

République Algérienne Démocratique Populaire

Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique



Université d'Ibn Khaldoun – Tiaret

Faculté des Mathématiques et de l'Informatique



Département Informatique

Mémoire

Présenté pour obtenir le diplôme de

Master en Informatique

Spécialité : génie informatique

Option : système d'information

Thème

Un système multi agent pour l'émergence du
langage

Rédigé par :

- BELHACHEMI Fatma
- SASSI Hakima

Dirigé par :

- MOKHTARI Ahmed

Année universitaire 2015-2016

Dédicace

Nous dédions ce modeste travail

à nos chers parents

pour leur soutien, leur patience, leur sacrifice

et leur amour, vous méritez tout éloge,

vous qui avez fait de nous ce que nous sommes maintenant.

Nous espérons être l'image que vous avez fait de

nous, que dieu vous garde et vous bénisse.

Nous dédions aussi ce travail à nos sœurs

pour leur affection et leur encouragement qui ont toujours

été pour nous des plus précieux.

Que ce travail soit pour vous le gage de notre profond amour

A tous nos amis

A tous ceux qui nous ont aidés.

A tous ceux que nous aimons, nous dédions ce travail ...

Remerciement

En tout premier lieu, je remercie «الله» le tout puissant à la sagesse, la volonté et au savoir infinis.

Nous tenons à remercier notre encadreur Mr MOKHTARI Ahmed, de nous avoir proposé le sujet du travail, pour ses conseils précieux, sa disponibilité et son soutien tout au long de notre travail.

Nous tenons à remercier également Dr CHADLI Abd Al Hafidh pour tout le temps qui nous a consacré de m'avoir honoré en acceptant de présider mon jury, et le membre de jury Mr ALEM Abd El Kader merci pour accepter d'évaluation de ce travail.

Enfin J'adresse mes remerciements aussi nos amis et nos collègues de la promotion du Master.

Résumé

L'objectif de ce travail est de émergé un langage qui permet la communication entre les agents, en l'occurrence l'émergence d'un lexique au sein d'une population d'agents. Ces agents sont soumis à des contraintes de réalisme fortes. Ils ne peuvent communiquer entre eux que par mots, et à leur création, ils ne connaissent ni mot ni catégorie.

Notre étude permet de représenter de manière très simplifiée les interactions langagières entre agents et les processus d'acquisition d'un langage par ces agents.

Nous Montrons comment ce lexique peut être transmis de génération en génération, comment il se modifie.

Nous utilisons la plateforme JADE pour la réalisation de notre système.

Mot Clé : Communication entre les agents, L' émergence du langage, Agent, Interactions langagières.

Sommaire

Introduction Générale	1
-----------------------------	---

<u>Chapitre I: Les systèmes multi agents</u>
--

1. Introduction	3
2. Agent	3
a. Définition.....	3
3. Architecture d'un agent :	4
i. Agents réactifs :	4
ii. Agents délibératifs :	4
iii. Agents hybrides :	4
b. Systèmes multi-agents	5
1. Définition :	5
2. Pourquoi les SMA ?	5
i. Interaction entre agents :	6
ii. Coopération entre agents	6
iii. Coordination entre agents	6
iv. Négociation entre agents.....	7
Conclusion.....	7

<u>Chapitre II: Émergence du langage</u>
--

1. Introduction	10
2. Définition du langage	10
3. Définition de la communication.....	10
4. Caractéristiques du langage	10
5. Origine du mot HOMO	13
6. L'acquisition du langage	15
7. Trois modèles informatiques pour l'émergence du langage	17
Conclusion.....	25

<u>Chapitre III: La modélisation</u>
--

1. Introduction	31
2. Présentation et justification du choix d'UML.....	31
a) Définition.....	31

b)	Que ce qu'un diagramme UML ?	31
3.	Diagramme de cas d'utilisation	32
4.	Diagrammes de séquences.....	32
a)	Nouvelle simulation	33
b)	Afficher les statistiques	34
c)	La matrice de codage.....	35
5.	Diagramme d'activité.....	36
a)	Nouvelle simulation	36
b)	Afficher les statistiques	37
c)	Afficher la matrice	38
6.	Diagramme de classe.....	39

<u>Chapitre IV: Implémentation.....</u>

Introduction	42
1. Outils exploités	42
2. Définition de la plate forme JADE.....	42
2.1. FIPA (Fondation for Intellegent Physical Agents).....	42
2.2. L'interface graphique	43
2.3. Pourquoi JADE	45
2.4. Un agent selon JADE	46
2.5. Langage de communication de la plate-forme JADE.....	46
3. Implémentation et simulation	47
4. Résultats Expérimentaux	53
Conclusion générale.....	54
Les références Bibliographies.....	

Liste des figures

Figure 1 : Évolution de la lignée humaine par rapport à celle des chimpanzés.....	13
Figure 2 : Origine du mot HOMO.....	15
Figure 3 : L'expérience de Werner et Dyer.....	19
Figure 4 : le dispositif des têtes parlantes.....	21
Figure 5 :Stabilisation d'un universel implicationnel.....	25
Figure 6: Diagramme de cas d'utilisation.....	32
Figure 7: Diagramme de séquence- nouvelle simulation.....	33
Figure 8 : Diagramme de séquence- afficher les statistiques.....	34
Figure 9: Diagramme de séquence- matrice de codage.....	35
Figure 10: Diagramme d'activité-nouvelle simulation.....	36
Figure 11: Diagramme d'activité - afficher les statistiques.....	37
Figure 12: Diagramme d'activité-Matrice de codage.....	38
Figure 13: Diagramme de classe.....	39
Figure 14 : L'interface utilisateur (GUI).....	44
Figure 15 : Echange de message entre les agents.....	45
Figure 16: Fenêtre principale.....	48
Figure 17: Nouvelle simulation.....	48
Figure 18:La trace de la simulation.....	49
Figure 19: Les messages échangés entre les agents de système.....	49
Figure 20: Matrice de codage.....	50
Figure 21: Présentation des statistiques.....	51
Figure 22: La courbe.....	52
Figure 23 : graphe- nombre de mot en fonction du seuil.....	54

Liste des Tableaux

Tableau 1: Structure d'un message ACL.....	47
--	----

Introduction Générale

Les langages possèdent un certain nombre de propriétés qui sont spécifiques au langage humain : elles le distinguent à la fois des autres systèmes de communication animale, et des autres modes d'expression et de communication existant dans les sociétés humaines. Ces propriétés relèvent de tous les niveaux linguistiques : phonologie, morphologie, syntaxe, lexique, etc.

Comment ont-elles pu émerger, et pourquoi ? Comment ont-elles pu évoluer ? Quelle est la nature des contraintes générales dans lesquelles s'est produite l'émergence du langage et son évolution ? Pour aborder ce problème, un certain nombre de chercheurs ont réalisé des modélisations de l'émergence du langage, la modélisation informatique permet de réaliser des systèmes simples d'interactions entre agents, dans lesquels on introduit un certain nombre de contraintes et l'on observe les propriétés du système de communication qui se stabilisent au cours des échanges. Ces modèles permettent de représenter de manière très simplifiée les interactions langagières entre agents et les processus d'acquisition d'un langage par ces agents. L'objectif essentiel de ces travaux n'est pas de modéliser de façon « réaliste » ces phénomènes hautement complexes, mais de rechercher les conditions minimales dans lesquelles peuvent émerger les propriétés linguistiques auxquelles on s'intéresse.

Les systèmes multi-agent constituent une nouvelle technique de modélisation qui place l'objet d'étude au centre de sa démarche. Ces modèles représentent les actions individuelles, les interactions entre les acteurs et les conséquences de ces interactions sur la dynamique du système.[1] et dans ce contexte en présente notre travail qui contient quatre chapitres : commençant par le premier chapitre « les Systèmes Multi Agent » qui définit le concept d'agent et leur environnement d'interaction produisant un SMA , passant au deuxième chapitre « Émergence du langage » nous présentons nos objectifs: faire émerger, au sein d'une population d'agents, un lexique leur permettant de désigner des objets du monde réel et faire que ce lexique, à la manière des langues naturelles, s'adapte culturellement pour gagner en expressivité et en concision.

