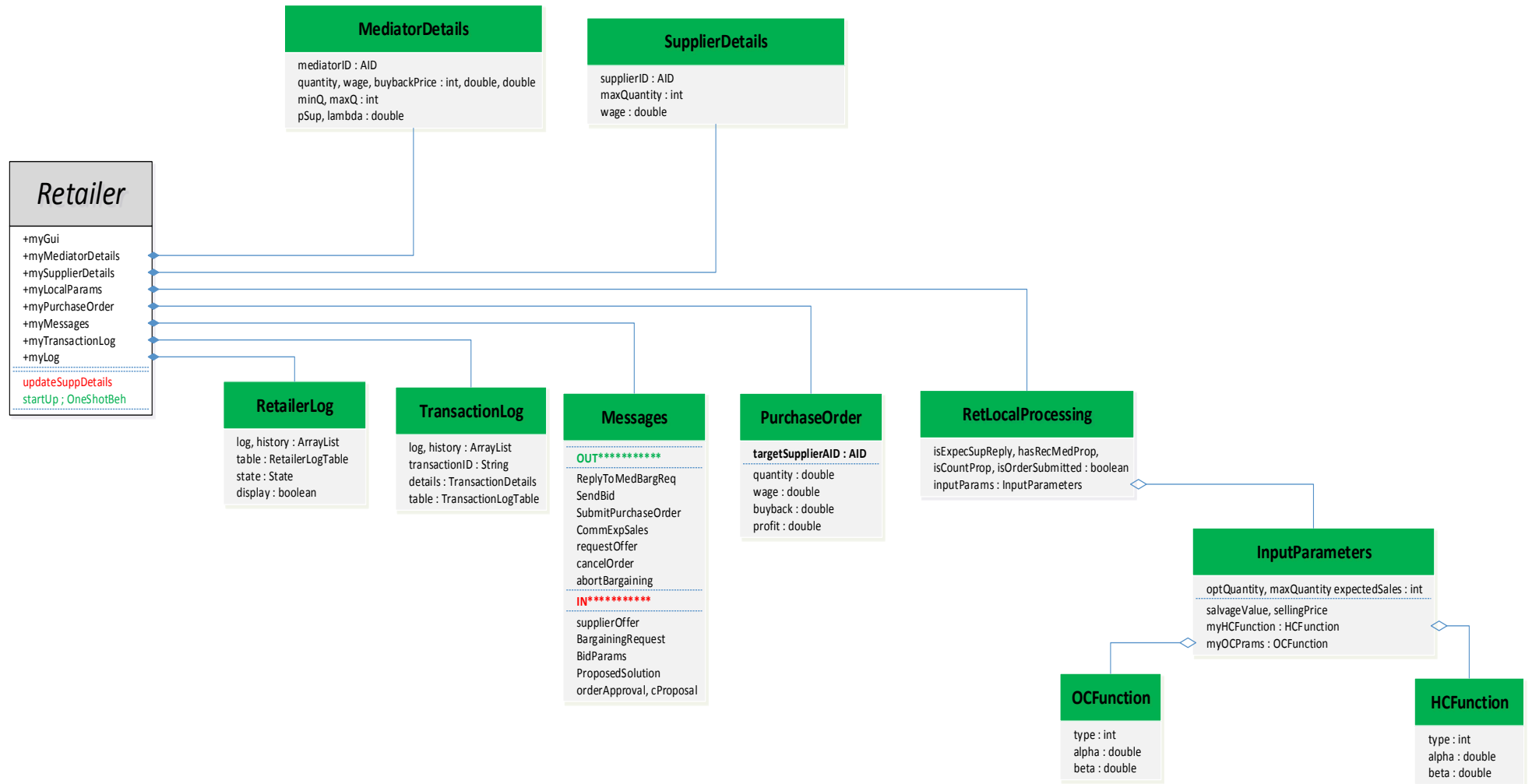


Figure 54: Retailer environment mod

4.4. Implémentation :

Le projet a été décomposé selon une vue agent. C'est-à-dire, un package pour chaque agent comportant ainsi des sous packages pour ses ressources, ses comportements, ses interfaces graphiques, son cerveau et sa classe. Avec quelques packages supplémentaires importés pour la visualisation graphique de certains courbes que l'utilisateur pourra également consulter.

Voici une fiche technique du projet :

| | |
|---|-----------------|
| Taille global du projet | 4.58 Mo |
| Nombre de classes implémentés | 91 |
| Nombre de classes open source importés | 11 |
| Nombre de lignes de codes | 10686 |
| Durée de développement | 50 jours |

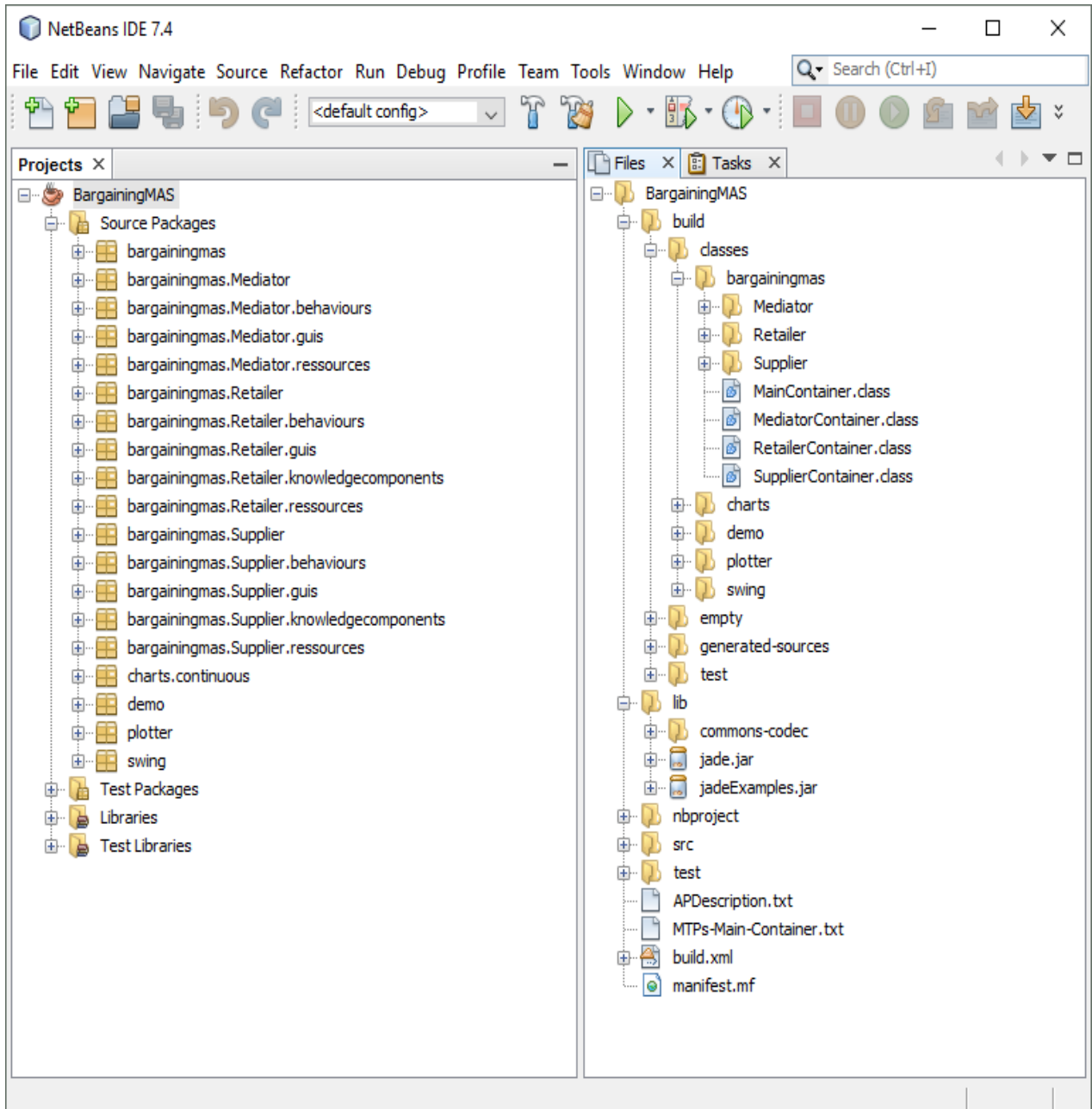


Figure 55: Project tree

4.4.1. Mécanisme de coordination

Là on parle du cerveau du médiateur et son raisonnement pour coordonner les deux participants. La médiation commence par une demande de la part de l'agent fournisseur. Le médiateur dans un premier temps vérifie la présence de l'agent distributeur dont le GUID est associé à la demande du fournisseur. Si l'agent distributeur existe dans la plateforme, le médiateur lui demande par message de coordonner. Une fois acceptée. Le médiateur informe le distributeur et initie le bargaining en envoyant l'intervalle du bargaining $K = [k_{min} ; k_{max}]$ calculé à partir des deux intervalles tolérées par les deux participants I et J.

$$k_{min} = \max\{i_{min} ; j_{min}\} \text{ et } k_{max} = \min\{i_{max} ; j_{ax}\}$$

- Son premier objectif vise à calculer le niveau de stock prévu $I = Q - S \in R$ pour ensuite déterminer les ventes prévues à perdre par le distributeur sous formes de pénurie ou excès $I = |Q - S| \in R_+$. cette dernière constitue la première information indispensable.

Pour cela, Le médiateur sait que l'acteur i doit lui envoyer à un point donné du bargaining Q_0 son seuil critique de profitabilité w_i déterminé à travers la résolution de l'équation suivante :

$$\pi_i(Q = Q_0, w = P_0, b = 0) \pm w_i Q \mp b_i I = 0 \quad \Leftrightarrow \quad \pi_i^0 + w_i Q \mp \frac{1}{\gamma} \ln\left(\frac{w_i}{2P_0}\right) I = 0$$

L'équation comporte deux inconnues I et π_i^0 et pour pouvoir les déterminer il faudra une deuxième équation impliquant ces deux inconnues. Donc, pour y arriver, le médiateur n'a besoin d'interroger l'un des deux acteurs par le biais de **deux** requêtes d'enchères. Il ne lui reste par la suite qu'à résoudre le système d'équation suivant :

$$\pi_i^0 + w_{i1} Q \mp \frac{1}{\gamma} \ln\left(\frac{w_{i1}}{2P_0}\right) I = 0 \quad \dots\dots (1)$$

$$\text{et } \pi_i^0 + w_{i2} Q \mp \frac{1}{\gamma} \ln\left(\frac{w_{i2}}{2P_0}\right) I = 0 \quad \dots\dots (2)$$

Ce système a comme solution :

$$I = Q_0 \frac{w_{i2} - w_{i1}}{b_{i2} - b_{i1}} \quad \dots\dots (A) \quad \text{et} \quad \pi_i^0 = \mp w_{i1} Q \pm b_{i1} I \quad \dots\dots (B)$$

Voici la portion du code Java permettant de calculer I et S :

```

public long calculateI(int q) {
    double dW = w2 - w1;
    double dB = b(-2, w2) - b(-1, w1);
    I = Math.round(q * dW / dB);
    //deduce S value
    S = Math.abs(q - I);
    return I;
}

```

Figure 56: Code Java pour le calcul de I et S.

- L'étape suivante est un « give 'N take » Avec un total de $k_{max} - k_{min} + 1$ requêtes d'enchère envoyées à chacun des deux agents. simultanément, le médiateur peut facilement d'après l'équation (B) déterminer π_r^0 et π_s^0 correspondants. Puisque le médiateur a pu calculer les utilités des participants on peut dire dès maintenant qu'on est arrivé.

-

Voici les portions du code Java permettant de calculer π_r^0 et π_s^0 :

Pour le calcul de π_r^0

```

public void calculateRetUtil(int q) { //Pr(0)
    double w = getRetBid(q);
    double transferCost = -w * q + Math.abs(S - q) * b(-1, w);
    if (utilRet == null) {
        utilRet = new HashMap();
    }
    utilRet.put(q, -transferCost);
}

```

Figure 57: code Java pour calculer les profits du retailer

Pour le calcul de π_s^0

```

public void calculateSupUtil(int q) { //Ps(0)
    double w = getSupBid(q);
    double transferCost = w * q - Math.abs(S - q) * b(-1, w);
    if (utilSup == null) {
        utilSup = new HashMap();
    }
    utilSup.put(q, -transferCost);
}

```

Figure 58: Code Java pour calculer le profit du fournisseur

- Ensuite, le médiateur doit déterminer le point optimal du bargaining

Qui est tout simplement le point Q^* pour lequel $\pi_r^0 + \pi_s^0$ est maximal.

```

public int getOptSolutionIndex() {
    Map sumUtil = new HashMap();
    for (int i = minQBar; i < maxQBar; i++) {
        if (wMed.get(i) != null) { //consider valid values only
            sumUtil.put(i, ((double) utilRet.get(i) + (double) utilSup.get(i)) / 2);
        }
    }
    int max = minQBar;
    for (int i = minQBar; i < maxQBar; i++) {
        if (sumUtil.get(i) != null && (double) sumUtil.get(i) > max) {
        }
    }
    return max;
}

```

Figure 59: Code Java pour calculer le profit du fournisseur.

- Maintenant, le médiateur doit appliquer le critère de convergence C3 développé antérieurement vérifiant pour restreindre l'intervalle de recherche de la solution optimale.

Pour y aboutir, il calcul d'abord la solution pareto-optimal à chaque point du bargaining ($\forall Q \in K$). C'est-à-dire :

$\exists ? b^{opt}$ tel que : $\pi_r^0 - \pi_s^0 + 2T = 0$ T : étant le prix de transfert.

```

public void calculateOptB() {
    BisectionMethod bisAlgo = new BisectionMethod() {
        @Override
        public double calculateBOpt(double b, int q) {
            double pr0_1 = (double) utilRet.get(q);
            double ps0_1 = (double) utilSup.get(q);
            double transferCost = -2 * supWage * q * Math.exp(-1 * b) + Math.abs(S - q) * b;
            double x = pr0_1 - ps0_1 + 2 * transferCost;
            return x;
        }

        @Override
        public double calculateBid(double x) {
            return 0;
        }
    };
    //prepare and send a reply message
    for (int i = minQBar; i <= maxQBar; i++) {
        bOpt.put(i, bisAlgo.initiateBisectionAlgorithm(0, 100, i));
    }
}

```

Figure 60: Détermination de la solution Pareto Optimal.

Ensuite, le médiateur fait appel à la méthode `setValidBargSet()` pour vérifier la convergence du bargaining par rapport au critère $C3$

```

public void setValidBargSet() {
    double corrBRet, corrBSup, maxB, b_Opt;
    for (int q = minQBar; q <= maxQBar; q++) {
        corrBRet = b(-1, getRetBid(q));
        corrBSup = b(-1, getSupBid(q));
        maxB = Math.max(corrBRet, corrBSup);
        b_Opt = getBOpt(q);
        if (maxB == corrBSup && b_Opt > 0) {
            wMed.put(q, w(-1, b_Opt));
            System.out.println("Valid Point : " + q);
        } else {
            wMed.put(q, null);
            System.out.println("Invalid Point : " + q);
        }
    }
}

```

Figure 61: Vérification de la convergence du bargaining.

- Enfin, il n'a qu'à prendre le b^{opt} qui correspond à l'indice Q^* de la solution optimal. Pour ensuite la proposer aux deux participants.

4.4.2. Mécanisme de détection de fraude

Un problème aussi reconnu lors de tout développement d'un SMA est celui de l'honnêteté des agents. L'agent distributeur peut dire qu'il prévoit de vendre 100 unités alors qu'il prévoit de vendre 50 unités uniquement. L'agent fournisseur peut de son tour soumettre un faux bid pour gagner plus d'argent.

Mais, la bonne nouvelle que ce problème a été surmonté dans notre système toujours grâce à notre buyback personnalisé.

En fait, le médiateur garde tous l'historique des bargaining pour la période du contrat. Donc, une fois le bid de chacun des acteurs est différent à un bid antérieurement soumis l'acteur est pénalisé automatiquement en retranchant une portion de son argent.

Ce qui donne le système l'avantage parfait d'être appliqué en réalité.

Séquence du bargaining :

```
//REGARDEZ POUR CHACUN DES TROIS CAS LA FIGURE CORRESPONDANTE
public BargainingSequence () {

    if (CustomerOrderIsOkay) { //Q_ret^*= Q_sup^*

        System.out.println(SeeFigure1_Below);

    } else if (RetailerRefusedToCoordinate) {

        System.out.println(SeeFigure2_Below);

    } else { //RetailerAcceptedToCoordinate

        System.out.println(SeeFigure3_Below);

    }

}
```

Figure 62: Les trois scénarios possibles

L'agent distributeur soumet un ordre coincident avec la quantité optimale de l'agent fournisseur ([1]REQUEST) ce qui pousse ce dernier à approuver l'ordre directement sans avoir recours au médiation ([2]ACCEPT-PROPOSAL)

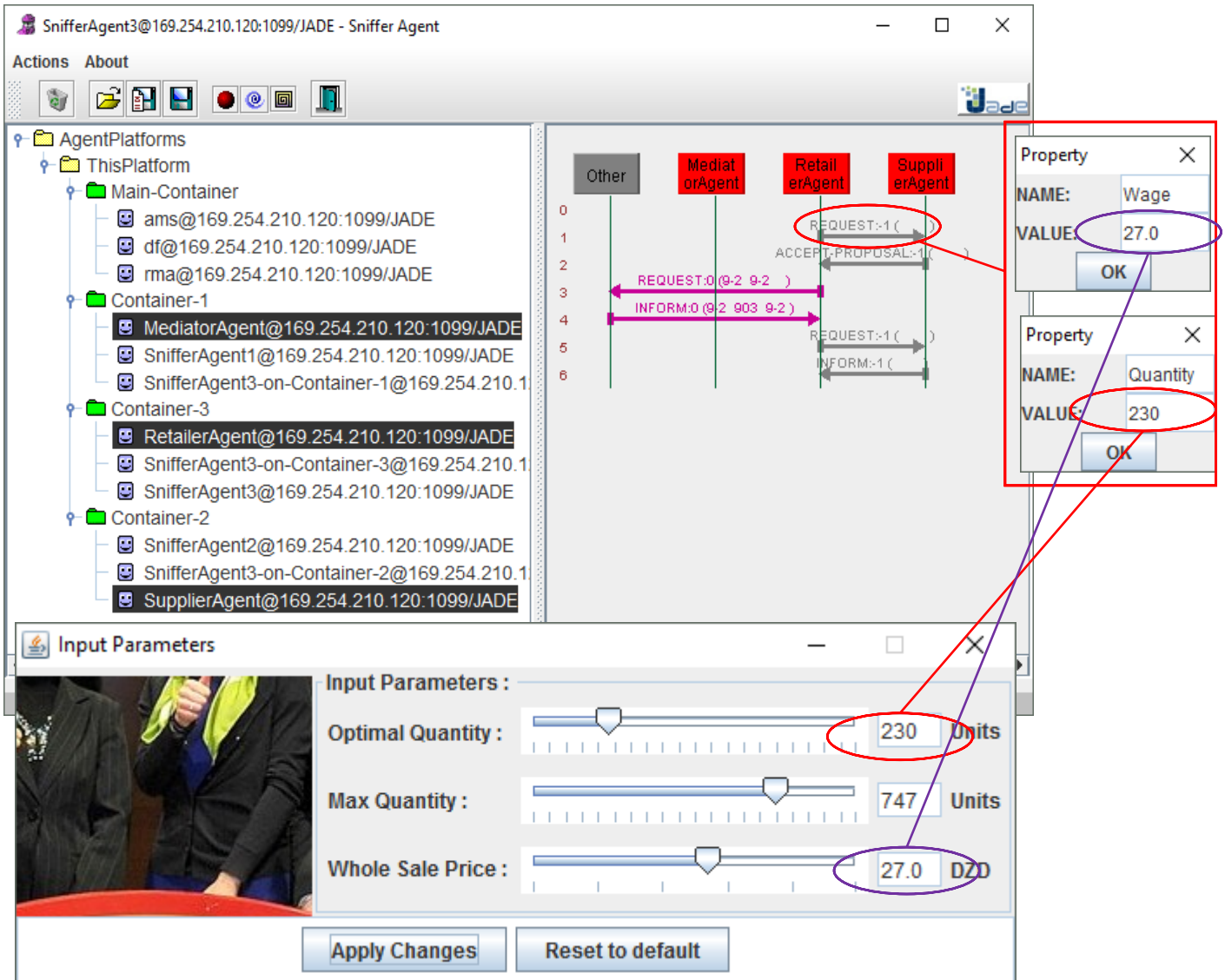


Figure 63: Approuve customer order

Cette fois là, $Q_{ret}^* \neq Q_{ret}^*$. donc, l'agent fournisseur demande un bargaining du médiateur ([2]REQUEST). Ce dernier après avoir vérifié d'abord l'existence de l'agent distributeur ([3]CONFIRM). Pour ensuite le contacter ([4]CFP). Le distributeur reçoit la requete et la refuse ([5]REFUSE) obligeant par conséquent, l'agent fournisseur de revenir à la commande initiale ([7]ACCEPT-PROPOSAL).

