



جامعة العربي التبسي - تبسة
Université Larbi Tébessi - Tébessa

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la
recherche scientifique

Université Larbi Tébessi - Tébessa

Faculté des Sciences Exactes et des Sciences de la Nature
et de la Vie

Département : Mathématiques et Informatique



Mémoire de fin d'études
Pour l'obtention du diplôme de *MASTER*
Domaine : Mathématiques et Informatique
Filière : Informatique
Option : Systèmes d'informations.

Thème :

**Elicitation des préférences dans un domaine de
négociation multi-issues non linéaire.**

Présenté Par :
Zerrougui Amani.

Devant le jury :

Mr : Sahraoui Abdelatif MCB Université Larbi Tébessi Président
Mr : Metrouh Abdelmalek MCB Université Larbi Tébessi Examineur
Mr : Betouil Ali Abdelatif MCB Université Larbi Tébessi Encadreur

Date de soutenance : 28/06/2020.

Remerciements

Tout d'abord je tiens à remercier ALLAH qui m'a donné la force, l'intelligence et la patience d'accomplir ce modeste travail.

*Mes remerciements et ma gratitude s'adressent à mon encadreur **Dr Betouil Ali Abdelatif** pour sa disponibilité et l'intérêt qu'il a porté à mon travail tout au long de semestre.*

*Je souhaite remercier **Dr Sahraoui Abdelatif** pour avoir accepté de présider le jury, qu'elle me permet de lui exprimer ma plus haute considération.*

*Je remercie également **Dr Metrouh Abdelmalek** pour leur investissement en tant qu'examineur et pour le temps qu'il a consacré à lire et évaluer ce travail.*

Un grand merci ainsi que tous les enseignants de département Mathématique et informatique.

Enfin, je remercie mes chers parents, ma famille et mes amis qui m'ont encouragée tout au long de mes études.

Leur soutien et leur aide sont pour beaucoup dans l'accomplissement de ce travail.

Dédicace

Du profond dans mon cœur, je dédie ce travail à tous ce qui est chères.

A mes chers parents

Mama... Papa, aucun dédicace ne serait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien être. Je vous remercie pour tout soutient, la motivation et l'amour que vous portez depuis mon enfance et j'espère que votre bénédiction m'accompagne toujours.

Que ce modeste travail soit l'exaucement de vos vœux tout formulés, le fruit de vos innombrables sacrifices. Puisse Dieu le très Haut, vous accorder santé, bonheur et longue vie.

A ma belle-sœur Meriem, et mes frères Mohamed Anis et Ilyas

C'est un moment de plaisir de dédier ce travail, en signe d'amour, de reconnaissance et de gratitude pour le dévouement dont vous avez fait toujours preuve à mon égard.

A mes meilleurs amis

Chayma, merci énormément pour ton soutient plus que précieux, merci pour ton grand cœur tous vos qualités qui serait trop longues à énumérer, ma vie ne serait pas aussi magnifique sans ton présence et ton amour.

Yasmine... Khaoula, qui n'ont jamais cessée de me soutenir, merci pour être dans ma vie je vous aime de fond du mon cœur.

A mon oncle Ali, et toute les personnes que j'aime.

Résumé :

La négociation est un processus par lequel toutes les parties peuvent parvenir à une solution. Dans le cadre de ce travail, nous abordons les mécanismes de négociation bilatérale multi issues dans un domaine non linéaire à fin d'obtenir une bonne solution de compromis.

Le but est de proposer une approche qui permet de modéliser les préférences d'adversaire en basant sur les négociations passées.

Dans ce travail, nous développons un agent basé sur l'apprentissage par renforcement (AR) et implémenté via l'algorithme Q-learning pour prédire les intentions d'autres agents et les actions futures possibles. Afin de maximiser le rôle de notre agent et en fonction de la décision de sélectionner la meilleure offre parmi plusieurs offres multi-issue non linéaires (y compris les stratégies acceptation / refus, concessions et proposition).

Nous montrons que notre agent est capable d'apprendre dans un environnement non linéaire multi-issue et obtient une valeur d'utilité meilleure que les autres agents disponibles dans la plateforme de simulation des négociations automatiques appelée GENIUS.

Les mots clés : négociation automatique, agent, apprentissage par renforcement, GENIUS, multi issues non-linéaire.

Abstract:

Negotiation is a process by which all parties can reach a solution. In this work, we approach the mechanisms of bilateral multi-issue negotiations in a non-linear field in order to obtain a good compromise solution.

The goal is to propose an approach that allows modeling of adversary preferences based on past negotiations.

In this work, we develop an agent based on reinforcement learning (RL) and implemented via the Q-learning algorithm to predict the intentions of other agents and possible future actions. In order to maximize the utility of our agent and depending on the decision to select the best offer among several non-linear multi-issue offers (including acceptance / refusal, concessions and proposal strategies).

We show that our agent is able to learn in a non-linear multi-issue environment and obtains a value in use better than the other agents available in the automatic trading simulation platform called GENIUS.

The key words: automatic negotiation, agent, reinforcement learning, GENIUS, non-linear multi-issue.

ملخص:

التفاوض عملية يمكن من خلالها لجميع الأطراف الوصول إلى حل. في إطار هذا العمل، نقترب من آليات المفاوضات الثنائية متعددة القضايا في مجال غير خطي من أجل الحصول على حل وسط جيد. الهدف هو اقتراح نهج يسمح بنمذجة تفضيلات الخصم على أساس المفاوضات السابقة. في هذا العمل، نقوم بتطوير وكيل يعتمد على التعلم المعزز (AR) ويتم تنفيذه عبر خوارزمية Q-learning للتنبؤ بنوايا وكلاء آخرين والإجراءات المستقبلية المحتملة. من أجل تعظيم دور وكيلنا واعتمادًا على قرار اختيار أفضل عرض من بين العديد من العروض غير الخطية متعددة القضايا (بما في ذلك القبول / الرفض، والامتيازات واستراتيجيات الاقتراح). نوضح أن وكيلنا قادر على التعلم في بيئة غير خطية متعددة القضايا ويحصل على قيمة في الاستخدام أفضل من العوامل الأخرى المتاحة في منصة المحاكاة للمفاوضات التلقائية تسمى GENIUS .

الكلمات المفتاحية: التفاوض الأوتوماتيكي، الوكيل، التعلم المعزز، GENIUS ، القضايا المتعددة غير الخطية.

Sommaire :

<i>Remerciements</i>	i
<i>Dédicace</i>	ii
<i>Résumé</i> :	iii
<i>Abstract</i> :.....	iv
<i>ملخص</i> :.....	v
<i>Glossaire</i> :	ix
Introduction générale :.....	1
Chapitre 1 : Les systèmes multi agents	1
Introduction :	2
1 Les systèmes multi agents (SMA) :.....	2
1.1. Le contrôle dans les systèmes multi agents :.....	3
2. Les agents :	4
2.1. Définition de l'agent :	5
2.2. Caractéristique des agents :	5
2.2.1. Caractéristiques primaires :	6
2.2.2. Caractéristiques secondaires :	6
2.2.3. Caractéristiques tertiaires :	6
2.3. L'intelligence d'agent :	7
2.4. La rationalité des agents :	8
2.5. La coopération des agents :	9
2.6. La capacité sociale des agents :	9
2.7. L'égoïsme d'un agent :.....	9
3. L'environnement :	9
4. L'organisation :	10
4.1. Traitement des organisations :.....	11
5. L'interaction :.....	11
5.2. Les types d'interactions :	12
5.1.1. Coopération entre agents :	12

5.1.2. Coordination entre agents.....	12
5.1.3. Négociation entre agents :.....	12
5.2. Communication entre agents :.....	13
5.3. Les éléments de communication :.....	13
Conclusion :.....	14
Chapitre 2 : Négociation multi agents.....	16
Introduction :.....	17
I. La négociation :.....	17
2. Les formes de négociation :.....	19
2.1. Les systèmes de vote :.....	19
2.2. Les enchères :.....	19
2.3. La négociation à base d'argumentation :.....	21
3. Les protocoles de négociation :.....	23
3.1. Définition.....	23
3.2. Classification des protocoles de négociation :.....	23
3.2.1. Les protocoles compétitifs vs coopératifs :.....	23
3.2.1. Les protocoles unidirectionnels vs bidirectionnels :.....	24
3.2. Quelques protocoles de négociation :.....	25
4. Evaluation des protocoles de négociation :.....	27
5. Stratégie de négociation :.....	28
II. Négociation multi-issues :.....	31
1. Négociation multi issues.....	31
2. Modèle de négociation multi issues :.....	31
Conclusion.....	33
Chapitre 3 : Les techniques de modélisation des adversaires dans une négociation multi agents.....	35
Introduction.....	36
1. Etat de l'art :.....	36
2. La modélisation de l'adversaire :.....	40
2.1. Stratégie d'acceptation :.....	41
2.3. Profil de préférence :.....	42
2.4. Stratégie d'enchère :.....	42
3. Les techniques de modélisation pour les modèles d'adversaire :.....	43
3.1. L'apprentissage bayésien :.....	43
3.2. Régression non linéaire :.....	44
3.3. Estimation de la densité du noyau.....	46

3.4. Réseaux de neurones artificiels :	47
Conclusion :	49
<i>Contribution</i>	0
Chapitre 4 : Négociation par renforcement basée agent	51
Introduction :	52
1. Description de notre agent :	52
2. L'apprentissage profond :	53
3. Apprentissage par renforcement :	53
4. Le champ des termes du RL dans notre modèle :	54
5. Explication du Q-learning dans notre modèle :	55
Le Q-table :	56
Dilemme exploration – exploitation :	58
6. La conception de notre agent négociateur :	58
6.1. Stratégie d'apprentissage :	59
6.2. La stratégie de proposition :	60
6.3. La stratégie d'acceptation :	61
6.4. La stratégie de concession :	61
Conclusion :	64
Chapitre 5 : Implémentation et résultats.....	65
Introduction	66
1. La plateforme GENIUS :	66
1.1. Présentation des interfaces du GENIUS :	67
2. La compétition des agents de négociation automatisés ANAC :	69
3. Eclipse ET java (JDK) :	70
4. Discussion des résultants :	71
4.1. Exemple d'application dans une session de négociation :	71
4.2. Résultats obtenus :	73
Conclusion :	76
Conclusion générale et perspectives :	65

Glossaire :

α	Coefficient d'apprentissage.
ε	Taux d'exploration.
γ	Coefficient d'actualisation.
P	Probabilité de transition.
R	Fonction de récompense.
r	Récompense scalaire.
t.....	Temps Période d'échantillonnage.
T.....	Deadline.
a.....	Action.
A.....	Ensemble d'action.
s	Etat courant.
S.....	Ensemble des états.
Q	Fonction d'utilité.
$U_A(w_{B \rightarrow A}^t)$	L'utilité de l'agent « A » qui lui gagner d'après la proposition de « B ».
$U_A(w_{A \rightarrow B}^t)$	L'utilité de l'agent « A » qui lui gagner d'après leur proposition chez l'agent « B ».
β	Une valeur réservé de l'utilité minimale.

Liste de figures :

Figure 2. 1. Interaction d'un agent avec son environnement.....	5
Figure 3. 1. L'architecture en couche d'une nvironnement	10
Figure 3. 2. Exemple de régression non linéaire basée sur un polynôme du deuxième degré.	45
Figure 3. 3: Exemple de la densité de noyau.	47
Figure 3. 4. Exemple d'un ANN symbolisant le XOR logique.	48
Figure 4. 1.. Exemple d'un agent utilise l'apprentissage par renforcement.....	54
Figure 4. 2.Explication de la fonction de Q-learning.	56
Figure 4. 3. Le Q-table.	56
Figure 4. 4.. Les composants de la stratégie de négociation.	59
Figure 4. 5. La stratégie de proposition.	62
Figure 4. 6. La stratégie d'acceptation.	63
Figure 4. 7. La stratégie de concession.....	63
Figure 5.1 composants BOA	67
Figure 5. 2.Les agents intégrés avec GENIUS.....	67
Figure 5. 3.composants BOA ; offrant à l'utilisateur la possibilité de créer et d'appliquer des composants nouvellement développés utiliser une interface utilisateur graphique	68
Figure 5. 4.Création d'une négociation session simple.....	68
Figure 5. 5.Choisir les paramètres d'une négociation..	69
Figure 5. 6.Le résultat de la négociation.	69
Figure 5. 7. La session de négociation est selon le protocole alternative, notre agent est de la coté « A » nommé« AMANI » avec l'agentKF.	71
Figure 5. 8.Les résultats selon la négociation session précédente.....	72
Figure 5. 9.Une section tournoi avec tous les agents compatibles avec les utilités non linéaires sous GENIUS.....	73
Figure 5. 10.Les résultats du tournoi	74
Figure 5. 11.Tableau de résultats.	74
Figure 5. 12.Suite tableau de résultats.....	75

Introduction générale :

Notre travail se concentre sur l'intelligence artificielle, en particulier l'intelligence artificielle distribuée et les systèmes multi-agents. Lorsque plusieurs agents interagissent, des conflits peuvent survenir, ce qui nécessite l'utilisation de mécanismes de résolution des conflits. Ces mécanismes comprennent la coordination, le système de vote et la négociation.

La négociation est un processus par lequel toutes les parties peuvent parvenir à une solution. Toutes les parties expriment d'abord leurs exigences et intègrent une série de concessions pour parvenir à un accord ou rechercher de nouvelles options. Négociation pour le commerce électronique, en particulier le partage de la bande passante dans les télécommunications par le biais d'enchères, etc., et l'allocation des tâches et des ressources dans les systèmes multi-agents (SMA), identification des conflits, Différences entre les objectifs et la détermination de la structure organisation : tout cela affectera la cohérence de la société de l'agent.

Les négociations sont divisées en deux catégories : la négociation compétitive et la négociation coopérative. Négociations compétitives entre agents concurrence, en essayant de maximiser son utilité locale. D'un autre côté, lors de la négociation de la coopération, les agents tenteront d'atteindre l'efficacité globale maximale, qui devrait être considérée calculez la valeur de toutes leurs activités. Cette forme de négociation est très différente de la négociation compétitive et peut être considérée comme un processus de recherche Décentralisé.

La négociation qui nous allons traiter dans cette mémoire est une négociation compétitive, car nous sommes intéressés à maximiser l'utilité de notre agent négociateur.

Selon Jennings et al. La recherche sur la négociation automatique peut être décomposée en trois larges thèmes :

- **Protocole de négociation** : un ensemble de règles pour contrôler l'interaction.
- **L'objectif de la négociation** : le périmètre standard de la solution doit être atteint. Dans un cas extrême, l'objectif ne peut contenir qu'une seule issue (par ex. Prix), mais il peut également contenir des multi issues (avec le prix, Qualité, délai, pénalités, conditions générales, etc.).
- **Les modèles décisionnels des agents** : méthode permettant à un agent d'atteindre ses objectifs de négociation tout en suivant les règles de négociation.

Complexité L'accord utilisé, la nature des objectifs de négociation et la gamme d'opérations qui peuvent être appliquées.

Dans ce mémoire nous intéressons au domaine des modèles décisionnels des agents, notre but est de fournir une stratégie d'apprentissage qui permet à l'agent d'anticiper l'intention et les futures actions possibles de l'autre agent adversaire au cours d'une négociation multi issues et non linéaire, où l'agent cherche de maximiser son utilité local et gagner le tour de négociation. Les modèles utilisés par les agents sont non linéaires, car les agents doivent établir leur propre représentation à travers de multiples interactions avec l'environnement.

Ainsi, de créer des stratégies d'acceptation ; c'est la condition choisi par l'agent selon leur veut de maximiser leur utilité, à côté d'une stratégie de concession qui présente un cas particulier le temps qu'il renonce à certain condamnations à son profit, et une stratégie de proposition faite où l'agent refus un offre d'adversaire et il lui proposée une contre-offre pour augmenter leur utilité.

Notre défi dans ce travail est d'effectuer d'une elicitions des préférences dans un domaine de négociation multi issues no linéaire selon une modélisation et un algorithme basé à l'apprentissage par renforcement.

Dans ce travail nous développerons un agent capable de gagner une négociation avec la maximum valeur d'utilité à l'utilisation des fonctions de l'apprentissage par renforcement et l'algorithme Q-learning.

Pour ce faire ; nous organisons notre mémoire en cinq chapitres :

- Dans le premier chapitre : nous présentons la notion de base des systèmes multi agent et ça composant de base, ainsi qu'une vue presque détaillée sur la notion d'agent et leur caractéristiques.
- Dans le deuxième chapitre : ce chapitre est composé de deux parties, le premier partie où nous présentés la négociation, leur composants de base, les modèles, les protocoles de négociation et une validation des protocoles, et les stratégies de négociation. Dans Le deuxième nous fournissons une définition sur la négociation multi issues et nous prend un modèle d'une négociation multi issues.
- Dans le troisième chapitre : nous appuyons sur l'apprendre des adversaires et les techniques de modélisation de ces derniers de plus nous fournissons un

revue d'état de l'art contient les plus connues méthodes utilisé pour modéliser les agents de négociation.

- Dans le quatrième chapitre : nous présentons une bref description de notre futur agent a modélisé ; nous appuyons sur l'apprentissage par renforcement ; ensuite nous confirmons le choix d'algorithme Q-learning et commencer à la conception de notre agent.
- Dans le cinquième chapitre : nous montrons la partie expérimentale de notre travail et discuter les résultats.

Chapitre 1 : Les systèmes multi agents

Résumé :

Un système multi agents est un ensemble des agents qui interagissent entre eux. Alors là dans ce chapitre on va présenter une vue générale sur les SMA, ainsi que les composants de base d'un SMA tel que : les agents, l'environnement, l'organisation et l'interaction entre ces derniers.

Introduction :

Les systèmes multi-agents (SMA) sont une branche de l'intelligence artificielle distribuée (IAD) ; alors on peut dire que les SMA sont un domaine multidisciplinaire, on fait, les concepts de ce domaine sont d'origine diverses.

SMA est un nouveau paradigme de programmation qui a occupé de plus en plus une place importante parmi les technologies de développement systèmes complexes et distribués. Cette importance peut se refléter sur l'évolution du marché mondial des agents logiciels. Dans ce chapitre on présentant les concepts généraux des systèmes multi agents.

1 Les systèmes multi agents (SMA) :

Par définition, le concept de systèmes multi-agents peut être promu. Concept "système". De toute évidence, le système multi-agents est un système composé d'un ensemble des agents. En effet, **un système multi agents est un ensemble d'agent en interaction** [1]. Cependant une analyse approfondie nous trouvons que cette définition est limitée.

Dans le livre de « **Les systèmes Multi-Agents : Vers une intelligence collective** », Ferber définit le système multi agents comme un ensemble composé de :

- Un environnement **E** : c'est-à-dire un espace disposant **généralement** d'une métrique.
- Un ensemble d'objets **O** situés : c'est-à-dire pour tout objet, il est possible, à un moment donné, d'associer une position dans **E**. Ces objets sont passifs, c'est-à-dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents.
- Un ensemble d'agents **A**, qui sont des objets particuliers, lesquels représentent les entités actives du système.
- Un ensemble de relations **R** qui unissent des objets (et donc des agents) entre eux.
- Un ensemble d'opérations **Op** permettant aux agents de **A** de percevoir, produire, consommer, transformer et manipuler des objets de **O**.
- Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification. » [2].

Maintenant cette définition est appliquée au monde réel qui l'entoure, nous pouvons identifier plusieurs exemples de systèmes multi-agents. Par exemple l'équipe de football on préfère un exemple typique d'un système multi-agents. Mais aussi les composants proposés par cette définition n'ont pas la même importance ; il est possible de négliger quelque composant dans certains types spécifiques d'un système multi agent.

Une autre définition : « un système multi agent (SMA) comporte plusieurs agents interagissent entre eux dans un environnement commun .certains de ces agents peuvent être des personnes ou leurs représentant (avatar), ou même que des machines mécaniques. S'il y a moins de trois agents, on parle plutôt d'interaction homme /machine ou machine/machine que des systèmes multi agents » [Anne Nicole] [3].

1.1. Le contrôle dans les systèmes multi agents :

Bien que les systèmes multi-agents soient des modèles de systèmes distribués, l'exécution de l'agent doit être cohérente. Donc, une question de base Cette zone comprend la sélection du type de contrôle pour différents agents, composez le système. Il existe deux méthodes dans la littérature professionnelle Différents agents de contrôle :

- **Le modèle de tableau noir** (approche centralisée) : Il s'agit d'un espace de travail commun dédié au transfert d'informations entre différents agents. Chacun peut le consulter à sa guise, où il peut collecter et stocker des objets modifiables. Ce tableau construit une modélisation du domaine d'application sous une forme spatiale d'hypothèses et de solutions.

Alors le SMA à contrôle centralisé possède en outre les propriétés suivantes :

- Pas de communication directe entre les agents.
- Interaction via le partage d'un même espace de travail qui est le blackboard.
- Mal adapté aux SMA large échelle.

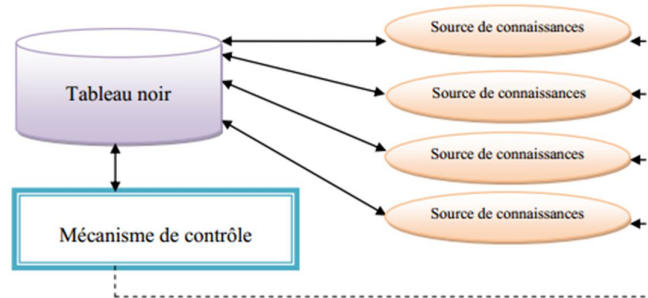


Figure 1.1. Architecture d'un système à base de tableau noir [1].