Le troisième chapitre est consacré à la modélisation UML de notre travail.

Le quatrième chapitre présente l'implémentation de l'application et une brève explication de la plateforme JADE. Nous finirons notre mémoire par une conclusion général qui résume les points essentiels de ce travail.

Chapitre I:

Les systèmes multi agent

1. Introduction

Les Systèmes Multi-agents (SMA) rassemblent les travaux qui portent sur l'étude et la conception d'organisations d'agents autonomes, capables d'agir sur leur environnement physique et/ou social, et de communiquer ou d'interagir pour accomplir collectivement leurs tâches.

Ce chapitre introduit, tout d'abord, les notions d'agents et des SMA, et détaille par la suite les différentes questions que soulèvent la problématique des SMA, en particulier: les interactions, la coopération, la coordination et l'organisation [1-3].

2. Agent

a. Définition

Dans la littérature, on trouve une multitude de définitions d'agents. Elles se ressemblent toutes, mais diffèrent selon le type d'application pour laquelle est conçu l'agent. Nous avons choisis celle de Jennings, Sycara et Wooldridge [1]:

« Un agent est un système informatique, **situé** dans un environnement, et qui agit d'une façon **autonome** et **flexible** pour atteindre les objectifs pour lesquels il a été conçu ».

Les notions situées, autonome et flexible sont définies comme suit:

- **Situé** : l'agent est capable d'agir sur son environnement à partir des entrées sensorielles qu'il reçoit de ce même environnement.
- **Autonome** : l'agent est capable d'agir sans l'intervention d'un tiers (humain ou agent) et contrôle ses propres actions ainsi que son état interne.
- **Flexible** : l'agent dans ce cas est capable de répondre à temps: il doit être capable de percevoir son environnement et élaborer une réponse dans les temps requis.

3. Architecture d'un agent :

Un agent peut toujours être vu comme une fonction liant ses perceptions à ses actions. Ce qui différencie les différentes architectures d'agents, c'est la manière dont les perceptions sont liées aux actions. Les deux grandes familles d'agents sont : les agents **réactifs** et les agents **délibératifs**[3]

i. **Agents réactifs** : Un agent réactif ne fait que réagir aux changements qui surviennent dans l'environnement. Autrement dit, un tel agent ne fait ni délibération ni planification, il se contente simplement d'acquiescer des perceptions et de réagir à celles-ci en appliquant certaines règles prédéfinies tant donné qu'il n'y a pratiquement pas de raisonnement, ces agents peuvent agir et réagir très rapidement. Cette catégorie regroupe les deux types d'architecture suivants :

- Les agents à réflexes simples.
- Les agents conservant une trace du monde.

ii. **Agents délibératifs** : Un agent délibératif est un agent qui effectue une certaine délibération pour choisir ses actions. Une telle délibération peut se faire en se basant sur les buts de l'agent ou sur une certaine fonction d'utilité. Elle peut prendre la forme d'un plan qui reflète la suite d'actions que l'agent doit effectuer en vue de réaliser son but. Ainsi, les trois types d'architecture suivants peuvent être regroupés sous cette catégorie :

- Les agents ayant des buts.
- Les agents utilisant une fonction d'utilité.
- Les agents **BDI**.

iii. **Agents hybrides** : Chacune des architectures précédentes est appropriée à un certain type de problème. Cependant, pour la majorité des problèmes, ni une architecture complètement réactive, ni une architecture complètement délibérative n'est appropriée. Dans ce cas, une architecture conciliant à la fois des aspects réactifs et délibératifs est requise. On parle alors d'architecture hybride.

b. Systèmes multi-agents

1. Définition :

Un système multi-agents est un système distribué composé d'un ensemble d'agents interagissant, le plus souvent, selon des modes de coopération, de concurrence ou de coexistence situés dans un environnement commun. Il possède les caractéristiques principales suivantes [1]:

- chaque agent à des informations ou des capacités de résolution de problèmes limitées, ainsi chaque agent a un point de vue partiel.
- il n'y a aucun contrôle global du système multi-agents.
- les données sont décentralisées.
- le calcul est asynchrone.

2. Pour quoi les SMA ?

Certains domaines requièrent l'utilisation de plusieurs entités, par exemple, il y a des systèmes qui sont géographiquement distribués. Les SMA procurent une façon facile et efficace de les modéliser.

Autre situation, où les sont requis, est lorsque les différents systèmes et les données qui s'y rattachent appartiennent à des organisations indépendantes qui veulent garder leurs informations privées et sécurisées pour des raisons concurrentielles.

Les SMA possèdent également les avantages traditionnels de la résolution distribuée et concurrente de problèmes [3]:

- La **modularité**, permet de rendre la programmation plus simple.

- La **vitesse**, due principalement au parallélisme.

- La **fiabilité**, qui peut être également atteinte, dans la mesure où le contrôle et les responsabilités étant

Partagés entre les différents agents, le système peut tolérer la défaillance d'un ou de plusieurs agents.

Finalement, les SMA héritent aussi des bénéfices envisageables du domaine de l'intelligence

Artificielle comme par exemple, le traitement symbolique (au niveau des connaissances).

i. **Interaction entre agents :**

Jacques Ferber donne la définition suivante de l'interaction : « Une interaction est la mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques... »[2] Les interactions entre agents peuvent varier selon les situations dans lesquelles se trouvent ces agents : coexistence, compétition ou coopération.

- S'ils ne font que **coexister**, alors chaque agent ne considère les autres agents que comme des composantes de l'environnement. Il n'y a aucune communication directe entre les agents
- S'ils sont en **compétition**, alors le but de chaque agent est de maximiser sa propre satisfaction. La compétition entre agents peut avoir plusieurs sources : Les buts des agents peuvent être incompatibles ou les ressources peuvent être insuffisantes.
- S'ils sont en **coopération**, alors le but des agents n'est plus seulement de maximiser leurs propres satisfactions mais aussi de contribuer à la réussite du groupe. Les agents travaillent ensemble à la résolution d'un problème commun.

ii. **Coopération entre agents**

La coopération est nécessaire quand un agent ne peut pas atteindre ses buts sans l'aide des autres agents. Cette situation est fréquente même chez des espèces primitives. Les buts nécessitant la coopération peuvent être.

iii. **Coordination entre agents**

Il y a interaction entre les agents soit parce qu'ils coopèrent, soit parce qu'ils sont en compétition. Dans les deux cas, une coordination peut être nécessaire pour améliorer le fonctionnement global du système[2].

Lorsque plusieurs agents travaillent sur le même lieu, utilisent les mêmes ressources, ou résolvent des sous problèmes qui ne sont pas complètement indépendants (conception d'un objet complexe par exemple), ils doivent accomplir, en plus des tâches liées directement au problème traité, des tâches de coordination. Ces tâches ne sont pas directement productives mais elles améliorent les tâches productives.

Les tâches de coordination peuvent être accomplies directement par les agents concernés quand elles sont relativement rares et qu'elles n'engagent pas un grand nombre d'agents en même temps. Sinon, elles sont prises en charge par des agents spécialisés qui recueillent les demandes et fixent les ordres de priorité ou d'autres contraintes.

iv. **Négociation entre agents**

Comme nous avons vus précédemment, en interagissant dans un environnement partagé, les agents doivent coordonner leurs actions et avoir des mécanismes pour la résolution des conflits. Le mécanisme favori pour la résolution des conflits *et* la coordination, inspiré du modèle des humains, est **la négociation**.

Dans les systèmes multi-agents, la négociation est une composante de base de l'interaction surtout parce que les agents sont autonomes ; il n'y a pas de solution imposée à l'avance et les agents doivent arriver à trouver des solutions dynamiquement, pendant qu'ils résolvent les problèmes [4].