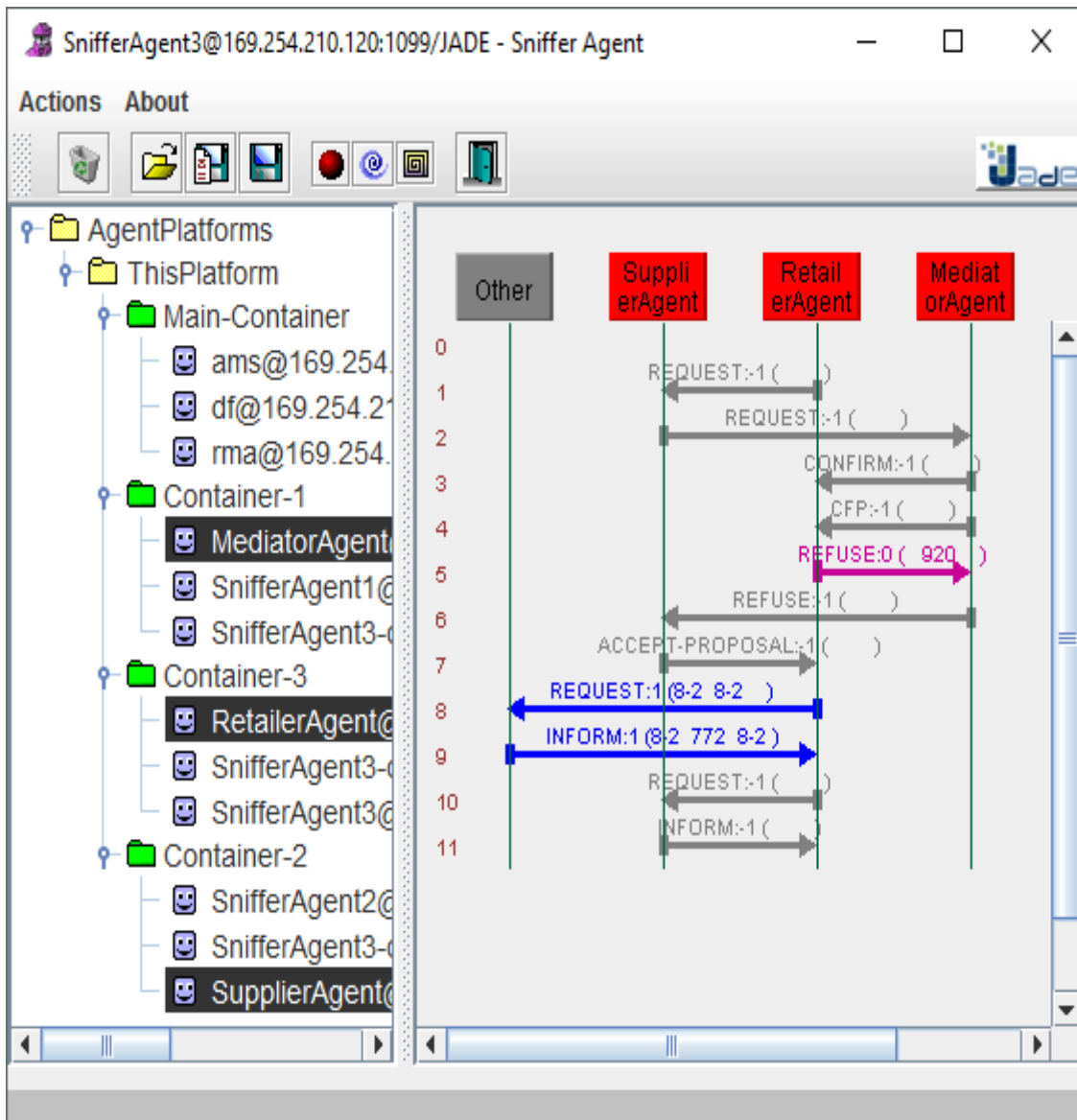


Figure 64: Retailer refuse to coordinate

Dans ce scénario, l'agent distributeur accepte la coordination ([5]AGREE). Simultanément il communique ses ventes prévues à l'agent fournisseur ([6] INFORM). Le médiateur expédie directement la réponse à l'agent fournisseur pour qu'il se prépare au bargaining([7]AGREE). Ensuite, Le médiateur envoie l'intervalle du bargaining K ([8 et 9] INFORM) puis il interroge le fournisseur par deux requetes d'enchère ([10 et 11] REQUEST) afin de déterminer la variable S.

Pour chaque point du bargaining, une requete d'enchère est envoyée à chaque agent demandant son seuil de profitabilité ([12 et 13] INFORM).

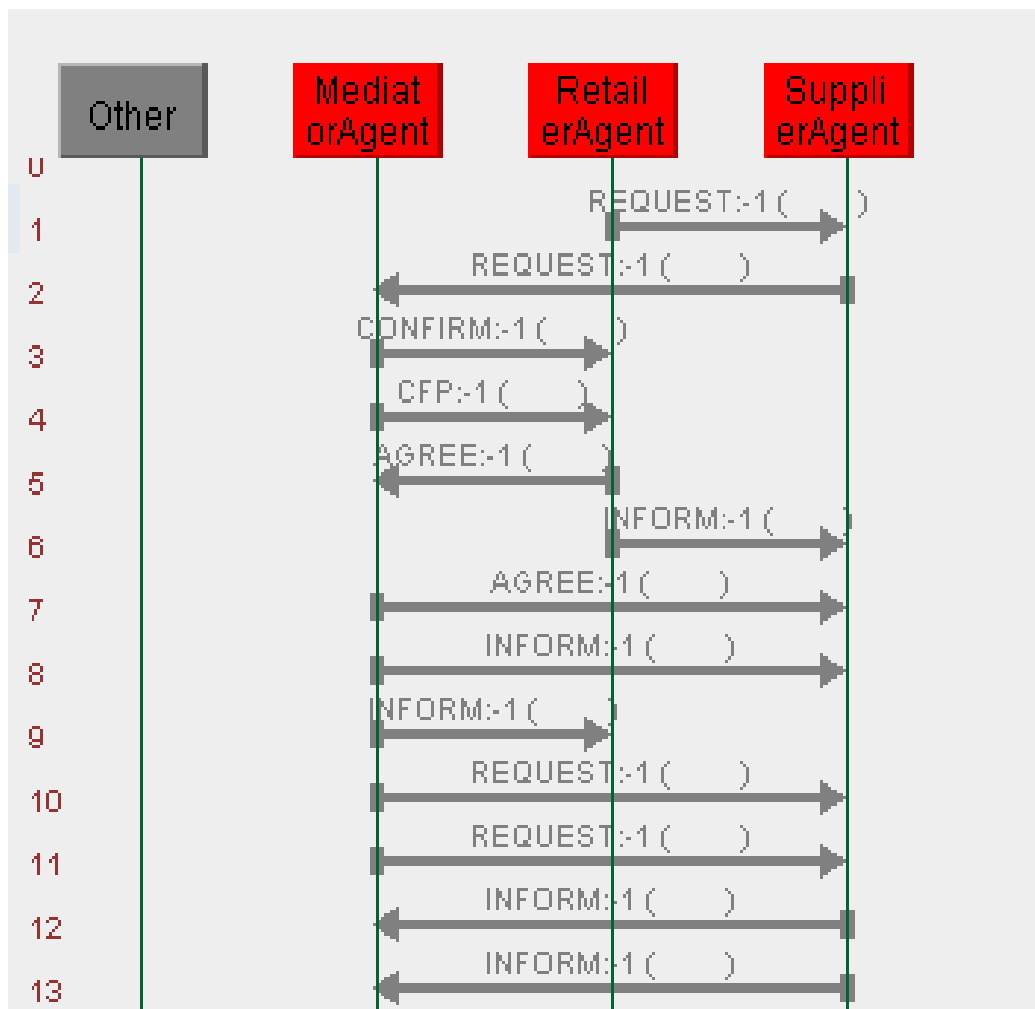


Figure 65 : Initiate bargaining

4.4.3. Le système respecte-il les concepts clés du paradigme agent ?

Dans cette section on va projeter notre système par rapport aux concepts liés au paradigme agent. On parle bien des concepts fondamentaux dont on cite :

Intelligence, Autonomie, réactivité, proactivité et sociabilité.

1. Intelligence :

Les agents sont intelligents dans le sens où ils simulent le raisonnement humain, par le biais de certains algorithmes que nous avons appelé antérieurement « knowledge components ».

Ainsi :

- L'agent distributeur ne va pas lancer la commande telle qu'elle est une fois requis par son utilisateur et doit d'abord vérifier si les paramètres saisis sont valides. Que la commande n'a pas dépassé la quantité maximale supportée par le fournisseur, que la valeur de liquidation est inférieure au prix de vente, que les coûts de lancement et de stockage sont positifs... etc.
- Les deux participants lors de la réception de la solution finale du médiateur ne vont pas l'accepter aveuglement et doivent d'abord l'évaluer par rapport à l'offre initial.

2. Autonomie et libre volonté des agents :

Ce concept est implémenté déjà par JADE de la manière que les agents ne communiquent jamais leur référence d'objet et s'identifient par contre par leurs GUIDs. Les GUIDs des trois agents du système sont :

RetailerAgent@169.264.210.120:1099/JADE

SupplierAgent@169.264.210.120:1099/JADE

MediatorAgent@169.264.210.120:1099/JADE

Mais, puisque l'approche de développement de JADE reste orientée objet il faudra ainsi assurer la sécurité des paramètres privés. Par exemple : Notre agent distributeur n'échange pas ainsi ses paramètres privés tel que les coûts de stockage et de lancement. L'agent fournisseur de son tour ne communique pas ses paramètres de coût de production.

L'agent distributeur ainsi n'est pas obligé d'accepter la proposition du médiateur et est libre de la prendre ou de la rejeter.

De plus, La queue des messages des 3 acteurs est ouverte pour tous les messages issus de n'importe quel destinataire, ayant n'importe quelles propriétés mais, chaque agent possède une structure de données que nous avons appelé « messages » comportant des filtres de sélection bien définis pour que l'agent ne considère que les messages qui lui intéressent et ignore tous les messages jugés non-importants.

A titre d'exemple :

- L'agent fournisseur n'accepte pas un ordre client jusqu'à la fin de l'ordre courant.
- L'agent distributeur ne considère pas un « bid request » s'il n'a pas encore lancé le comportement « AwaitForBidRequest ».
- L'agent médiateur ignore un « mediation request » issu du distributeur car c'est l'agent fournisseur seul qui pourra lui demander d'initier le bargaining.

3. Réactivité :

Rappelons que la réactivité consiste à répondre à un stimulus extérieur que l'agent juge important sans raisonnement. Voici quelques exemples illustrant ce concept :

- L'agent fournisseur répond immédiatement à la requête d'offre issue de l'agent distributeur sans avoir effectué aucun raisonnement.
- Il revient ainsi pour approuver l'ordre initial de son client si le bargaining a été annulé avant qu'il termine. Et annule tout comportement lié au bargaining si son client annule sa commande à certain moment.

4. Proactivité :

La proactivité d'un agent étant définie comme lui orienté vers un but précis à atteindre peut être illustré dans notre par les exemples suivants :

- L'agent fournisseur soumet une contre-offre si l'ordre client reçu est invalide, il l'approuve s'il lui convient et demande une médiation s'il n'est pas optimal agissant donc comme une entité *orienté-but* dont l'objectif est de maximiser le profit de son utilisateur.
- Le médiateur lorsqu'il est appelé à initier le bargaining envoie d'abord un message à l'agent distributeur dont le nom et l'adresse sont joints à la requête de l'agent fournisseur pour vérifier d'abord que le distributeur est présent dans la plateforme.
- L'agent distributeur quand il reçoit une contre-offre se questionne en disant que son fournisseur a changé son offre récemment de manière que la quantité demandée a dépassé la quantité maximale supportée par le fournisseur ou bien que son prix de vente a changé. Donc, il lance instantanément une nouvelle requête d'offre en exécutant le comportement « InstantlyUpdateSupplierDetails » avant de soumettre un nouvel ordre.
- L'agent distributeur après avoir reçu l'accord de l'utilisateur pour s'impliquer dans le bargaining envoie un message « AGREE » avec lequel il associe son intervalle toléré et simultanément il communique ses ventes prévues à l'agent fournisseur sans que ces informations soient demandées.
- Les deux acteurs prennent la solution finale fournie par le médiateur comme une proposition à évaluer d'abord par rapport à la commande initial.
- Le médiateur est un agent purement proactif raisonnant dès le début de bargaining jusqu'à sa fin tel qu'il est déjà expliqué dans la section précédente.

5. Comportement social des agents :

La figure ci-dessous montrant une portion d'une conversation entre les trois acteurs du système, elle illustre bien la sociabilité des agents de notre système. Ainsi, à titre d'exemple le message intitulé « **inform** » envoyé par l'agent distributeur vers l'agent fournisseur porte dans son contenu les ventes prévues par le fournisseur. Par contre, dans un paradigme orienté objet cette information est accédée directement.

La sociabilité s'étend ainsi à ce que le mot société lui-même signifie, un ensemble d'individus non seulement interagissant mais gouvernés par une réglementation. Punis quand ils violent les lois et peuvent aussi être récompensés pour leur bonne conduite. Notre Système implémente bien cette propriété. En effet, L'agent lorsqu'il accepte le contrat, il accepte également les engagements qui sont lui associés. Ces engagements sont les suivants :

- L'agent (fournisseur ou distributeur) n'est pas obligé d'accepter la solution du médiateur.
- L'agent (fournisseur ou distributeur) peut quitter le contrat à n'importe quel instant.
- L'agent (fournisseur ou distributeur) n'est pas obligé de partager ses paramètres privés.
- Un retard de 120 secondes d'envoi de l'intervalle de bargaining K ou des paramètres du bid signifie que le bargaining s'est arrêté.
- Toute fraude est punie d'une somme disons de 10 000 DZD.
- Le service de médiation est gratuit.

4.5. Conclusion

A travers ce chapitre nous avons pu mettre en pratique notre solution, pour qu'elle soit exploitée réellement dans la pratique.

Reste à dire qu'un système multi-Agents et du fait qu'il se situe entre l'Intelligence Artificielle qui simule les mécanismes de la vie courante et les systèmes distribués décentralisés, il présente une combinaison parfaite pour arriver à simuler les comportements des individus (autonomes, intelligents, réactifs, proactifs et sociables) tels qu'ils interagissent dans une société selon les lois et les règlements en vigueur.

Conclusion générale

A travers cette étude nous avons pu mettre en avant une approche multi agents par l'un des mécanismes de négociation qui est le bargaining et qui demeure le plus adéquat par rapport au contexte. Ceci pour une chaîne logistique à deux niveaux avec un seul distributeur et un seul fournisseur. Pour la formulation des séquences du bargaining nous nous sommes inspirés des contrats de la chaîne logistique qui s'incluent dans les approches de collaboration et de coordination. Avec une combinaison entre deux contrats qui sont le buyback et le revenu sharing pour assurer la coordination dans le cas décentralisé.

Ce projet comporte plusieurs étapes clés. L'étape la plus importante est la première et qui concerne la définition de la problématique. Elle regroupe deux volets, le premier concerne le SCM où nous nous sommes d'abord familiarisés avec les concepts clés et les principes de ce domaine pour ensuite entamer la coordination de la chaîne logistique pour ressortir les principaux limites et problèmes discutés dans la littérature par rapport à ce volet. Le deuxième volet s'intéresse aux systèmes multi agents, une bonne période était consacrée à leur découverte pour ensuite faire l'analogie entre le SCM et les SMAs pour bien voir l'adéquation de la solution SMAs avec le SCM. Ce qu'il nous a poussé à aller plus loin dans les mécanismes d'interaction, principalement dans la négociation (théorie des jeux, enchères, etc.) vue leurs importance pour faire face aux limites du premier volet. Aussi dans le développement des SMAs à savoir, une vue d'ensemble sur les méthodologies de développements avec la maîtrise de Gaia, et encore les environnements de développements avec la réalisation d'un TP (Travaux pratiques) dans JADE sur l'exemple célèbre dans les SMAs, il s'agit du Book Trading (Annexe)

Après et une fois que nous avons eu une vue d'ensemble sur la problématique nous avons débuté la deuxième étape qui s'intéresse à la définition du problème en précisant un contexte bien définie dans la chaîne logistique. Dans ce sens notre point de départ était l'analyse des études faites dans le LIP6 par l'équipe RO. Après cela, nous avons procédé à la construction de la solution autour du problème. Plusieurs solutions ont été proposées, la première proposition assure la proactivité des agents mais reste totalement centralisée puisque elle regroupe l'EOQ (Wagelmans), le partage de coût de stockage et le buyback contract. La deuxième proposition constitue un pas vers la décentralisation, dans ce sens nous avons gardé les coûts respectifs aux acteurs privés sauf que l'utilité est partagée selon une échelle, cette dernière sera utilisé pour déterminer la solution optimale à travers la théorie des jeux avec l'algorithme de Lemke and Houson algorithme (Codenotti et al 2008). La dernière proposition correspond à celle qui a été présenté dans le rapport (Bargaining avec médiateur).

La troisième étape correspond à la modélisation du système suivant la méthodologie Gaia, ainsi que son implémentation avec la plateforme JADE avec la réalisation des tests et des simulations nécessaires pour s'assurer du bon fonctionnement du système.

Le projet a été mené avec succès vu que nous avons pu modéliser et implémenter la solution malgré la limite du temps. Les tests de la solution ont montré le bon fonctionnement ainsi que les résultats considérables que nous pouvons atteindre. A partir de notre travail nous pouvons distinguer plusieurs apports, à savoir des apports conceptuels et des apports pratiques. Par rapport au premier point notre travail offre une description sur la place des SMAs dans le SCM, une analogie entre les deux aussi bien que la valeur que peut apporter les SMAs dans le SCM, ce qui peut constituer une base sur laquelle peuvent se développer d'autres réflexions et des nouvelles pistes de développements. Pour ce qui concerne l'apport pratique, nous pouvons mettre en avant notre solution qui peut faire l'objet d'un modèle générique dans sans contexte, reste à l'adapté par rapport aux paramètres, à la situation de mise en œuvre et le mode de fonctionnement.

Nous aurions souhaité mener plus de projets dans ce contexte, surtout avec la nouveauté de ces applications qui nous ont ouvert plusieurs portes à chaque avancement. Toutefois, en raison de la contrainte de temps, il a bien fallu clôturer notre travail.

Reste à dire que le modèle comporte plusieurs limites à contourner dans le futur. D'abord ça sera très intéressant d'accompagner le modèle avec un système d'apprentissage automatique de façon que les agents peuvent apprendre de l'environnement dans lequel ils opèrent pour s'adapter avec lui automatiquement, cela peut éliminer plusieurs problèmes comme les aspects comportementaux (fiabilité de l'information, confiance, etc.) mais aussi permet d'améliorer significativement la performance du modèle puisque les résultats seront beaucoup plus réalistes, surtout pour le médiateur pour qu'il s'assure que les acteurs sont honnête dans leurs décisions. Une autre voie d'amélioration très importante est l'extension du modèle vers plusieurs fournisseurs et plusieurs distributeurs (N fournisseurs, M distributeurs) pour représenter plusieurs phénomènes qui se passent dans la réalité dans notre contexte. Nous pouvons citer le regroupement des acteurs pour augmenter leurs profits respectifs et qui peut être résolu par la formation des coalitions dans le cadre de la théorie des jeux coopérative, comme par exemple lors du lancement du plan d'approvisionnement par un distributeur au prêt d'un fournisseur il peut faire appel à d'autres distributeurs qui font leurs commande au prêt du même fournisseur pour bénéficier de l'économie d'échelle. Aussi l'apprentissage des acteurs, l'un de l'autre dans

ce qui concerne le mode de fonctionnement, comme dans le cas d'un nouvel acteur qui rentre et qui essaie de suivre les anciens acteurs, ceci peut se faire à l'aide des algorithmes génétique par exemple.