- **Le modèle d'acteur** (contrôle distribuée) : L'acteur est une entité autonome, Communiquez via des messages asynchrones. Alors non Contrôle centralisé, mais chaque acteur peut se contrôler. En conséquence, l'acteur a eu du mal à atteindre Objectifs mondiaux avec connaissances locales.

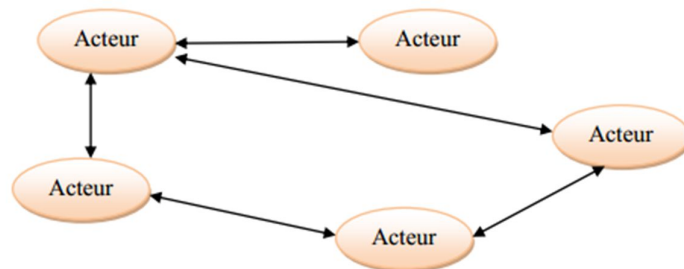


Figure1.2. Architecture d'un système à base d'acteurs [1].

Les systèmes multi agents introduisent plusieurs nouveaux concepts, La méthode VOYELLES en fait partie Une façon de présenter les concepts SMA dans un seul cadre. Alors, Selon cette méthode, un système multi-agents se compose d'agents (A), interaction (I), environnement (E) et organisation (O). Dans la suite de ce chapitre, nous Présentons ces éléments de base de SMA.

2. Les agents :

Les agents représentent la brique de base utilisée pour la réalisation des SMA, alors on peut considérer l'agent comme le concept le plus controversé dans ce domaine.

2.1. Définition de l'agent :

Un agent est une entité (physique ou abstraite) caractérisée par le fait qu'elle est autonome dans la prise de décision, par ses connaissances sur elle-même et sur les autres, et par sa capacité d'agir.

Ce peut-être un processus (en gestion des processus dans les systèmes d'exploitation), un robot (dans un environnement industriel), un être humain (en sociologie), etc.

Pour [Weiss1999], un agent est une "**entité computationnelle**", comme un programme informatique ou un robot, qui peut être vue comme percevant et agissant de façon autonome sur son environnement. [4]

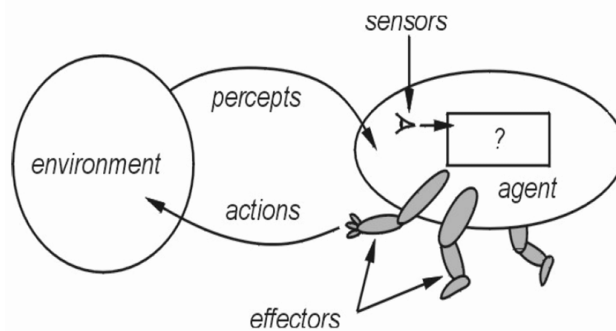


Figure2.1. Interaction d'un agent avec son environnement [4].

2.2. Caractéristique des agents :

En utilisant quelques définitions de la littérature, il est possible de créer un ensemble de caractéristiques d'agent primaires, secondaires et tertiaires [5].

La fonctionnalité principale est inhérente à la plupart des définitions d'agents, et la fonctionnalité secondaire est un ensemble de fonctionnalités liées aux agents, qui sont généralement étendues. Le troisième ensemble de caractéristiques contient des caractéristiques idéales plus abstraites similaires aux humains.

2.2.1. Caractéristiques primaires :

- ❖ **Autonomie** : c'est la caractéristique la plus importante des agents, ou ils peuvent effectuer certaines tâches spontanément et peuvent prendre l'initiative.
- ❖ **Communicative/ sociale** : Un agent devrait avoir un niveau élevé de communication avec d'autres agents. Le protocole de communication entre agent le plus commun KQML (Knowledge Query and Manipulation Language) [5].
- ❖ **Réactivité** : Un agent doit être capable de détecter son environnement (qui peut être le monde physique, un utilisateur via une interface graphique ou une collection d'autres agents) et réagir à ses changements, qu'il s'agisse d'une modification des objectifs de l'utilisateur ou d'une modification des ressources disponibles.

2.2.2. Caractéristiques secondaires :

- ❖ **Proactive** : Les agents ne répondent pas seulement à l'environnement, ils peuvent montrer un comportement déterminé en prenant l'initiative.
- ❖ **Adaptatif** : Les agents adaptatifs sont ceux qui peuvent contrôler et ajuster leurs compétences (communication, comportement, etc.) en réponse à des changements dans les connaissances internes ou l'environnement.
- ❖ **Intentions** : Ces agents ont un plan d'action interne clair pour atteindre un objectif ou un ensemble d'objectifs.
- ❖ **Persistance** : les agents persistants ont un état interne qui reste cohérent [5].
- ❖ **Mobilité** : Les agents mobiles peuvent choisir de migrer vers un autre ordinateur ou réseau tout en conservant la persistance.

2.2.3. Caractéristiques tertiaires :

- ❖ **Emotion** : A la capacité d'exprimer l'émotion ou l'humeur comme un être humain ; De tels agents peuvent également avoir une forme ou des caractéristiques anthropomorphiques.
- ❖ **L'Intelligence** : Avec la capacité de raisonner, d'apprendre et de s'adapter avec le temps. Un agent est intelligent s'il peut effectuer des actions flexibles et autonomes pour atteindre les objectifs qui lui sont fixés. La flexibilité signifie réactivité, initiative et attitudes sociales [5].

2.3. L'intelligence d'agent :

A partir des caractéristiques citées dans la section précédente on peut les considérer comme des critères de classification des agents.

Dans cette section nous allons détailler une classification basée sur le critère de la **granularité** des agents parce qu'elle est la classification la plus adoptée par les chercheurs de domaine. Par la granularité, nous entendons la complexité des comportements et des connaissances des agents [1].

- ❖ **Les agents cognitifs** : Ces agents font partie de l'intelligence symbolique. Ils représentent globalement leur environnement et les agents avec lesquels ils communiquent, tout en considérant leurs actions antérieures. Chaque agent possède une base de connaissances qui contient toutes les informations nécessaires pour accomplir ses tâches et interagir avec l'environnement et d'autres agents.

On note que les SMA constitué de cette catégorie est composé d'un petit nombre des agents hétérogènes [1].

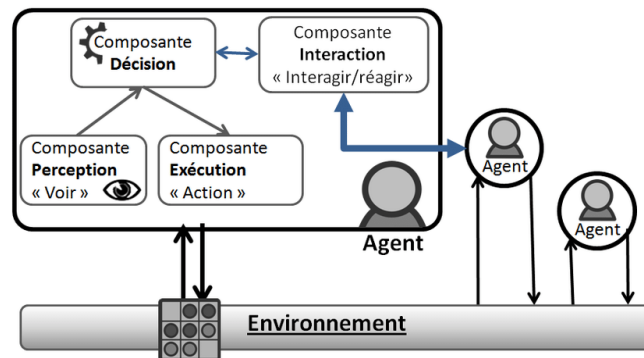


Figure2.3.1. Structure d'un agent cognitif dans un environnement MA [6].

- ❖ **Les agents réactifs** : Ces agents font partie de l'intelligence artificielle réactive ; Cette école de l'intelligence artificielle est basée sur la possibilité de concevoir des comportements intelligents à partir de comportements simples [1].

Les agents sont simples et ne représentent plus ni leur environnement ni leur mémoire, ce qui les prive de leur apprentissage et de toute attente d'événements. Ils ne se caractérisent par aucune structure organisationnelle initiale prédéfinie et les agents peuvent agir naturellement lorsqu'une action est nécessaire. Leur comportement est du

type "stimuli-réponses". A l'inverse des agents cognitifs, un système multi-agents basé sur les agents réactifs est constitué d'un grand nombre d'agents homogènes [1].

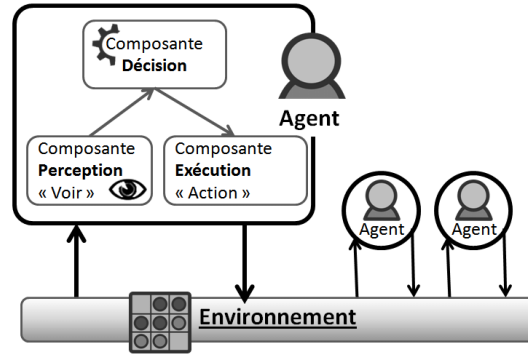


Figure2.3.2. Structure d'un agent réactif dans un environnement MA [7].

- ❖ **Les agents hybrides :** Ce type d'agent combine les deux classements précédents, qui sont généralement distribués sur plusieurs couches. Ces agents sont répartis comme suit : les couches de hauts niveaux rassemblent purement des agents cognitifs qui sont responsables de capacités cognitives complexes comme les aspects sociaux (manipulent des connaissances). Par contre les couches de bas niveaux contiennent les agents réactifs qui exécutent des tâches élémentaires à partir de l'ordre de la supérieures couches ou par leur initiative (manipulent directement des données). En conséquence, des couches intermédiaires (peut être regroupé les deux types des agents) sont responsables de transformation des données en connaissances.

2.4. La rationalité des agents :

À travers un détecteur et agit sur l'environnement à travers des effecteurs. Nous voulons qu'un agent intelligent effectue la tâche pour nous tout en agissant de manière rationnelle.

Alors **un agent rationnel** est un agent qui agit d'une manière lui permettant d'obtenir le plus de succès possible dans la réalisation des tâches qu'on lui a assignées.

2.5. La coopération des agents :

FERBER dans son livre a défini la coopération comme suit : « On dira que plusieurs agents coopèrent, ou encore qu'ils sont en situation de coopération, si l'une de ces conditions est vérifiée :

- 1) L'ajout d'un nouvel agent permet d'accroître différenciellement la performance de groupe ;
- 2) L'action des agents sert à éviter ou à résoudre des conflits potentiels ou actuels. » [2].

Il a ensuite ajouté : « c'est parce qu'ils coopèrent que les agents peuvent accomplir plus que la somme de leur action, mais c'est aussi à cause de leur multitude qu'ils doivent coordonner leurs actions et résoudre des conflits » [2]. Généralement, la coopération est presque une attitude intentionnelle ; Alors **un agent coopératif** est un agent qui engage dans des activités collectives après avoir identifié et adopté un but commun [12].

2.6. La capacité sociale des agents :

Cette notion est abordée d'une manière détaillée lorsqu'on traite les SMA. Alors on dit qu'un agent est **un agent social** lorsqu'il est capable d'interagir pour résoudre des problèmes et d'aider les autres agents dans leur activité [13]. On peut le définir aussi comme suit : « est une entité interagissant autonome qui exerce des actions dans un contexte social. » **Source spécifiée non valide.** Ces derniers sont appartenant à un ou plusieurs communautés.

2.7. L'égoïsme d'un agent :

Lorsqu'un conflit apparaît, un agent de tempérament égoïste privilégiera à tout prix son propre intérêt supérieur [10]. Ils travailleront pour accroître la résilience du système et développeront des stratégies pour s'assurer qu'ils disposent des ressources et des compétences dont ils ont besoin au bon temps.

3. L'environnement :

Par définition, un agent est une entité qui agit sur l'environnement. En effet, cette dernière est un élément important d'un SMA, d'après l'environnement on fait référence

aux objectifs passifs qui compose leur monde .Habituellement, les agents se développe dans ce monde.

Malgré l'importance de l'environnement, l'environnement n'as pas reçu l'attention qu'il mérite par les chercheurs de domaine que récemment. Elle est considéré la partie implicite et intégré au SMA de manière ad hoc ; alors, changer une partie de système signifie changer l'environnement.

Une conception qui met en accent l'importance d'un environnement multi-agents, considère ce dernier comme une entité de première classe. On entend par une entité de première classe, un module logiciel indépendant qui fournit une abstraction (ou un mécanisme de masquage d'information) permettant le changement de son Les implémentation sans modifier les autres modules [1].

Et pour limiter cette confusion, un environnement peut modéliser par une architecture de couches.la figure suivante illustre la modélisation en 3 couches dans un environnement multi agents.

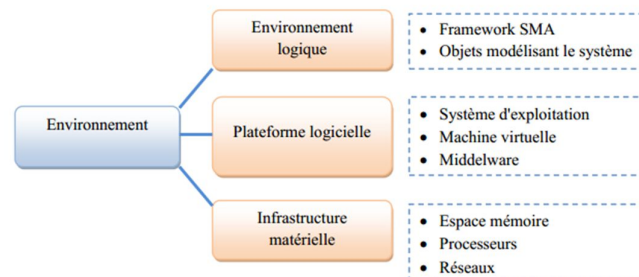


Figure3.1. L'architecture en couche d'un environnement [1].

4. L'organisation :

Au cours de l'étude des SMA nous pouvons jamais d'ignorer la société dans laquelle l'agent est intégré.

Nous étudions généralement les agents selon deux aspects Différentes dimensions : micro-niveau et macro-niveau. Le niveau micro indique L'architecture interne d'un agent avec ces mécanismes d'inférence. Au niveau Macro, nous positionnons le système multi-agents comme une organisation. Pendant l'étude de comportements de système on doit concentrer sur les fonctions des agents et non sur leur comportements (**fait quoi pas comment**). Dans l'étude des organisations nous distinguons deux types :

- ✓ **Organisation structurelle** : est composée par un ensemble des rôles avec des relations entre eux. Elle ne représente que les caractéristiques abstraites d'une organisation (représente que l'organigramme de l'organisation) [1].
- ✓ **Organisation concrète** : représente une instanciation de l'organisation structurelle. Ainsi, dans cette organisation nous trouvons les agents qui jouent chaque rôle [1].

4.1. Traitement des organisations :

Les organisations des SMA sont traitées selon les axes suivants :

- ❖ **L'analyse structurelle** : Ce que cette analyse fait est d'identifier la forme Organisation. En effet, un système multi-agents peut être organisé dans les conditions suivantes En tant que **hiérarchie, groupe** ou **coalition**.
- ❖ **L'analyse fonctionnelle** : Cette analyse détermine les fonctions de Membres de l'organisation. Bien que les rôles qui existent dans un étroitement lié à certains rôles génériques peut être couramment utilisé comme facilitateur, courtier et le médiateur. Ses rôles fournissent la communication, les services d'annuaire et / ou coordination dans un système multi-agents [1].
- ❖ **Les paramètres de concrétisation** : Cette dimension concerne la performance des organisations ADM de l'organisation structurelle à l'organisation spécifique.

Le problème de base avec cet axe est Attribué des rôles à différents agents qui composent un SMA. Deux normes peuvent guider les concepteurs dans la résolution de ce problème : spécialisation et redondance. Selon la norme Spécialisation, le concepteur doit décider si l'agence doit être donnée Spécialisé dans un seul rôle ou universel dans ses capacités Jouez plusieurs rôles. Cependant, la norme de redondance spécifie La possibilité d'attribuer un rôle unique à un groupe d'agents.

5. L'interaction :

L'interaction est la condition importante des agents pour atteindre leurs objectifs. Alors qu'un agent qui ne peut pas d'atteindre les objectifs n'est qu'une entité handicapée. C'est pour cela l'interaction des agents est considère le raison d'être des SMA ; En fait, l'interaction oblige les agents de raisonner sur les capacités des autres agents, de leurs croyances, de leurs buts,...etc. [2]. De plus, des interactions mal gérées peuvent conduire à des situations indésirables voire chaotiques.

Définition : «**une interaction est une mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'action réciproques** » [8].

Il reste maintenant aux agents la possibilité de communiquer entre eux ; il peut y avoir plusieurs objectifs liés aux actes de communication entre agents.

5.2. Les types d'interactions :

5.1.1. Coopération entre agents : S'ils coopèrent, l'objectif de l'agent n'est pas seulement de maximiser leur satisfaction, mais aussi de contribuer au succès de l'équipe. Les agents travaillent ensemble pour résoudre un problème commun.

5.1.2. Coordination entre agents : La coordination est le problème central de l'analyse des SMA et des systèmes distribués. En effet, l'absence de coordination causé rapidement une collection chaotique d'individus. Pour assurer un comportement cohérent du groupe d'agent serait de le faire par un centralisateur ; cet agent peut créer des plans d'actions et assigné les tâches aux divers agents du groupe. Cette dernière approche est pratiquement impossible à cause de la difficulté à obtenir de tel agent centralisé sans tenir compte des objectifs, des connaissances et des activités de chaque agents sans considérer que les avantages d'un SMA composé d'agent autonome seront perdus.

On distingue deux composants fondamentaux de la coordination entre agents ; ce sont :

- ✓ **l'allocation de ressources rares :** pour l'allocation des ressources partagées, les agents doivent être capables de faire des transferts de ressources.
- ✓ **La communication des résultats intermédiaires :** les agents doivent être capables de communiquer entre eux de façon à pouvoir échanger les résultats intermédiaires [9].

5.1.3. Négociation entre agents : la négociation est une composante de base de l'interaction ; car elle joue un rôle fondamental dans les activités coopératives, permettant aux personnes de résoudre les conflits qui peuvent mettre en danger le comportement coopératif. Généralement

les chercheurs d'IAD l'utilisent comme un mécanisme pour coordonner un groupe d'agents.

5.2. Communication entre agents :

Les agents peuvent interagir en effectuant des opérations linguistiques (communication entre eux). Soit en effectuant des opérations non-linguistiques qui modifient leur environnement. En cour de communication, les agents peuvent échange des informations et coordonnée leur activités ; cela besoin un langage de communication commun entre agents. Dans les SMA on trouve deux principales stratégies utilisées pour supporter la communication entre agents :

- ❖ **Envoi de message (communication directe) :** les agents peuvent échange des messages directement.
- ❖ **Partage de ressources (communication indirecte) :** les agents accèdent à une base de données partagée où dans laquelle les informations sont publiées.

5.3. Les éléments de communication :

La communication est généralement basée sur les trios éléments suivantes :

- ✓ **Protocole d'interaction entre agents :** Ceci est lié aux stratégies avancées poursuivies par les agents logiciels interagissant avec d'autres agents. L'interaction entre agent exige un ensemble de messages convenus, de règles pour des actions basée sur la réception de divers messages et d'acceptation de voies de transmission [9]. Ces contraintes peuvent être soustraites et formalisées en AIP (Agent Interaction Agreement), qui constitue la base de la négociation et des agents coopératives. En utilise AIP pour représenter les messages et les restrictions correspondantes sur le contenu de ces messages.
- ✓ **Protocole de transport des messages :** Ceci est le mécanisme de transmission réel pour la communication ; en utilisant le langage de communication. Pour assurer un bien échange entre les agents, il y a plusieurs moyens : au niveau le plus bas, il existe des *sockets* qui permettent aux différents agents codées en Java de communiquer entre eux ; il existe aussi d'autre technologie (en Java notamment). On peut citer

l'invocation de méthode distantes RMI (pour Remote Method Invocation) et la technologie COBRA (Common Request Broker Architecture) [9].

- ✓ **Langage de communication entre agents** : C'est un moyen de transmettre des attitudes sur le contenu des messages échangés.
 - **ACL (Agent Communication Language)** : Conçu pour assurer l'interopérabilité entre les agents autonomes et distribués. ACL a trois composantes : le vocabulaire, le langage de communication entre les agents et le langage du contenu spécifié, appelée KIF (Knowledge Access Format).
 - **KOML (Knowledge Query and Manipulation Language)** : est issu d'un projet de la DARPA [9]. Il s'agit d'un langage conçu pour définir un ensemble standard et utile de comportements langagiers. Ces actes de langage (également appelés performatives) sont utilisés par les agents pour échanger des informations. Un message est divisée en trois couches : la couche communication, la couche message et couche contenu ; il a été conçue comme format de message et de protocole qui permet l'identification.
 - **FIPA-ACL** : créer par FIPA (Fondation for Phisycal Intelligent Agent) été conçu pour pallier des faiblesses des différents versions de KQML.

Conclusion :

Dans ce chapitre nous avons présentées une vision globale des systèmes multi agents et des agents. Ces systèmes c'est un réseau d'agent qui interagissent, communiquent et coopèrent entre eux pour accomplir un objectif bien précise ; et nous avons décrivons tous les éléments constituant un SMA on part de l'agent à l'organisation en passant par l'environnement et finalement, les interactions et communication entre agents.

Le chapitre suivant va être consacré à la présentation détaillée de négociation des agents.

Chapitre 2 : Négociation multi agents.

Résumé :

Ce chapitre est composé de deux parties, le premier partie où nous présentés la négociation, leur composants de base, les modèles, les protocoles de négociation et une validation des protocoles, et les stratégies de négociation. Dans Le deuxième nous fournissons une définition sur la négociation multi issues et nous prend un modèle d'une négociation multi issues.

Introduction :

Dans le cas des SMA, où les agents sont autonomes, la négociation est la meilleure solution pour résoudre les conflits entre eux.

Alors la négociation est un mécanisme inspiré du modèle humains pour résolution des conflits ; ainsi qu'elle permet à tous les agents d'atteindre leurs buts à l'arrive à un accord commun pour les deux parties négociateurs. Dans ce chapitre on va détailler qu'est-ce qu'une négociation, leur composants de base ainsi que les modèles et les formes de négociation automatique ; dans le deuxième partie de ce chapitre on va toucher la négociation multi issue.