Conclusion

Les interactions entre les agents et la résolution distribuée qu'elles mettent en place induisent alors une contradiction avec le principe d'autonomie de décision propre au concept d'agent. Les agents peuvent avoir conscience de cette résolution collective (agents cognitifs) ou non (agents réactifs), mais de toute façon, ils utiliseront ou subiront à un moment ou à un autre des contraintes issues de leurs interactions avec les autres agents du système.

Toute la difficulté de concevoir des systèmes multi-agents consiste alors à gérer cette contradiction entre le principe d'autonomie des agents et la résolution collective par le système qu'ils composent. Tous les agents doivent s'intégrer au même système pour former un tout cohérent résolvant le problème à traiter, même si les formalismes qui modélisent les différents agents du système sont hétérogènes. Cette nécessité d'intégration et d'interaction avec les autres fait intervenir des mécanismes et des notions permettant la coordination de la résolution distribuée du problème pour obtenir un comportement global cohérent et efficace du système. Les notions d'interactions et d'organisation permettent d'appréhender cette nécessité.

Chapitre II:

Émergence du langage

1. Introduction

L'origine et l'évolution du langage et des langues constituent un vaste domaine d'étude, qui a vu naître en son sein nombre des courants de recherche actuels en linguistique. De plus en plus, il implique en outre la convergence de disciplines extérieures qui ouvrent progressivement de nouvelles perspectives sur un sujet difficile[5].

2. Définition du langage

Le langage est un moyen de communication purement humain et non instinctif, pour les idées, les émotions et les désirs, par l'intermédiaire d'un système de symboles créés à cet effet. Ces symboles sont en premier lieu auditifs et sont produits par ce qu'on nomme « les organes de la parole»[6], c'est la notion la plus large, il comprend toutes les façons de s'exprimer, de dire, de communiquer, d'échanger (il peut s'agir de gestuelle, du langage d'action, d'appui...)[7]

De nombreuses définitions ont été proposées par les différentes disciplines étudiant le langage (la linguistique, mais aussi la neurobiologie, la psychologie, la philosophie, la biologie, etc.), chacune s'intéressant à une de ses particularités[8].

3. Définition de la communication

Action, fait de communiquer, d'établir une relation avec autrui.

4. Caractéristiques du langage[7]

- **Langage, langue et parole.**

Il y a deux façons d'étudier une langue

- **La diachronie** : évolution au cours de l'histoire étymologie.
- **La synchronie** : à un moment donné du temps, c'est dans cette perspective que nous allons nous placer.

La langue: cela renvoie à un code propre à une communauté qui rassemble des mots et un agencement particulier de ceux-là, c'est un produit social car c'est une propriété collective. Une langue est coercitive car cela impose des mots une syntaxe et si on ne les emploie pas on risque de ne pas être compris.

La parole: c'est la mise en œuvre par les individus (individuellement) de la langue,

L'appropriation par l'individu de la langue et sa mise en œuvre sont indépendantes de langue puisque par exemple il existe des langues mortes qui ne sont plus parlées.

L'usage de la parole varie en fonction du lexique, des registres de langue mais renvoie aussi à des aspects psychologiques.

« Je veux un bonbon ». (Je veux un bonbon ou je souhaite que l'on s'occupe de moi).

- **Les mots sont des signes.**

La langue est la capacité de symboliser, d'utiliser des signes.

Les mots sont conventionnels, ils nous permettent des symboles par similitude (la balance symbolise la justice).

Les mots sont des signes de pure convention : une table ne ressemble pas à une table déjà le mot n'a pas de réalité visible comme l'objet table

Le mot comporte une double face le signifiant et le signifié.

Le signifiant c'est l'aspect phonique, Saussure parle d'image acoustique c'est alors la notion d'image qui a de l'importance. Les mots de la langue sont dans notre tête et non pas dans la réalité physique. Ils sont dans notre pré-conscience, prêts à être mobilisés.

- **L'arbitraire du signe.**

Il est arbitraire, il ne ressemble pas à ce qu'il évoque et c'est un lien de pure convention, (le mot chien n'aboie pas et ne renvoie à aucun chien réel car sa généralité ne renvoie qu'à une classe et non pas à un chien. La classe chien est créée par la pensée humaine pour comprendre la réalité.

Le mot est le produit d'une convention ce qui se retrouve dans la diversité des langues (cheval en français, horse en anglais...)

La diversité des langues fonctionne aussi pour les onomatopées.

L'arbitraire est déjà dans signifiant et le signifié.

Sœur, le son n'a aucune parenté avec l'idée de sœur

La langue est marquée par un caractère de convention. Celle-ci est nécessaire, à partir du moment où un mot existe, on ne peut pas décréter que le mot dise autre chose que le concept dont il est porteur.

Les mots ont un sens, le pouvoir d'agir sur ce sens n'est dépendant que de la collectivité dans un temps long.

Les êtres humains ont tendance à penser que les mots de sa langue sont naturels et non pas arbitraires. L'usage fait un effet de naturel (spaghetti).

- **La double articulation[3].**

Les mots sont des unités mais il y a plus petites unités

Les plus petites unités de sens sont les monèmes.

On a également les lexèmes (racines) les morphèmes (terminaisons, suffixes, préfixes).

J'ai mal à la tête : six mots, six monèmes

Travaillons : radical + terminaison, 2 monèmes en un seul mot.

Ces unités permettent une économie : un lexème va permettre de trouver d'autres mots : travailler, travailleur.

Un nombre d'éléments réduits permettent un grand nombre de possibilités.

Les phonèmes divisent les monèmes, ce sont des unités de sons : le phonème, il n'a pas de signifié, il n'a de valeur que dans sa différence avec d'autres phonèmes. Il en existe, en français, un nombre réduit 31 et 34 phonèmes.

Les unités n'apparaissent pas quand on parle d'où la frontière difficile à concevoir par les enfants, en français l'élision amplifie cela.

Les linguistes différencient 4 composantes du langage[3] :

- **Phonétique** = concerne les sons du langage ou phonèmes.
- **Sémantique** = concerne les mots et leur signification (lexique).
- **Syntaxe** = règles d'associations des éléments du langage entre eux.

-Pragmatique = étude des actes de parole en situation (prise de parole, échange conversationnel, etc.).

5. Origine du mot HOMO[9]

Si la plupart des chercheurs pensent que la faculté de langage n'est pas apparue après l'émergence de notre espèce, le consensus est moins net sur la question de savoir si elle est apparue avant, autrement dit, si d'autres espèces d'hominidés, comme notamment les Neandertaliens, possédaient un langage de même nature que le nôtre.

Le mot Homo est le nom du genre biologique qui regroupe toutes les espèces humaines. Elles sont toutes éteintes à l'exception de l'Homme.

Le terme Homo, humain en latin, dérive d'une racine de l'Indo-européen commun .

Il a été choisi par Carl von Linné, dans sa méthode de classification de la nature, L'homme y est décrit sous le nom d'homme sapiens.

Le langage a été le résultat d'un processus très long, sur des millions d'années certainement, un processus dont le point de départ peut très bien être des facultés cognitives et sémiotiques comparables à ce qu'on trouve chez les chimpanzés. Avec, en plus, les modifications du système phonatoire liées à la station debout permanente et la démarche bipède de l'homme.

Nous aurons besoin, au moins, de prendre en considération les étapes suivantes dans l'évolution de l'homme[9] :

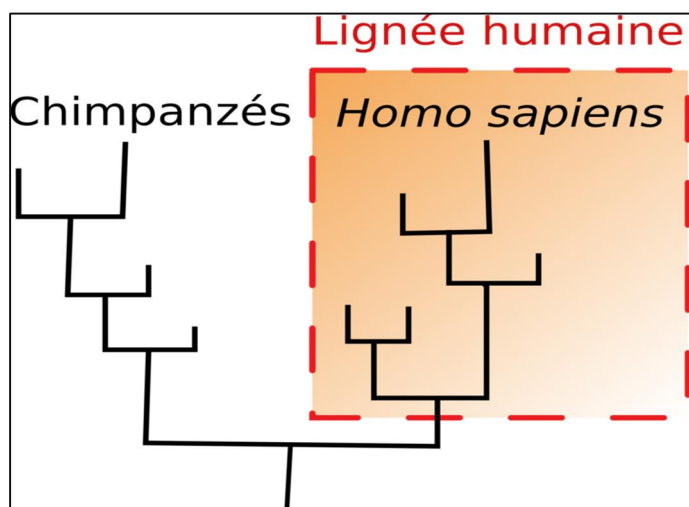


Figure 1 : Évolution de la lignée humaine par rapport à celle des chimpanzés[10].