Enfin, et pour finir le rapport, il faut dire que nous avons eu la chance d'avoir un encadrement d'une qualité exceptionnelle et dans les deux volets de notre travail, que ce soit dans le SCM ou bien dans les SMAs. Cela nous a permis de se développer dans plusieurs sens. A savoir un développement d'un ensemble de compétences très importantes à nos jours, essentiellement la conception des systèmes multi agents, la modélisation des approches de collaboration de la chaîne logistique, aussi bien que la prise de décision collective. De plus que le développement du relationnel surtout lors de la collaboration valorisante avec une équipe de travail référente du LIP6, puisque nous avons l'habitude de travailler dans un cadre managérial d'entreprise alors que là, nous avons eu à travailler avec des chercheurs de haut niveau, et le défi était de bénéficier de la recherche sans autant lâcher l'esprit pratique. Nous pouvons ajouter aussi le développement méthodologique, principalement la démarche à suivre pour mener un travail de recherche, l'importance de la définition du problème et la valorisation du travail.

BIBLIOGRAPHIE

1. Agarwal, Ashish, Dixit Garg, et Ashwini Sharma. « Quality management in supply chains: The literature review ». *International Journal for Quality Research* 6, n° 3 (2012): 193–206.
2. Aknine, Samir, Dhouha Anane, et Suzanne Pinson. « Les approches agents pour la coordination d'activités dans les chaînes graphiques », 2009. <https://hal.archives-ouvertes.fr/hal-00877029/>.
3. Aknine, Samir, Anane Dhouha, et Suzanne Pinson. « Coordination d'Activités dans les Chaînes Logistiques : une Approche Multi-Agents par Formation de Coalitions ». *REVUE INGENIERIE DES SYSTEMES D'INFORMATION*, 2009, 33.
4. Amgoud, Leila, et Martin Caminada. « On the evaluation of argumentation formalisms ». *Artificial Intelligence* 171, n° 5 (2007): 286–310.
5. Anupindi, Ravi, et Yehuda Bassok. « Supply contracts with quantity commitments and stochastic demand ». In *Quantitative models for supply chain management*, 197–232. Springer, 1999. http://link.springer.com/chapter/10.1007/978-1-4615-4949-9_7.
6. Aryanezhad, Mir-Bahador, Maryam Esmaeili, et Panlop Zeepongsekul. « A game theory approach in seller–buyer supply chain ». *European Journal of Operational Research* 195, n° 2 (2009): 442–448.
7. Aviv, Yossi. « The effect of collaborative forecasting on supply chain performance ». *Management science* 47, n° 10 (2001): 1326–1343.
8. Aydoğan, Reyhan, Koen V. Hindriks, et Catholijn M. Jonker. *Multilateral mediated negotiation protocols with feedback*. Springer, 2014. http://link.springer.com/chapter/10.1007/978-4-431-54758-7_3.
9. Banerjee, Avijit. « A joint economic-lot-size model for purchaser and vendor ». *Decision sciences* 17, n° 3 (1986): 292–311.
10. Banerjee, Avijit, Ruo Du, et Seung-Lae Kim. « Coordination of two-echelon supply chains using wholesale price discount and credit option ». *International Journal of Production Economics* 143, n° 2 (2013): 327–334.
11. Barbati, Maria, Giuseppe Bruno, et Andrea Genovese. « Applications of agent-based models for optimization problems: A literature review ». *Expert Systems with Applications* 39, n° 5 (2012): 6020–6028.
12. Belgasmi, Nabil, Khaled Ghédira, et Lamjed Ben Saïd. « Multiobjective Analysis of the Multi-Location Newsvendor and Transshipment Models ». *International Journal of Information Systems and Supply Chain Management (IJISSCM)* 6, n° 4 (2013): 42–60.

13. Bellifemine, Fabio Luigi, Giovanni Caire, et Dominic Greenwood. *Developing multi-agent systems with JADE*. Vol. 7. John Wiley & Sons, 2007.
14. Bellifemine, Fabio, Agostino Poggi, et Giovanni Rimassa. « JADE—A FIPA-compliant agent framework ». In *Proceedings of PAAM*, 99:33. London, 1999. <http://www.cs.unicam.it/merelli/Calcolo/PAAM.pdf>.
15. Briot, Jean-Pierre, et Yves Demazeau. *Principes et architecture des systèmes multi-agents*. Lavoisier., 2016. <http://www.lavoisier.fr/livre/informatique/principes-et-architecture-des-systemes-multi-agents/briot/descriptif-9782746203365>.
16. Briot, Jean-Pierre, et Zahia Guessoum. « From active objects to autonomous agents ». *IEEE concurrency*, n° 3 (1999): 68–76.
17. Cachon, Gérard P., et Martin A. Lariviere. « Contracting to assure supply: How to share demand forecasts in a supply chain ». *Management science* 47, n° 5 (2001): 629–646.
18. Cachon, Gerard P., et Serguei Netessine. « Game theory in supply chain analysis ». In *Handbook of Quantitative Supply Chain Analysis*, 13–65. Springer, 2004. http://link.springer.com/chapter/10.1007/978-1-4020-7953-5_2.
19. Cachon, Gérard P., et Paul H. Zipkin. « Competitive and cooperative inventory policies in a two-stage supply chain ». *Management science* 45, n° 7 (1999): 936–953.
20. Chaib-Draa, Brahim, Sophie D'Amours, et Thierry Moyaux. « Multi-agent coordination based on tokens: Reduction of the bullwhip effect in a forest supply chain ». In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 670–677. ACM, 2003. <http://dl.acm.org/citation.cfm?id=860683>.
21. Chaib-Draa, Brahim, Imed Jarras, et Bernard Moulin. « Systèmes multi-agents: principes généraux et applications ». *Edition Hermès*, 2001, 1030–1044.
22. Chaib-Draa, Brahim, et Bernard Moulin. « An overview of distributed artificial intelligence ». *Foundations of distributed artificial intelligence* 1 (1996): 3–55.
23. Chau, Patrick YK, Zhimin Huang, Susan X. Li, et Joe Zhu. « Cooperative advertising, game theory and manufacturer–retailer supply chains ». *Omega* 30, n° 5 (2002): 347–357.
24. Chen, Frank, Zvi Drezner, Jennifer K. Ryan, et David Simchi-Levi. « Quantifying the bullwhip effect in a simple supply chain: The impact of forecasting, lead times, and information ». *Management science* 46, n° 3 (2000): 436–443.

-
25. Chen, Yao, Wade D. Cook, Juan Du, Liang Liang, et Joe Zhu. « A bargaining game model for measuring performance of two-stage network structures ». *European Journal of Operational Research* 210, n° 2 (2011): 390–397.
 26. Cheng, TC Edwin, Zhimin Huang, Shouyang Wang, Xiaoqi Yang, et Yingxue Zhao. « Coordination of supply chains by option contracts: A cooperative game theory approach ». *European Journal of Operational Research* 207, n° 2 (2010): 668–675.
 27. Cho, Namjae, Shima Mohebbi, et Rasoul Shafaei. « e-Supply Network: The Design of Intelligent Agents for ». *Electronic Supply Network Coordination in Intelligent and Dynamic Environments: Modeling and Implementation: Modeling and Implementation*, 2010, 250.
 28. Christopher, Martin. *Logistics and Supply Chain Management*. Financial Times Prentice Hall, 2011.
 29. Chu, Chi-Leung, et V. Jorge Leon. « Scalable methodology for supply chain inventory coordination with private information ». *European Journal of Operational Research* 195, n° 1 (2009): 262–279.
 30. Chyr, Fuchiao, Chin-Fu Ho, et Tsong-Ming Lin. « A new approach to the dynamic lot size model ». *Engineering costs and production economics* 20, n° 3 (1990): 255–263.
 31. Codenotti, Bruno, Stefano De Rossi, et Marino Pagan. « An experimental analysis of lemke-howson algorithm ». *arXiv preprint arXiv:0811.3247*, 2008. <http://arxiv.org/abs/0811.3247>.
 32. Cook, Robert L., Brian J. Gibson, et John T. Mentzer. « Supply chain management: the pursuit of a consensus definition ». *Journal of business logistics* 26, n° 2 (2005): 17–25.
 33. Cost, Scott, Ye Chen, Bill Chu, Tim Finin, Yannis Labrou, Yun Peng, Rongming Sun, et R. Willhelm. « A negotiation-based multi-agent system for supply chain management ». *Working Notes of the agents* 99 (1999).
 34. Costas, José, David de la Fuente, Raúl Pino, Borja Ponte, et Julio Puche. « Applying Goldratt's Theory of Constraints to reduce the Bullwhip Effect through agent-based modeling ». *Expert Systems with Applications* 42, n° 4 (2015): 2049–2060.
 35. Dada, Maqbool, et Nicholas C. Petruzzi. « Pricing and the newsvendor problem: A review with extensions ». *Operations Research* 47, n° 2 (1999): 183–194.
 36. D'Amours, Sophie, Pascal Forget, et Jean-Marc Frayret. « Multi-behavior agent model for planning in supply chains: An application to the lumber industry ». *Robotics and Computer-Integrated Manufacturing* 24, n° 5 (2008): 664–679.

-
37. D'Amours, Sophie, Pascal Forget, Jean-Marc Frayret, Steve Harvey, Alain Rousseau, et Pascale Plamondon. « Agent-based supply chain planning in the forest products industry ». In *Information Technology For Balanced Manufacturing Systems*, 17–26. Springer, 2006.
 38. Davis, Randall, et Reid G. Smith. « Negotiation as a metaphor for distributed problem solving ». *Artificial intelligence* 20, n° 1 (1983): 63–109.
 39. Dayong, Sang, et Wang Ying. « Multi-agent framework for third party logistics in E-commerce ». *Expert Systems with Applications* 29, n° 2 (2005): 431–436.
 40. Demazeau, Yves. « From interactions to collective behaviour in agent-based systems ». In *In: Proceedings of the 1st. European Conference on Cognitive Science. Saint-Malo*. Citeseer, 1995. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.47.7968>.
 41. Denko, Mieso K., Tao Sun, et Isaac Woungang. « Trust management in ubiquitous computing: A Bayesian approach ». *Computer Communications* 34, n° 3 (2011): 398–406.
 42. Dennis, Nacken, Gasquet Olivier, et Hans-Jörg Naumer. « Les vagues longues de la prospérité ou le sixième cycle de Kondratiev ». Allianz Global Investors. Consulté le 9 juin 2016.
 43. Dewitt, William, James S. Keebler, John T. Mentzer, Soohong Min, Nancy W. Nix, Carlo D. Smith, et Zach G. Zacharia. « Définir le supply chain management ». *Logistique & Management* 9, n° 2 (2001): 3–18.
 44. DeWitt, William, James S. Keebler, John T. Mentzer, Soonhong Min, Nancy W. Nix, Carlo D. Smith, et Zach G. Zacharia. « Defining supply chain management ». *Journal of Business logistics* 22, n° 2 (2001): 1–25.
 45. Dodd, Chad, et Soundar RT Kumara. *A distributed multi-agent model for value nets*. Springer, 2001. http://link.springer.com/chapter/10.1007/3-540-45517-5_79.
 46. Durand, Rodolphe, Zahia Guessoum, et Lilia Rejeb. « Using adaptive multi-agent systems to simulate economic models ». In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 1*, 68–75. IEEE Computer Society, 2004. <http://dl.acm.org/citation.cfm?id=1018795>.
 47. El Azizi, Mohamed Berrada. « Modélisation multi-agents de la coopération au sein des chaînes logistiques à deux échelons: application à la distribution de produits pharmaceutiques au Maroc ». Université Paris-Nord-Paris XIII, 2014. <https://tel.archives-ouvertes.fr/tel-01227979/>.
 48. Ertogral, Kadir, et S. David Wu. « A bargaining game for supply chain contracting ». *preprint*, 2001.

-
49. ESPINASSE, Bernard. « Coordination et Négociation dans les Systèmes Multi-Agents ». Consulté le 9 juin 2016. <http://www.lsis.org/espinasseb/Supports/SMAI-2008/SMA-BE5-2012-4p.pdf>.
 50. Evenchik, Amir, Sarit Kraus, et Katia Sycara. « Reaching agreements through argumentation: a logical model and implementation ». *Artificial Intelligence* 104, n° 1 (1998): 1–69.
 51. Eymann, Torsten. « Co-evolution of bargaining strategies in a decentralized multi-agent system ». In *AAAI fall 2001 symposium on negotiation methods for autonomous cooperative systems*, 126–134, 2001.
 52. Faratin, Peyman, Nicholas R. Jennings, Alessio R. Lomuscio, Simon Parsons, Carles Sierra, et Michael J. Wooldridge. « Automated negotiation: prospects, methods and challenges ». *Group Decision and Negotiation* 10, n° 2 (2001): 199–215.
 53. Ferber, Jacques. « Les systèmes multi-agents: un aperçu général ». *Techniques et sciences informatiques* 16, n° 8 (1997).
 54. Florea, Adina Magda. « Agents et systèmes multi-agents ». *Université Politehnica de Bucarest*. Consulté le 9 juin 2016.
 55. Forrester, Jay W. « Industrial dynamics: a major breakthrough for decision makers ». *Harvard business review* 36, n° 4 (1958): 37–66.
 56. Gan, Xianghua, Suresh P. Sethi, et Houmin Yan. « Coordination of supply chains with risk-averse agents ». In *Supply Chain Coordination under Uncertainty*, 3–31. Springer, 2011.
 57. Ghodsypour, Seyed Hassan, Ashkan Hafezalkotob, et Davood Mohammaditabar. « A game theoretic analysis in capacity-constrained supplier-selection and cooperation by considering the total supply chain inventory costs ». *International Journal of Production Economics*, 2015.
 58. Gosling, James. *The Java language specification*. Addison-Wesley Professional, 2000.
 59. Goyal, S. K. « An integrated inventory model for a single supplier-single customer problem ». *The International Journal of Production Research* 15, n° 1 (1977): 107–111.
 60. Grzybowska, Katarzyna, et Gabor Kovacs. « Supply Chain Coordination between Autonomous Agents—a game theory approach ». In *Computer Science and Information Systems (FedCSIS), 2015 Federated Conference on*, 1623–1630. IEEE, 2015.

-
61. Guangxing, Wei, Qin Yanhong, et Yin Yaxian. « Buy Back Contract Considering Fairness Preference under the Framework of Nash-Bargain in a Two Stage Supply Chain ». Consulté le 9 juin 2016.
 62. Guessoum, Zahia. « Un environnement opérationnel de conception et de réalisation de systèmes multi-agents ». Université Pierre et Marie Curie, 1996.
 63. Gunasekaran, A., A. Spalanzani, et I. Zouaghi. *Impact of supply chain relational interactions on information sharing*, 2011. <https://halshs.archives-ouvertes.fr/halshs-00665920>.
 64. Haghpanah, Yasaman. « A trust model for supply chain management ». In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 3*, 1375–1376. International Foundation for Autonomous Agents and Multiagent Systems, 2011. <http://dl.acm.org/citation.cfm?id=2034573>.
 65. Harsanyi, John C., others, et Reinhard Selten. « A general theory of equilibrium selection in games ». *MIT Press Books 1* (1988).
 66. Haucap, Justus, Ulrich Heimeshoff, Gordon J. Klein, Dennis Rickert, et Christian Wey. *Bargaining Power in Manufacturer-Retailer Relationships*. 107. DICE Discussion Paper, 2013.
 67. Höhn, Michaela Isabel. « Literature Review on Supply Chain Contracts ». In *Relational Supply Contracts*, 19–34.
 68. Hou, Jing, Amy Z. Zeng, et Lindu Zhao. « Achieving better coordination through revenue sharing and bargaining in a two-stage supply chain ». *Computers & Industrial Engineering* 57, n° 1 (2009): 383–394.
 69. Huang, Lihua, Sijie Li, et Zhanbei Zhu. « Supply chain coordination and decision making under consignment contract with revenue sharing ». *International Journal of Production Economics* 120, n° 1 (2009): 88–99.
 70. Hwang, Hark-Chin, et Wilco van den Heuvel. « Improved algorithms for a lot-sizing problem with inventory bounds and backlogging ». *Naval Research Logistics (NRL)* 59, n° 3-4 (2012): 244–253.
 71. Jaber, Mohamad Y., et Ibrahim H. Osman. « Coordinating a two-level supply chain with delay in payments and profit sharing ». *Computers & Industrial Engineering* 50, n° 4 (2006): 385–400.

-
72. Jennings, Nicholas R. « An agent-based approach for building complex software systems ». *Communications of the ACM* 44, n° 4 (2001): 35–41.
73. Jennings, Nicholas R., Katia Sycara, et Michael Wooldridge. « A roadmap of agent research and development ». *Autonomous agents and multi-agent systems* 1, n° 1 (1998): 7–38.
74. Jiang, Y. C., Z. Y. Xia, Y. P. Zhong, et S. Y. Zhang. « Autonomous trust construction in multi-agent systems—a graph theory methodology ». *Advances in Engineering software* 36, n° 2 (2005): 59–66.
75. Kahraman, Cengiz, et Sezi Çevik Onar. *Intelligent Techniques in Engineering Management*. Springer, 2015.
76. Kedad-Sidhoum Safia, Pascual Fanny, Phouratsamay Siao-Leu « *Coordination des décisions de planification dans une chaîne logistique* », ROADEF 2015, Marseille, France.
77. Kedad-Sidhoum Safia, Pascual Fanny, Phouratsamay Siao-Leu « *Solving two-echelon lot sizing problems with inventory bounds* », IWLS 2015, Montréal, Canada.
78. Kersten, Wolfgang. *Pioneering Solutions in Supply Chain Management: A Comprehensive Insight Into Current Management Approaches*. Erich Schmidt Verlag GmbH & Co KG, 2010.
79. Kolen, Antoon, Stan Van Hoesel, et Albert Wagelmans. « Economic lot sizing: an $O(n \log n)$ algorithm that runs in linear time in the Wagner-Whitin case ». *Operations Research* 40, n° 1-1 (1992): S145–S156.
80. Kumar, Vivek, et S. Srinivasan. « A review of supply chain management using multi-agent system ». *International Journal of Computer Science Issues* 7, n° 5 (2010): 198–205.
81. Lai, Kin Keung, Stephen CH Leung, et Zhong Yao. « Manufacturer’s revenue-sharing contract and retail competition ». *European Journal of Operational Research* 186, n° 2 (2008): 637–651.
82. Lambert, Douglas M. *Supply chain management: processes, partnerships, performance*. Supply Chain Management Inst, 2008.
83. Lambert, Douglas M., et Martha C. Cooper. « Issues in supply chain management ». *Industrial marketing management* 29, n° 1 (2000): 65–83.
84. Lariviere, Martin A., et Evan L. Porteus. « Selling to the newsvendor: An analysis of price-only contracts ». *Manufacturing & service operations management* 3, n° 4 (2001): 293–305.