I. La négociation :

La négociation est toujours un sujet très intéressant dans les SMA. Selon le point de vue de MH.VERRON dans [10], il y a négociation lorsqu'il y a une discussion, des propositions entre les protagonistes et lorsque l'accord final satisfait au mieux tous les participants. VERRON a donné aussi une autre définition dans [10] : « Une négociation met en jeu des ressources, qui seront rassemblées afin d'être négociées dans un contrat et un ensemble de personnes qui participent à cette négociation. Il y a toujours un ou plusieurs manageurs (vendeur ou autre) et un ou plusieurs contractants (acheteurs ou autre) ».

Autre définition ; une négociation est un octuple (G, R, P, L, S, G, H et Ct) où :

- **G** est l'ensemble des individus impliqué dans la négociation.
- **R** est l'ensemble des ressources disponibles.
- **P** est un ensemble de relation de préférence local de chaque agent.
- **L** est l'ensemble des actes de langage autorisé pour la négociation.
- **S** est un ensemble de règles de séquencement.
- **G** est un graphe dans les nœuds sont des règles de séquencement.
- **H** est l'histoire de la négociation.
- **Ct** est un contrat qui est l'objet de la négociation. [11]

Pour assurer une négociation réussite, La modélisation des processus doit prendre en compte plusieurs Aspect. Ces aspects doivent être clairement spécifiés :

- **Le langage de négociation** : comme on a vu précédemment que la communication est le support de l'interaction. les agents ne peuvent pas négocier sans échange des messages.
Autre que ça, le langage de communication doit inclure l'expression primitive de communication. Par exemple ; en utilisant le langage de communication, un agent doit être capable d'exprimer son avis concernant une proposition (acceptation ou refus) comme il doit être capable de proposer ou de refuser la participation à une négociation [1].
- **Le protocole de négociation** : le protocole de négociation précise les règles de rencontre entre les agents négociateurs. Ces protocoles spécifient pour chaque moment du processus de négociation un ensemble valide de réponses ou d'interactions. Par exemple, dès réception d'une demande de participation à la négociation, l'agent a droit de participer ou de refuser la participation. C'est inacceptable et non valide d'envoyer un message d'échec à l'initiateur à ce niveau. En plus, les protocoles déterminent la terminaison de négociation avec réussite ou terminaison échouée.
- **L'objet de négociation** : il faut spécifier clairement l'objet de négociation avant le début de processus. L'agent sera une entité perdue et sans objectif. Un agent vendeur qui minimise le prix d'un produit « X » pour arriver à un accord avec un agent qui cherche un produit « Y » et qui n'est pas intéressé par le produit « X » est un agent irrationnel. On note que l'objet de la négociation peut être spécifié par un seul attribut (par exemple, le prix) ou plusieurs attributs (comme le type de produit, la couleur, la qualité et le prix) [1].
- **Le processus de décision** : Bien sûr, les décisions d'agent durant le processus de négociation ne sera pas une décision aléatoire. Les agents doivent suivre une stratégie spécifique au cours de ce processus appelée **stratégie de négociation**. Cette stratégie permet aux agents de prendre les bonnes décisions à tout moment. Par exemple, un acheteur qui négocie pour acheter un produit avec le moindre prix possible, quand est-il décide que le prix proposé par le vendeur à un instant donné est le prix minimum ? Pour prendre ces décisions, l'agent doit être capable de raisonner sur le raisonnement des autres agents. Il doit être aussi capable d'identifier les

stratégies suivies par les autres agents. Un joueur d'échec ne raisonne pas seulement sur ses mouvements possibles mais aussi sur les mouvements possibles de son adversaire [1].

- **La cardinalité des participants de la négociation** : négociation un-à-un, un-à plusieurs ou bien plusieurs-à-plusieurs. [13]

2. Les formes de négociation :

Il y a plusieurs techniques de négociation on peut les traiter dans le domaine des SMA, dans cette partie on va citer quelque formes de négociation :

2.1. Les systèmes de vote :

Le système de vote est utilisé pour choisir un suppléant parmi les différents électeurs. Le plus simple consiste à choisir entre des alternatives et des alternatives Statu quo. Il s'agit de proposer des alternatives et de recueillir les votes et les bulletins de vote contre ce choix. Les systèmes plus complexes impliquent de nombreux plus de deux alternatives, alors les électeurs doivent choisir l'alternative qu'ils choisissent comme ça. Le terme choix social est utilisé pour indiquer un choix qui remplit une condition Le plus peuplé.

Les systèmes de vote sont des systèmes de négociation à un seul tour de parole (les participants envoient leur liste de préférences et l'initiateur réalise la procédure choisie) qui permettent de choisir parmi toutes les alternatives possibles. Il faut donc que chacun évalue l'ensemble des alternatives et les classe par ordre de préférence avant que le vote n'ait lieu. Cela est alors très coûteux en temps et en espace. « *GeNCA : un modèle général négociation de contrats entre agents* ». [14]

2.2. Les enchères :

La vente aux enchères est un mécanisme de négociation où les agents rivaliser avec plusieurs autres agents pour acheter ou vendre des biens. Cette concurrence permettra de déterminer objectivement le prix d'un objet. Ce mécanisme est bien connu depuis l'Antiquité, de plus il est utilisé dans différent site de vente en ligne.

La philosophie générale d'enchère est la proposition des prix par des participants. Ensuite, l'initiateur va décider la meilleure proposition. Bien

entendu, le nombre de tours (c'est-à-dire le nombre de fois que l'agent peut proposer un prix), la méthode d'annonce des propositions (publique ou privée) et la méthode de choix de vainqueur sont les caractéristiques qui font la différence entre les protocoles d'enchère [1]. Cependant, il y a des choses en commun dans la plupart des protocoles, par exemple s'il existe un prix de réserve, ce prix est le plus donnée par l'initiateur pour un objet. Par conséquent, les participants n'ont pas le droit d'annoncer en dessous du prix de réserve. Si tous les participants fournir le prix au prix de réserve le protocole va terminer par un échec.

Dans le domaine des SMA, il existe quatre catégories principales d'enchères :

❖ **Les enchères ascendantes (Anglaise) :**

Ces enchères appellent ascendante car la stratégie de ce protocole est d'augmenter les prix avec le temps. Elles sont aussi dites enchères ouvertes, orales ou anglaise. Ce protocole s'applique aux tours successifs, ou le prix est successivement augmenter jusqu'il reste l'agent vainqueur, celui qui propose le plus grand offre et gagner l'article.

Autrement dit, il y a une discussion entre le vendeur et les acheteurs, ce qui est bien à notre sens de négociation. Cette négociation peut se formaliser comme suit : le vendeur commence l'enchère par l'annonce de prix de réservation, ensuite les autres agents (les acheteurs) vont annoncer publiquement un prix plus grand que le prix proposé dernièrement, les agents continuer la proposition des prix jusqu'à l'obtention d'un prix gagnant. Un prix est gagnant si aucun agent ne peut augmenter par rapport à ce dernier. Le protocole est terminé, l'acheteur doit acquit l'objet, il ne peut plus le rendre [15].

❖ **Les enchères descendantes (Hollandaises) :**

Ces enchères dites descendants car le prix va diminuer dans chaque tour. Elles sont utilisées notamment aux **Pays-Bas** pour la vente des fleurs, c'est pourquoi les économistes les appellent aussi enchères hollandaises [15].

Ces enchères est considérer aussi comme une négociation car elles se déroulent sur plusieurs tours. Le protocole se définit comme suit : le vendeur présente leur article avec un prix élevé à un ensemble d'acheteurs ; si l'acheteur accepte la proposition il a remporté le contrat. Si tous les acheteurs refusent, le vendeur va poser leur article avec

un prix inférieur. Le processus se termine quand l'acheteur accepte la proposition, ou lorsque le prix de réserve est atteint et que personne n'accepte la proposition.

❖ **Les offres scellées au premier meilleurs prix :**

Ces enchères se déroulent sur un seul tour où les acheteurs proposent un prix une seule fois « prix unique » sans connaître les propositions des autres agents, c'est pour cela elles dites offres scellées. Le vainqueur ici celui qui propose le plus élevé prix et le paie.

Comme les acheteurs ne peuvent plus faire une deuxième tour, ces enchères ne sont pas des négociations au sens strict. Alors le protocole est très simple : le vendeur propose un article à un ensemble des acheteurs. Ceux-ci peuvent soit accepter la proposition où il propose un prix, soit refuser la proposition. Le gagnant de l'enchère celui qui propose le plus élevé prix (supérieur au prix réservé). L'une des différences avec les précédentes enchères est qu'il n'y a pas de contreproposition possible [15].

❖ **Les offres scellées au second meilleur prix (vickrey) :**

Ce protocole est similaire au protocole de premier meilleur prix, mis à part que l'agent gagnant paie le deuxième prix proposée. Ici aussi, il n'y a pas une seconde proposition possible implique y a pas une négociation proprement dites. Le protocole est identique à celui des offres scellées au meilleur prix, la seule différence étant le prix à payer par le gagnant du contrat [15].

2.3. La négociation à base d'argumentation :

La négociation à base d'argumentation est utilisée chez les agents logiques qui possèdent une base de connaissance avec des prédicats et des règles d'inférences [16]. L'argumentation est donc conçu pour changer les croyances des autres agents afin qu'il adopte le même point de vue, les mêmes croyances et intentions que l'agent argumentant.

Au temps de négociation les agents peuvent utilisés plusieurs types d'arguments ; ainsi que chaque argument à leur propre précondition pour l'utilisé. L'agent doit choisir le bon argument et l'utiliser lorsqu'il est satisfait, c'est pour cela l'agent besoin d'une stratégie pour effectuer leur choix. A partir de qui ce précède on trouve les types d'arguments suivants :

❖ **Appels à une promesse passée :**

Le négociateur A rappelle à B une promesse passée concernant l'objet de négociation. Autrement dit l'agent B a promis dans une négociation antérieure à l'agent A d'offrir ou effectuer un objet de négociation.

- Préconditions : l'agent a doit vérifier si une promesse concernant un objet de négociation a été reçue dans le passé à l'occasion d'une négociation conclue avec succès.

❖ **Promesse d'une récompense future :**

Le négociateur A promet de faire l'objet de négociation pour un autre agent B à un moment futur.

- Préconditions : l'agent A doit trouver un désir de l'agent B pour un moment futur, si possible un désir qui peut être satisfait par une action (service) que A peut effectuer mais que B ne peut pas effectuer.

❖ **Appels au propre intérêt :**

L'agent A croit que la conclusion d'un accord sur l'objet de négociation est dans l'intérêt de B et essaye de convaincre B de cela.

- Préconditions : l'agent A doit trouver (ou déduire) un des désirs de B qui sera satisfait si l'agent B a l'objet de négociation ou A doit trouver un autre objet de négociation *ON'* qui a été offert auparavant sur le marché et démontrer que l'objet de négociation proposé est plus intéressant que *ON'*.

❖ **Appel à une pratique fréquente :**

L'agent A croit que B a refusé la proposition parce que B croit que la proposition contredit un de ses buts. Dans ce cas, l'agent A donne à B l'exemple d'une pratique fréquente qui démontre que l'acceptation de la proposition ne contredit pas le but de B.

- Précondition : l'agent A doit trouver un autre agent ayant le même but que B, qui a été déjà d'accord avec une proposition semblable et qui a vu que l'accord n'a pas nui à ses buts.

❖ **Menace :**

Le négociateur menace de refuser à faire/offrir quelque chose à B ou il menace de faire quelque chose qui contredit les désirs de B.

- Préconditions : l'agent A doit trouver un des désirs de B directement satisfiable par un objet de négociation que l'agent A peut offrir ou A doit

trouver une action qui est contradictoire avec ce qu'il croit être un des désirs de B [16].

3. Les protocoles de négociation :

3.1. Définition

Un protocole de négociation n'est pas loin d'un protocole d'interaction, seulement que la négociation nécessite un conflit entre les agents. Alors nous adoptons la définition suivante : « Chaque forme de négociation possède son propre protocole, qui définit le déroulement du processus de négociation, c'est-à-dire les actes de langage utilisés et leur séquençement » [10]

3.2. Classification des protocoles de négociation :

Dans cette section on va citer quelques propositions de classifications des protocoles de négociation :

3.2.1. Les protocoles compétitifs vs coopératifs :

Selon P. Maes, il identifie deux classes de protocoles.

- La première classe regroupe les protocoles distributifs qui se basent sur le principe du partage des gains entre les participants. Chacun des participants cherche à maximiser sa part de profit, la réussite de l'acheteur signifie la défaite du vendeur, et vice versa.

Dans la théorie de jeu, on les appelle des jeux à somme nulle. P. Maes les considère comme des protocoles compétitifs, car c'est l'esprit qui domine dans ce genre de protocoles. Les protocoles de vente aux enchères sont considérés comme des protocoles compétitifs.

- La deuxième classe regroupe les protocoles intégratifs, dont l'objectif de ses participants est d'essayer d'élargir l'espace d'entente, et de chercher de nouvelles solutions qui maximisent les profits mutuels des différents participants. Dans la théorie de jeu, on les appelle des jeux à somme non nulle, ils sont considérés comme des protocoles coopératifs en intégrant plusieurs paramètres de négociation. [15]

3.2.1. Les protocoles unidirectionnels vs bidirectionnels :

- Dans les protocoles unidirectionnels, nous sommes limités à accepter ou rejeter la proposition. Cette catégorie comprend les protocoles de ventes aux enchères et Contract Net.
- D'autre côté les protocoles bidirectionnel (son scénario sont plus complexes) est basé sur l'échange des messages (offre et contre-offre). Ce type de protocoles fournit plus négociation flexible on modifiant d'un ou plusieurs attributs peut être négocié via une contre-proposition ou bien nous changeons la structure de la proposition. Par exemple nous ajoutons de nouveaux attributs négociables.

Il y a d'autre classification pour les protocoles de négociation, sont les suivants :

- Wurman et al, proposent une classification basée sur les critères suivants :
 - **Simple ou double** : un protocole est dit simple si le type des enchérisseurs est soit des agents acheteurs ou soit des agents vendeurs. Dans la double enchère on trouve des agents acheteurs et des agents vendeurs qui participent à la même enchère.
 - **Privé ou public** : dans le premier cas, les propositions soumises par les participants ne sont connues qu'après la clôture de l'enchère. Dans le deuxième cas, les propositions sont publiques, connues par tous les participants.
- **Conditionné ou inconditionné** : Un protocole est conditionné si les propositions sont régies par des règles, par exemple, le prix proposé doit être ascendant ou descendant.
- **London Classification** est un modèle de classification des systèmes de négociation, défini à l'issue du workshop *Negotiations in Electronic Markets*. Il traite l'aspect participant, l'aspect produit, l'aspect décisionnel et l'aspect processus de négociation. Ce dernier aspect est défini par un ensemble d'attributs, parmi lesquels :
 - ❖ Le type d'agent initiateur de la négociation qui peut être l'agent vendeur ou l'agent acheteur.
 - ❖ Le nombre de phase : un protocole est dit à simple phase si les règles de négociation ne changent pas tout au long du processus de négociation, dans le cas contraire, nous le considère comme un protocole multi-phases.

- ❖ L'ordonnancement dans la négociation multi-attributs peut être une seule étape si tous les attributs sont négociés en même temps ou multi étapes si on négocie chaque attribut à part.
- ❖ La publication des offres ou des propositions, si la proposition est entendue par tous les participants le protocole est dit public (ou open cry pour les protocoles de vente aux enchères). Dans le cas contraire on l'appelle protocole privé ou sealed bid.
- ❖ Le prix que paie le gagnant dans l'enchère dépend des règles du protocole utilisé : dans les protocoles discriminatifs le gagnant paie le prix de la proposition qu'il a soumis (par exemple l'enchère anglaise). Dans le protocole non discriminatifs le gagnant paie le prix plus bas que celui qu'il avait proposé. Par exemple, dans le protocole Vickrey le gagnant paye le prix de la deuxième proposition.
- ❖ Le désistement d'un participant sur un engagement qu'il avait pris au cours d'une négociation peut être permis ou non par le protocole de négociation.
- ❖ Sandholm aussi identifie cinq types de protocoles de négociation : les systèmes de vote, les enchères, marchandage, systèmes contractuels et la formation de coalition.

3.2. Quelques protocoles de négociation :

Il existe plusieurs protocoles de négociation, on va parler sur les deux protocoles les plus populaires :

- **Le protocole Contract Net :**

Le Protocole *Contract Net* proposé par Smith en 1980 est un mécanisme de négociation entre deux types d'agents : contractant et gestionnaire [13]. Le Contract Net permet après quelque discussion avec un groupe d'agents, le gestionnaire conserve le service d'un agent qui s'appelle un contractant pour effectuer une tâche (contact). Le protocole est qualifié « sélection mutuelle » car pour « signer un contact », l'agent sélectionné doit s'engager pour l'exécution de la tâche et le gestionnaire ne sélectionne que l'agent Fournit les propositions les plus favorables. La version originale du protocole comporte trois étapes principales :

1. L'appel d'offre
2. La soumission de proposition
3. L'attribution de contrat.

La section suivante présente l'algorithme de protocole Contract Net détaillée en sept étapes ; regroupe les traitements du gestionnaire et des contractants, ainsi que les échanges entre eux.

Pour appliquer ce protocole, le concepteur doit détailler le contenu des messages échangés, le temps d'expiration ainsi que les fonctions « évalue-annonce » et «évalue-soumission » [15].

Etant donné une tâche, un gestionnaire, un groupe de (n-1) soumissionnaire :

- 1- le gestionnaire envoie un message de type « annonce- tâche » à un groupe d'agents (ou fait un 'broadcast').
- 2- Chaque agent évalue la tâche annoncée à l'aide d'une fonction locale 'évalue-annonce'.
- 3- L'évaluation précédente permet à certains agents de soumettre une proposition à l'aide d'une 'soumission- tâche' au gestionnaire.
- 4- Si une proposition est jugée satisfaisante (l'aide de la fonction 'évalue-soumission'), alors le gestionnaire envoie un message de type 'acceptation' à celui dont la proposition est retenue. Il envoie également un message de type 'refus' aux autres agents dont les propositions n'ont pas été retenues.
- 5- Le gestionnaire peut mettre fin à la période d'acceptation de proposition si le temps d'expiration est dépassé.
- 6- C'est aucune proposition n'a été retenue, alors le gestionnaire fait parvenir à tous les agents non retenus un message de type 'refus' pour indiquer le rejet de chacune des propositions.
- 7- Il peut alors se retirer de la négociation, retenir la proposition la plus acceptable redémarrer un nouvel appel d'offre (nouveau 'annonce- tâche') ou prolonger le temps d'expiration de la période d'acceptation de proposition.

L'agent ayant obtenu le contrat, remet un rapport d'exécution lorsque la tâche est complétée [15].

- **Le protocole Kasbah** : a été développé au MIT Media Lab par Pattie Maes en 1996. C'est un système où les utilisateurs créent des agents pour négocier la vente et l'achat de biens pour leur compte sur Internet. Ces biens sont classifiés, reprenant ainsi l'idée des petites annonces classées par type, voici un exemple de négociation avec le protocole Kasbah [10]. Lors de la création d'un agent pour acheter ou vendre, l'utilisateur spécifie le type d'attribut Négociateur, la date à laquelle il veut trader, le

prix attendu, le plus Petit (ou grand) prix acceptable et stratégie de négociation choisis parmi 3 propositions Correspond aux fonctions linéaires, quadratiques et exponentielles pour le calcul Les prix changent au fil du temps. L'utilisateur spécifie également si l'agent doit demander Consentement avant de conclure un accord et si vous souhaitez être averti par e-mail lorsqu'un accord est conclu. Une fois un accord conclu, des transactions physiques peuvent avoir lieu et doivent être fabriqués par des agents humains.

4. Evaluation des protocoles de négociation :

On doit choisir le meilleur protocole pour chaque système ou domaine d'application ; pour assurer le choix d'un bon protocole il faut être capable d'évaluer les protocoles proposés, cette dernière effectuée seulement si on suit les critères d'évaluation suivants :

- **Rationalité individuelle** : on dit qu'un protocole est rationnel pour un agent en temps qu'il participe à la négociation ; il garantir un gain précise. Ou bien dit, l'agent ne perd rien en participant à la négociation [1]. Un agent apparemment cherchera à travers la participation à la négociation à accroître leur fonction d'utilité. Donc si la participation aux négociations n'augmentera pas les revenus d'agent, il n'y a pas un besoin pour participer.
- **Le bien-être social** : Parfois, nous pouvons choisir de négocier un protocole Non pas parce qu'il est rationnel envers l'agent, mais parce qu'il est le meilleur par rapport à la société. Les fonctions de choix social agrègent les utilités des individus. Alors que maximiser le bien-être utilitaire (la somme des utilités) vise à satisfaire la société dans son ensemble, maximiser le bien- être égalitaire (l'utilité minimale) consiste à réduire les inégalités dans la population, i.e. améliorer la satisfaction de l'agent le moins bien doté. Maximiser le bien-être de Nash (le produit des utilités) est un compromis entre ces deux règles de décision collective. Une alternative qui maximise le bien-être utilitaire ou le bien-être de Nash est optimale au sens de Pareto [18]. Par conséquent, nous comparerons les protocoles par rapport aux critères d'efficacité globale.
- **L'efficacité Pareto** : Une solution (résultat) x à un problème de négociation est efficace Pareto (Pareto optimale), s'il n'y a pas une autre solution x' telle qu'au moins un agent ait une meilleure situation avec x' qu'avec x et aucun agent n'ait une plus

mauvaise situation avec x' qu'avec x [17]. En d'autres termes, s'il n'y a pas d'autre résultat qui peut faire économiser plus d'argent à l'agent, et le coût d'un autre agent (il profite moins de cette solution), le résultat de la négociation est optimal Pareto. L'efficacité Pareto mesure les avantages globaux et ne nécessite pas de comparaison des services publics.