- **Homo habilis[9] (env. – 1, 8 million d’années)**
- Utilise les outils
- Construit des gîtes/des places pour dormir
- Chasse collectivement (Langaney 1988: 30)

Cette description ne distingue pas l’homo habilis des chimpanzés et ne présuppose pas un langage -Les loups et les hyènes aussi chassent en groupes -mais elle peut suggérer les débuts d’une faculté langagière.

- **Homo erectus[9] (env. – 1, 6 million d’années)**
- Fabrique des outils
- Possède une vie sociale
- Utilise le feu depuis env. – 400 000 (ib. 31)

L’homo erectus est le premier humain indiscutable : la station debout est permanente, il fabrique ses outils, ne se contente plus de ramasser une pierre ou une branche pour ses besoins immédiats, et il se sert du feu. Une capacité langagière est probable et une sorte de langage possible. Le seul fait de fabriquer les outils semble présupposer des processus d’apprentissage, qui, à leur tour, présupposent une sorte de langage.

- **Homo sapiens néandertaliens[9] (env. – 140 000 (□□120 000) à □□40 000 (□□30 000) ans)**
- Fabrique des outils
- Possède une vie sociale
- Utilise le feu
- Enterre ses morts

La place des néandertaliens dans l’évolution de l’homme est controversée, mais ils sont souvent considérés comme une branche parallèle de l’arbre évolutionnaire. Leur organisation sociale assez avancée et leurs cérémonies funèbres laissent supposer qu’ils ont communiqué par un langage articulé.

- **Homo sapiens [9](env. – 80 000 ans, en Europe depuis env. – 30 000 ans, l’homme de Cro Magnon)**

Est aujourd’hui le seul représentant de l’espèce humaine et ne se distingue en rien de l’homme contemporain. L’espèce a probablement possédé un langage : environ – 50 000 ans, le langage humain existe certainement. Le scénario total supposé est présenté dans la figure 2.

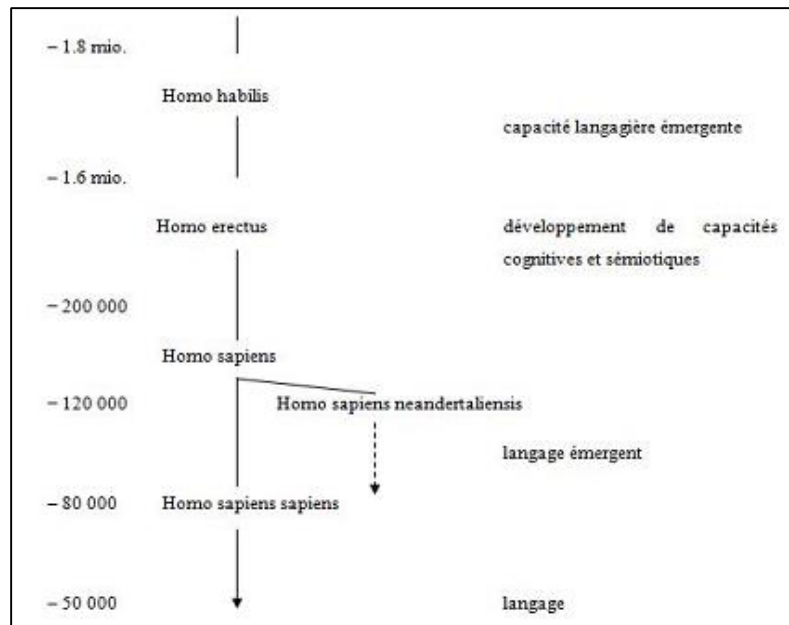


Figure 2 : Origine du mot HOMO[11].

6. L’acquisition du langage

- **Le langage humain inné ou acquis**

Le débat Piaget-Chomsky, en quoi s’oppose-t-il ?

Pour Piaget le langage est acquis pas inné, pour Piaget il ne y a pas d’universaux du langage. Selon Chomsky, il existe des compétences mentales innées, inscrites dans le cerveau de l’homme qui explique notamment ses capacités linguistiques universelles.

Piaget soutient que les capacités cognitives de l’humain ne sont ni totalement innées ni totalement acquis, elles résultent d’une construction progressive où l’expérience et la maturation interne se combinent.

Piaget ouvre la discussion : la pensée ne fonctionne pas par un simple enregistrement des données (comme le suppose les empiristes).[12]

On peut entrevoir quelques éléments de l'origine du langage en étudiant l'acquisition du langage par les enfants et en utilisant avec précaution la loi biogénétique de Haeckel. Selon celle-ci, l'ontogenèse reflète la phylogenèse. Les stades que parcourt l'acquisition du langage chez les enfants pourraient être une sorte d'image de l'évolution du langage chez l'espèce. Cette hypothèse comporte au moins deux aspects intéressants[9] :

- Anatomiquement parlant, le nouveau-né n'est pas tout à fait prêt. Les organes phonatoires ne sont pas en place, le larynx ne descend qu'à l'âge d'un an. L'anatomie du très petit enfant ressemble en effet à celle de l'homme de Néandertal (Lieberman, Crelin 1971).
- L'acquisition même du langage se produit à une vitesse incroyable à travers des stades qu'on peut supposer analogues à ceux de l'évolution du langage (Lyons 1977: 85ss.; Fry 1977: 101ss.)[9]:
 - ✓ Un premier stade où l'enfant babille, explore le spectre phonique et acoustique, et développe une sensibilité aux contours prosodiques.
 - ✓ Vers la fin de la première année, on constate la présence de signaux holophrastiques, l'enfant semble programmé pour la communication sonore. Vers l'âge de deux ans, des énoncés de deux et de trois mots sont produits.
 - ✓ A travers les stades deux et trois, jusqu'à l'âge de quatre ans environ, le langage est pleinement développé. L'enfant arrive à maîtriser le système phonologique en acquérant d'abord les distinctions et oppositions basales, dominantes et non-marquées (Jakobson 1944), alors que les traits récessifs, ou marqués, tels que les consonnes isolées [r] et [l] ne sont acquis que plus tard, quand l'enfant parle proprement.

Un tel scénario pour l'origine et l'évolution du langage chez l'espèce ne me semble pas trop irréaliste[9].

- **Apprentissage de la communication [13]**

Le langage n'est pas seulement un moyen d'encoder les significations ; il est aussi un des instruments qui permettent la communication entre les individus ; par son intermédiaire, on agit

sur autrui, et on est soumis à l'action d'autrui. On parle à l'autre pour l'informer, pour obtenir de lui des informations, pour influencer son comportement et ses opinions, pour le convaincre, pour le blâmer ou le féliciter, etc. Le langage ne relie pas seulement un contenu et une forme, il installe aussi une relation entre deux ou plusieurs individus, qui tour à tour, se font locuteurs et auditeurs, au sein de cet échange très structuré qu'est le dialogue.