-
85. Lee, Hau L., Venkata Padmanabhan, et Seungjin Whang. « Information distortion in a supply chain: The bullwhip effect ». *Management science* 50, n° 12_supplement (2004): 1875–1886.
86. Lee, Hau L., et Seungjin Whang. « Information sharing in a supply chain ». *International Journal of Manufacturing Technology and Management* 1, n° 1 (2000): 79–93.
87. Lee, Hau, et Seungjin Whang. « Decentralized multi-echelon supply chains: Incentives and information ». *Management Science* 45, n° 5 (1999): 633–640.
88. Leyton-Brown, Kevin, et Yoav Shoham. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.
89. Li, Xiuhui, et Qinan Wang. « Coordination mechanisms of supply chain systems ». *European journal of operational research* 179, n° 1 (2007): 1–16.
90. Li, Xueping, et Shima Mohebbi. « Coalitional game theory approach to modeling suppliers' collaboration in supply networks ». *International Journal of Production Economics* 169 (2015): 333–342.
91. Lin, Fu-ren, et You-yu Lin. « Integrating Multi-Agent Negotiation to Resolve Constraints in Fulfilling Supply Chain Orders ». *Electronic Commerce Research and Applications* 5, n° 4 (décembre 2006): 313–22. doi:10.1016/j.elerap.2006.04.003.
92. Lin, Qiang, Yanhong Qin, et Guangxing Wei. « A new buy-back contract coordinating dual-channel supply Chain under stochastic demand ». *International Journal of Computer Science Issues* 10, n° 2 (2013): 637–643.
93. Lorentziadis, Panos L. « Optimal bidding in auctions from a game theory perspective ». *European Journal of Operational Research* 248, n° 2 (2016): 347–371.
94. Lu, Meng, Suresh Sethi, et Houmin Yan. « Robustness of Supply Chain Coordination Contracts: Taxonomy, Examples, and Structural Results ». *Examples, and Structural Results (April 1, 2015)*, 2015.
95. Lummus, Rhonda R., et Robert J. Vokurka. « Defining supply chain management: a historical perspective and practical guidelines ». *Industrial Management & Data Systems* 99, n° 1 (1999): 11–17.
96. MANDIAU, René, et Emmanuelle STRUGEON. « Systèmes multiagents ». *Techniques de l'ingénieur. Informatique industrielle* 1, n° S7216 (2002): S7216–1.
97. Manzini, Paola, et Marco Mariotti. « Two-stage bargaining solutions ». *Available at SSRN 924738*, 2008.

-
98. Metters, Richard. « Quantifying the bullwhip effect in supply chains ». *Journal of operations management* 15, n° 2 (1997): 89–100.
 99. Minner, Stefan. « Multiple-supplier inventory models in supply chain management: A review ». *International Journal of Production Economics* 81 (2003): 265–279.
 100. Mohebbi, Shima, et Rasoul Shafaei. « e-Supply network coordination: the design of intelligent agents for buyer-supplier dynamic negotiations ». *Journal of intelligent Manufacturing* 23, n° 3 (2012): 375–391.
 101. Nagarajan, Mahesh, et Greys Sošić. « Game-theoretic analysis of cooperation among supply chain agents: Review and extensions ». *European Journal of Operational Research* 187, n° 3 (2008): 719–745.
 102. Neyret, Alexandre, et Jinxing Xie. « Co-op advertising and pricing models in manufacturer–retailer supply chains ». *Computers & Industrial Engineering* 56, n° 4 (2009): 1375–1385.
 103. Nilsson, Nils J. *The quest for artificial intelligence*. Cambridge University Press, 2009.
 104. Nisan, Noam, Tim Roughgarden, Eva Tardos, et Vijay V. Vazirani. *Algorithmic game theory*. Vol. 1. Cambridge University Press Cambridge, 2007.
 105. Norvig, Peter, et Stuart J. Russell. *Artificial Intelligence: A Modern Approach*. 2^e éd. Pearson Education, 2003.
 106. Nudelman, Eugene, Ryan Porter, et Yoav Shoham. « Simple search methods for finding a Nash equilibrium ». *Games and Economic Behavior* 63, n° 2 (2008): 642–662.
 107. Papazoglou, Mike P. « Agent-oriented technology in support of e-business ». *Communications of the ACM* 44, n° 4 (2001): 71–77.
 108. Parunak, H. Van Dyke. « Applications of distributed artificial intelligence in industry ». *Foundations of distributed artificial intelligence* 2 (1996).
 109. Pendharkar, Parag C. « Game theoretical applications for multi-agent systems ». *Expert Systems with Applications* 39, n° 1 (2012): 273–279.
 110. Prakash, Sumit, Ajit Singh, et Pankaj Singh Sammal. « Implementation of Distributed Multi Agent System using JADE Platform ». *International Journal of Computer Applications* 105, n° 1 (2014).
 111. Quiggin, John. « A theory of anticipated utility ». *Journal of Economic Behavior & Organization* 3, n° 4 (1982): 323–343.

-
112. Radu, Serban. « An Adaptive Negotiation Multi-Agent System for e-Commerce Applications ». Consulté le 9 juin 2016.
113. Raff, Horst, et Nicolas Schmitt. « Manufacturers and retailers in the global economy », 2011. http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1883402.
114. Raiffa, Howard. *The art and science of negotiation*. Harvard University Press, 1982.
115. Russell, Stuart J., et Peter Norvig. *Artificial Intelligence: A Modern Approach 3rd edition*, 2009.
116. Sanchez-Palencia, Evariste. *Le dilemme du prisonnier revisité et la méthode dialectique*. Récupéré du site de l'Institut de France/Académie des sciences: http://www.academiesciences.fr/activite/hds/textes/lpa_Sanchez1.pdf, 2013.
117. Schneeweiss, Christoph, et Kirstin Zimmer. « Hierarchical coordination mechanisms within the supply chain ». *European Journal of Operational Research* 153, n° 3 (2004): 687–703.
118. Searle, John R. *Speech acts: An essay in the philosophy of language*. Vol. 626. Cambridge university press, 1969.
119. Shapiro, Jeremy F. « Modeling and IT perspectives on supply chain integration ». *Information Systems Frontiers* 3, n° 4 (2001): 455–464.
120. Shubik, Martin. *Game theory in the social sciences: Concepts and solutions*. Vol. 155. JSTOR, 1982. <http://www.jstor.org/stable/pdf/1911883.pdf>.
121. Smith, Reid G. « The contract net protocol: High-level communication and control in a distributed problem solver ». *IEEE Transactions on computers*, n° 12 (1980): 1104–1113.
122. Snyder, Lawrence V., et Zuo-Jun Max Shen. *Fundamentals of supply chain theory*. John Wiley & Sons, 2011.
123. Song, Shiji, Cheng Wu, et Long Zhang. « Supply chain coordination of loss-averse newsvendor with contract ». *Tsinghua Science & Technology* 10, n° 2 (2005): 133–140.
124. Spalanzani, A., et I. Zouaghi. « Les effets de la dynamique relationnelle inter-organisationnelle sur l'alignement des systèmes d'information au sein d'une supply chain », 2011.
125. Stadtler, Hartmut. « Supply chain management and advanced planning—basics, overview and challenges ». *European journal of operational research* 163, n° 3 (2005): 575–588.

-
126. Sterman, John D. *Business dynamics: systems thinking and modeling for a complex world*. Vol. 19. Irwin/McGraw-Hill Boston, 2000.
127. Sucky, Eric. « A single buyer-single supplier bargaining problem with asymmetric information-theoretical approach and software implementation ». In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, 10–pp. IEEE, 2003.
128. Tsay, Andy A. « The quantity flexibility contract and supplier-customer incentives ». *Management science* 45, n° 10 (1999): 1339–1358.
129. Tubbs, Stewart L. *A Systems Approach to Small Group Interaction*. Addison-Wesley, 1984.
130. Von Stengel, Bernhard. « Computing equilibria for two-person games ». *Handbook of game theory with economic applications* 3 (2002): 1723–1759.
131. Weibull, Jörgen W. *Evolutionary game theory*. MIT press, 1997.
132. Weiss, Gerhard. *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT press, 1999.
133. Weng, Z. Kevin. « The power of coordinated decisions for short-life-cycle products in a manufacturing and distribution supply chain ». *IIE Transactions* 31, n° 11 (1999): 1037–1049.
134. Wooldridge, Michael. *An introduction to multiagent systems*. John Wiley & Sons, 2009.
135. Zangwill, Willard I. « A backlogging model and a multi-echelon model of a dynamic economic lot size production system-a network approach ». *Management Science* 15, n° 9 (1969): 506–527.
136. Zaraté, Pascale. *Tools for collaborative decision-making*. John Wiley & Sons, 2013.
137. Zouaghi, Iskander. « Maturité supply chain des entreprises : conception d'un modèle d'évaluation et mise en oeuvre ». Phdthesis, Université de Grenoble, 2013. <https://tel.archives-ouvertes.fr/tel-01015950/document>.

LISTE DES ANNEXES

Annexe 1 : Algorithme de Wagelmans et al (1992) pour le problème de lot-sizing EOQ

Annexe 2 : Vue générale sur JADE

Annexe 3 : Mode d'emploi JADE

Annexe 4 : Les outils de JADE

Annexe 5 : FIPA-ACL Performatifs

Annexe 6: Book trading project

Annexe 7: Modélisation

Annexe 1 : Algorithme de Wagelmans et al (1992) pour le problème de lot-sizing EOQ :

Au début le distributeur va déterminer la taille des lots des périodes à venir t , qui satisfait la demande à un coût minimal.

Après ils les envoient au fournisseur qui lui de son côté fait la même chose.

La formulation mathématique du problème est la suivante :

- Les différents coûts considérer dans l'étude à chaque **période t** , sont :

p_t : le coût de production, h_t : le coût de stockage et f_t : le coût de lancement.

- Les variables de décisions sont :

x_t : la production, s_t : le stockage, y_t (**binaire**): le lancement et d_{in} : la demande a la période i .

$$\text{Minimize } \sum_{i=1}^n (p_i x_i + f_i y_i + h_i s_i)$$

subject to

$$x_i + s_{i-1} - s_i = d_i \quad \text{for } i = 1, \dots, n$$

$$d_{in} y_i - x_i \geq 0 \quad \text{for } i = 1, \dots, n$$

$$s_0 = s_n = 0$$

$$x_i \geq 0, \quad s_i \geq 0, \quad y_i \in \{0, 1\} \quad \text{for } i = 1, \dots, n.$$

Figure 67: Algorithme de Wagelmans - Initialisation (Wagelmans et al 1992)

L'**algorithme de résolution** choisit est bien celui de Wagelmans et al (1992) et qui demeure le meilleur pour ce problème avec une complexité de $O(n \log n)$, il se présente comme suit :

Entrées : $p, h \in \mathbb{R}^n / f, d \in \mathbb{R}^n \geq 0$

Sorties : $x \in \mathbb{R}^n, y \in B^n$

Initialisation :

- Calculer $c_i = p_i + \sum_{t=i}^n h_i$ $i=1, \dots, n$.
- $G(i)$ Le coût de la solution optimale dans un horizon de planification de périodes 't' a 'n'.

$$G(i) = \text{Min } f_i + c_i d_{i,t-1} + G(t) \quad \text{if } d_i > 0$$

$$\text{Min}[G(i+1), \text{min } f_i + c_i d_{i,t-1} + G(t) \quad \text{if } d_i = 0$$

Avec $G(n+1) = 0$;

- Trouver l'ensemble des périodes efficaces L.

D'abord il faut définir la fonction $(g(z))$ qui correspond au coût de la fraction de demande (z) satisfaite par la production (x) : $z = \frac{x}{d}$

Les valeurs de (z) ou $(z, g(z)) = (d_{in}, G(i))$ sont notées par : $(p)=1, \dots, r$;

$t(p)$: correspond aux périodes efficaces.

- L'ensemble $t(q)$ est l'ensemble des périodes efficaces ou la pente de (g) est inférieur à celle de (c_i) .

$$q = \min \left[r, \min \left\{ p \text{ avec } 1 \leq p < r \text{ et } \frac{G(t(p)) - G(t(p+1))}{d_{t(p),t(p+1)-1}} < c_t \right\} \right]$$

- Pour le passage de $G(i)$ a $G(i-1)$, il faut déterminer le nouvel ensemble des périodes efficaces,

pour le faire on définit : $t(s)$ qui correspond a la période efficace la plus petite pour laquelle la pente du segment $(d_{in}, G(i))$, $(d_{t(s),n}, G(t(s)))$ est supérieur a $(d_{t(s),n}, G(t(s)))$, $(d_{t(s+1),n}, G(t(s+1)))$, pour cela le nouveau ensemble est bien : les périodes i + les périodes de $t(s)$ a $t(r)$.

$$s = \min \left[q, \min \left\{ p \text{ avec } 1 \leq p < q, d_{i,t(p)-1} > 0 \text{ et } \frac{G(i) - G(t(p))}{d_{i,t(p)-1}} > \frac{G(t(p)) - G(t(p+1))}{d_{t(p),t(p+1)-1}} < c_t \right\} \right]$$

Itérations :

```

For  $i := n$  down to 1 do
begin
  search for  $q(i) := \min \left[ n + 1, \right.$ 
     $\min \left\{ p \in L \mid p < n + 1 \right.$ 
       $\left. \text{and } \frac{G(p) - G(l(p))}{d_{pn} - d_{l(p),n}} < c_i \right\} \left. \right],$ 
   $G(i) := f_i + c_i [d_{in} - d_{q(i),n}] + G(q(i)),$ 
  if ( $d_i = 0$  and  $G(i + 1) < G(i)$ ) then,
    begin
       $G(i) := G(i + 1)$ 
       $s := l(i + 1)$ 
    end
  else
    begin
      if  $d_i > 0,$  then  $s := i + 1$  else  $s := l(i + 1)$ 
      while
         $\left( \frac{G(i) - G(s)}{d_{in} - d_{sn}} \leq \frac{G(s) - G(l(s))}{d_{sn} - d_{l(s),n}} \text{ and } s < q(i) \right)$ 
          do  $s := l(s)$ 
        end
      delete all  $p$  with  $i + 1 \leq p < s$  from  $L$ 
      add  $i$  to  $L$ 
    end.

```

Figure 68: Algorithme de Wagelmans - Itérations itérations. (Wagelmans et al 1992).

Construction de la solution optimale :

```

 $i := 1$ 
while  $i \leq n$  do
  if ( $d_i = 0$  and  $G(i) = G(i + 1)$ ), then  $i := i + 1$  else
    begin
       $x_i := d_{in} - d_{q(i),n}$ 
       $y_i := 1$ 
       $i := q(i)$ 
    end.

```

Figure 69: Algorithme de Wagelmans - Construction de la solution (Wagelmans et al 1992)

Annexe 2 : Vue générale sur JADE

Main gui (Remote monitoring agent).

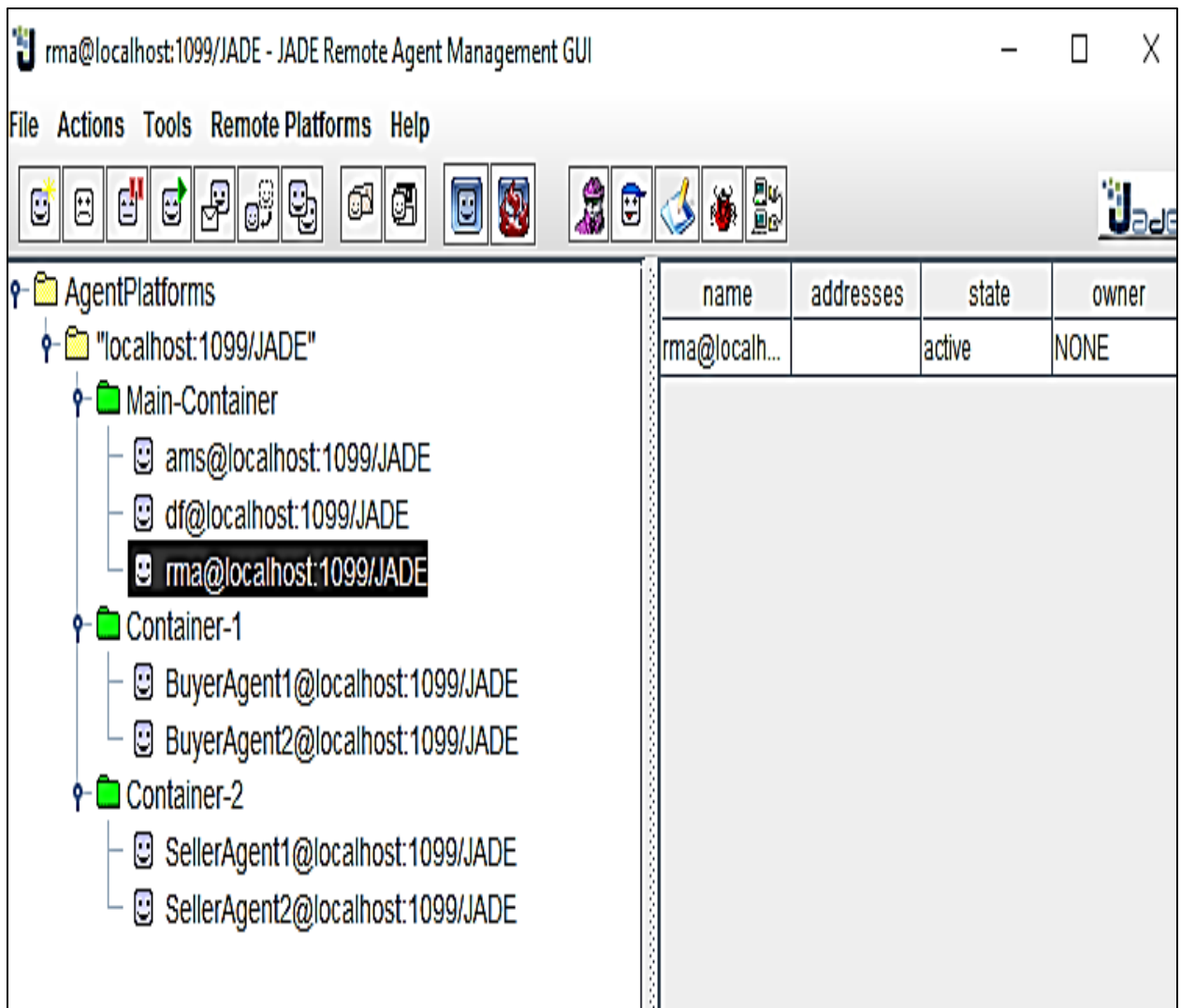


Figure 70: L'interface principale de JADE.

Les messages JADE :

```
send(message_1,msg1_ontology, msg1_content, msg1_performative, msg1_receiver);  
send(message_2,msg2_ontology, msg2_content, msg2_performative, msg2_receiver);  
send(message_3,msg3_ontology, msg3_content, msg3_performative, msg3_receiver);
```

```
//setting up message parameters  
message.setOntology("marketing");  
message.setContent("message content");  
message.setPerformative(ACLMessage.REQUEST);  
//adding receivers  
message.add(receiver _1);  
message.add(receiver _2);  
message.add(receiver _3);  
//sending the message  
send(message);
```

Figure 71: Les messages de JADE.