- **Stabilité** : Nous avons prédit qu'un agent en interaction doit être en mesure de discuter les actions possibles des autres agents. La négociation est traitée dans plusieurs domaines comme les guerres, les jeux stratégiques ou football c'est pour cela un agent doit être capable de connaître la stratégie de l'adversaire ; cette dernière on peut le trouver dans le domaine de théorie de jeux dans le théorème d'**Equilibre de Nash** :

- Deux stratégies, S_1 de l'agent A et S_2 de l'agent B, sont dans un équilibre Nash si :
 - dans le cas où l'agent A adopterait S_1 l'agent B ne peut pas faire mieux que d'utiliser S_2
 - dans le cas où l'agent B adopterait S_2 l'agent A ne peut pas faire mieux que de d'utiliser S_1 [17].

Par exemple, dans un match de football si l'un de deux équipes choisi la stratégie d'attaque pour gagner des buts, l'équipe adverse automatiquement doit adopter la stratégie de défense pour éviter ces buts-là.

- **La simplicité de calcul** : on dit qu'un protocole est simple si les agents peuvent facilement choisir une stratégie parfaite. De plus, un protocole doit être efficace du point de vue des calculs nécessaires pour déterminer la stratégie optimale.

5. Stratégie de négociation :

La stratégie de négociation d'un agent est une spécification de la séquence d'actions (généralement offres ou réponses) que l'agent envisage de faire pendant la négociation. Il y aura généralement de nombreuses stratégies compatibles avec un protocole particulier, dont chacune peut produire un résultat différent.

Pour étudier la stratégie de négociation on trouve trois approches principales, la théorie des jeux fournit des modèles théoriques abstraits dont les propriétés des accords et des processus ont été formellement prouvées. Au lieu de cela, l'heuristique fournit

Modèles basés sur des hypothèses réalistes, telles que la rationalité limitée, qui sont évalués empiriquement et mis en œuvre dans des applications pratiques. Enfin, la méthode d'argument Les lieux où les agents échangent des offres et des arguments.

➤ **La théorie des jeux :** Cette stratégie relie les fonctions d'utilité au protocole de concession monotone. En cas de désaccord, la perte de l'agent reconnu est inférieure à la perte de l'agent qui a fait moins de concessions. Le facteur de risque de ZEUTHEN (1930) évalue la propension d'un agent à risquer le désaccord. Il correspond au rapport entre sa perte maximale d'utilité s'il concède et sa perte d'utilité en cas de désaccord [18].

Par conséquent, les négociations commencent avec l'offre préférée. Ensuite, il suffit de négocier ce n'est pas encore fini, l'agent calcule à chaque fois ses facteurs de risque et les facteurs de risque de la contrepartie. Où ses facteurs de risque sont inférieurs ou égaux à ceux de son adversaire, il fait donc des concessions assez minimum. Sinon, il répétera la citation précédente. Grâce à suffisamment d'offres minimales, nous désigne l'offre fera basculer de l'ordre vers la prochaine série de facteurs de risque (en supposant L'adversaire restera ferme) tout en transférant une utilité minimale à l'adversaire.

Ainsi, une négociation entre deux agents qui adoptent cette stratégie consiste en une alternance de concession minimale. On peut remarquer que cette stratégie suppose que l'agent dispose d'une information parfaite, c.-à-d. qu'il connaisse les préférences de son interlocuteur.

➤ **Approche heuristique :** il s'agit d'une version continue du protocole d'enchère alternatif avec une date limite. Lorsqu'un offre reçu, il va évaluer par la fonction utilitaire multi-attribut. Les offres sont générées par une combinaison linéaire de fonctions simples (appelées tactiques). Les tactiques calculent les valeurs d'attribut (prix, quantité, qualité, etc.) en fonction de critères. Ainsi, trois familles de tactiques sont suggérées :

- Les tactiques temporelles qui déterminent quand un agent doit concéder.
- Les tactiques qui contraignent les offres en fonction des ressources disponibles.
- Les tactiques imitatives qui déterminent le comportement de l'agent en fonction de celui de son opposant [18].

Selon la fonction de concession d'une tactique de temps, l'agent initie la négociation en proposant une valeur maximale (si l'utilité diminue à mesure que la valeur d'attribut diminue), puis des concessions Jusqu'à ce que la valeur de rétention soit atteinte avant la date limite (cette fonction est paramétrée). Les polynômes ont initialement produit plus rapidement que les fonctions exponentielles. Ainsi, on distingue deux tactiques temporelles :

- la tactique ***Boulware*** qui consiste à rester sur ses positions jusqu'à la date butoir pour y proposer sa valeur de réserve ;
- la tactique ***Conceder*** qui consiste à concéder rapidement [18].

Selon la fonction de concession réciproque, l'agent reproduit proportionnellement le comportement de son opposant. La fonction peut être affinée en pseudo-aléatoire ou par calcul du comportement de l'adversaire dans la fenêtre de temps.

- **L'approche argumentative** : l'argumentation est une approche qui permet aux agents, en plus d'échanger des propositions, d'apporter des justifications concernant les propositions qu'ils soumettent et leurs décisions (i.e., acceptation ou rejet d'une proposition). Chaque agent peut fournir des arguments accompagnant ses propositions dans le but d'expliquer les raisons pour lesquelles les autres devraient les accepter. Lorsqu'un agent rejette une proposition, il peut aussi formuler des critiques expliquant les raisons pour lesquelles cette proposition est inacceptable. Cela permet à certains agents de modifier leurs croyances et préférences. Ces échanges d'informations additionnelles conduisant les agents à s'influencer mutuellement, peuvent aussi avoir un impact important sur le résultat de la négociation [19].

En résumé, Une stratégie est constituée de trois composants :

- une règle d'acceptabilité qui décide si une offre est admissible
- une stratégie d'offre qui détermine la concession en fonction des préférences, de l'historique et éventuellement d'une valeur de réserve, du temps et du modèle d'opposant (si disponible).
- éventuellement une stratégie d'apprentissage ou d'adaptation qui s'appuie sur un modèle d'opposant [18].

II. Négociation multi-issues :

En plus de ce qui était dénommer dans la partie précédente, il y a d'autres formes de négociation moins connues de grand public, ces négociation sont tout aussi nombreuse et variées [15] Parmi ces négociations nous trouve la négociation multi issues.

1. Négociation multi issues

La négociation multi issues comme leur nom indique, sont des négociations qui impliquent des problèmes indépendants les uns des autres .elle sont directement opposée aux enchères qui s'impliquent un seul attribut d'un seul problème. Cette forme de négociation est cependant très rependue et à la base de nombreuse variantes de négociation.

2. Modèle de négociation multi issues :

Nous prenons le modèle de la négociation multi issues suivant ; Supposons que l'acheteur, **b**, et le vendeur, **S**, qui ont des délais inégaux, négocient le prix de deux biens / services distincts, **X** et **Y**. Ici ; **T_a** dénote la date limite de l'agent, **a**, pour parvenir à un accord sur les deux issues.

La négociation sur toutes les issues doit être achevée avant la première des deux date limites. Nous considérons deux biens / services afin de simplifier la discussion, mais ceci est un cadre général qui travaille pour plus de deux biens / services.

➤ Etat d'information des agents :

En dénote les valeurs de réservation de l'acheteur pour X et Y par RP_x^b et RP_y^b et l'autre du vendeur sont respectivement RP_x^s et RP_y^s , aussi S_x^a dénote la stratégie de l'agent pour l'issue X et S_y^a indique la stratégie de l'agent a pour l'issue Y. alors l'état d'information de l'acheteur est :

$$I^b = \{ RP_x^b, RP_y^b, T^b, U^b, S_x^b, S_y^b, L_t^s, L_x^s, L_y^s \}$$

$L_t^s ; L_x^s$ et L_y^s sont trois distributions de probabilité qui dénote ses croyances sur les paramètres de l'adversaire ; une autre fois $L_t^s ; L_x^s$ et L_y^s dénote les croyances de

l'acheteur concernant le délais de vendeur, sa valeur de réservation pour X et sa valeur de réservation pour Y respectivement.

De manière similaire, l'état des informations du vendeur est défini comme :

$$I^s = \{RP_x^s, RP_y^s, T^s, s, S_x^b, S_y^b, L_t^b, L_x^b, L_y^b\}$$

L'état d'information de chaque agent est sa connaissance privée.

➤ **Protocole de négociation :**

Nous utilisons un protocole de négociation d'offres alternatives. L'un des agents commence par faire une offre combinée. Un autre agent peut accepter ou rejeter une partie de l'offre (un seul issue) ou l'offre complète. S'il rejette l'offre complète, il envoie une contre-offre combinée. Le processus de proposition d'une fusion se poursuit jusqu'à ce qu'un accord soit conclu sur l'un des points. Après L'agent ne fait qu'une offre sur les issues restantes (c'est-à-dire qu'une fois qu'un accord est conclu sur l'issue, il ne peut être renégocié). Lorsqu'un accord est conclu sur deux questions ou que le délai est atteint, les négociations sont conclues. Soit $S_{0x}^b(t)$ le prix généré par le meilleur agent de l'agent, b, Stratégie d'émission X au temps t. Par conséquent, l'action telle que prise par l'agent, s, au moment t d'une offre unique. Son action sur une offre combinée, $A_s(t, X_{b \rightarrow s}^t, Y_{b \rightarrow s}^t)$, est défini comme:

$$A^s(t, X_{b \rightarrow s}^t, Y_{b \rightarrow s}^t) = \begin{cases} \text{Quitter} & \text{si } t > T^s ; \\ \text{Accepter } X_{b \rightarrow s}^t & \text{si } X_{b \rightarrow s}^t \geq S_{0x}^b(t), \\ \text{Accepter } Y_{b \rightarrow s}^t & \text{si } Y_{b \rightarrow s}^t \geq S_{0y}^b(t), \\ \text{Offre } S_{0x}^s(t') \text{ à } t' & \text{si } X_{b \rightarrow s}^t \text{ n'est pas accepté,} \\ \text{Offre } S_{0y}^s(t') \text{ à } t' & \text{si } Y_{b \rightarrow s}^t \text{ n'est pas accepté.} \end{cases}$$

La fonction d'utilité des agents est comme suit :

$$U^a(P_x, P_y, t) = \begin{cases} (RP_x^b - P_x)(\delta_x^b)^t + (RP_y^b - P_y)(\delta_y^b)^t & \text{Pour b} \\ (P_x - RP_x^s)(\delta_x^s)^t + (P_y - RP_y^s)(\delta_y^s)^t & \text{Pour s} \end{cases} \quad [20].$$

Conclusion

Les négociations jouent un rôle vital dans le cas où les agents doivent résoudre des conflits pouvant mettre en danger les comportements coopératifs. Dans ce chapitre nous avons présenté les différents concepts de négociation qui permettent à comprendre parfaitement la négociation entre les agents. Nous pouvons dire que par la négociation, nous permet de résoudre les conflits sans touches l'aspect autonomie de l'agent car ce dernier peut continuer ou s'arrêter la négociation.

Dans le chapitre suivant nous allons présenter un état de l'art sur les méthodes de modélisations de l'adversaire dans la négociation.

Chapitre 3 : Les techniques de modélisation des adversaires dans une négociation multi agents

Résumé :

Nous appuyons sur l'apprendre des adversaires et les techniques de modélisation de ces derniers de plus nous fournissons un revue d'état de l'art contient les plus connues méthodes utilisé pour modéliser les agents de négociation.

Introduction

La négociation est un processus dans lequel les parties interagissent pour régler un souci mutuel d'améliorer leur statu quo. Traditionnellement ; la négociation est une activité indispensable, mais le temps de négociation est long et le coût est élevé.

La technique de modélisation de l'adversaire est applicable à divers protocoles de négociation. Ces derniers peuvent différer de plusieurs façons, notamment la zone de configuration, le nombre d'agents, l'état du problème et la disponibilité des informations.

Dans ce chapitre nous avons proposées une revue de l'état de l'art sur les techniques de modélisation d'une négociation, nous focalisant sur la variante la plus élémentaire des méthodes : apprentissage bayésien, régression non linéaire, estimation de la densité du noyau, réseaux de neurones artificiels.

1. Etat de l'art :

Dans ce chapitre nous présentons des différents types de modèles d'adversaires basés sur plusieurs techniques d'apprentissage. Y compris l'apprentissage bayésien, régression non linéaire, estimation de la densité du noyau et les réseaux de neurones artificiel. Nous proposons cette revue de l'état de l'art qui rassemble un ensemble de recherches déjà fait sur ces derniers techniques.

- **Apprentissage bayésien :**

Zeng et Sycara dans leur travail propose une méthode d'apprentissage bayésienne pour estimer la valeur de la réservation, en utilisant les données de précédentes négociations. Une seule issue quantitative est négociée, pour laquelle on suppose que les agents ont des préférences opposées. Avant la négociation, on a un ensemble d'hypothèses $H = \{h_1, \dots, h_n\}$ sur la valeur de réservation de l'adversaire est généré. Chaque hypothèse H_i est de la forme $r_v = v_i$, où v_i est l'une des valeurs possibles pour la valeur de réservation de l'adversaire r_v . Les hypothèses, les valeurs v_i et leur vraisemblance a priori sont toutes déterminées sur la base des connaissances du domaine dérivées des négociations précédentes. En appliquant l'apprentissage bayésien pendant la négociation, les probabilités des hypothèses sont mises à jour en fonction du

comportement observé et des connaissances du domaine disponibles. Intuitivement, l'idée est qu'une offre au début de la négociation est susceptible d'être loin de la valeur de la réservation. La valeur de la réservation est estimée en utilisant la somme pondérée des hypothèses en fonction de leur probabilité. Cette méthode est largement appliquée ; par exemple dans les travaux de Ren et Anumba et Zhang et al.

On addition Sim et al dans leur article : Appliquer la même procédure lorsque l'adversaire est contraint d'utiliser une tactique temporelle particulière, mais avec un délai privé, en considérant le cas où un agent utilise BL et date limite pour l'élaboration de sa prochaine proposition dans les trois combinaisons de délais, (court, moyen), (moyen, moyen) et (Long, moyen). La performance a montré que :

1) En cas (court, moyen), BLNA peut apprendre l'adversaire délai est plus long que son adversaire presque exactement.

2) Dans le cas (Mid, Mid), BLNA peut apprendre l'adversaire délai n'est pas plus court que son adversaire avec certains les erreurs.

3) Dans le cas (Long, Mid), BLNA peut apprendre les RP et délai avec des erreurs plus élevées.

Alors ; La fonction de décision de l'opposant est supposée revêtir une forme particulière dans laquelle la valeur de la réservation et le délai sont liés, en ce sens que l'un peut être dérivé de l'autre. Nous le trouve aussi dans autre articles.

- **Régression non _ linéaire :**

Une approche différente est adoptée par Hou, qui présente une méthode pour estimer la tactique de l'adversaire dans une négociation sur un seul problème quantitatif avec des délais privés. On suppose que l'adversaire utilise une tactique qui dépend du temps, du comportement ou des ressources.

La régression non linéaire est utilisée pour estimer lequel des trois types de stratégies est utilisé et pour estimer les valeurs des paramètres associés à la tactique, y compris la valeur de réservation. Une approche similaire est suivie par Agrawal et Chari, qui modélisent la fonction de décision de l'opposant en tant que fonction exponentielle. Lorsque le délai est connu du public, la méthode de Haberland peut être utilisée pour estimer la valeur de réservation de l'opposant supposant que l'adversaire utilise une tactique dépendante du temps.

Pour améliorer la fiabilité des estimations, Yu et al. Combinent la régression non linéaire avec l'apprentissage bayésien pour estimer la valeur de réserve de l'adversaire, ainsi que le délai. Dans leur modèle, l'adversaire est supposé utiliser une tactique dépendante du temps avec des paramètres inconnus.

Chaque tour, les paramètres sont estimés à l'aide d'une régression non linéaire. Ensuite, l'estimation de cette ronde est utilisée pour créer un ensemble plus fiable d'hypothèses sur la valeur de la réservation et le délai de l'opposant en utilisant l'apprentissage bayésien.

- **Estimation de la densité du noyau :**

L'interpolation de la probabilité d'acceptation ne se répercute pas directement sur un paramètre de négociation à problèmes multiples, car l'espace de préférence à problèmes multiples n'a pas la structure du cas à problème unique avec des préférences opposées. L'approche clé pour surmonter ce défi est celle d'Oshrat et al. Et s'appuie sur une base de données de négociations contre un ensemble de négociateurs humains avec des profils de préférences connus. Au cours de la négociation, il est supposé que la position de profil de préférence est connue, les auteurs ont ensuite utilisé l'estimation de la densité du noyau pour estimer la probabilité d'acceptation de toutes les autres offres.

Robert et Nicholas ont montré que les préférences d'une négociation d'adversaire dans les négociations bilatérales multi-issues peuvent être efficacement apprises en utilisant l'estimation de la densité du noyau. Plus précisément, en l'appliquant au modèle de compromis de Faratin, ils ont montré qu'il peut rendre les négociations plus efficaces (en termes d'utilité et de délai d'accord). En choisissant l'estimation de la densité du noyau comme paradigme d'apprentissage, nous n'avons pas eu à faire d'hypothèses explicites sur la relation entre le temps, l'historique des négociations et les préférences de l'adversaire (comme beaucoup d'autres méthodes d'apprentissage).

- **Les réseaux de neurones artificiels :**

Fang et al. Présentent une technique moins connue de négociation sur plusieurs questions, qui suppose que chaque offre présentée est également acceptable pour l'opposant. L'ensemble des offres acceptables des négociations antérieures est utilisé pour former un réseau neurone simple qui peut ensuite tester si une offre particulière est acceptable ou non.

Pour le cas général dans la négociation multi issues, des perceptrons multicouches (MLP) peuvent être utilisés, comme le font Masvoula et al. Les MLP sont des réseaux de neurones artificiels où certains nœuds ont une fonction d'activation non linéaire. Masvoula et al tester deux réseaux dans un cadre expérimental :

Un réseau dans lequel chaque problème est approximé par un MLP distinct et un réseau où un seul MLP est utilisé pour tous les problèmes. La quantité de neurones d'entrée et de neurones de couche cachée est déterminée empiriquement pour les deux réseaux, après quoi il est démontré qu'ils prédisent de manière fiable l'offre suivante de l'adversaire, le réseau MLP unique entraînant l'erreur moyenne la plus faible.

Dans des travaux plus récents, Masvoula étudie les performances de deux réseaux de neurones artificiels qui apprennent la stratégie de l'adversaire sans s'appuyer sur des connaissances historiques. Le premier modèle est un MLP simple qui est recyclé à chaque tour, en utilisant la trace de négociation complète de l'adversaire. Le second est plus avancé (et surpasse le premier), car la structure du réseau neuronal est optimisée à chaque tour, en utilisant un algorithme génétique qui évalue les réseaux de neurones en fonction de leur complexité et de leur erreur de prédiction.

Les méthodes ci-dessus ne limitent pas explicitement la stratégie de l'adversaire. S'il est connu que l'adversaire utilise une tactique dépendante du temps, les travaux de Rau et al ; et Papaioannou et al Peut être utilisé. En raison de l'espace de recherche réduit des tactiques dépendant du temps, Rau et al. Constatent que la tactique de concession et le poids de chaque question offerte par l'opposant peuvent être tirés de ce processus de manière exacte. Rau et al. Constatent que la tactique de concession et le poids de chaque problème offert par l'opposant peuvent être tirés de ce processus de manière exacte. Papaioannou compare les performances de cinq estimateurs pour la stratégie d'enchères de l'adversaire, dont nous avons déjà examiné trois estimateurs basés sur la régression dans la section précédente. Des deux estimateurs restants, un est basé sur un réseau neuronal perceptron multicouche, et l'autre utilise un réseau neuronal à fonction de base radiale Ce dernier estimateur surpasse tous les autres estimateurs en ce qui concerne prédire les offres futures de l'adversaire. Il obtient l'erreur globale la

plus faible et son application se traduit par le plus grand nombre de négociations réussies.

2. La modélisation de l'adversaire :

Les négociations bilatérales peuvent être considérées comme un jeu d'informations incomplètes entre deux personnes. Le modèle relatif n'est qu'une description abstraite du joueur (ou de son comportement) pendant la partie. En négociation, la modélisation de l'adversaire s'articule généralement autour de trois questions :

- Que veut l'adversaire ?
- Que fera l'adversaire ?
- Et de quel type est l'adversaire ?

Ces considérations sont généralement étroitement liées. Par exemple, pour comprendre le comportement de l'adversaire, l'agent doit d'abord savoir ce que veut l'adversaire.

Les modèles adversaires sont différents selon un aspect important « leur temps de création » :

- **Offline opponent model** → sont construits avant la négociation en utilisant des données historiques.
- **Online opponent model** → sont construits au cours de la négociation par interprétation de l'échange des offres.