La question des fonctions du langage n'est assurément pas neuve. Sans vouloir en retracer ici le développement, on signalera qu'elle a intéressé tant les linguistes que les psychologues. En linguistique, la classification la plus généralement retenue est celle de Jakobson (1956), qui proposera de distinguer les six fonctions suivantes [13]:

- ✓ **La fonction expressive** : le locuteur est centré sur ses besoins et ses émotions
- ✓ **La fonction référentielle** : le locuteur décrit son environnement
- ✓ **La fonction conative** : on tente d'agir sur autrui par le langage
- ✓ **La fonction phatique** : le langage sert à assurer que la communication reste maintenue
- ✓ **La fonction poétique** : la destination du langage est ludique ou artistique
- ✓ **La fonction métalinguistique** : le langage devient objet d'analyse

7. Trois modèles informatiques pour l'émergence du langage[11]

- **Pourquoi des modèles informatiques ?**

Qui a décidé d'appeler table l'objet que l'on désigne ainsi ? La réponse est simple : personne ! Alors comment ce mot s'est-il imposé ? On peut répondre qu'il a une histoire, qu'il vient du mot latin tabula (qui voulait dire planche), qui lui-même provenait sans doute d'un mot d'une langue encore plus ancienne... Oui, mais comment tout cela a-t-il commencé ? Y a-t-il eu au départ une personne ou un groupe pour choisir quels mots allaient désigner chaque chose ? Sinon par quel miracle les gens se sont-ils compris ?

Pour aborder ce type de questions, des chercheurs utilisent l'informatique. Ils simulent sur ordinateur, de manière bien sûr extrêmement simplifiée, des communautés d'agents qui interagissent entre eux et ils essaient de déterminer dans quelles conditions apparaît telle ou telle capacité de communication, telle ou telle propriété que l'on retrouve ou non dans nos langues. On dispose ainsi de modèles expérimentaux, modifiables à volonté, qui permettent de réfléchir concrètement à l'émergence des propriétés du langage. Cette activité de modélisation peut se révéler très fructueuse dans les recherches sur l'origine et l'évolution du langage, à

condition de bien mesurer ses limites : l'objectif n'est pas de modéliser de façon « réaliste » ces phénomènes hautement complexes, mais de rechercher les conditions minimales dans lesquelles peuvent émerger les propriétés linguistiques auxquelles on s'intéresse.

Etant donné le caractère foisonnant de ce domaine de recherche, il est pratiquement impossible d'en donner une vue exhaustive qui ne se réduise pas à un survol superficiel et plutôt frustrant. Nous avons donc choisi de nous centrer sur quelques modèles, représentatifs de l'ensemble du champ, de manière à mieux illustrer l'intérêt de ce type de recherches.

- **Un premier modèle d'émergence d'un lexique par adaptation « génétique »[11]**

Parmi les premières recherches dans ce domaine, c'est sans doute le travail de Werner et Dyer qui a eu le plus grand impact. Ces chercheurs ont en effet montré, avec un dispositif assez simple, qu'un lexique rudimentaire pouvait émerger dans une population d'agents à la seule condition que ce lexique joue un rôle « vital » pour la perpétuation de cette population au cours du temps. Le système conçu par Werner et Dyer s'inscrit dans ce que l'on appelle les modèles de vie artificielle. Il s'agit de systèmes informatiques qui simulent un mécanisme de sélection naturelle : des agents évoluent dans un espace dans lequel ils interagissent et se reproduisent suivant un algorithme génétique. Chaque agent possède un « génome » qui régit son comportement, et tout agent « nouveau-né » hérite du patrimoine génétique de ses « parents » avec des mutations aléatoires qui engendrent en permanence de la diversité dans la population. Ainsi, toute mutation qui rend un agent plus performant (c'est-à-dire qui augmente son succès reproductif) va se propager dans la population. Après un certain nombre de générations, seuls subsistent de façon stable des génomes qui confèrent aux agents un comportement adapté à la survie de la population.

La mise en scène de l'expérience de Werner et Dyer est la suivante. La population d'agents est composée de « mâles » et de « femelles » disposés sur les cases d'une grande grille carrée.

Les femelles restent toujours immobiles alors que les mâles peuvent se déplacer (cf. fig. 3). Si un mâle rencontre une femelle, ils se reproduisent, donnant naissance à deux nouveaux agents, un mâle et une femelle (on supprime aussi aléatoirement un mâle et une femelle de la grille, de manière à ce que la taille de la population reste constante). Les femelles sont dotées d'une capacité « visuelle » : elles perçoivent tout agent mâle qui se situe dans une case voisine (dans un périmètre de 5x 5 cases autour d'elles). Elles peuvent alors émettre des sons qui sont

perçus par les mâles qui passent à proximité. Les mâles, eux, ne voient rien, mais ils ont la capacité de choisir leur action parmi quatre mouvements possibles (aller tout droit, tourner à droite ou à gauche, rester immobile) en fonction des sons qu'ils perçoivent[11].

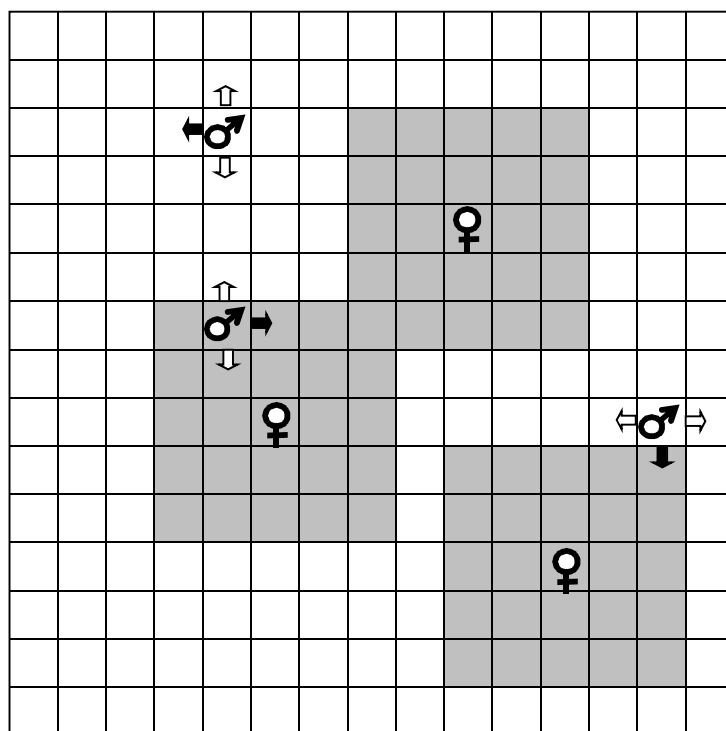


Figure 3 : L'expérience de Werner et Dyer[14]

Les zones grisées représentent le champ de perception visuelle des femelles et les flèches l'orientation (en noir) et les déplacements possibles des mâles. L'expérience a été menée avec 1600 agents sur une grille de 200x200 cases.

Chaque agent est muni d'un génome complet, constitué de deux parties : une partie utilisée par les mâles, qui détermine la direction de leurs mouvements en fonction des sons perçus, et une partie utilisée par les femelles, qui détermine le type de son émis en fonction de la position des mâles dans leur champ visuel. Au départ de l'expérience, les génomes sont choisis aléatoirement pour chaque agent. Il n'y a donc aucune cohérence entre les sons émis par les femelles et les déplacements des mâles, et les rencontres entre mâles et femelles sont entièrement fortuites. Mais au cours de la simulation on voit se mettre en place progressivement un comportement cohérent des agents. Les femelles, dans leur grande majorité, émettent des sons différenciés suivant la position et l'orientation du mâle dans leur champ visuel, et les mâles réagissent la plupart du temps de manière adaptée à ces sons, en se dirigeant vers la femelle émettrice. C'est donc un véritable code de communication, constitué de signes arbitraires et partagé par une majorité d'agents, qui a émergé grâce au mécanisme de sélection implémenté

par l'algorithme génétique. Chaque son a pris la signification d'une instruction précise (« tout droit ! », « à droite ! », « à gauche ! », « ne bouge plus ! ») qui est émis à bon escient par les femelles, et qui est correctement interprété par les mâles.