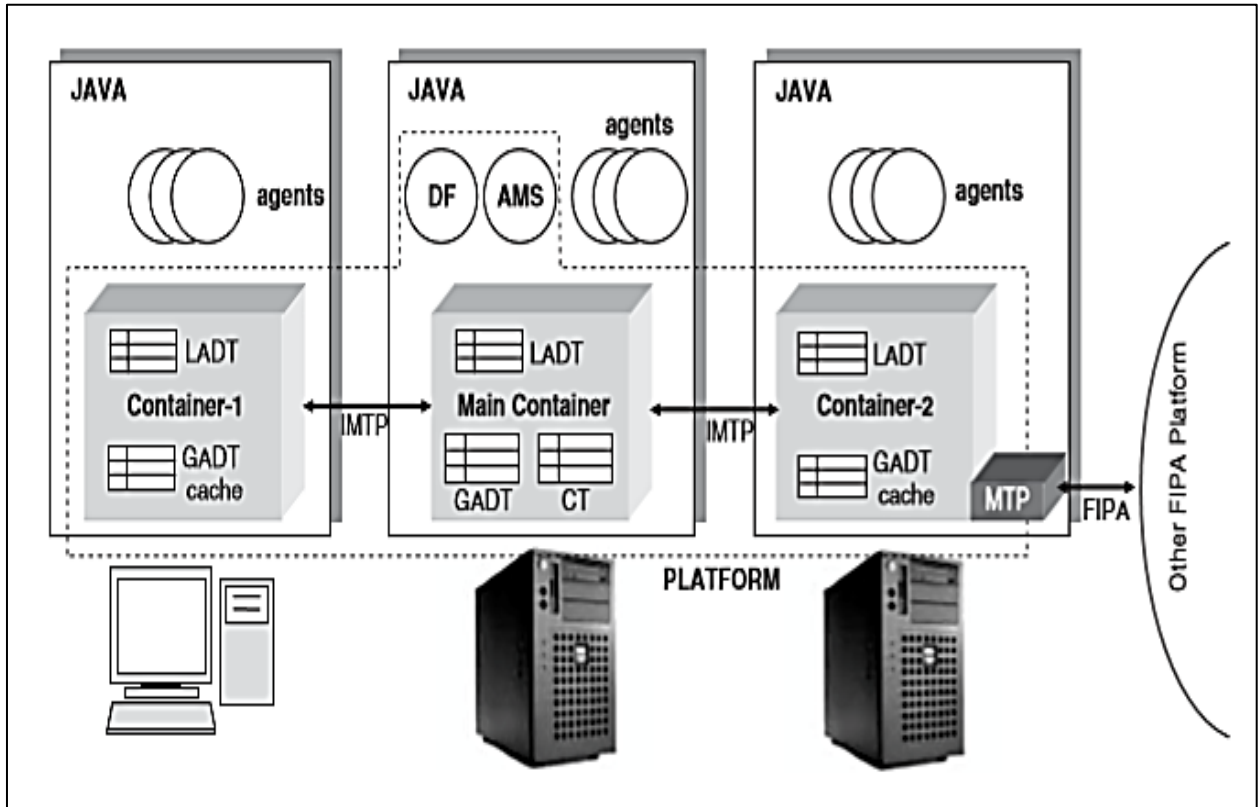


Figure 72: Architecture JADE (Bellifemine et al, 2004)

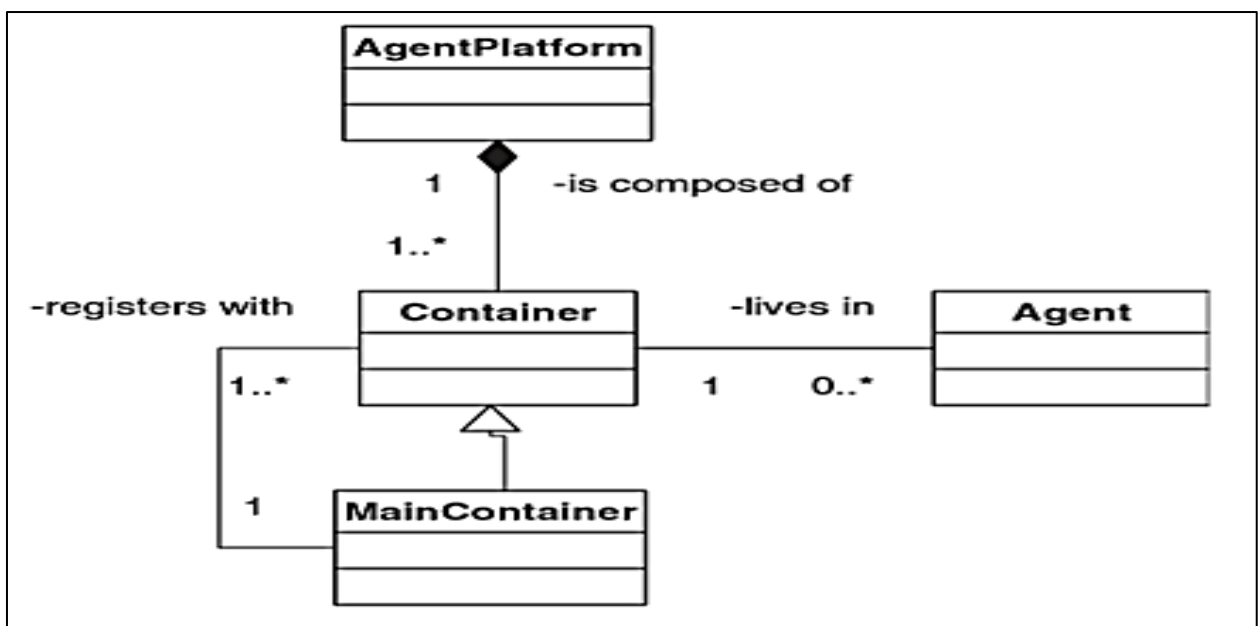


Figure 73: Relation entre les éléments de l'architecture (Bellifemine et al, 2004)

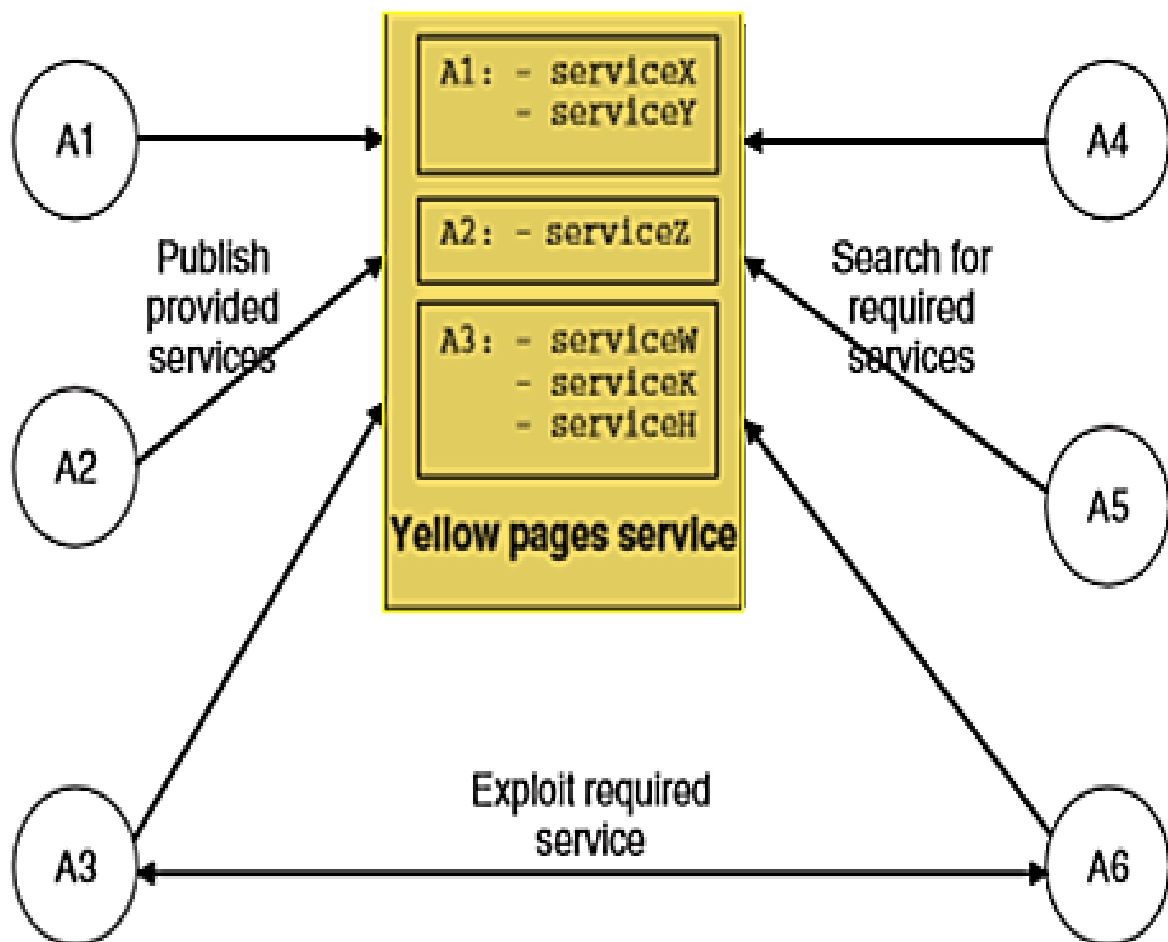


Figure 74: Les services de la page jaune (Bellifemine et al, 2004)

Annexe 3 : Mode d'emploi JADE

Installation et configuration :

Sources disponibles à : <http://jade.tilab.com>

Ajouter dans le classpath les archives *java* (*.jar*) se trouvant dans le répertoire *lib*

Archives nécessaires :

jade.jar
jadeTools.jar

Archives optionnelles en fonction du type de communication inter-plateformes :

http.jar
iiop.jar

Lancement de base :

java jade.Boot [liste agents]

Lancement avec interface graphique :

java jade.Boot -gui [liste agents]

Lancement d'un agent :

java jade.Boot
myAgt:myPackage.myAgent

Annexe 4 : Les outils de JADE

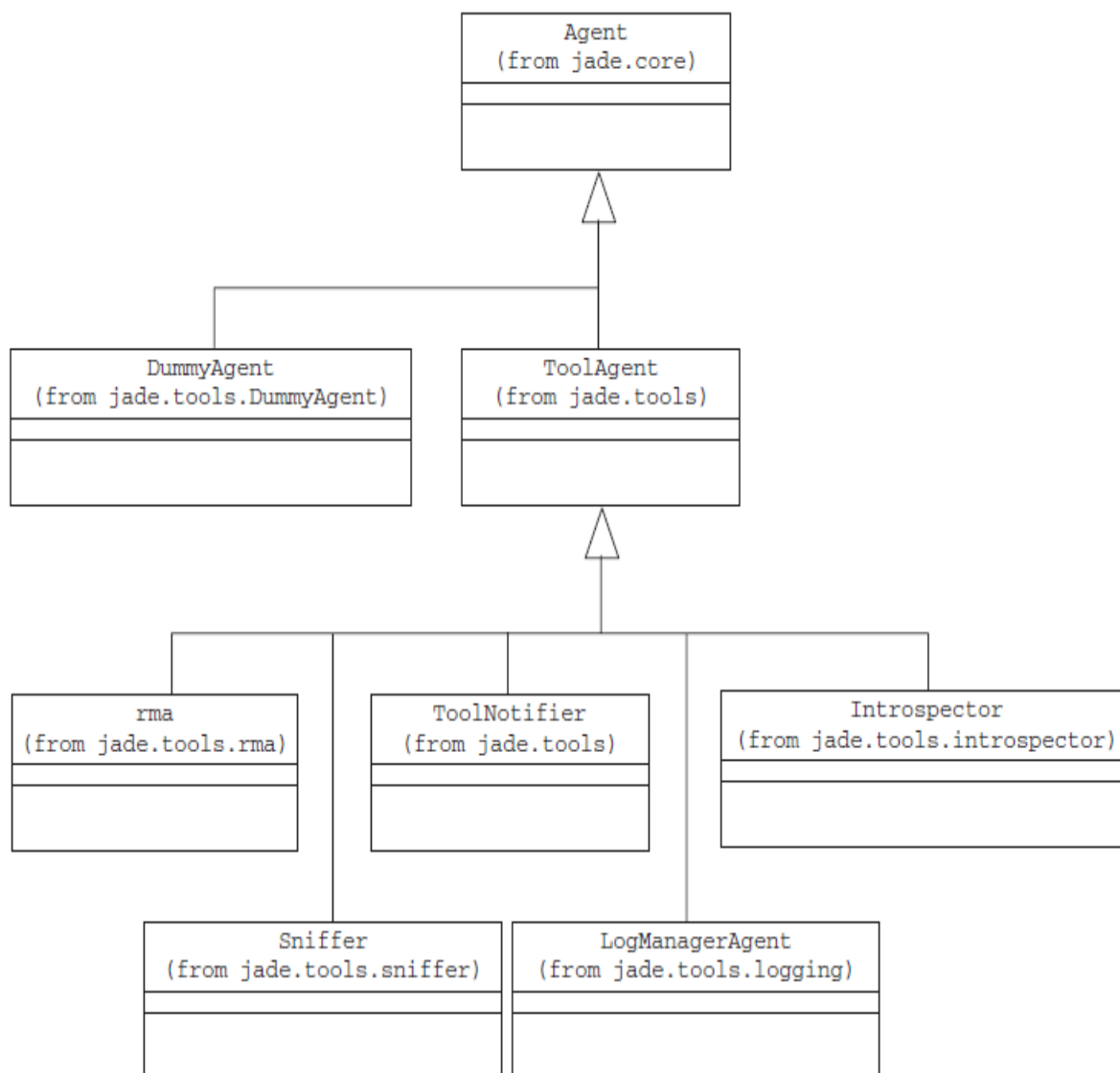


Figure 75: Le diagramme de class des outils de JADE (Bellifemine et al, 2004)

1. Remote Monitoring Agent (RMA) :

Lancement:

Par l'option `-gui` en ligne de commande

Ou `java jade.Boot monInterface:jade.tools.rma.rma`

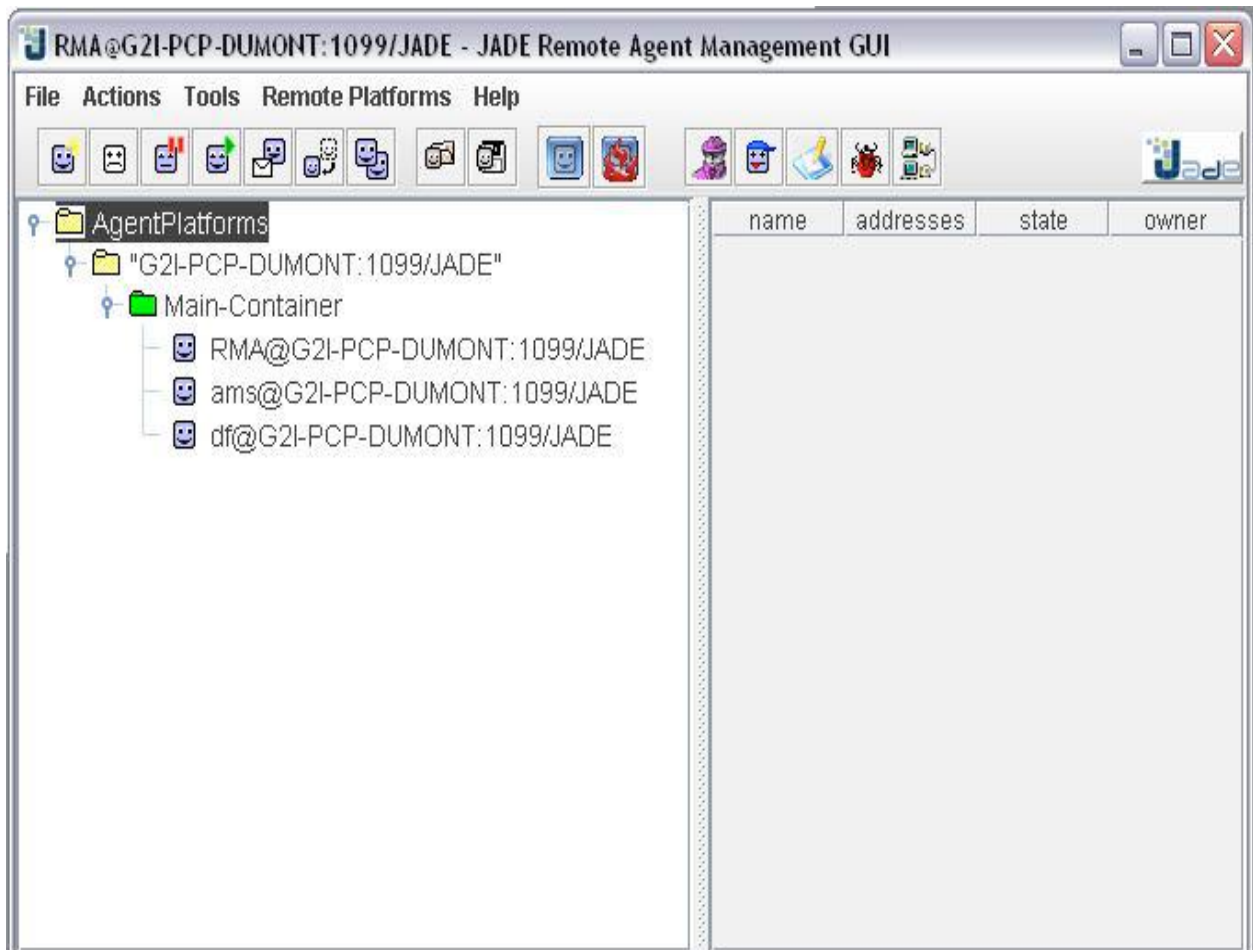


Figure 76: JADE RMA GUI

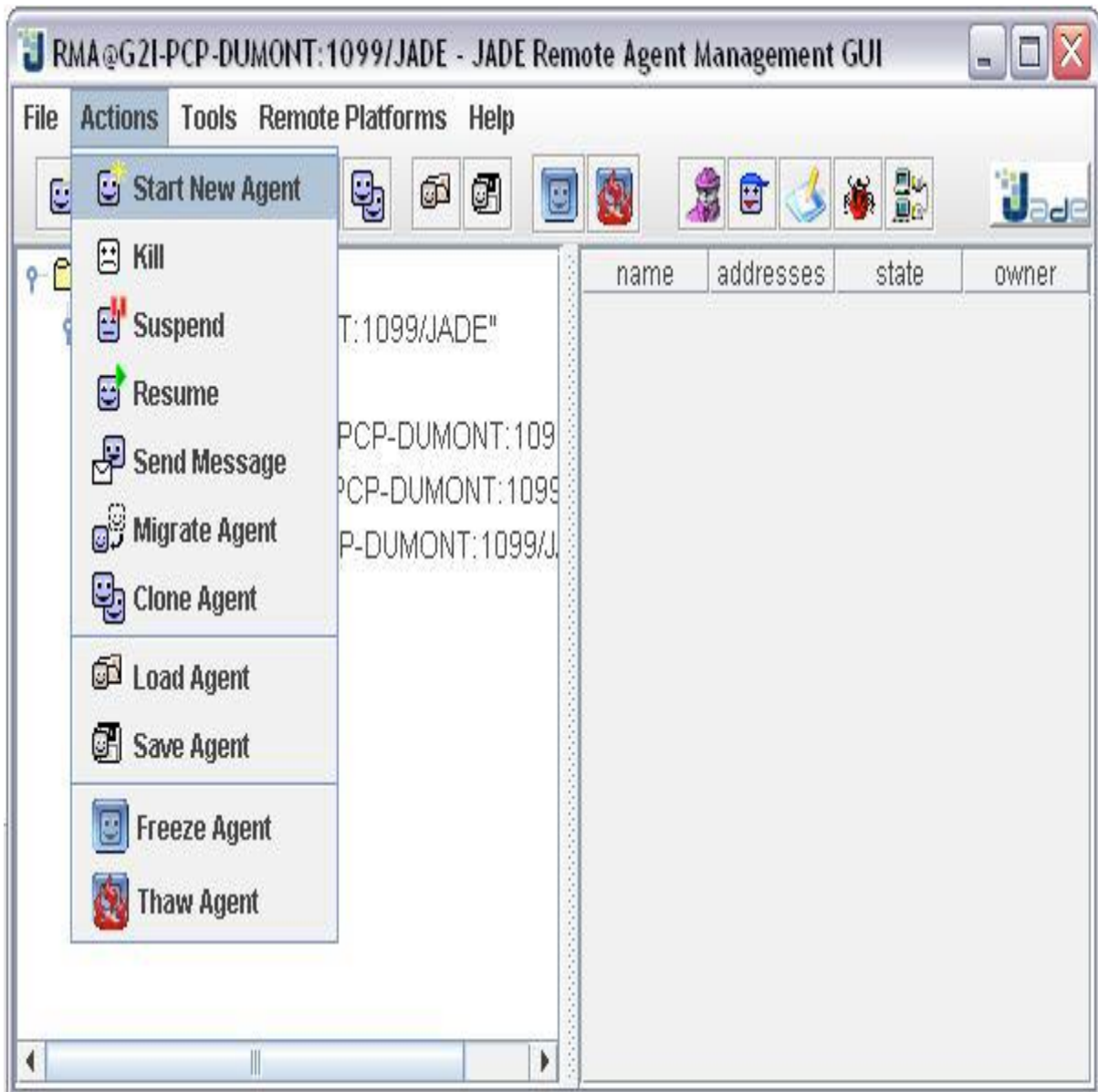


Figure 77: Démarrer un agent.