Les défis majeurs pour les modèles en ligne sont les « deadlines » au temps réel ; ce qui limite le temps disponible pour la maintenance du modèle ; même s'il y a une grande différence entre les modèles adversaires ils ont un ensemble commun pour l'utiliser :

- ✓ Minimisez les coûts de négociation : généralement nous avons besoin du temps et des ressources pour démarrer une négociation, Par conséquent, les accords rapides sont généralement préférés. L'apprentissage peut aider à identifier les transactions prometteuses et ainsi faciliter les transactions rapides.

- ✓ Adapte à l'adversaire : les agents peuvent adapter leur comportement en fonction de leur adversaire.
- ✓ Atteindre des accords Win -Win : Dans un environnement coopératif, l'objectif des agents est d'obtenir des résultats équitables. L'estimation de la préférence de l'adversaire permet de déterminer le résultat de l'avantage mutuel.

2.1. Stratégie d'acceptation :

L'acceptation d'une offre est déterminée par l'acceptation de la stratégie d'un agent négociateur. Après avoir accepté l'offre, la négociation se termine par un accord, sinon l'agent continuera à échanger des offres. Apprendre la stratégie d'accepter des adversaires peut être très utile car il peut aider à trouver le protocole le plus efficace, et l'utilité peut toujours être acceptée par l'adversaire. Dans les négociations d'un seul issu où l'opposant a des préférences opposées des autres connus l'estimation de la valeur de la réservation est suffisante pour déterminer toutes les offres acceptables. La valeur de la réserve peut être apprise en déduisant les concessions de l'opposant, par exemple en appliquant l'apprentissage bayésien. Alternativement, un modèle peut supposer que l'adversaire utilise une fonction de décision particulière dont l'inconnu les variables peut être estimé à l'aide d'une régression non linéaire.

Dans les négociations multi-issues, nous pouvons estimer la probabilité d'acceptation pour chaque résultat possible. En gardant une trace des offres qui ont été proposées et acceptées lors des négociations précédentes, un agent peut estimer la probabilité qu'une offre soit acceptée. Comme il est peu probable qu'une telle estimation puisse être dérivée pour toutes les offres possibles, des méthodes de régression peuvent être appliquées pour déterminer la probabilité d'acceptation pour l'ensemble de l'espace des résultats.

2.2. Deadline (la durée de temps) :

Deadline d'une négociation se réfère à la période pendant laquelle un accord doit être conclu ce qui est meilleur que la meilleure alternative pour chaque partie. Chaque agent peut avoir son propre délai privé, mais les délais de négociation partagés sont également très courants. Le délai peut considérer comme un nombre maximum du tour ou bien un point de coupure en temps réel.

Il est important de connaître les délais privés car les agents sont susceptibles de faire de fortes concessions autour de l'échéance pour éviter de parvenir à un accord. En raison de la relation étroite avec la valeur réservée, la plupart du processus d'apprentissage de la valeur réservée peut également être appliqué ici.

2.3. Profil de préférence :

Le profil de préférence d'un agent représente une évaluation privée des résultats de négociation possible. En plus, aide à trouver des résultats mutuellement bénéfiques et reconnaissent la possibilité de concessions significatives.

Jusqu'à présent, il y a quatre approches ont été utilisées pour estimer les informations de préférence de l'adversaire :

- ✓ L'importance des issues : En générale, il est plus facile d'estimer le poids de toutes les issues au cours de négociation plutôt que la préférence sur tous les résultats. Le truche est d'analyser les concessions de l'opposant ; en supposant que des concessions plus fortes sont accordées sur des issues qui sont valorisées moins.
- ✓ Classifiez la trace de négociation : Compte tenu des actions de négociation de l'adversaire, nous pouvons déterminer le type d'opposant le plus probable, puis appliquer un algorithme de classification pour classer les préférences de l'adversaire.
- ✓ Données de négociation agrégées : Lorsque des données hors ligne sont disponibles, nous pouvons dériver le profil de préférence de l'adversaire à partir d'une grande base de données de traces de négociation d'opposants similaires - mais pas identiques.
- ✓ Importance des résultats : Une technique populaire est l'heuristique d'analyse de fréquence. L'idée principale est que les valeurs préférées d'un problème sont proposées relativement plus souvent dans une trace de négociation.

2.4. Stratégie d'enchère :

La stratégie de négociation détermine l'offre d'un agent dans tout compte tenu de l'état de négociation. Les stratégies peuvent aller de simples concessions en fonction du temps à une prise de décision complexe cela dépend du

comportement de l'adversaire. L'apprentissage de la stratégie d'enchère de l'adversaire permet à un agent d'anticiper de manipuler le comportement de l'adversaire. Alors, la stratégie d'enchère de l'adversaire se résume à créer une prévision des offres futurs à l'aide d'ensemble de techniques statistiques et les méthodes de lissage [21].

3. Les techniques de modélisation pour les modèles d'adversaire :

Un ensemble complet de techniques d'apprentissage a été appliqué à la négociation automatisée. Ci-dessous nous présenterons les méthodes de la base la plus utilisées. Les deux premières parties impliquent l'apprentissage bayésien et la régression non linéaire ; ces derniers ont été principalement utilisés comme technique d'apprentissage en ligne, parce qu'ils n'ont pas besoin d'une étape de formation pour obtenir une estimation raisonnable, et leur estimation peuvent être progressivement améliorées au cours de processus de négociation.

D'un autre côté, les deux autres méthodes sont l'estimation de la densité du noyau et les réseaux de neurones artificiels ; nécessite généralement une phase de formation et sont principalement utilisées l'historique des négociations disponibles. Grace à ces méthodes, moins cher d'utiliser les informations obtenues lors des négociations.

3.1. L'apprentissage bayésien :

L'apprentissage bayésien est la méthode probabiliste la plus importante lors de la modélisation des adversaires. Cette méthode est basée sur la règle suivante :

$$P(H|E) = \frac{P(E|H) * P(H)}{P(E)}$$

La règle de Bayes est un outil pour mettre à jour la probabilité que l'hypothèse H soit basée sur Preuve E. Dans la formule donnée, P (H | E) est la probabilité postérieure de l'hypothèse H contient la preuve donnée E, et P (E | H) est appelée la probabilité conditionnelle de l'événement E se produit lorsque

l'hypothèse H est donnée. $P(H)$ représente la probabilité a priori de l'hypothèse, Avec ou sans preuve, de même, $P(E)$ représente la probabilité antérieure de preuve Observez E .

L'apprentissage bayésien est couramment utilisé pour identifier l'hypothèse la plus probable H_i Supposons que l'ensemble $H = \{H_1, \dots, H_n\}$. Dans la littérature de négociation, Les hypothèses sont des hypothèses, telles que le type d'adversaire. Dans ce cas, Étant donné l'hypothèse de la preuve donnée, l' E observé peut être déterminé en utilisant la formule Règle bayésienne :

$$P(H_i|E) = \frac{P(H_i|E) * P(H_i)}{\sum_{j=1}^n P(E|H_j) * P(H_j)}$$

L'agent peut formuler un ensemble d'hypothèses indépendantes sur la propriété de l'opposant, Utilisez des preuves pour découvrir quelle hypothèse pourrait être correcte. L'idée est à chaque fois En observant de nouvelles preuves E , nous pouvons utiliser la deuxième équation pour mettre à jour et calculer une amélioration estimation de la probabilité postérieure $P(H_i | E)$. Après avoir traité les preuves, l'agent peut déduire l'hypothèse la plus probable.

Un inconvénient d'utiliser l'apprentissage bayésien est sa complexité de calcul. Mettre à jour un Étant donné les preuves E_k , supposons seulement que H_i a une faible complexité de calcul ; Cependant, il peut exister de nombreuses hypothèses H_i et preuves E_k Par exemple Lors de la modélisation des préférences de l'adversaire, nous pouvons personnaliser cet ensemble d'hypothèses, ou générer à partir de structures fonctionnelles qui devraient modéliser les préférences. Même dans Scénarios de négociation avec fonctions utilitaires additives linéaires, la modélisation des préférences nécessite un ensemble de profils de préférence pour chaque question négociable. Cela conduit déjà à un certain nombre d'hypothèses qui est exponentielle dans le nombre de problèmes.

3.2. Régression non linéaire :

La régression non linéaire utilisée pour dériver un ensemble de fonctions « le mieux adaptée » pour observer les données échantillonnées. Il est utilisé lorsque nous voulons que les données affichent certain relation fonctionnelle entre l'entrer et la sortie, à partir de laquelle nous pouvons insérer

de nouveaux points de données. Une application de négociation typique consiste à estimer le comportement futur de l'adversaire sur la base de l'historique des transactions ; nous supposons que la stratégie d'enchère de l'adversaire utilise une formule connue avec des paramètres inconnus.

Un modèle de régression non linéaire est composé de quatre éléments principales : la relation de dépendance (ou repense), variable indépendante (ou variable prédictive), formule (non linéaire) et ensemble. Pour illustrer ce point, supposons que nous avons un ensemble d'observations. Dans le but de trouver la relation entre X et Y afin de prédire la valeur Y' de la nouvelle valeur de X. supposons que la formule de relation soit : $Y'(x)=ax^2+bx+c$ / où a et b sont des paramètres de valeur inconnue. Dans cette formule, Y' est la variable dépendante et X la variable indépendante. En appliquant une régression non linéaire, nous pouvons estimer les paramètres a et b tel que l'erreur entre les valeurs y' prédites et les valeurs y observées soit minimisée, utiliser la fonction de perte pour calculer l'erreur. Dans la littérature de négociation, La formule de calcul est la somme des différences au carré entre la valeur prédite et la valeur observée. La figure 3.1 représente un exemple d'une régression non linéaire.

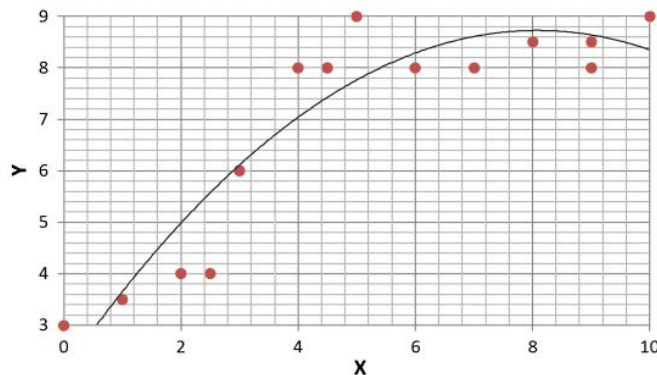


Figure 3.1. Exemple de régression non linéaire. Le meilleur ajustement est montré comme la ligne noire [22].

Les fonctions de perte alternative peuvent par exemple calculer des différences absolues ou traiter des effets positifs et des erreurs négatives différemment. Les paramètres de la formule quadratique discutée dans cet exemple peuvent être utilisés une formule fermée.

La régression non linéaire généralement utilisée lorsqu'il y a par exemple un grand nombre de paramètres qui ont une relation non linéaire avec la solution. Le calcul des paramètres est basé sur l'estimation initiale des paramètres, après lequel un algorithme de mise à niveau d'itération est utilisé pour affiner l'hypothèse jusqu'à ce que l'erreur devienne triviale. Les algorithmes couramment utilisés sont la méthode Marquardt et l'algorithme simplex. Mais le problème de l'utilisation des algorithmes d'escalade est qu'ils peuvent retourner un optimum local au lieu de l'optimum global. De plus, dans des cas extrêmes, l'algorithme peut même ne pas converger du tout. Cela peut être résolu en utilisant plusieurs estimations initiales et en sélectionnant le meilleur ajustement après un nombre spécifié d'itérations.

3.3. Estimation de la densité du noyau :

L'estimation de la densité du noyau (KDE) est une technique mathématique utilisée pour estimer la probabilité distribution d'une population à partir d'un ensemble d'échantillons de population [22]. La base de KDE consiste à convertir chaque échantillon à une fonction dite « noyau », cette dernière est une distribution de probabilité qui quantifie l'incertitude de l'observation. Les choix communs pour les noyaux sont la distribution normale et standard ou la distribution de probabilité uniforme ; la deuxième étape de KDE est d'accumuler tous les noyaux pour l'estimation distribution de probabilité de population. L'un des problèmes les plus importants dans ce travail est *bandwidth* (la bande passante) ou la propagation du noyau. La figure 3.2. Présente, *bandwidth* plus petite entraînera des noyaux plus élevés mais plus petits, tandis que *bandwidth* plus élevée entraînera des noyaux plus grands et plus faibles. Ici, nous utilisons *solve-the-equation* (STE) suggérée par [22] pour calculer cette bande passante.

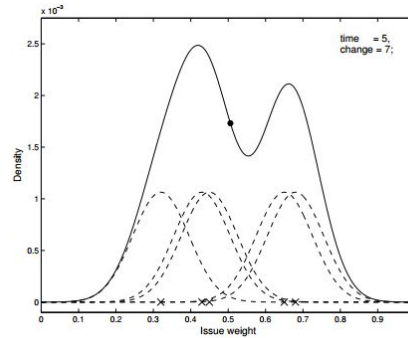


Figure 3.2. Exemple de la densité de noyau [23].

- les noyaux ; — l'estimation ; xxx observation individuelle ;
- le poids prévu pour la question de négociation.

Cette méthode prend une estimation initiale de la bande passante et calcule une nouvelle valeur, en utilisant l'erreur dans l'estimation de densité résultante, jusqu'à ce que le processus converge. Les raisons du choix de cette méthode sont ses bonnes performances sur notre domaine, et la facilité de calcul. La raison du choix de cette méthode est sa bonne performance dans notre domaine et sa facilité de calcul [23].

3.4. Réseaux de neurones artificiels :

Les réseaux de neurones artificiels sont des réseaux d'unités de calcul simples qui peuvent résoudre des problèmes complexes. Ci-dessous, nous fournissons une brève introduction aux réseaux de neurones artificiels (ANN) basée. [22]. Un ANN est un modèle informatique avec la capacité d'apprendre la relation entre la entrée et sortie en minimisant l'erreur entre le signal de sortie de l'ANN et la sortie attendue. Étant donné que tout ce qui est requis est un mécanisme pour calculer l'erreur, ANN peut être appliquée lorsque la relation entre l'entrée et la sortie est inconnue. Les ANN ont été utilisées à plusieurs fins, y compris la classification, la mémorisation et la structuration des données. Un réseau neuronal est constitué d'unités de calcul appelées neurones, qui sont connectées par des bords pondérés. La figure 3.3 représente un réseau neurone simple composé de six neurones. Un seul neurone peut avoir plusieurs fronts entrants et sortants. Lorsqu'un neurone a reçu toutes les entrées, il les combine selon une règle de combinaison, par exemple la somme des entrées. Ensuite, il teste s'il est déclenché par cette entrée en utilisant une fonction d'activation ; par exemple, si un seuil a été dépassé ou non. Si le neurone est déclenché, il propage

le signal combiné sur les lignes de sortie, sinon il envoie un signal prédéfini. L'ensemble des neurones fonctionne dans un environnement qui fournit les signaux d'entrée et traite les signaux de sortie de l'ANN. L'environnement calcule l'erreur utilisée par le réseau de neurones pour mieux connaître la relation entre l'entrée et la sortie en ajustant le poids sur les bords entre les neurones de la sortie, que le réseau neuronal utilise pour mieux connaître la relation entre l'entrée et la sortie en ajustant les poids sur les bords entre les neurones.

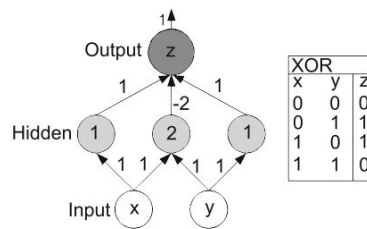


Figure 3.3. Exemple d'un ANN symbolisant le XOR logique. Les neurones d'entrée attendent une valeur de 0 ou 1. Les bords indiquent les poids. Les fonctions d'activation sont des fonctions de seuil, dont les valeurs sont représentées à l'intérieur des nœuds. La règle de combinaison est la somme des entrées. Si l'entrée combinée est supérieure ou égale à la valeur seuil l'entrée combinée est propagée. Sinon, la valeur 0 se propage sur la ligne de sortie [22].

Les neurones peuvent être ordonnés en couches successives en fonction de leur profondeur. Sur la figure 3.3, chaque couche a une couleur unique. La première couche est appelée couche d'entrée, la dernière est la couche de sortie. Les neurones d'entrée et de sortie n'ont généralement pas de fonction d'activation. Les couches intermédiaires sont appelées couches cachées car ils ne sont pas directement connectés à l'environnement. Pour illustrer le fonctionnement d'un réseau neurone simple, supposons que les entrées $x = 0$ et $y = 1$ sont envoyées au réseau de la figure 3.3. Dans ce cas, le neurone gris clair le plus à gauche reçoit une entrée 0, ce qui donne la sortie 0, car le neurone n'est pas déclenché. Le neurone gris clair le plus à droite est cependant déclenché car il reçoit l'entrée 1 et propage donc la sortie 1. Le neurone gris clair moyen reçoit l'entrée 0 et 1, qui est intégrée à l'aide de la règle de combinaison. Le signal combiné est insuffisant pour déclencher le neurone, résultant en un 0 en sortie. Étant donné que le neurone gris clair le plus à droite est le seul neurone qui a produit un non

nul sortie, la sortie finale est 1. La quantité de neurones et leur topologie détermine la complexité de la relation entrée-sortie que l'ANN peut apprendre. Globalement, plus il y a de neurones et de couches, plus la flexibilité l'ANN. Cependant, plus l'ANN est complexe, plus l'algorithme d'apprentissage est complexe et, par conséquent, plus le coût de calcul de l'apprentissage est élevé.

Un ANN est généralement utilisée lorsqu'il existe une grande quantité d'échantillons de données disponibles et lorsqu'il est difficile de saisir la relation entre l'entrée et la sortie dans la description fonctionnelle ; par exemple, lors de négociations contre les humains [22].

Conclusion :

Le domaine de recherche de la modélisation des adversaires en négociation est en constante évolution, porté par plus de vingt ans d'intérêt pour la négociation automatisée et la conception d'agents de négociation en particulier. Dans ce chapitre nous examinons une taxonomie des modèles d'opposants existants en fonction des attributs connus des opposants, des méthodes spécifiques utilisées pour modéliser l'attribut, ainsi que des techniques d'apprentissage générales utilisées tel que l'apprentissage bayésien ; régression non linéaire ; estimation de la densité noyau et les réseaux neurones artificiel. Il existe une relation claire entre l'attribut de chaque adversaire et les techniques d'apprentissage appropriées. Par exemple, l'apprentissage bayésien est la technique la plus populaire et peut être utilisé pour apprendre l'un des quatre attributs opposés que nous les citons dans notre chapitre.

Le chapitre suivant va contenir une conception d'une modélisation d'un agent négociateur basée sur la technique de modélisation émergente « Apprentissage par renforcement profond ».

Contribution

Chapitre 4 : Négociation par renforcement basée agent

Résumé :

Nous présentons une bref description de notre futur agent a modélisé ; nous appuyons sur l'apprentissage par renforcement ; ensuite nous confirmons le choix d'algorithme Q-learning et commencer à la conception de notre agent.

Introduction :

Nous avons présenté dans le chapitre précédent des différentes techniques de modélisation des adversaires au cours d'une négociation multi agents qu'ils sont presque tous des techniques qui se soucient « que veut l'adversaire ? ». Dans ce chapitre, nous allons bâtir un agent négociateur basé sur la technique d'apprentissage par renforcement, cette technique est soucieuse de ce que fera l'adversaire cette dernière est plus difficile que découvrir ce qu'il veut.

Dans ce chapitre ; nous présentons une bref description de notre agent, ensuite nous allons voir l'approche d'apprentissage par renforcement ; finalement, nous proposons une conception d'un agent négociateur, ainsi que leur stratégie d'apprentissage, d'acceptation /refus, d'offre et de concession.

1. Description de notre agent :

Dans ce mémoire, nous nous intéressons particulièrement à la modélisation d'un agent capable de négocier un agent adversaire et de maximiser sa utilité pour gagner le tour de négociation à l'aide d'une stratégie d'apprentissage par l'apprentissage par renforcement qui lui permet de connaître ce que fera l'adversaire et lui proposera des offres plus utilitaires chez nous et satisfaisante chez l'adversaire.

Notre agent doit vérifier les propriétés suivantes :

- ✚ **La réactivité** : l'agent doit être capable à répondre aux événements extérieurs.
- ✚ **L'autonomie** : il doit être capable d'agir sans l'intervention d'un tiers.
- ✚ **La rationalité** : l'agent être capable d'effectuer l'action qui lui permet de maximiser leur performance à la base de ses connaissances précédentes et ses séquences perspectives.
- ✚ **La possession** d'un ou plusieurs objets.
- ✚ **La situation**, l'agent peut percevoir son environnement à travers ses captures et peut agir le résultat de son effectuer.
- ✚ **Le tempérament égoïste** : l'agent privilégiera à tout prix son propre intérêt [10].

2. L'apprentissage profond :

L'apprentissage profond est un sous type de l'apprentissage automatique qui vise à former et à enseigner la machine à exécuter des fonctions humaines telles que la distinction d'objets visuels et l'identification du son et de l'image. Au lieu d'organiser les données, l'apprentissage profond définit ses propres paramètres de base qui permettent à la machine d'apprendre de manière indépendante.