Ainsi cette expérience prouve bien qu'une convention peut effectivement s'établir au cours d'interactions individuelles dans une population, dans des conditions tout à fait minimales : il suffit en somme que le système de communication soit potentiellement utile à la survie des individus, ou plutôt de leur génome, pour que le mécanisme de sélection génétique aboutisse à son émergence. Le modèle de Werner et Dyer présente bien sûr un certain nombre de caractéristiques qui ne conviennent pas pour la modélisation de l'émergence du langage humain, même si l'on fait abstraction de la « mise en scène » du modèle, forcément caricaturale, pour ne retenir que les principes d'interaction entre agents et les mécanismes d'adaptation mis en œuvre. Notamment, le code obtenu est inscrit directement dans le génome des agents, ce qui ne saurait être le cas pour le lexique de nos langues. Mais il n'en reste pas moins que ce travail a joué un rôle fondateur en montrant tout l'intérêt potentiel de ce type de recherches[11].

- **Les « têtes » parlantes[14]**

A la suite de Werner et Dyer, de nombreux chercheurs ont proposé des modèles d'émergence d'un lexique dans une communauté d'agents, en utilisant différents mécanismes et les scénarios les plus divers. Nous allons nous centrer ici sur les travaux de Steels et Kaplan, qui illustrent bien ce courant de recherche.

Dans ces modèles, les agents sont munis d'une mémoire associative, qui s'enrichit au cours des interactions. Autrement dit, contrairement au modèle de Werner et Dyer, les agents ne « naissent » pas avec un code langagier inscrit dans leur génome : ils « apprennent », tout au long de leur existence, à associer des sons avec des sens. Les interactions entre agents consistent en « jeux de dénomination » (naming games). Les détails précis du jeu varient d'une expérience à l'autre, mais typiquement ils se déroulent de la manière suivante. Un premier agent, le « locuteur », sélectionne un objet de son environnement et le nomme en prononçant un « mot ». Le deuxième agent, « l'interlocuteur », entend le mot émis, et doit désigner l'objet correspondant. Si l'objet désigné est bien celui qu'avait sélectionné le locuteur, le jeu est un succès, sinon c'est un échec. L'interlocuteur modifie alors sa mémoire associative en fonction du résultat du jeu.

Dans les premières expériences, tout le processus était entièrement simulé sur ordinateur. Par la suite, ces chercheurs ont construit de véritables robots, des « têtes parlantes »

(talkingheads : voir la photo fig. 4). Des agents logiciels viennent tour à tour manipuler les robots pour jouer à un jeu de dénomination avec différentes formes géométriques présentes sur un tableau. Les mots qui constituent le lexique émergent désignent diverses propriétés de ces formes : triangle, carré, rouge, vert, petit, en haut, tout à droite, etc.



Figure 4 : le dispositif des têtes parlantes[14]

Ce sont des systèmes robotiques munis de caméras orientables et d'un système de traitement d'image. Ils peuvent donc percevoir et « pointer » (par l'orientation de la caméra) différentes formes géométriques sur un tableau.

Dans ce type d'expérience robotique dite « située » ou « ancrée dans le réel », de nombreux facteurs d'imprécision interviennent qui rendent la communication plus incertaine : variations naturelles de luminosité, variations des positions des objets sur le tableau, mouvements approximatifs des caméras, imprécision du système de perception (erreurs de segmentation de l'image, par exemple), etc. De plus, la diversité des propriétés permettant de désigner les objets introduit aussi une source de confusion : un même objet peut être désigné par sa couleur par le locuteur et par sa position par l'interlocuteur... On se trouve donc dans

des conditions de communication assez réalistes qui rendent l'expérimentation bien plus intéressante[14].

L'une des expériences a été menée « en vrai grandeur » pendant plusieurs mois, avec le concours de centaines d'internautes. Plusieurs plate-formes robotiques ont été installées dans des villes différentes (Paris, Bruxelles, Tokyo, etc.). Un millier d'agents logiciels, créés par les internautes, pouvaient migrer de plate-forme en plate-forme, formant ainsi une vaste communauté interagissant 24 heures sur 24. Qui plus est, les internautes pouvaient enseigner des mots de leur langue à leurs agents, ce qui augmentait le caractère ludique de l'expérience. Cette expérience a constitué un succès assez spectaculaire. Comme le souligne Kaplan :

« Malgré les erreurs causées par le caractère ancré et situé des interactions, malgré le flux perpétuel de nouveaux agents, malgré les scènes régulièrement renouvelées, malgré le caractère incohérent des enseignements divulgués par les utilisateurs à leur agent, un lexique a pu émerger. Ce lexique, mélange de mots inventés par les agents et de mots enseignés par des humains, s'est transmis culturellement de façon très stable. Le cœur de ce lexique est constitué par des mots correspondant à des sens qui optimisent la robustesse, la généralité, la facilité d'apprentissage et la réutilisabilité. Ils sont parfaitement adaptés aux scènes que les agents rencontrent dans les environnements qui leur sont présentés. »[11]

Ainsi ces jeux de dénomination fournissent un cadre dans lequel peut émerger un lexique conventionnel qui possède notamment des propriétés dynamiques d'apprentissage et d'évolution que l'on peut rapprocher de phénomènes observables dans le lexique des langues humaines. Ce modèle reste bien sûr très frustré et limité, mais cela suffit à prouver que ces propriétés peuvent être acquises sans que l'on ait besoin de faire appel à des mécanismes cognitifs spécialisés, encore moins à des facultés innées[11].

- **Émergence de propriétés syntaxiques[11]**

Les chercheurs se sont aussi penchés sur la modélisation de l'émergence de propriétés structurelles des langues, notamment aux niveaux phonologique et syntaxique. On sait qu'au-delà de l'extraordinaire diversité des langues on a mis en évidence un certain nombre de régularités auxquelles obéissent toutes les langues.

Nous ne prendrons qu'un exemple, dans le domaine de la syntaxe, mais il est très représentatif de ce type de phénomènes. Il s'agit des propositions relatives. La plupart des langues possèdent des pronoms relatifs. Mais ces pronoms sont plus ou moins contraints suivant

les langues. Certaines admettent des pronoms relatifs sujets (l'équivalent de qui en français) ; d'autres admettent des pronoms compléments d'objet (l'équivalent de que). Mais ce qui est remarquable, c'est que toutes les langues qui possèdent un pronom objet possèdent aussi un pronom sujet, alors que l'inverse n'est pas vrai. Tout se passe comme s'il fallait d'abord être capable de fabriquer des relatives avec un pronom sujet, telle que L'homme qui a épousé Marie, avant de pouvoir fabriquer des relatives avec un pronom objet, comme L'homme que Marie a épousé. C'est ce que l'on appelle un universel implicationnel ou hiérarchique.

En fait, dans le cas des propositions relatives, la hiérarchie de se limite pas aux pronoms sujet et objet. Elle s'applique à toutes les fonctions syntaxiques possibles du pronom, qui « apparaissent » dans les langues dans l'ordre suivant : sujet, objet direct, objet indirect, circonstanciel, complément de nom, comparatif. Par exemple, pour qu'une langue puisse utiliser un pronom complément de nom (comme dans : L'homme dont Marie est l'épouse), il faut qu'elle possède tous les pronoms qui le précèdent dans la hiérarchie, notamment l'objet indirect (à l'œuvre dans : L'homme à qui Marie a dit oui).

Comment expliquer cette curieuse propriété ? Certains y ont vu l'effet d'un mécanisme inné : on naîtrait tous avec un « organe du langage » qui imposerait cette hiérarchie. Ensuite, suivant la langue que l'on acquière, cet organe s'ajusterait pour ne produire que les pronoms possibles dans la langue en question : il suffirait donc d'un seul paramètre pour fixer la « barre » qui sépare les pronoms acceptables dans cette langue de ceux qui ne le sont pas. L'acquisition de ces universaux implicationnels, a priori assez bizarres et complexes, serait donc en fait très simple : toute l'apparente complexité proviendrait de l'organe inné. Ce serait même l'une des principales « preuves » de l'existence du caractère inné de la faculté de langage.

C'est dans le but de contester cette explication « génétique » que Simon Kirby a construit un modèle informatique de ce type de phénomène. Comme dans le cas du lexique, il a pu montrer que de tels universaux implicationnels pouvaient émerger dans certaines conditions dans des populations d'agents, sans que l'on ait besoin d'invoquer un mécanisme inné.