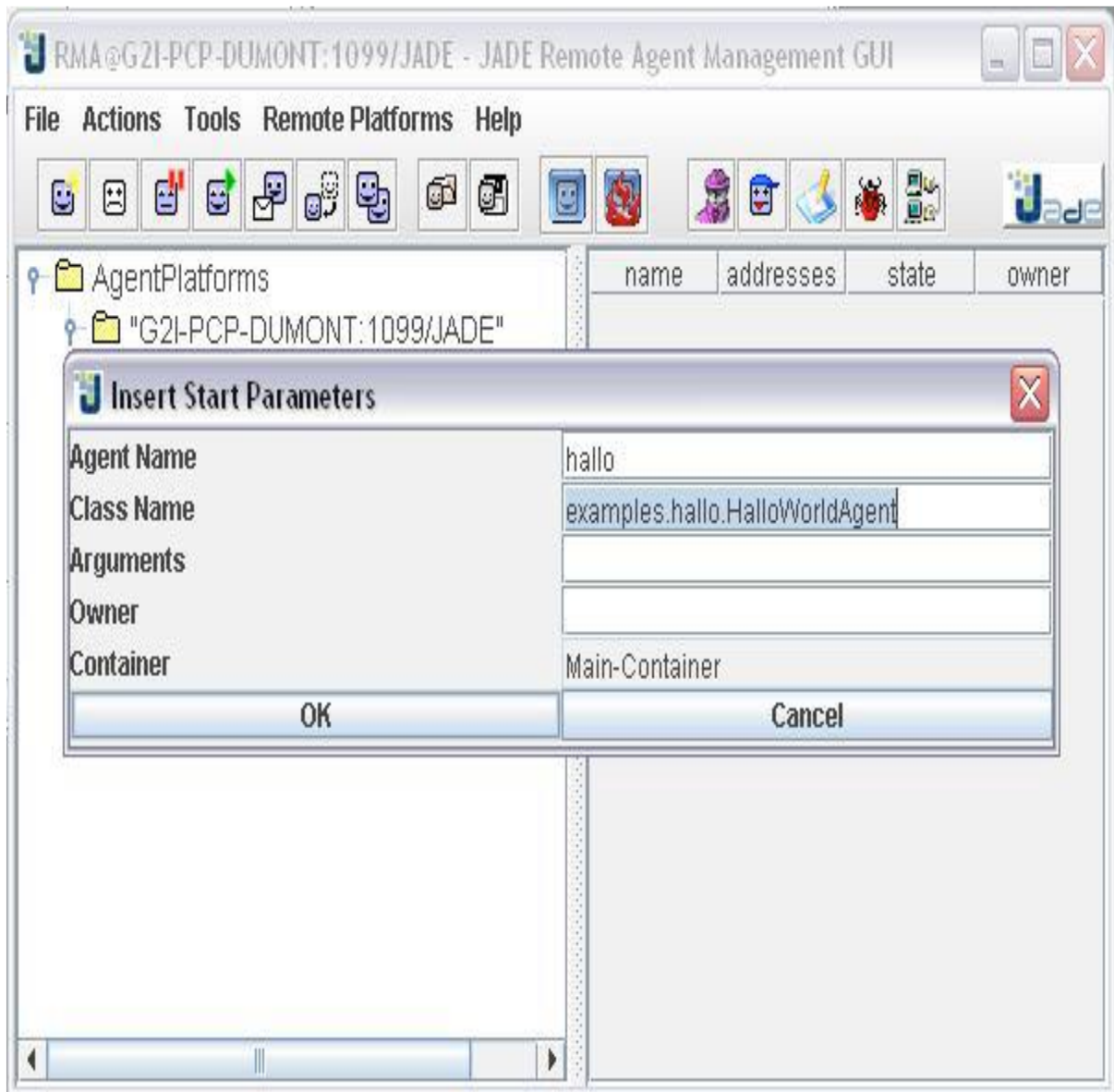


Figure 78: Définir les attributs d'un agent

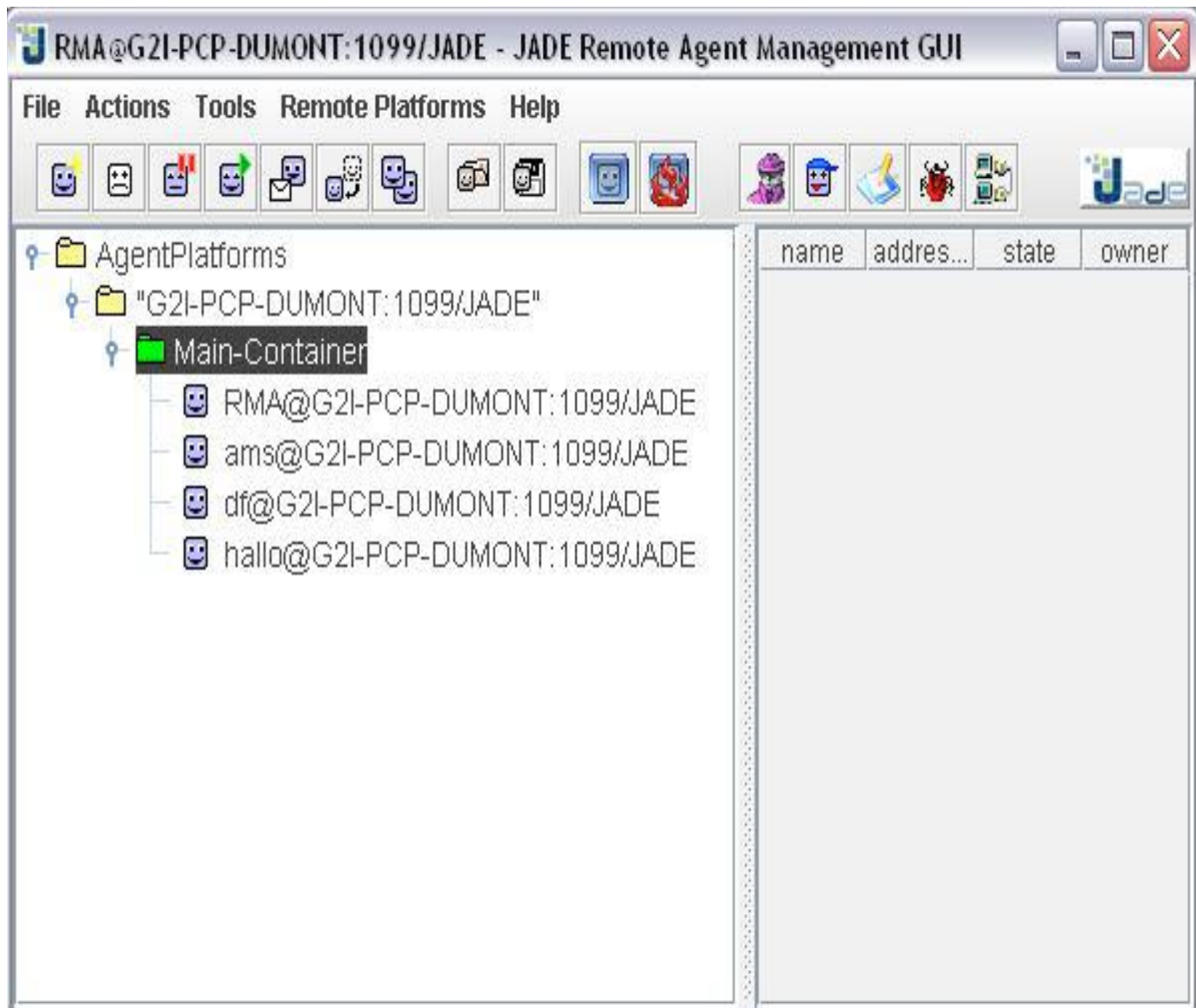


Figure 79: Création de l'agent dans le conteneur

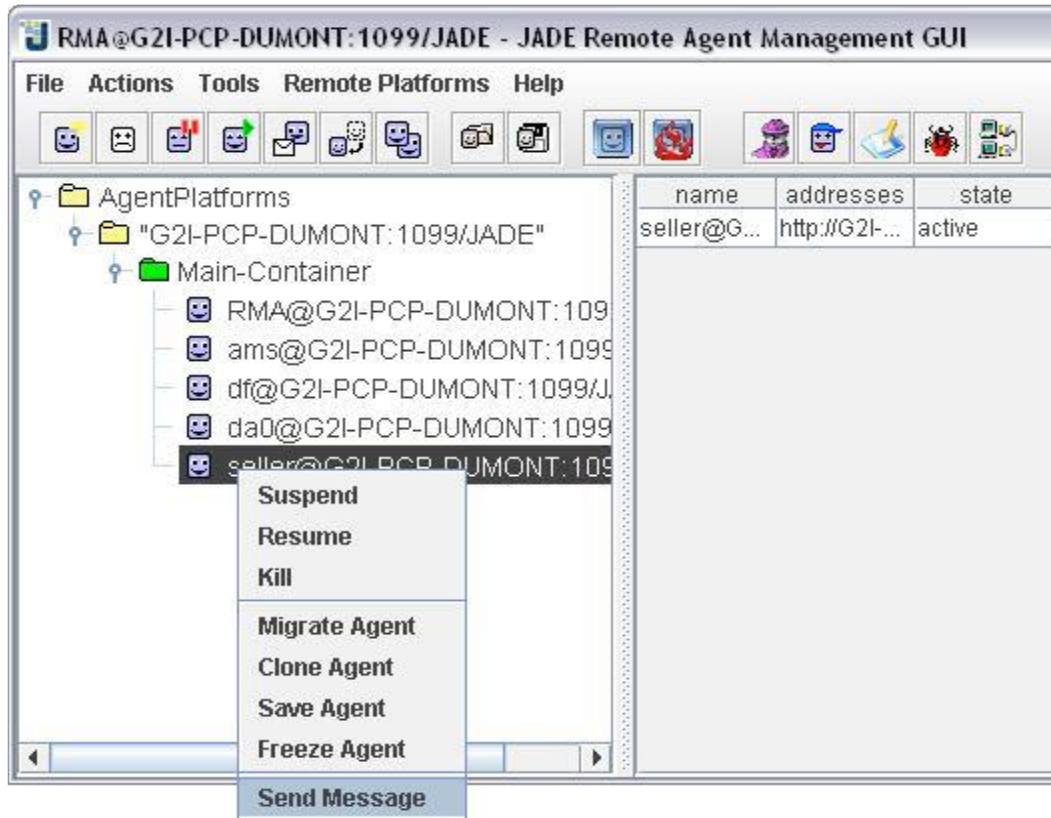


Figure 80: Envoi de message

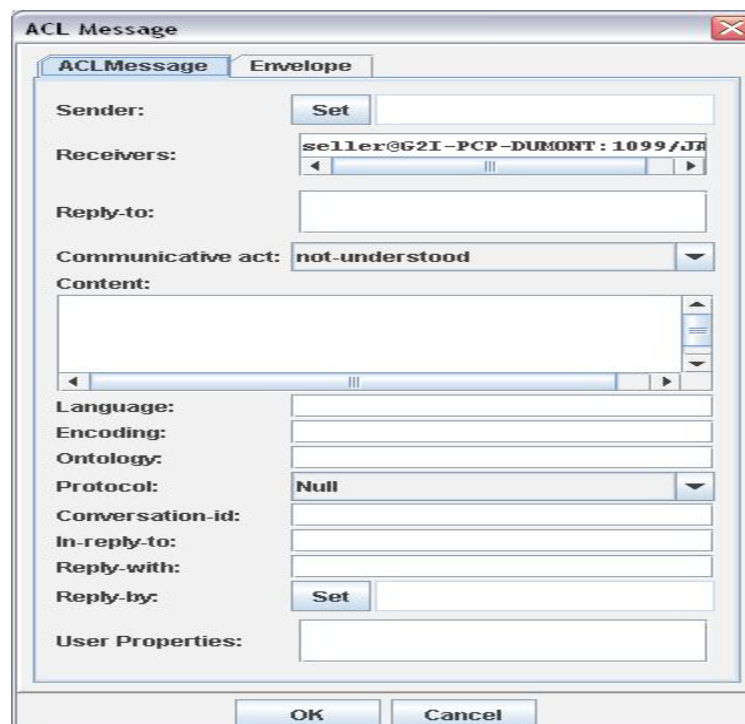


Figure 81: Interface message (ACL-FIPA)

2. Dummy agent :

Lancement :

```
java jade.Boot
monDummy:jade.tools.DummyAgent.DummyAgent
```

Ou par l'intermédiaire du GUI du RMA

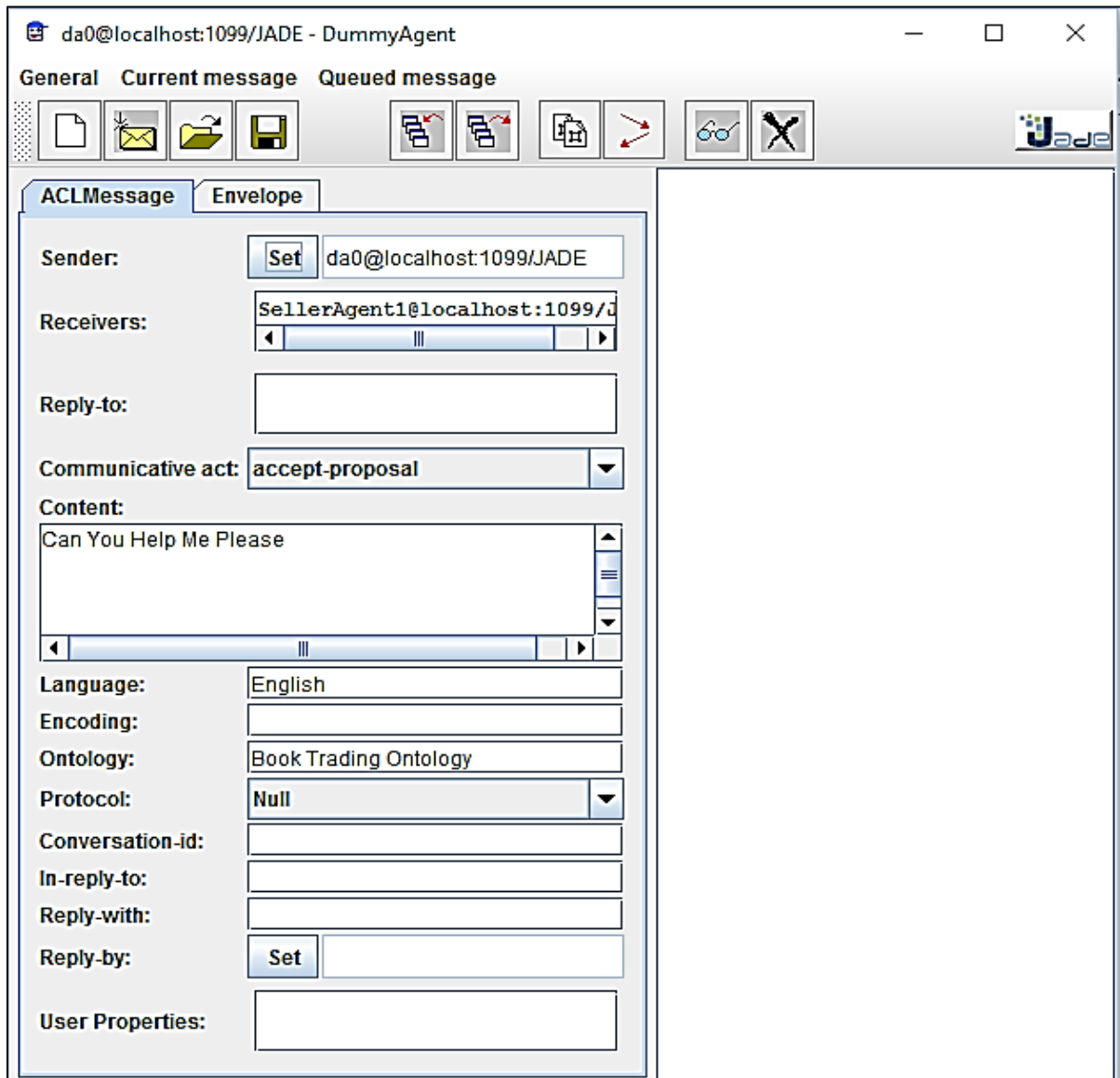


Figure 82: Dummy agent.

3. Inspector Agent :

Lancement :

```
java jade.Boot
monIntrospector:jade.tools.introspector.Introspector
```

Ou par l'intermédiaire du GUI du RMA

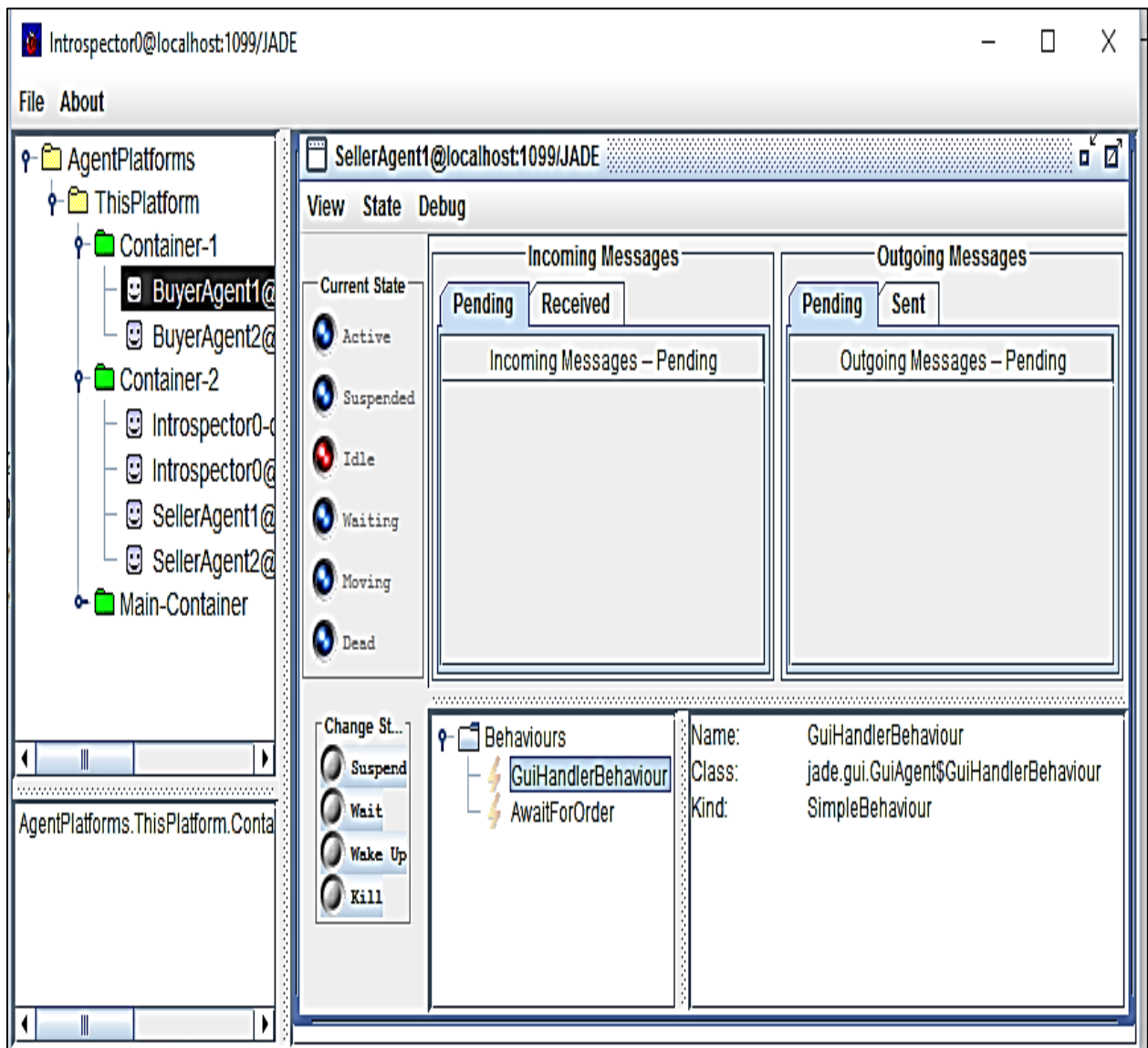


Figure 83 : Inspector agent.