Le *deep learning*, ou apprentissage profond, appartient à la grande famille de l'intelligence artificielle. Plus précisément, il constitue un sous-ensemble de la *machine learning*, et fait appel à des types particuliers de réseaux de neurones artificiels. Il présente donc des caractéristiques similaires à ces techniques, notamment la capacité d'apprentissage de façon autonome. Pour cela on a trois approches d'apprentissage selon le cas d'utilisation :

- Apprentissage supervisé.
- Apprentissage non supervisé.
- **Apprentissage par renforcement.**

3. Apprentissage par renforcement :

L'apprentissage par renforcement est un domaine de l'apprentissage automatique. Renforcement. Il s'agit de prendre des mesures appropriées pour maximiser les bénéfices dans une situation donnée. Il est utilisé par divers logiciels et machines pour trouver le meilleur comportement possible ou le meilleur chemin à suivre dans une situation donnée. L'apprentissage par renforcement diffère de l'apprentissage supervisé en ce sens que, dans l'apprentissage supervisé, les données de formation ont la clé de réponse ; le modèle est donc formé avec la bonne réponse alors que dans l'apprentissage par renforcement, il n'y a pas de réponse, mais l'agent de renforcement décide quoi faire. Pour effectuer la tâche donnée. En l'absence de jeu de données de formation, il est tenu d'apprendre de son expérience [24].

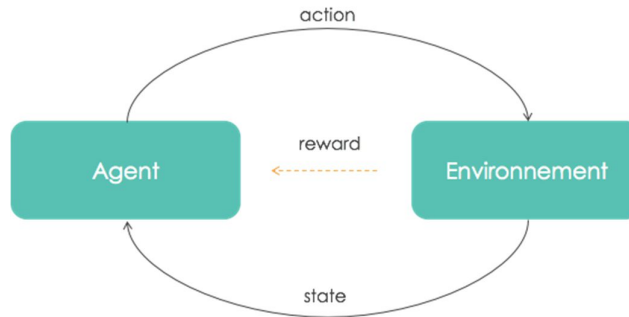


Figure 4.1. Exemple d'un agent utilise l'apprentissage par renforcement [24].

Presque tous les problèmes du RL peut être formalisées à l'aide du processus de décision de Markov (MDP), est un processus sans mémoire avec une séquence d'états aléatoire $S_1 \dots S_t$ qui utilise la propriété de Markov qu'il dit que « **la nouvelle action et l'état de l'agent ne dépend pas du passé actions ou états** », ce dernier se compose d'un tuple de quatre éléments (S, A, P, R) ; où :

- S : l'ensemble d'états finis.
- A : l'ensemble des actions finis.
- P : la probabilité d'une transition d'état $s \in S$ à s' lorsque l'action $a \in A$ est choisie :

$$P = (s'|s, a) = P(s_{t+1} = s|s_t = s; a_t = a).$$

- R : la récompense reçu après la transition de l'état s à l'état s' à cause d'une action a .

La **figure 4.1** se présente ce processus, un agent interagit avec l'environnement et effectuer des actions, à chaque action l'environnement réagit et génère un nouvel état.

4. Le champ des termes du RL dans notre modèle :

- ✚ **Agent ()** : une entité qui peut percevoir → notre agent négociateur.
- ✚ **Environnement ()** : si où notre agent est présent ou tourné → la plateforme de simulation de de négociation GENIUS.
- ✚ **Action ()** : est les actions effectuées par notre agent négociateur au cours de négociation → accepte/ refus, contre-offre, concession.
- ✚ **State ()** : la situation retourné par l'environnement après chaque action de notre agent → gagner le tour / perdu le tour.

- ✚ **Reward ()** : un retour d'information renvoyé à l'agent depuis l'environnement pour évaluer l'action de l'agent → et toujours prend la valeur 0 sauf dans le cas où Reward= utilité du final accord $r = 1/$ avec $r \in [0,1]$.
- ✚ **Policy ()** : la stratégie appliqué pour la prochaine action en fonction de l'état actuel de notre agent.
- ✚ **Valeur ()** : attendu à long terme avec le facteur d'actualisation et opposée à la récompense à courte terme.
- ✚ **Q-valeur ()** : similaire à la valeur, mais elle prend un paramètre supplémentaire comme action courante a .

Les algorithmes d'apprentissage par renforcement sont utilisés principalement dans le domaine des applications d'IA et les applications de jeu. Dans notre mémoire et pour traiter le problème d'elicitations des préférences dans une négociation multi issues non linéaire nous choisissons d'appliquer l'algorithme **Q-learning**.

5. Explication du Q-learning dans notre modèle :

Est un algorithme d'apprentissage par renforcement populaire sans modèle basé sur « **l'équation de Bellman** » ci-dessous :

$$V(s) = \max [R(s, a) + \gamma \sum_{s'} P(s, a, s') V(s')]$$

L'objectif principal de l'algorithme est d'apprendre des stratégies qui peuvent informer les agents dans quelles circonstances ils doivent prendre des mesures pour maximiser les récompenses ; et dans notre cas nous cherchons de maximiser l'utilité de notre agent négociateur.

Il apprend la fonction de valeur $Q(s, a)$; ce qui signifie à quel point il est bon d'agir 'a' à un état 's', dans notre cas veut dire quel offre est le meilleur parmi les offres proposées par l'adversaire.

L'organigramme suivant va exprimer la fonction du Q-learning :

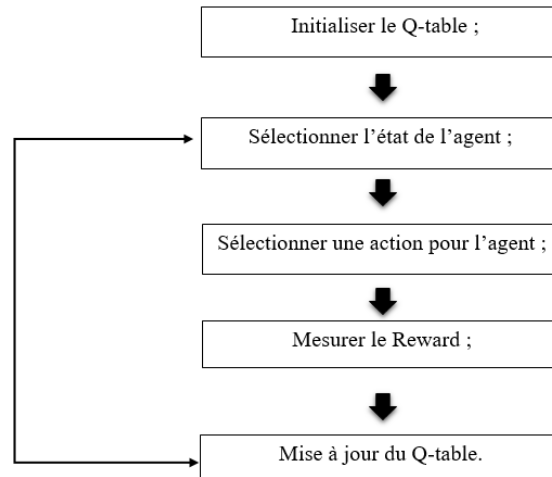


Figure 4.2. Explication de la fonction de Q-learning [25].

Le Q-table :

Une table ou une matrice Q est créée pendant le Q-learning. La table suit les paires d'état et d'action (s, a) et initialise la valeur à zéro. Après chaque opération, la table est mise à jour et les valeurs q sont stockées dans la table.

L'agent RL utilise cette table Q comme table de référence pour sélectionner la meilleure opération en fonction des valeurs q.

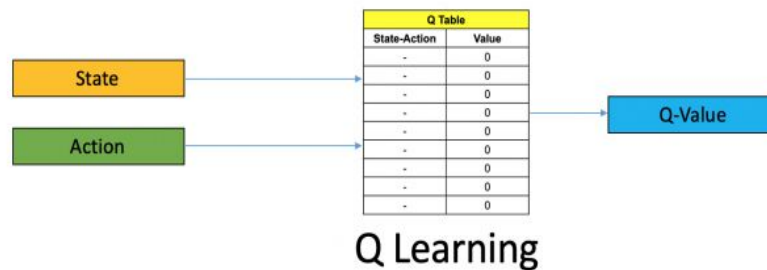


Figure 4.3. Le Q-table [26].

Nous détaillons de plus dans l'algorithme ci-dessous le principe de Q-learning, est de mettre à jour itérativement, à la suite de chaque transition (s_n, a_n, s_{n+1}, r_n) , la fonction de valeur courante Q_n pour le couple (s_n, a_n) , où s_n représente l'état courant, a_n l'action sélectionnée et réalisée, s'_n l'état résultant et r_n la récompense immédiate. Cette mise à jour se fait sur la base de l'observation des transitions instantanées et de leur récompense associée. N_{tot} est t un paramètre initial fixant le nombre d'itérations. Le taux d'apprentissage $\alpha_n(s, a)$ est spécifique à chaque couple (s, a) , et diminue vers

0 à chaque passage. La fonction Simuler renvoie dynamiquement le nouvel état et les récompenses associées en fonction du système [27].

Le Q-learning

*/ α_n est un taux d'apprentissage */

Initialiser (Q_0)

```

pour n ← 0 jusqu'à  $N_{tot}-1$  faire
   $s_n \leftarrow$  choixEtat
   $a_n \leftarrow$  choixAction
  ( $s'_n, a_n$ ) ← simuler ( $s_n, a_n$ )
  {MàJ de  $Q_n$  ;}
  Début
     $Q_{n+1} \leftarrow Q_n$ 
     $\delta_n \leftarrow r_n + \gamma \max_{a'} Q_n(s'_n, a') - Q_n(s_n, a_n)$ 
     $Q_{n+1}(s_n, a_n) \leftarrow Q_n(s_n, a_n) + \alpha_n(s_n, a_n)\delta_n$ 
  Fin

```

Retourner (Q_{tot})

La valeur Q est spécifiée quelle action est la meilleure parmi les autres, d'après cette dernière l'agent prend sa prochaine décision. Afin d'effectuer l'action, l'agent recevra la récompense $R(s, a)$, et il sera également dans un certain état, donc l'équation pour la valeur Q est :

$$Q(s, a) = R(s, a) + \gamma \sum_{s'} (P(s, a, s') \max_{a'} Q(s', a'))$$

Pour garantir que les repenses donnée toujours sont meilleures, ceci induit une activité d'exploitation de la part de l'agent. Par conséquent, il est nécessaire que ce dernier traverse son environnement pour déterminer dans quelles circonstances sera-t-il récompensé. Quelle est la séquence d'actions qu'il prend Obtenez des récompenses au lieu de punitions. Dans tous les processus d'apprentissage, la source de cette exploration est renforcée, c'est-à-dire le dilemme provoqué par le compromis entre exploration et développement.

Dilemme exploration – exploitation :

Pour régler la politique de façon à maximiser sa récompense sur le long terme, la phase d'apprentissage se trouve confrontée à la nécessité de trouver un compromis entre l'exploitation qui consiste à refaire les actions dont on sait déjà qu'elles donnent lieu à une récompense et l'exploration qui consiste à parcourir de nouveaux couples (état, action) à la recherche d'une récompense cumulée plus grande mais au risque d'adopter parfois un comportement sous optimal. En effet, tant que l'agent n'a pas parcouru la totalité de son environnement, il n'est pas certain que la meilleure politique qu'il connaît soit la politique optimale [28].

Les méthodes d'exploration les plus employées sont d'une part les méthodes « Softmax » et d'autre part les méthodes « ϵ -greedy » ou « glouton »

→ Les méthodes ϵ -greedy : elles choisissent l'action optimale avec une probabilité $1-\epsilon$ et tire une commande au hasard avec une probabilité ϵ [28].

6. La conception de notre agent négociateur :

On expérimentera sur la plateforme GENIUS, consacré à simuler les négociations automatique entre des agents intelligents le problème posé dans ce mémoire est d'élaborer un agent capable d'effectuer une elicitation des préférences au cours d'une négociation multi issues non linéaire.

L'espace de résultats dans une négociation est le produit cartésien des issues, représenté par : $\Omega = \{W_1 \dots W_N\}$ où W_i représente les résultats possibles et N est le nombre total de résultats possibles. Chaque agent relie une utilité à un résultat w, en fonction de son profil de préférence, noté par $U_A(w) \in [0, 1]$ pour l'agent A, l'espace de résultat définit un scénario de négociation avec le profil de préférence.

Chaque agent a une valeur réservée qui est l'utilité minimale accepté par l'agent, il n'acceptera jamais qui ne correspond pas à cette valeur, nous la dénotons par β .

Il n'est pas facile d'élaborer un cadre d'apprentissage commun pour le renforcement des négociations multi-issues. Un défi majeur est de permettre à un espace d'état et d'action commun d'être représenté de manière compacte et simultanément contesté arbitrairement par le concepteur d'agent pour s'adapter à différents paramètres de négociation.

Et pour faciliter ce défi de l'état et l'action, nous devons impliquer trois composants principaux dans la stratégie de l'agent négociateur : une stratégie de proposition pour décider quoi offrir, un modèle d'adversaire pour estimer l'utilité par différentes offres, et une stratégie d'acceptation qui aide à prendre une décision d'accepter l'offre de l'adversaire ou proposer la prochaine proposition. L'astuce ici est d'associer la stratégie de négociation en une stratégie de proposition, un modèle d'adversaire et une stratégie d'acceptation.

Afin de fournir une négociation par une interface d'apprentissage par renforcement, il faut que nous décrivions les états et les actions qui favorisent l'interaction avec l'environnement d'une manière indépendante du domaine. Ceci est rendu difficile par deux caractéristiques de Ω :

- 1- Sa taille est exponentielle par le nombre des issues.
- 2- L'espace de résultat doit être unique pour chaque combinaison des issues [29].

Notre stratégie de négociation est constituée d'une stratégie d'apprentissage, une stratégie d'acceptation, une stratégie de proposition et une stratégie de concession sont présentés ci-dessous dans l'organigramme suivant :

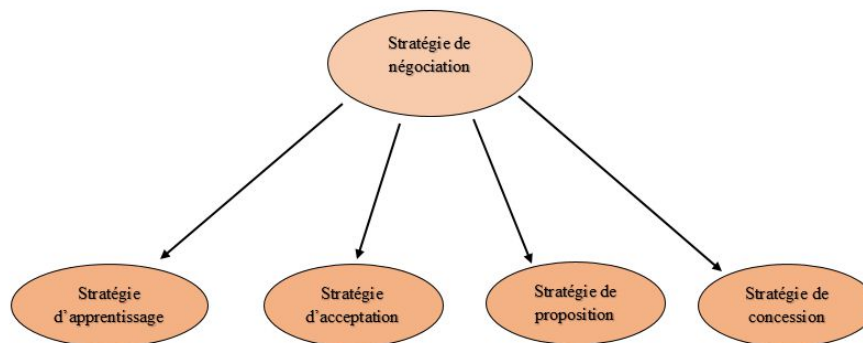


Figure 4.5. Les composants de la stratégie de négociation.

Nous allons détailler chaque une des stratégies précédente dans le paragraphe suivant :

6.1. Stratégie d'apprentissage :

Nous décomposons la stratégie d'apprentissage de notre agent en quatre étapes principales : l'étape de mémorisation, de renforcement, d'évaluation et de décision ; Le fonctionnement de ce modèle organisé comme suit :

- **L'étape de mémoire** : la mémoire permet de stocker la valeur d'utilité courante ; à l'utilisation de Q-table de dimension nombre d'état s et nombre d'action à ces dernier enregistre donc la valeur d'utilité $Q(s, a)$ pour toute les situations S possibles et les actions A .
- **L'étape de renforcement** : la fonction de renforcement de Q-Learning basées sur des algorithmes itératifs à l'objectif d'optimiser la valeur de la fonction d'utilité pour chaque action d'état (s, a) pour en tirer la meilleure stratégie de contrôle. Par conséquent, le rôle de ce module est d'optimiser l'estimation de gain $Q(s, a)$ pour un état " s " et action " a ". Pour ce faire, nous utilisons l'équation de mise à jour de Q-learning suivante :

$$Q_{k+1}(s_i, a_i) = Q_k(s_i, a_i) + \alpha \left(r_i + \gamma \max_{a \in A(s_{i+1})} Q_k(s_{i+1}, a) - Q_k(s_i, a_i) \right) [28]$$

- **L'étape d'évaluation** : Le but de ce module d'évaluation est d'établir une connexion entre la mémoire et les fonctions de prise de décision. Fournissez simplement la fonction de décision, L'estimation actuelle du retour attendu de chaque action dans l'état actuel du système.
- **L'étape de décision** : Nous avons vu que l'algorithme d'apprentissage Q converge vers la direction optimale est assuré sous la contrainte unique de parcourir tous les états et actions indéfiniment possible (Q-table). L'agent doit toujours sélectionner la commande à exécuter. Il est donc nécessaire faites un compromis entre exploitation et exploration. Dans notre cas, nous avons choisi le type fonction de décision ϵ - greedy. Ce principe conserve le principe du choix d'un comportement gourmand dans la plupart des cas.

6.2. La stratégie de proposition :

Pour la stratégie de proposition, notre agent va apprenait que veut l'adversaire et lui proposé une offre, premièrement nous allons déclarer une valeur $d=0.9$ qui représente l'utilité objective, en deuxième place il faut calculer la distance euclidienne entre l'utilité de chaque offre par rapport à la dernière offre d'adversaire ; notre choix est la valeur « min-espace-euc » avec $d=0.9$ c'est le temps où notre agent offre une proposition.

Le pseudo code suivant résume cette stratégie :

Tant que $U_A(W_{B \rightarrow A}^t) < d$ et $t > T$ faire :

/d la valeur d'utilité demandée /

$d := 0.9$;

→ A := effectuer un choix pour proposer la plus proche offre de la dernière offre d'adversaire ;

(Calculer la distance euclidienne)

→ A := choisir le résultat min de l'espace euclidienne ET $d=0.9$;

→ $U_A(W_{A \rightarrow B}^{t+1})$.

6.3. La stratégie d'acceptation :

La stratégie d'acceptation est basée sur la combinaison de condition de temps ($t > T$) et utilité (U_A), elle est représentée comme suit :

Tant que $t > T$ faire :

Si $U_A(W_{B \rightarrow A}^t) > d$ alors

→ A := accepte l'offre de l'opposant ;

Sinon si [$\alpha U_A(W_{B \rightarrow A}^t) > U_A(W_{A \rightarrow B}^{t+1})$]

Si non A := refus l'offre de l'opposant ;

6.4. La stratégie de concession :

Nous proposons une condition pour effectuer une concession selon le temps, ainsi quand il reste 10% de deadline l'agent commence à faire des concessions de la façon suivante : il diminue la valeur de l'utilité objective jusqu'elle égale la valeur de l'utilité réservée

Tant que $d > \beta$ et $t \leq 10\%T$ faire

$d := d - 10\% d$;

Nous résume chaque stratégie parmi les stratégies précédentes par les organigrammes suivantes :

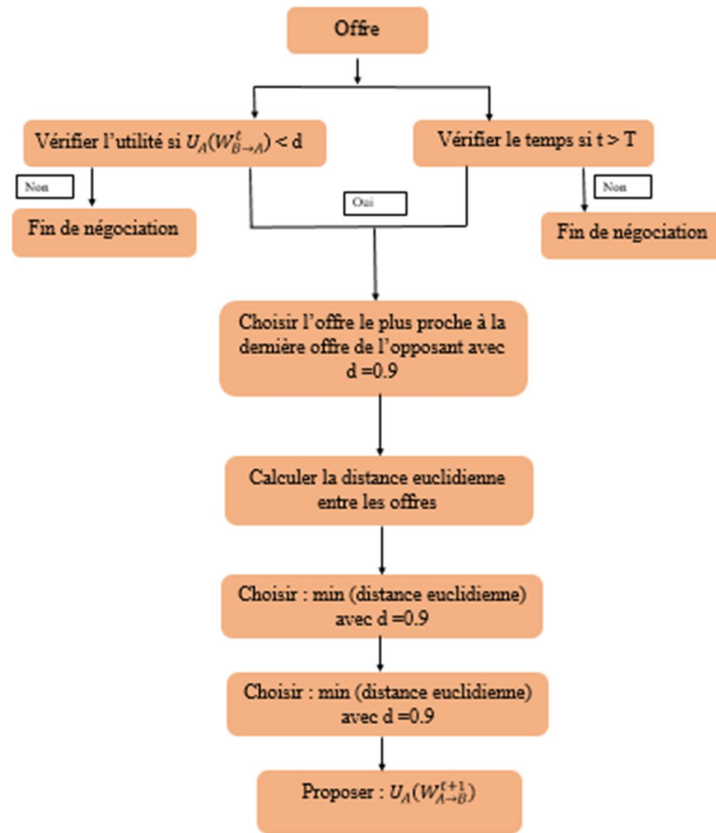


Figure 4.6. La stratégie de proposition.

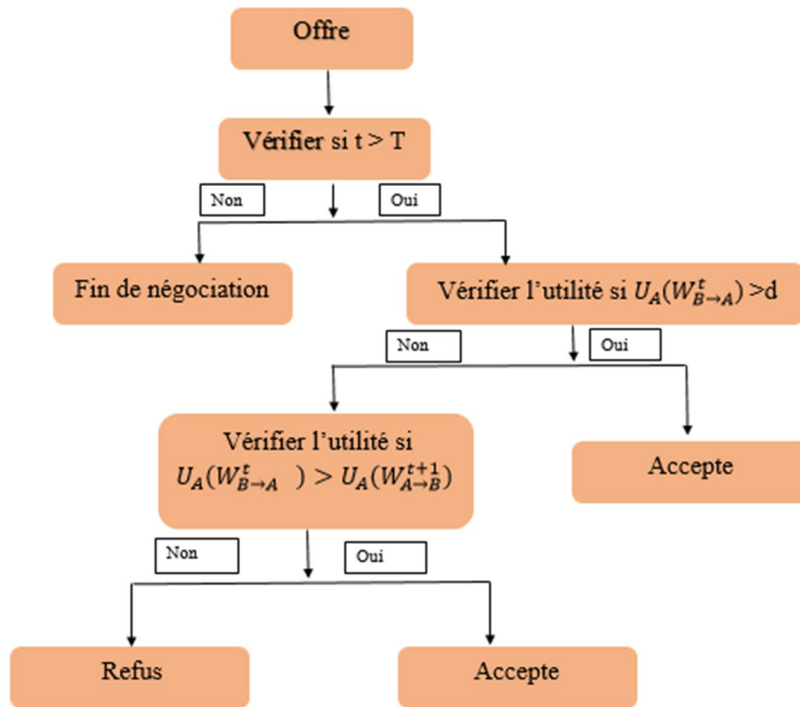


Figure 4.7. La stratégie d'acceptation.