Le principe général des simulations est le suivant. On se donne une population composée de deux générations d'agents :

- La génération des locuteurs, munis chacun d'une « grammaire » qui leur permet de produire des phrases obéissant aux règles de cette grammaire. Dans notre cas, les grammaires servent à produire des propositions relatives. Chaque grammaire permet ou non tel type de pronom.

- La génération des apprenants, qui doivent acquérir une grammaire. Pour ce faire, chaque apprenant est confronté à un certain nombre de productions de locuteurs, et il construit sa grammaire en fonction de cet échantillon. Ainsi un apprenant confronté à des relatives dont le pronom est en position sujet, objet direct et indirect construira la grammaire qui permet ces trois fonctions, tandis qu'un apprenant confronté à des exemples ne comportant que des relatives avec des pronoms objets directs construira la grammaire qui ne permet que ces pronoms (c'est donc une grammaire « déviante » au sens où on ne trouve pas une telle grammaire dans les langues humaines)[11].

A chaque cycle évolutif, chaque locuteur produit donc des énoncés obéissant aux règles de sa grammaire et chaque apprenant sélectionne un échantillon parmi les énoncés produits pour acquérir sa propre grammaire. Les apprenants remplacent alors les locuteurs, une nouvelle génération d'apprenants est créée et un nouveau cycle peut commencer. Après un certain nombre de cycles, on observe quelles grammaires se sont stabilisées au cours du processus.

Nous n'entrerons pas ici dans le détail des expériences. Disons simplement que l'universel implicationnel émerge, si les quatre conditions suivantes sont réalisées :

- existence d'une pression de sélection par les apprenants qui privilégie les productions langagières de complexité structurelle plus faible (la complexité structurelle croît avec la difficulté d'analyse syntaxique d'une phrase) ;
- existence d'une pression antagoniste de sélection par les locuteurs de formulations de complexité morphologique plus faible (la complexité morphologique croît avec le nombre d'unités linguistiques nécessaires présents dans une phrase) :
- prise en compte d'une répartition spatiale des agents dans laquelle chaque apprenant n'interagit qu'avec un petit nombre de locuteurs voisins ;
- introduction de facteurs de variation (variantes syntaxiques, réductions morphologiques, grammaticalisations, etc.) qui changent le poids relatif des deux types de complexité.

Sous ces quatre conditions, on obtient non seulement l'universel implicationnel (cf. fig. 5), mais aussi un certain nombre de caractéristiques intéressantes de l'évolution des langues (notamment le rôle du bilinguisme à la frontière entre deux aires de répartition de langues de propriétés différentes).

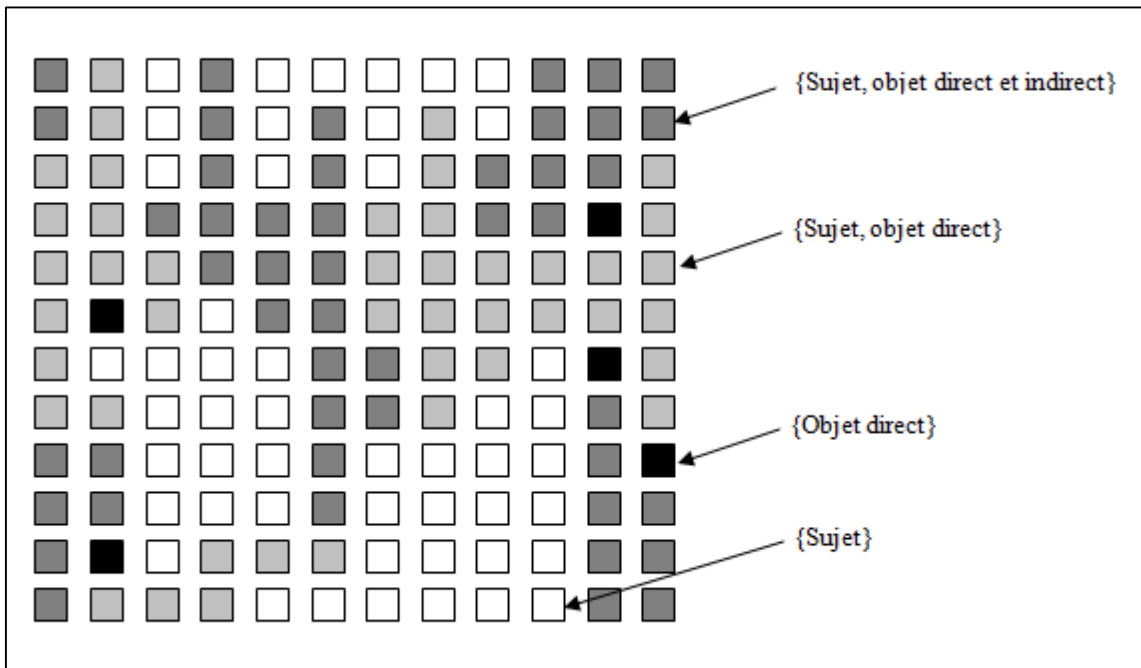


Figure 5 :Stabilisation d'un universel implicationnel[11]

Chaque case représente un agent et sa couleur indique la grammaire qu'il a acquise. Une seule grammaire déviante (couleur noire : {objet direct}) apparaît aux frontières entre deux grammaires acceptables mais de manière transitoire, lors du basculement d'une grammaire acceptable à une autre[11].

Conclusion

Dans cette partie, nous avons introduit l'émergence des langages, une question qui a été toujours posée par les chercheurs dans les différents domaines.

La problématique est étudiée en informatique dans beaucoup de travaux dans lesquels des modèles sont proposés pour modéliser l'émergence du langage ou autres propriétés. Nous avons présenté quatre modèles de l'émergence du langage: le modèle de la cohérence de vocabulaire,

le modèle des réseaux de neurones, le modèle de l'apprentissage réitéré et le modèle de mémoire associative.

Dans ces modèles, le langage est considéré comme étant un système complexe adaptatif. Les jeux de langage sont un modèle général de simulation pour étudier les phénomènes linguistiques comme l'émergence, l'évolution et la stabilité.

Chapitre III:

La modélisation

1. Introduction

La réalisation d'une application doit être impérativement précédée d'une méthodologie d'analyse et de conception qui a pour objectif de permettre de formaliser les étapes préliminaires du développement d'une application.

La phase d'analyse permet de lister les résultats attendus, en termes de fonctionnalités. et La phase de conception permet de décrire de manière non ambiguë, le plus souvent en utilisant un langage de modélisation, le fonctionnement futur du système, afin d'en faciliter la réalisation.

2. Présentation et justification du choix d'UML

a) Définition[15]

Unified Modeling Language est un langage unifié de modélisation objets. Ce n'est pas une méthode, il ne donne pas de solution pour la mise en œuvre d'un projet. C'est avant tout un formalisme graphique issu de notations employées dans différentes méthodes objets.

b) Que ce qu'un diagramme UML ?[16]

UML utilise l'approche objet en présentant un langage de description universel. Il permet grâce à un ensemble de diagrammes très explicites, de représenter l'architecture et le fonctionnement des systèmes informatiques complexes en tenant compte des relations entre les concepts utilisés et l'implémentation qui en découle.

UML est avant tout un support de communication performant, qui facilite la représentation et la compréhension de solutions objet :

- Sa notation graphique permet d'exprimer visuellement une solution objet, ce qui facilite la comparaison et l'évaluation de solutions.

L'aspect formel de sa notation, limite les ambiguïtés et les incompréhensions.

Son indépendance par rapport aux langages de programmation, aux domaines d'application et aux processus, en fait un langage universel.

UML est donc bien plus qu'un simple outil qui permet de "dessiner" des représentations mentales... Il permet de parler un langage commun, normalisé mais accessible, car visuel.