4. Sniffer Agent :

Lancement :

```
java jade.Boot
monSniffer:jade.tools.sniffer.Sniffer
```

Ou par l'intermédiaire du GUI du RMA

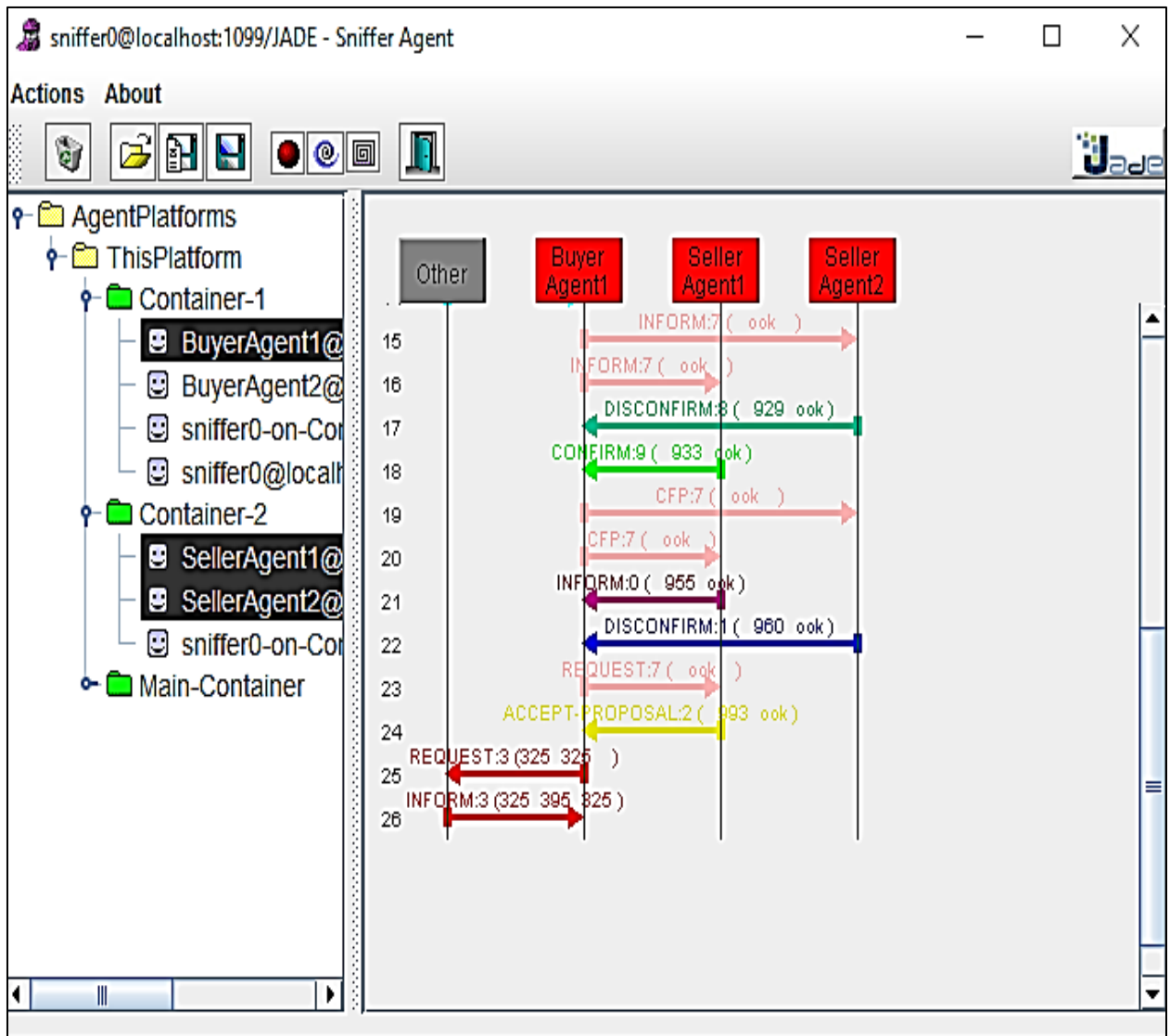


Figure 84: Sniffer Agent

5. Directory Facilitator :

Lancement :

Le DF est automatiquement lancé au démarrage de la plateforme

Lancement de l'interface DF GUI par l'intermédiaire du RMA

Lancement par envoi de message au DF :

(request :content (action DFName (SHOWGUI)) :ontology JADE-Agent-Management :protocol fipa-request)

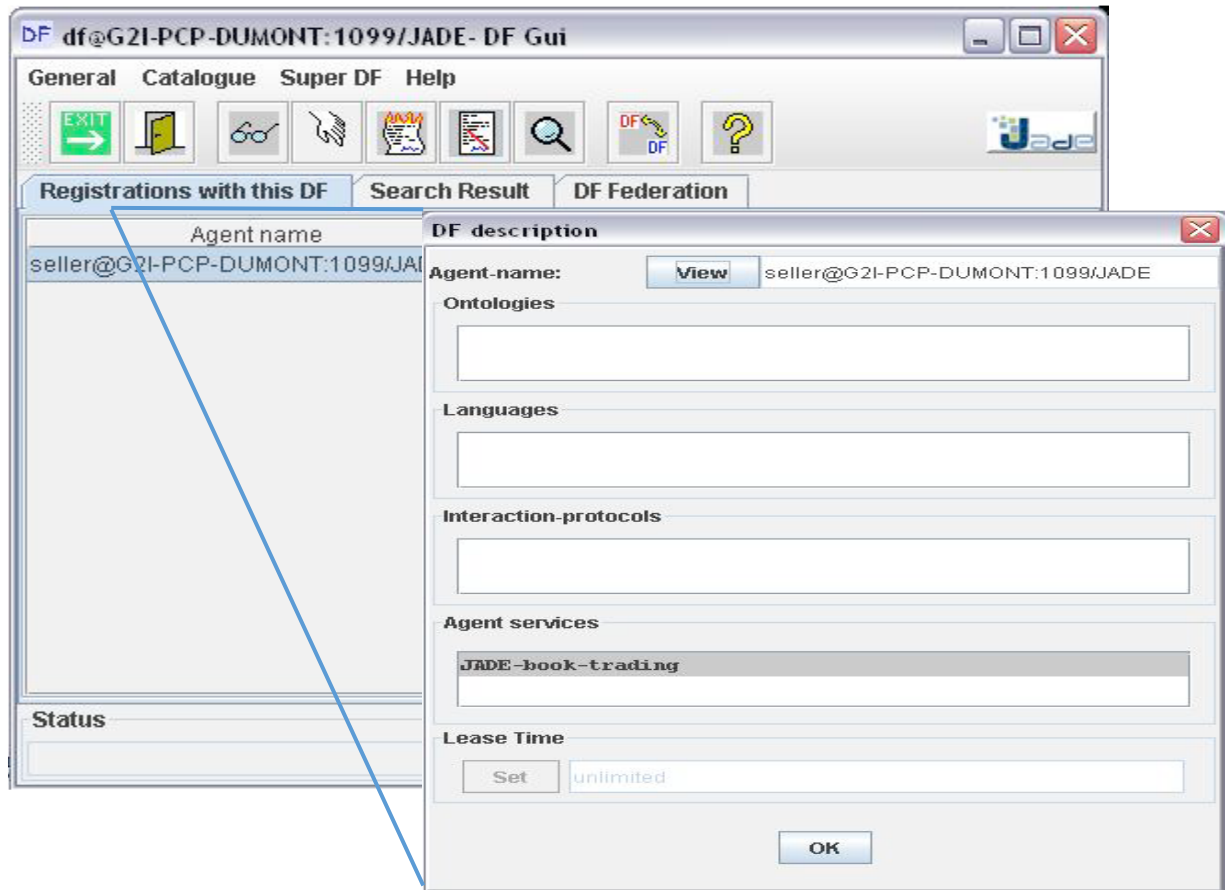


Figure 85: Directory facilitator avec description.

6. Log manager :

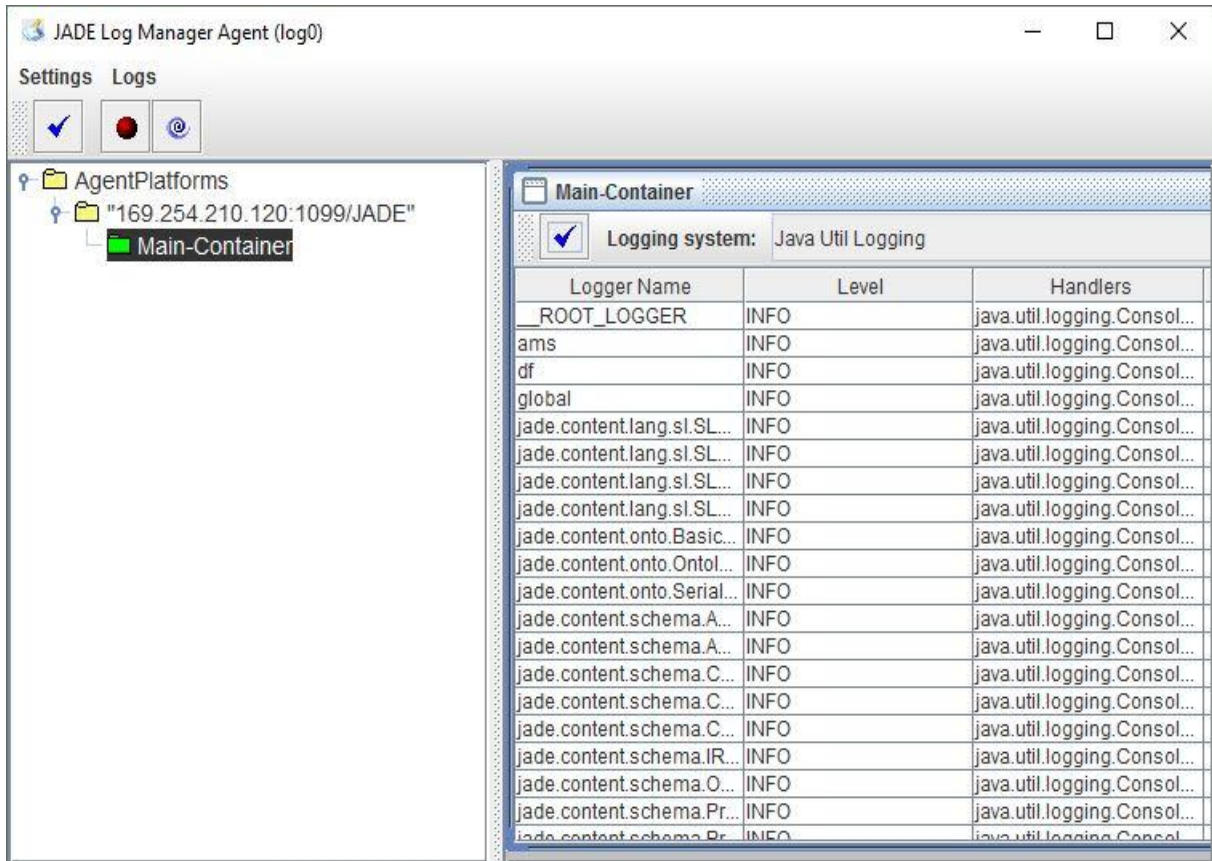


Figure 86: Log manager GUI

Annexe 5: FIPA-ACL Performatifs

Table 25: FIPA-ACL Performatifs

| Primitives | Description |
|------------------------|---|
| <i>acceptproposal</i> | L'action d'accepter une proposition précédemment soumise pour effectuer une action. |
| <i>agree</i> | L'action d'accepter d'effectuer une certaine action, probablement à l'avenir. |
| <i>cancel</i> | L'action de décommander une certaine action précédemment requis qui a l'ampleur temporelle (i.e. : n'est pas instantanée). |
| <i>cfp</i> | L'action de la réclamation des propositions pour effectuer une action donnée. |
| <i>confirm</i> | L'expéditeur informe le récepteur qu'une proposition donnée est vraie, où le récepteur est connu pour être incertain au sujet de la proposition. |
| <i>disconfirm</i> | L'expéditeur informe le récepteur qu'une proposition donnée soit fausse, où le récepteur est connu pour croire, ou le croient probablement, que la proposition est vraie. |
| <i>failure</i> | L'action de dire à un autre agent qu'une action a été essayée mais la tentative a échoué. |
| <i>inform</i> | L'expéditeur informe le récepteur qu'une proposition donnée est vraie. |
| <i>inform-if</i> | Une macro action pour l'agent de l'action pour informer le destinataire si une proposition est vraie. |
| <i>inform-ref</i> | Une macro action pour que l'expéditeur informe le récepteur qu'un objet correspond à un descripteur défini (par exemple un nom). |
| <i>notunderstood</i> | L'expéditeur de l'acte (par exemple i) informe le récepteur (par exemple j) qu'il a perçu que j a effectué une certaine action, mais que je n'ai pas compris quel j a juste fait. |
| <i>propose</i> | L'action de soumettre une proposition pour effectuer une certaine action, donnée certaines conditions préalables. |
| <i>query-if</i> | L'action demandant à un autre agent si une proposition donnée est vraie. |
| <i>query-ref</i> | L'action de demander à un autre agent, avec l'objet qui est rapportée par une expression. |
| <i>refuse</i> | L'action de refuser d'effectuer une action donnée, et d'expliquer la raison du refus. |
| <i>rejectproposal</i> | L'action de rejeter une proposition pour effectuer une certaine action pendant une négociation. |
| <i>request</i> | L'expéditeur invite le récepteur à effectuer une certaine action. Une classe importante des utilisations de l'acte de demande est d'inviter le récepteur à exécuter un autre acte communicatif. |
| <i>request-when</i> | L'expéditeur veut que le récepteur effectue une certaine action quand une certaine proposition donnée devient vraie. |
| <i>requestwhenever</i> | L'expéditeur veut que le récepteur effectue une certaine action dès qu'une certaine proposition deviendra vraie et ensuite chaque fois que la proposition devient vraie. |
| <i>requestwhomever</i> | L'expéditeur veut une action effectuée par un certain agent autre que lui-même. L'agent de réception devrait effectuer l'action ou la transmettre à un autre agent. |
| <i>subscribe</i> | L'acte de demander une intention persistante d'informer l'expéditeur de la valeur d'une référence, et de l'annoncer encore, toutes les fois que l'objet identifié par la référence change. |

Annexe 6: Book trading project

Seller agent features:

L'utilisateur commence par insérer un ensemble de livres dans le catalogue. Le but de l'agent est de les vendre à chaque opportunité possible et notifier l'utilisateur quand le livre est vendu.

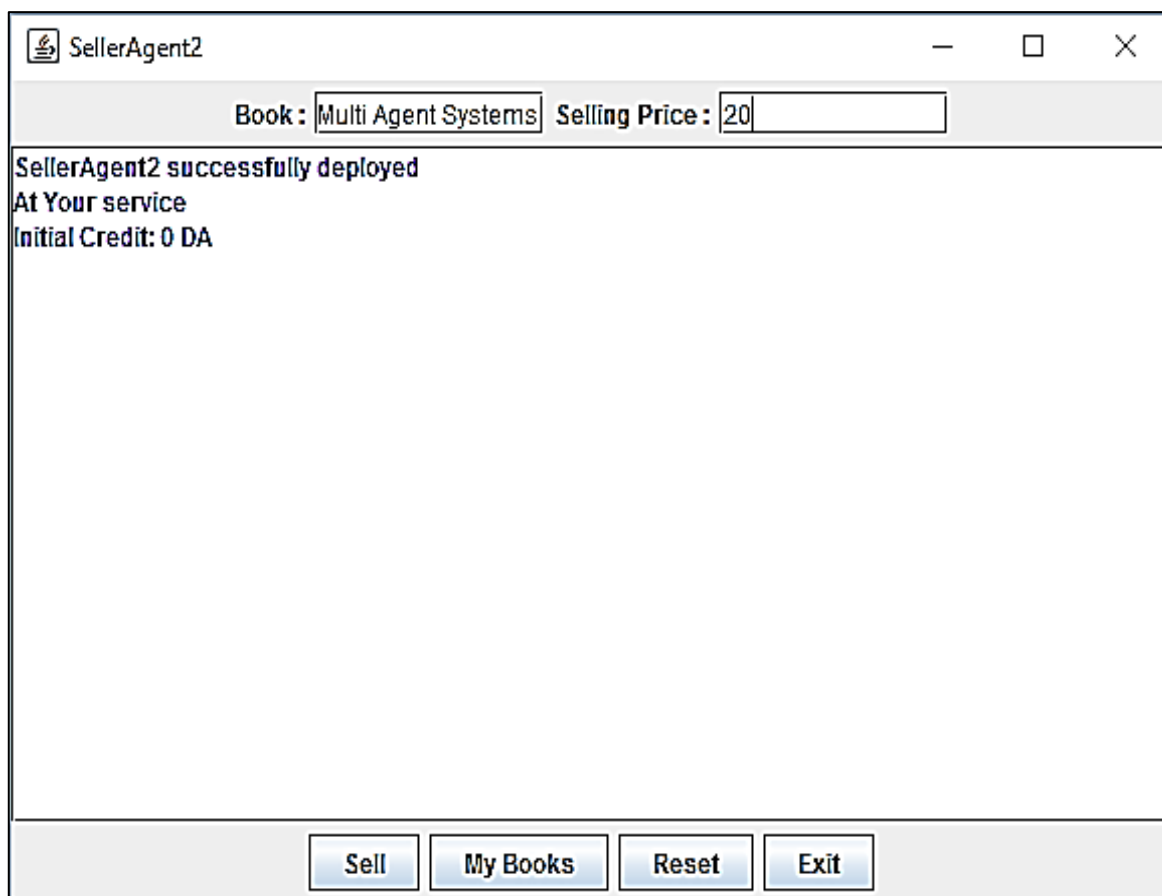


Figure 87: Seller agent gui

Buyer agent features :

L'utilisateur commence par demandé des livres a achetés. Le but de l'agent est de demandé les livres depuis les sellers disponibles et l'acheter a un minimum de prix. Toute en s'assurant que le prix ne dépasse pas le maximum défini par l'utilisateur.