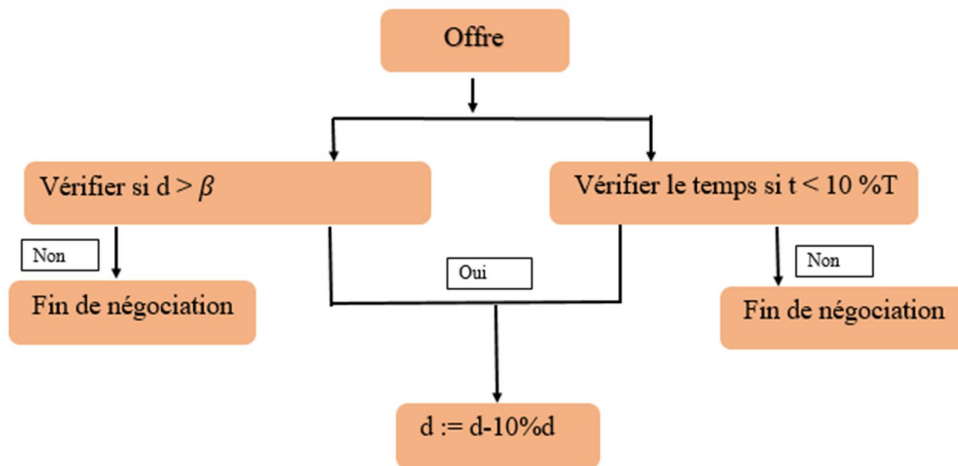


Figure 4.8. La stratégie de concession.

Conclusion :

Dans ce chapitre nous avons établi les principes de notre modèle de conception d'agent négociateur à travers l'algorithme Q-learning, et défini également l'apprentissage par renforcement ainsi que la confirmation de choix d'algorithme Q-learning.

L'objectif de chapitre suivant est de valider notre modèle de conception dans la plateforme de simulation de la négociation automatique GENIUS.

Chapitre 5 : Implémentation et résultats

Résumé :

Nous montrons la partie expérimentale de notre travail et discuter les résultats.

Introduction

Dans le chapitre précédent nous avons présenté le principe de notre modèle de conception d'un agent négociateur. Ce chapitre 5 est consacré à l'étude comportementale de cet agent sur négociation multi-issues non linéaire dans la plateforme GENIUS.

Dans ce chapitre, nous appliquerons l'un des algorithmes d'apprentissages par renforcement pour la négociation multi-issues non linéaire sous la forme de test avec différents agents compatibles sous la plateforme utilisé ; ainsi que nous présentés les logiciels utilisés.

1. La plateforme GENIUS :

(Dans cette mémoire nous utilisons la version 5.1.1)

L'environnement GENIUS (Generic Environment for Negotiation with Intelligent multi-purpose Usage Simulation) est Conçu pour promouvoir des stratégies de négociation. Avec GENIUS, les programmeurs peuvent se concentrer sur la conception de stratégies. À cette fin, GENIUS fournit un environnement flexible et facile à utiliser pour la mise en œuvre d'agents et des mécanismes de support pour la conception et l'analyse de stratégies d'agents. De plus, le cœur de GENIUS peut être intégré dans un système de support de transaction plus large qui peut prendre en charge l'intégralité de la négociation du début à la fin. GENIUS se concentre sur les négociations bilatérales, c'est-à-dire les négociations entre deux parties ou agents A et B. Les agents négocient sur les issues qui entrent dans le champ de la négociation, et chaque issue à un éventail d'alternatives ou de valeurs. Les résultats des négociations comprennent chaque issue est mappée sur une valeur et tous les ensembles Ω possibles est appelé l'espace des résultats. La partie négociatrice connaît le domaine de résultat et reste inchangée pour une seule session de négociation.

Le résultat est appelé l'espace des résultats. La partie négociatrice connaît le domaine de résultat et reste inchangée pour une seule session de négociation. Nous supposons en outre que les deux parties ont certaines préférences spécifiées dans le profil de préférence Ω . Ces préférences peuvent être modélisées à l'aide de la fonction d'utilité normalisée U , qui mappe le résultat possible $\omega \in \Omega$ aux nombres réels dans la

page [0, 1]. Contrairement à l'espace de résultats, le profil de préférence de l'agent est une information privée. [30]

1.1. Présentation des interfaces du GENIUS :

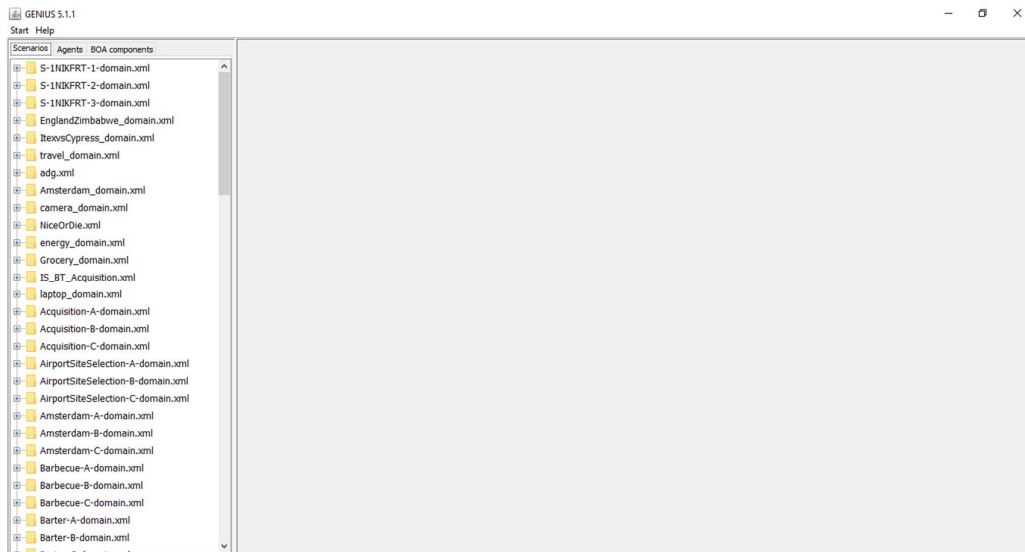


Figure 5.1. GENIUS juste après le démarrage, à la gauche on a le panneau de composants à la droite Le panneau d'état.

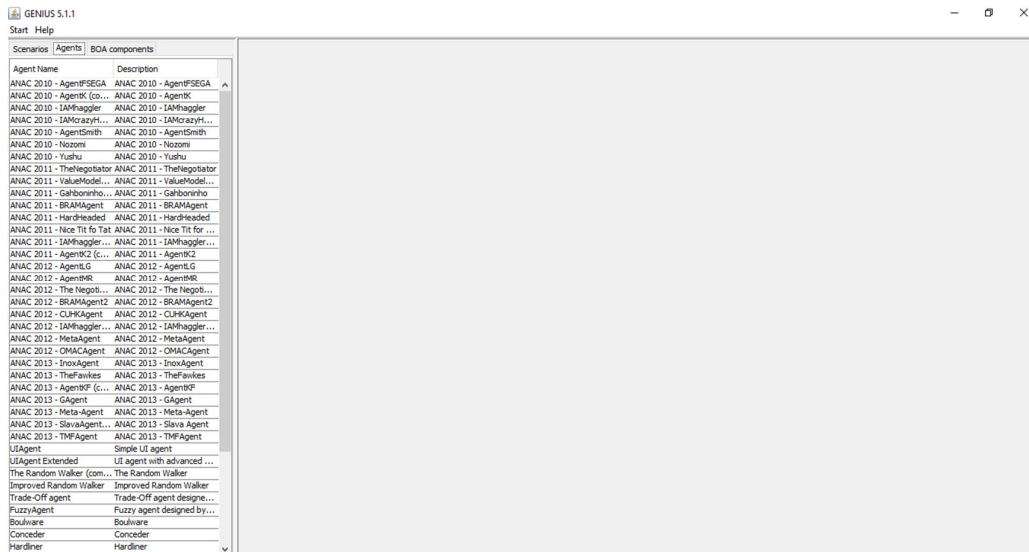


Figure 5.2. Les agents intégrés avec GENIUS.

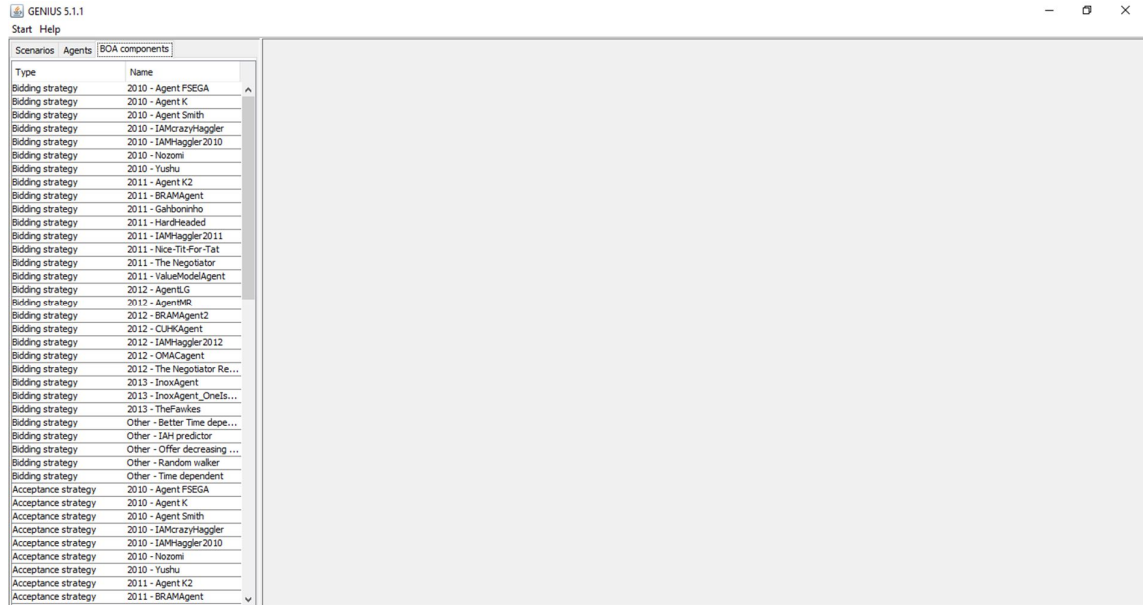


Figure 5.3. composants BOA ; offrant à l'utilisateur la possibilité de créer et d'appliquer des composants nouvellement développés utiliser une interface utilisateur graphique.

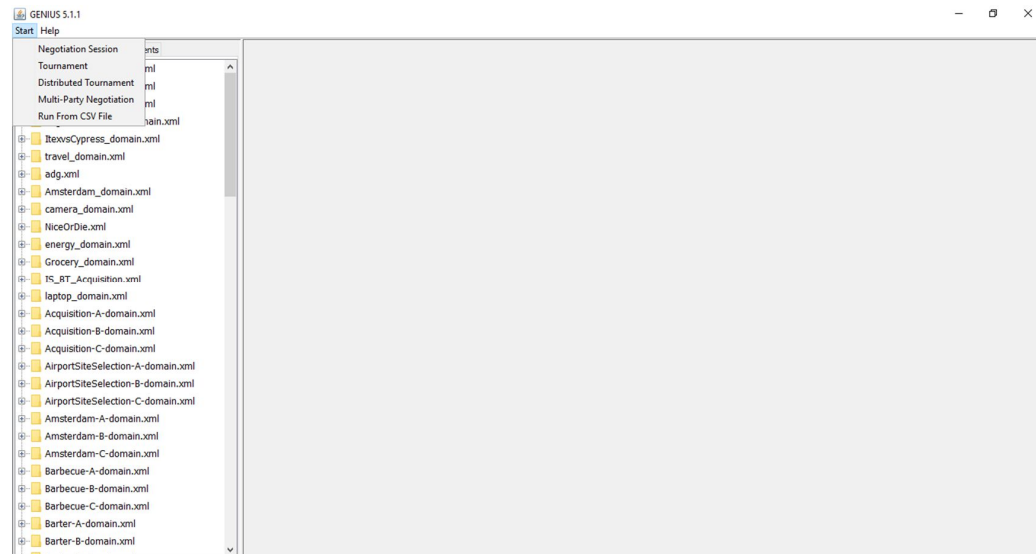


Figure 5.4. Création d'une négociation session simple.

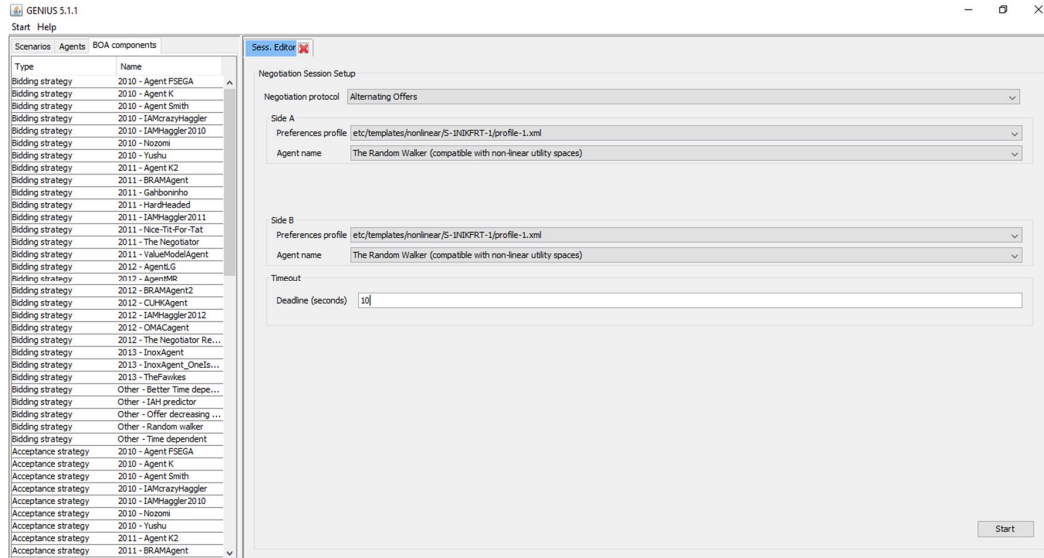


Figure 5.5. Choisir les paramètres d'une négociation, ainsi que le protocole de négociation, deadline et les agents et leurs préférences, enfin, cliqué « start » pour commencer la négociation.

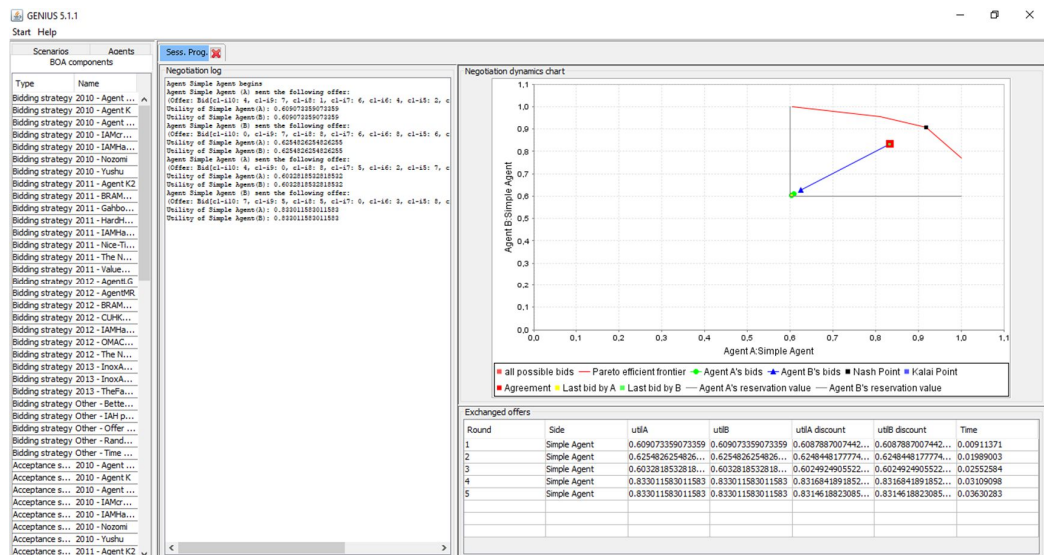


Figure 5.6. Le résultat de la négociation.

2. La compétition des agents de négociation automatisés ANAC :

Le Concours international des agents de négociation automatisés (ANAC) est un événement annuel, organisé conjointement avec la Conférence conjointe internationale

sur les agents autonomes et les systèmes multi-agents (AAMAS), ou la Conférence conjointe internationale sur l'intelligence artificielle (IJCAI). Le concours de l'ANAC rassemble des chercheurs de la communauté de la négociation et fournit des repères uniques pour évaluer des stratégies de négociation pratiques dans des domaines à enjeux multiples. Les concours ont donné naissance à de nouvelles recherches en IA dans le domaine de la conception d'agents autonomes, accessibles à la communauté de recherche au sens large.

L'ANAC met les chercheurs au défi de développer des négociateurs automatisés efficaces pour les scénarios où les informations sur l'adversaire sont incomplètes. Avec ce concours, la communauté de la recherche dans la négociation automatisée oriente la recherche de ses chercheurs, encourage la conception d'agents de négociation génériques capables de fonctionner dans une variété de scénarios et fournit des repères de performance [31].

Le premier concours a été organisé conjointement avec la neuvième conférence internationale sur les agents autonomes et les systèmes multi-agents (AAMAS-10) et comprenait sept équipes. ANAC est déjà lancer 11ème compétition en 29 mai jusqu'à 14 juillet 2020.

3. Eclipse ET java (JDK) :

Eclipse est un environnement de développement (IDE) historiquement destiné au langage Java, même si grâce à un système de plugins il peut également être utilisé avec d'autres langages de programmation, dont le C/C++ et le PHP. Eclipse nous permet de programmer même des agents à partir un lien de connexe entre Genius et Eclipse.

Eclipse nécessite une machine virtuelle Java (JRE) pour fonctionner. Mais pour compiler du code Java, un kit de développement (JDK) est indispensable. [32]

Le kit de développement Java (JDK) est un environnement de développement logiciel utilisé pour développer des applications et des applets Java. Il comprend l'environnement d'exécution Java (JRE), un interpréteur / chargeur (java), un compilateur (javac), un archiveur (jar), un générateur de documentation (javadoc) et d'autres outils nécessaires au développement Java. [33]

4. Discussion des résultants :

4.1. Exemple d'application dans une session de négociation :

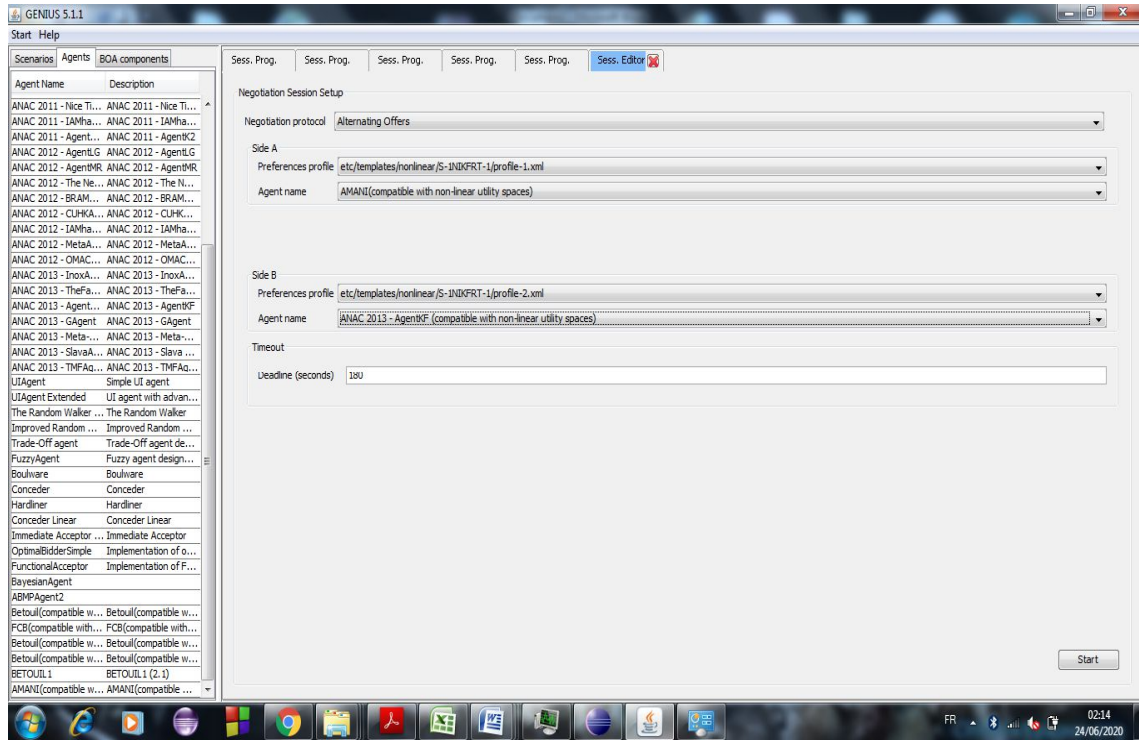


Figure 5.7. La session de négociation est selon le protocole alternative, notre agent est de la coté « A » nommé « AMANI » avec l'agentKF.