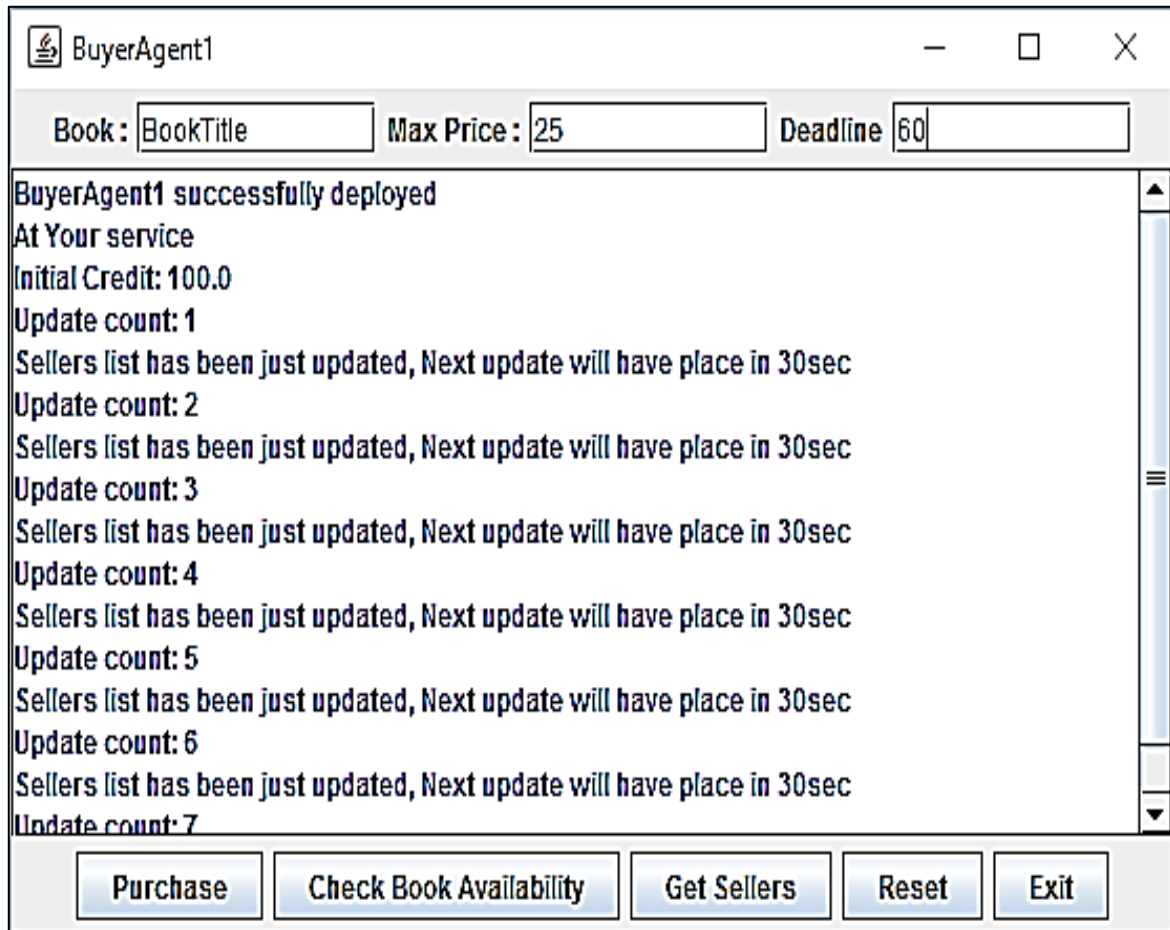


Figure 88: Buyer agent gui

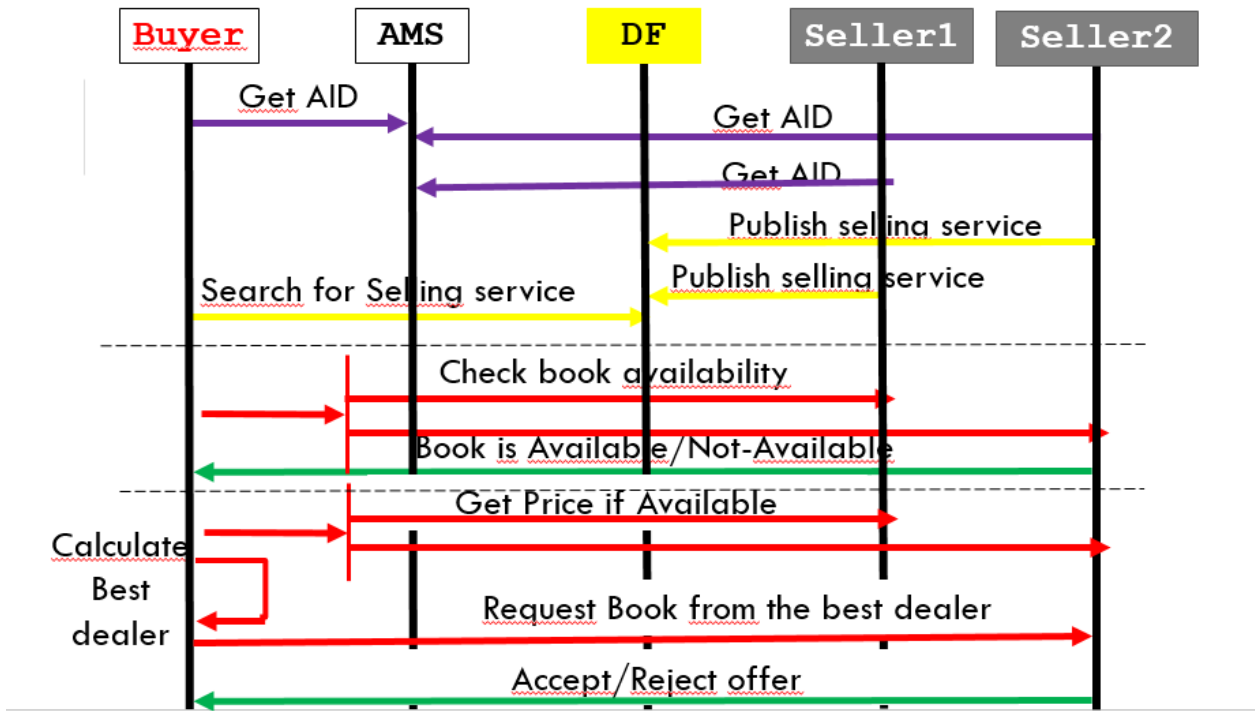


Figure 89: Le diagramme d'interaction

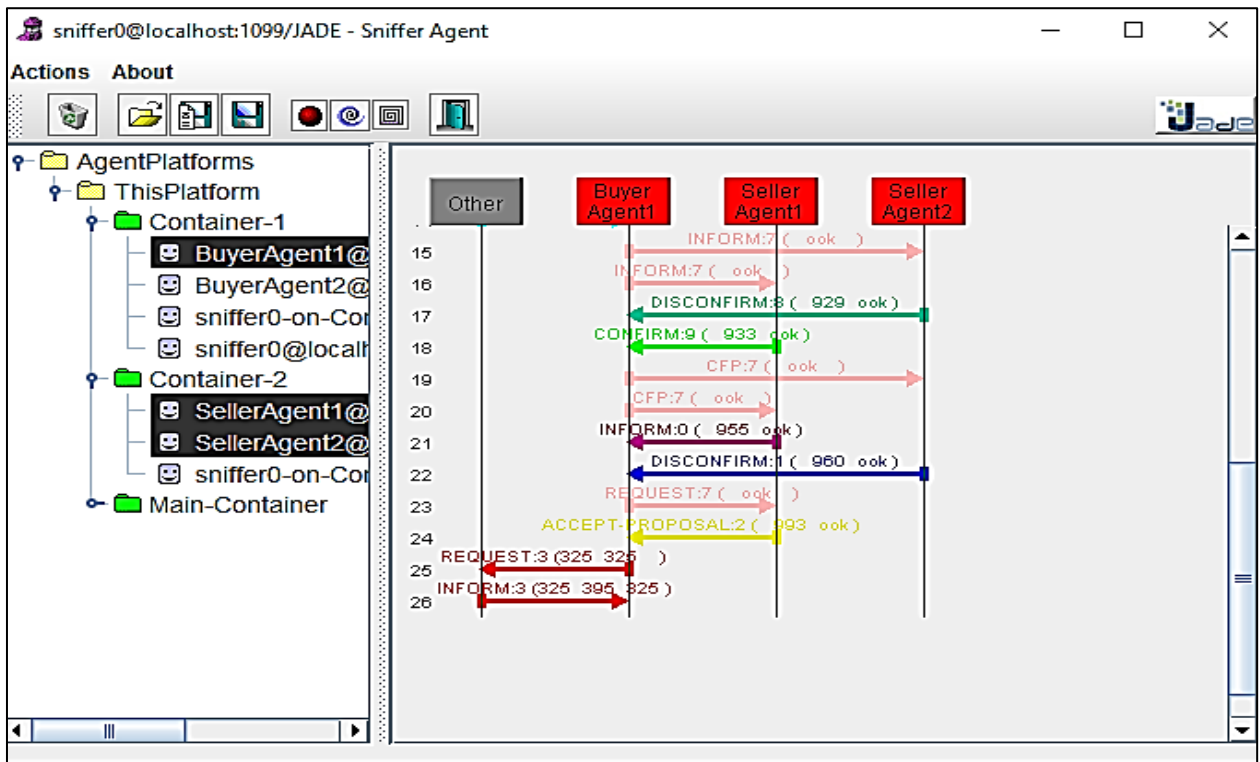


Figure 90: Le processus d'interaction par 'Sniffer Agent'

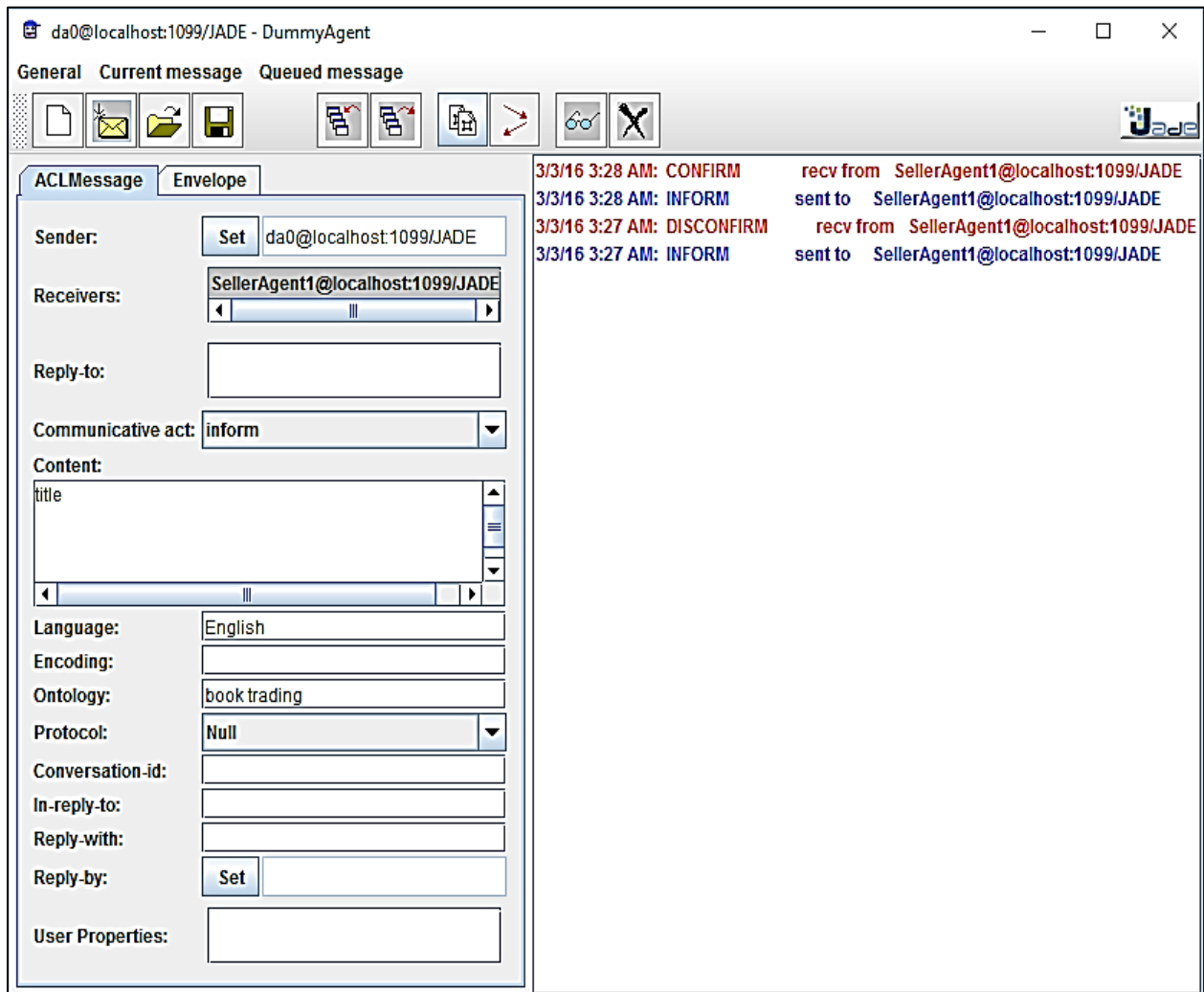


Figure 91: Dummy agent book trading

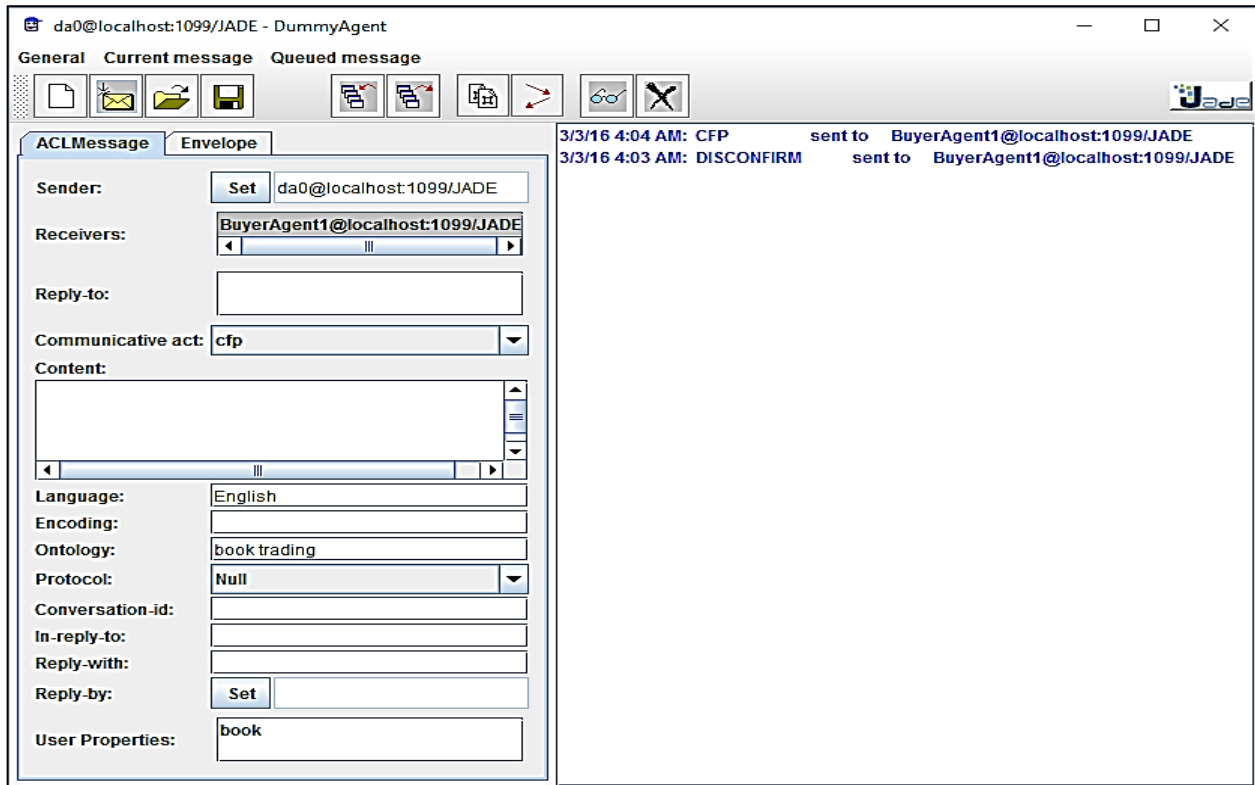


Figure 92 : Dummy agent Book trading two

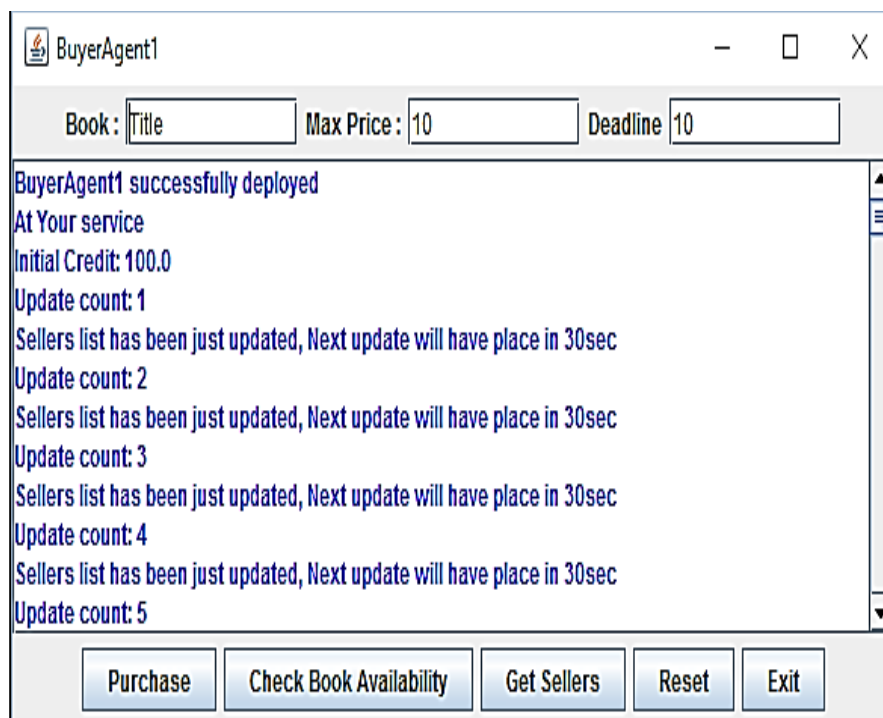


Figure 93: Buyer agent information of deployment

Complete protocol model

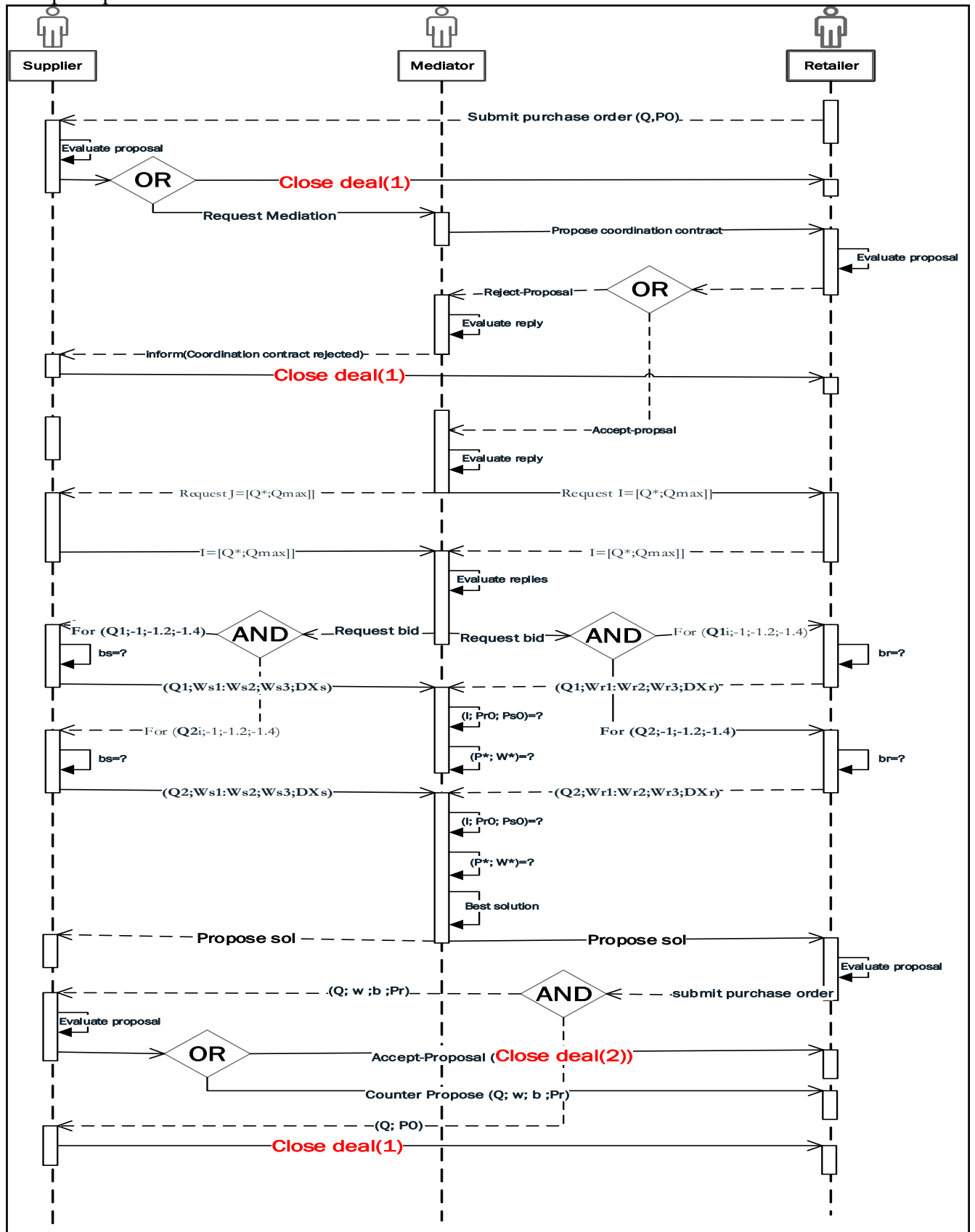


Figure 95: Complete protocol model