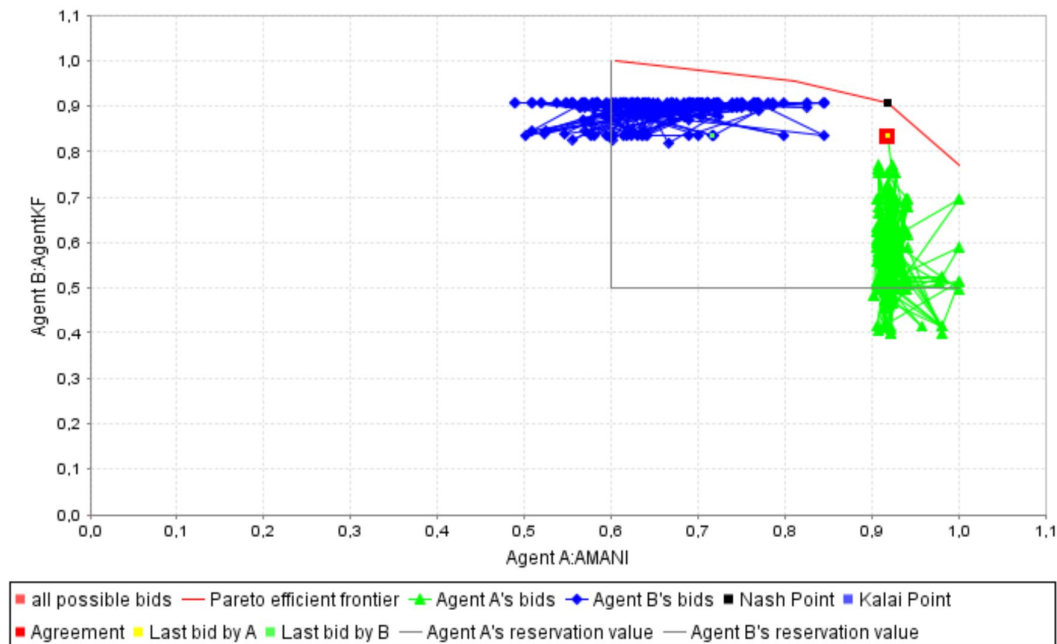


Figure 5.8. Les résultats selon la négociation session précédente. Nous remarquons que notre agent est gagné la session avec une utilité max environ 0.917.

Les offres de notre agent au cours la négociation est comme suit :

Agent AMANI begins

Agent AMANI (A) sent the following offer:

(Offer: Bid [c1-i10: 8, c1-i9: 6, c1-i8: 9, c1-i7: 5, c1-i6: 2, c1-i5: 4, c1-i4: 1, c1-i3: 6, c1-i2: 7, c1-i1: 7,])

Utility of AMANI (A): 0.9111969111969112

Utility of AgentKF (B): 0.5539772727272727

Agent AgentKF (B) sent the following offer:

(Offer: Bid [c1-i10: 3, c1-i9: 2, c1-i8: 3, c1-i7: 5, c1-i6: 7, c1-i5: 8, c1-i4: 2, c1-i3: 5, c1-i2: 8, c1-i1: 7,])

Utility of AMANI (A): 0.583976833976834

Utility of AgentKF (B): 0.9076704545454546

.....

Agent AMANI (A) sent the following offer:

(Offer: Bid [c1-i10: 7, c1-i9: 7, c1-i8: 5, c1-i7: 6, c1-i6: 3, c1-i5: 7, c1-i4: 1, c1-i3: 7, c1-i2: 8, c1-i1: 6,])

Utility of AMANI (A): 0.9391891891891891

Utility of AgentKF (B): 0.5071022727272727

Agent AgentKF (B) sent the following offer:

(Offer: Bid [c1-i10: 5, c1-i9: 8, c1-i8: 7, c1-i7: 3, c1-i6: 7, c1-i5: 8, c1-i4: 6, c1-i3: 6, c1-i2: 8, c1-i1: 6,])

Utility of AMANI (A): 0.7171814671814671

Utility of AgentKF (B): 0.8338068181818182

Agent AMANI (A) sent the following offer:

(Offer: Bid [c1-i10: 6, c1-i9: 5, c1-i8: 9, c1-i7: 6, c1-i6: 6, c1-i5: 6, c1-i4: 1, c1-i3: 6, c1-i2: 8, c1-i1: 7,])

Utility of AMANI (A): 0.917953667953668

Utility of AgentKF (B): 0.8338068181818182

4.2. Résultats obtenus :

Nous avons testé notre agent « AMANI » par tous les scénarios non linéaires disponibles dans GENIUS avec tous les agents compatibles avec les utilités non linéaires sous forme d'un tournoi.

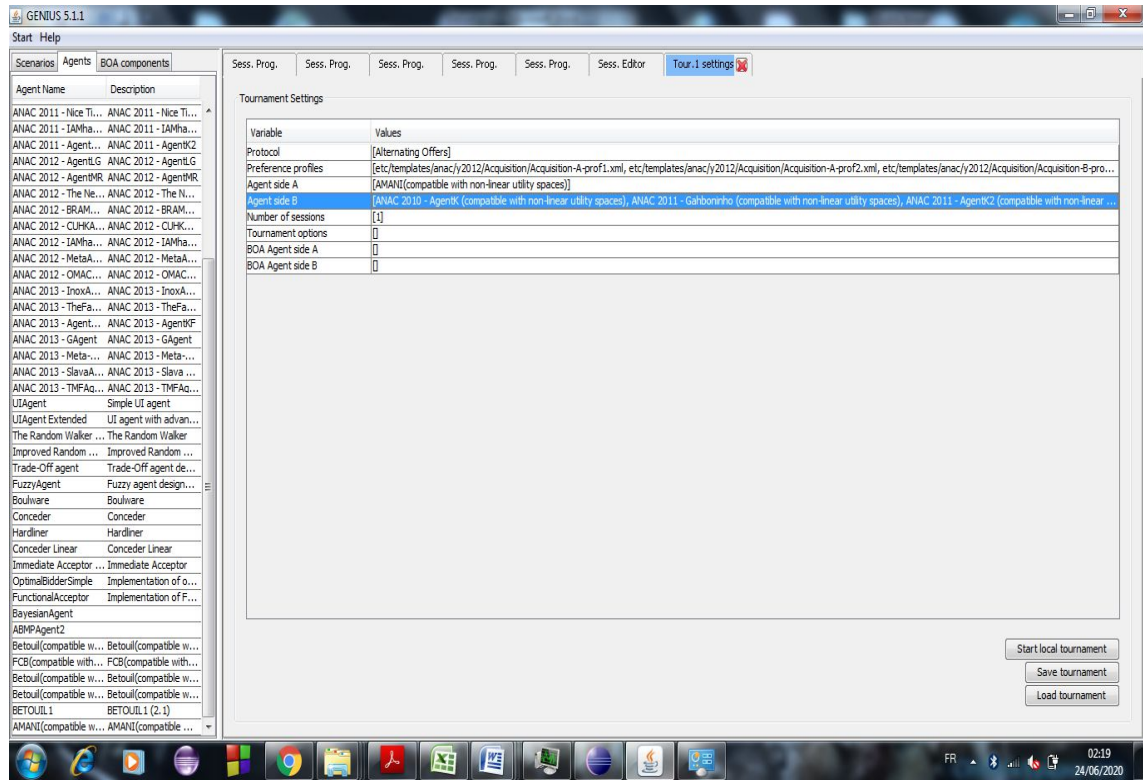


Figure 5.9. Une session tournoi avec tous les agents compatibles avec les utilités non-linéaires sous GENIUS.

Après avoir cliqué sur « start local tournament », Le panneau est ensuite échangé pour une progression du tournoi.

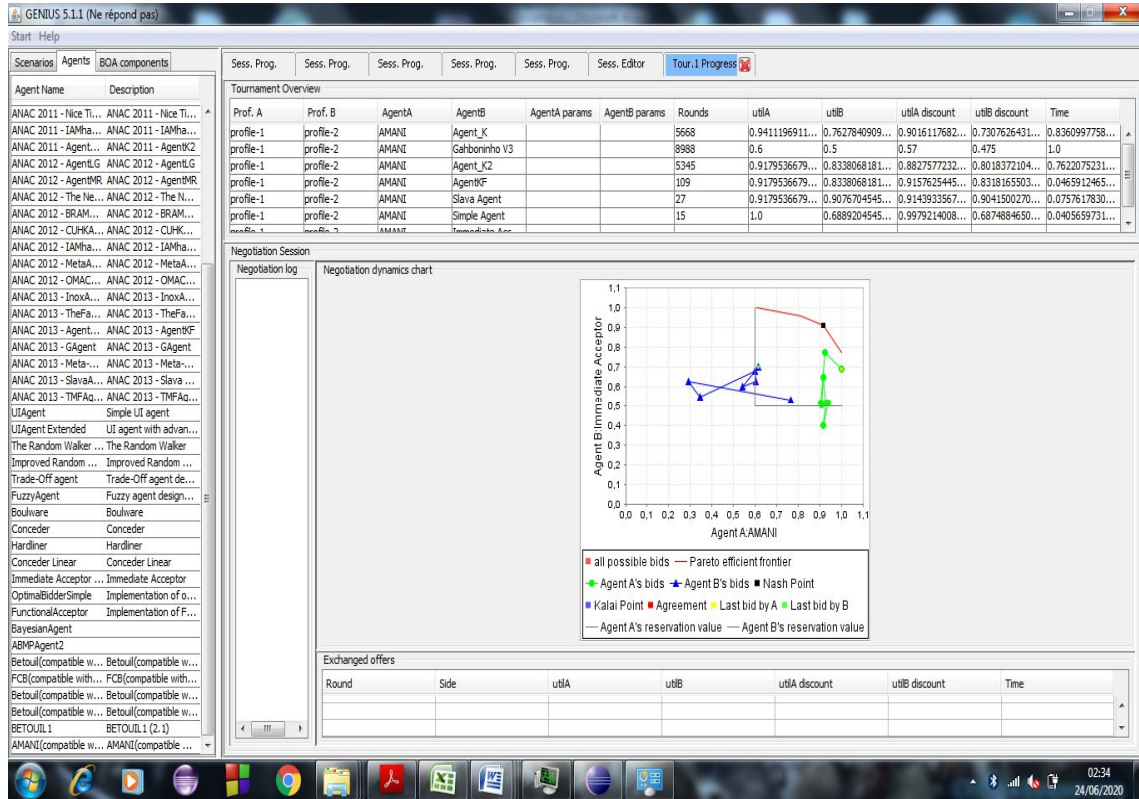


Figure 5.10. Les résultats du tournoi ; La boîte à outils donne de précieux graphiques Informations pendant les sessions de négociation, y compris : solutions optimales Pareto, produit Nash.

Les tableaux suivants montrent tous les résultats de la session :

Prof. A	Prof. B	AgentA	AgentB	AgentA params	AgentB params	Rounds	utilA	utilB	utilA discount	utilB discount	Time
profile-1	profile-2	AMANI	Agent_K			5668	0.941119691119...	0.762784090909...	0.901611768205...	0.730762643108...	0.836099775855...
profile-1	profile-2	AMANI	Gahboninho V3			8988	0.6	0.5	0.57	0.475	1.0
profile-1	profile-2	AMANI	Agent_K2			5345	0.917953667953...	0.833806818181...	0.882757723241...	0.80183721045775	0.762207523122...
profile-1	profile-2	AMANI	AgentKF			109	0.917953667953...	0.833806818181...	0.915762544508...	0.831816550337...	0.046591246561...
profile-1	profile-2	AMANI	Slava Agent			27	0.917953667953...	0.907670454545...	0.914393356712...	0.904150027060...	0.075761783044...
profile-1	profile-2	AMANI	Simple Agent			15	1.0	0.688920454545...	0.997921400883...	0.687488465097...	0.040565973122...
profile-1	profile-2	AMANI	Immediate Acceptor			0	0.6	0.5	0.57	0.475	1.0
profile-2	profile-1	AMANI	Agent_K			4238	0.955965909090...	0.810810810810...	0.916144582021...	0.777036005473...	0.829505301288...
profile-2	profile-1	AMANI	Gahboninho V3			10991	0.5	0.6	0.475	0.57	1.0

Figure 5.11. Tableau de résultats.

Prof. A	Prof. B	AgentA	AgentB	AgentA params	AgentB params	Rounds	utilA	utilB	utilA discount	utilB discount	Time
profile-1	profile-2	AMANI	Agent_K2			5345	0.917953667953...	0.833806818181...	0.882757723241...	0.80183721045775	0.762207523122...
profile-1	profile-2	AMANI	AgentKF			109	0.917953667953...	0.833806818181...	0.915762544508...	0.831816550337...	0.046591246561...
profile-1	profile-2	AMANI	Slava Agent			27	0.917953667953...	0.907670454545...	0.914393356712...	0.904150027060...	0.075761783044...
profile-1	profile-2	AMANI	Simple Agent			15	1.0	0.688920454545...	0.997921400883...	0.687488465097...	0.040565973122...
profile-1	profile-2	AMANI	Immediate Acceptor			0	0.6	0.5	0.57	0.475	1.0
profile-2	profile-1	AMANI	Agent_K			4238	0.955965909090...	0.810810810810...	0.916144582021...	0.777036005473...	0.829505301288...
profile-2	profile-1	AMANI	Gaboninho V3			10991	0.5	0.6	0.475	0.57	1.0
profile-2	profile-1	AMANI	Agent_K2			414	0.907670454545...	0.845559845559...	0.906014357569...	0.844017072963...	0.035603563766...
profile-2	profile-1	AMANI	AgentKF			11	0.907670454545...	0.845559845559...	0.907493068155...	0.845394597470...	0.003810429616...

Figure 5.12. Suite tableau de résultats.

Nous observons que noter agent termine le tournoi avec la plus grande utilité, finalement nous pouvons dire que notre agent « AMANI » capable d'apprenait leur opposant et maximiser leur valeur d'utilité.

Conclusion :

Dans ce chapitre, nous avons testé notre agent «AMANI» dans la plateforme GENIUS avec différents agents qui sont compatibles avec les utilités non linéaire pour les paramètres d'apprentissage dans une négociation multi-issues non linéaire. Enfin, ces travaux nous ont aidés à finaliser la conception d'un agent négociateur par Q-learning.

Conclusion générale et perspectives :

Le point que nous avons adoptées au cours de ce mémoire pour la conception de notre agent négociateur est celui qui agit de manière intelligente et apprend les veus de leur adversaire par l'algorithme Q-learning au but de maximiser leur utilité toujours et gagner le tour de négociation, nous testons notre agent « AMANI » sur la plateforme GENIUS est nous obtenons des résultats satisfaisantes chez notre agent et l'adversaire.

Finalemnt ce travail, nous motive a pensé pour tester notre agent dans différents plateformes de simulations des négociations automatiques selon différents stratégie d'apprentissage et des buts maximisés, comme la bibliothèque Negmas dans Python nous souhaitons de tester notre agent dans cette dernière.

Bibliographie :

[1]T. MARIR, «les système multi agents,» oum Ibouaghi, unversité larbi Ben Mhidi, 2016/2017.

[2]J.Ferber, «les systèmes multi agents : vers une intelligence collectif,» inter edition, 1995.

[3]HANACHI, «Introduction au système multi agents,» Université Toulouse & IRIT.

[4]I.M.Tani, «mémoire online stratégie de rendez-vous dans les système multi agent,» 2017. [En ligne]. Available: https://www.memoireonline.com/08/08/1510/m_strategie-de-rendez-vous-dans-les-systemes-multi-agents1.html. [Accès le 07 02 2020].

[5]M.Beggas, «mémoire online Modéisation par un système multi agents d'un hypermédia éducatif adaptatif dynamique.,» 2005. [En ligne]. Available: https://www.memoireonline.com/07/08/1413/m_modelisation-systeme-multi-agents-hypermedia-educatif20.htm. [Accès le 20 02 2020].

[6]W.Bouazza« mémoire online, Pilotage de cycle de vie de produit à base de produits intelligents : application à l'industrie pharmaceutique, chapitre 2 : Spécification et modélisation d'une solution à base SMA » [En ligne]. Available: <https://www.memoireonline.com/12/14/8930/Pilotage-de-cycle-de-vie-de-produit-a-base-de-produits-intelligents--application-a-l55.png>.

[7] W.Bouazza« mémoire online, Pilotage de cycle de vie de produit à base de produits intelligents : application à l'industrie pharmaceutique, chapitre 2 : Spécification et modélisation d'une solution à base SMA » [En ligne]. Available: <https://www.memoireonline.com/12/14/8930/Pilotage-de-cycle-de-vie-de-produit-a-base-de-produits-intelligents--application-a-l55.png>.

[8]M. S. eddine, «vers une architecture générique à base d'agent pour la création d'un virtuel,» 2009.

-
- [9]I. h. bouknafet samir, «une approche multi agent d'identification précoce des interactions entre les aspects,» skikda, juin 2018.
- [10]A. Florea, «agents intelligent cour web interactif, chapitre 2: architecture des agents et langage,» Politechnia University of Bucharest, 2002.
- [11]N.Bakhta, «Modèle multi agents pour la conception de système d'aide à la décision collective, Thèse de doctorat en informatique,» université d'oran , oran, 2013/2014.
- [12]A. Bakhta, «utilisation des systèmes multi agent pour la prise de décision dans les chaines logistiques.,» Université Abdel Hamid Ibn Badis, Mostaghanem , 2010/2011.
- [13] S, Mourad et C, Brahim. « *Stratégies de négociation entre agents dans le domaine du transport* ». Département d'Informatique, Faculté des Sciences et de Génie, Université Laval, Ste-Foy, PQ, Canada, G1K 7P4.
- [14]J. Jamont, «Démarche modèle et outils pour l'ingenierie des collectifs cyber-physiques, Memoire,» Université Grenoble alpes, 2016.
- [15]B. samir, «une approche multi agent d'identification précoce des interactions entre les aspects,» skikda, juin 2009.
- [16]M. H. Verrons, «GENAC: un modèle générale de négociation de contrats entre agent,» 12 jan 2005.
- [17]S. Samia, «une approche formelle pour la négociation automatique entre (application e-commerce),» Biskra , 2020.
- [18] «Critères d'évaluation , Technologies de l'information et de la communication et appropriation des savoirs ; Agent intelligents cours Web interactif » [Enligne].Available :http://turing.cs.pub.ro/auf2/html/chapters/chapter5/chapter_5_2_2.html. [Accès le 28 02 2020].
- [19]M. Morgue, «Contributions à la négociation multi agents,» université de Lille, 2020.
- [20]N. A. Diego, «Mécanisme de négociation multilatérale pour la prise de décision collectifs,» université de Lion, 2018.
- [21]M. W. N. R. .. Shahin Fatima, «An agenda based frame work for multi issue negotiation,» *Elsevier*, p. 45, 2003.
-

[22]T. Baarslag et al, «A Survey of Opponent Modeling Techniques in Automated negotiation,»

[23]T. Baarslag et al, «Learning about the opponent in automated bilateral negotiation: a comprehensive survey of pponent modeling techniques,» *Springer*, p. 50.

[24]A. Mansour « Apprentissage profond(Deep Learning) pour la classification des lames anapath numérisées,Mémoire pour l'obtention de master 2 » tebessa , Université Echikh larbi tebessi ,2019.

[25]«javaTpoint tutoriel apprantissage par renforcement,» [En ligne]. Available: <https://www.javatpoint.com/reinforcement-learning>. [Accès le 21 06 2020].

[26]«MLQ, deep reinforcement learning guide to deep Q learning,» [En ligne]. Available: <https://www.mlq.ai/deep-reinforcement-learning-q-learning/>. [Accès le 21 06 2020].

[27]k. B. e. v. d. smagt, «An introduction to neural networks. Amsterdam,» University of amesterdam, Amesterdam, 1993.

[28]B. KARIMA, «CONCEPTION D'AGENT INTELLGENT AVEC APPRENTISSAGE PAR RENFORCEMENT,» Guelma , 2008.

[29]j.Bakker et al, «RLBOA: A Mo dular Reinforcement Learning Framework for Autonomous Negotiating Agents,» montérial ,canada, 2019.

[30]S. K. e. a. RAZ LIN, «G E N I U S: AN INTEGRATED ENVIRONMENT FOR SUPPORTING THE DESIGN OF GENERIC AUTOMATED NEGOTIATORS,» *computinal intelligent*, p. 23, 2012.

[31]«tUDelfet, ANAC International negotiation competition » [En ligne]. Available: <http://ii.tudelft.nl/genius/?q=article/releases>. [Accès le 24 06 2020].

[32]« Introduction à Eclipse, cour licence informatique » Université de Lille 1 -2013-2014.

[32]o. s. e. frédèrik, «chapitre3 :apprentissage par renforcement».

[33]«techopedia, java developement kit (JDK)» [En ligne]. Available: <https://www.techopedia.com/definition/5594/java-development-kit-jdk>. [Accès le 26 06 2020].