Il représente un juste milieu entre langage mathématique et naturel, pas trop complexe mais suffisamment rigoureux, car basé sur un métamodèle. Une autre caractéristique importante d'UML, est qu'il cadre l'analyse. UML permet de représenter un système selon différentes vues complémentaires : les diagrammes.

3. Diagramme de cas d'utilisation

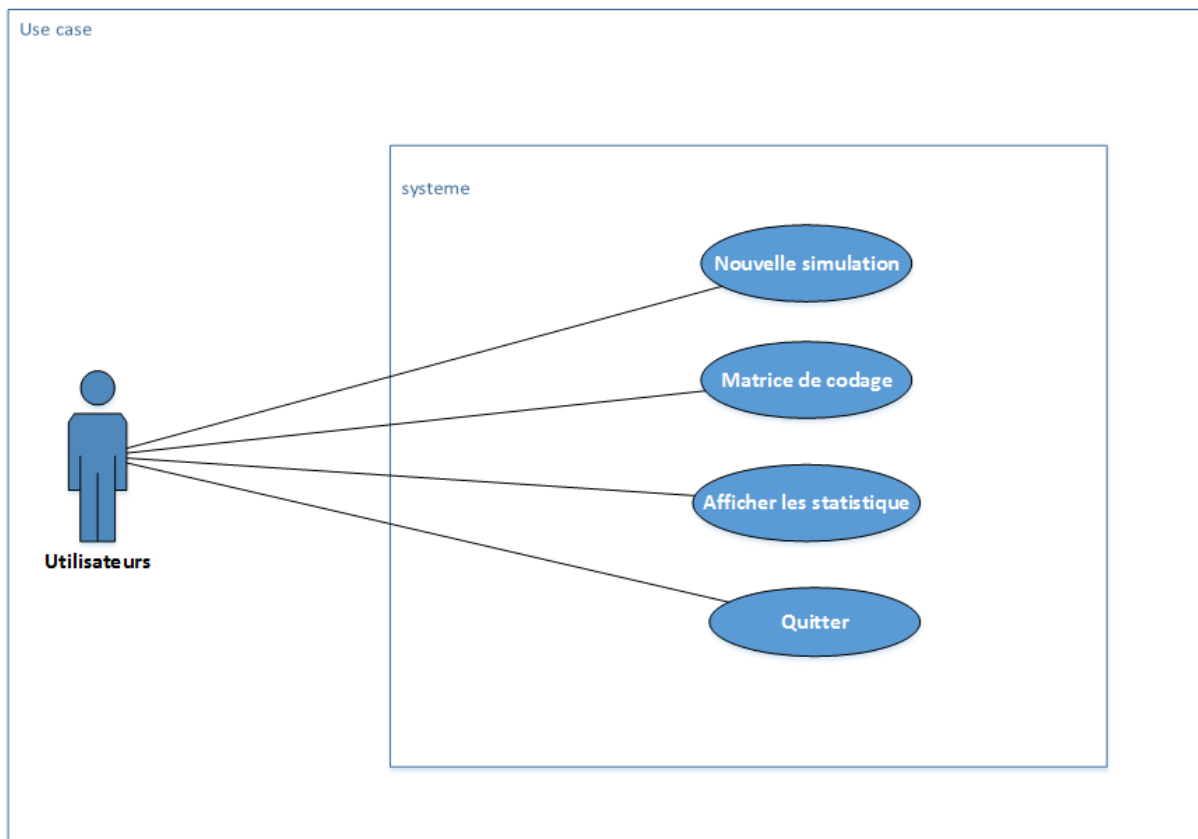


Figure 6: Diagramme de cas d'utilisation.

4. Diagrammes de séquences

Les principales informations contenues dans un sont les messages échangés entre les lignes de vie, présentés dans un ordre chronologique, le temps y est représenté explicitement par une dimension (la dimension verticale) et s'écoule de haut en bas.

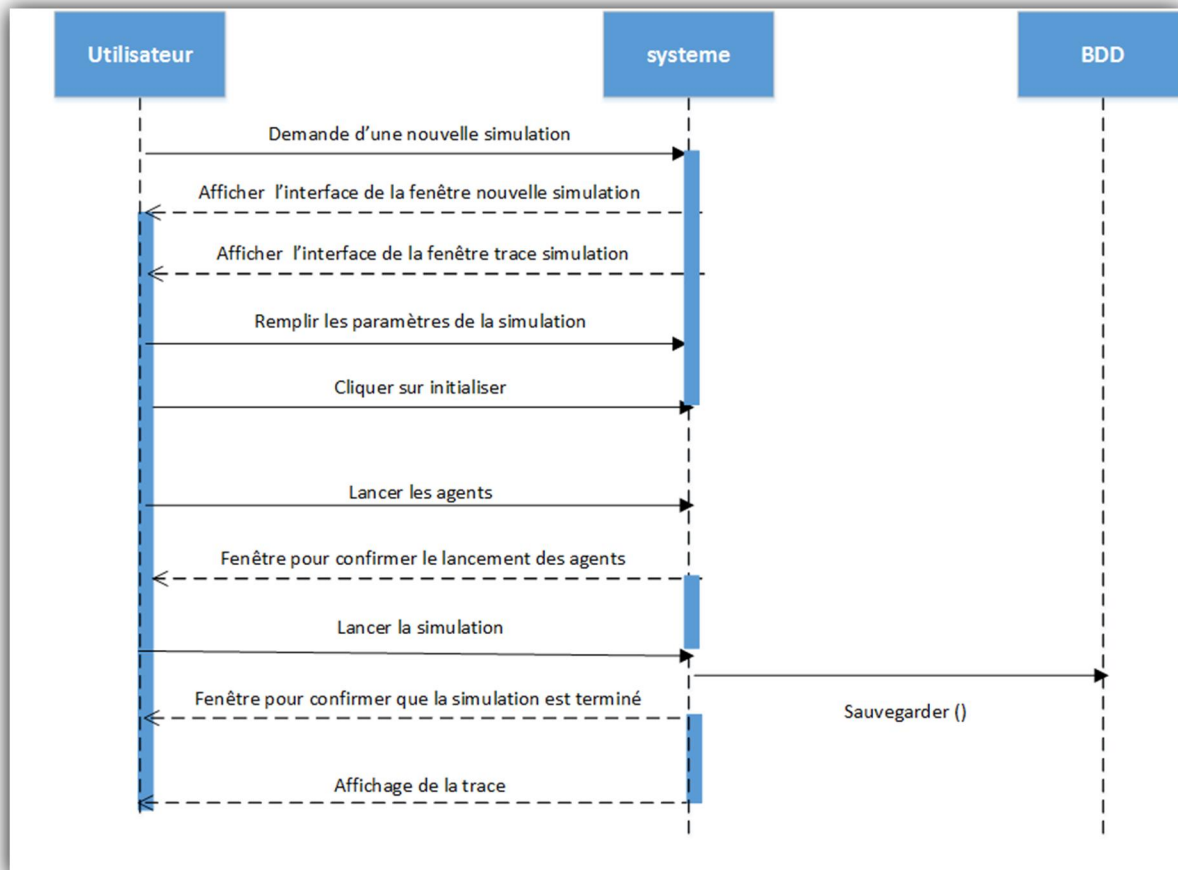


Figure 7: Diagramme de séquence- nouvelle simulation.

a) Nouvelle simulation

Le diagramme précédent montre les étapes à suivre pour la fonction nouvelle simulation :

1. l'utilisateur demande un accès aux fonctionnalités nouvelle simulation
2. le système affiche à l'utilisateur la fenêtre de la fonction nouvelle simulation
3. l'utilisateur remplit les champs nécessaires qui représentent les paramètres de la simulation
4. l'utilisateur clique sur le bouton initialiser

5. l'utilisateur clique sur le bouton lancer les agents
6. un pop-up apparaît pour informer que les agents sont bien lancés
7. l'utilisateur clique sur le bouton lancer la simulation, les paramètres de la simulation seront enregistrés dans la BDD
8. un pop-up apparaît pour informer que la simulation est bien terminée
9. le système affiche la trace de la simulation

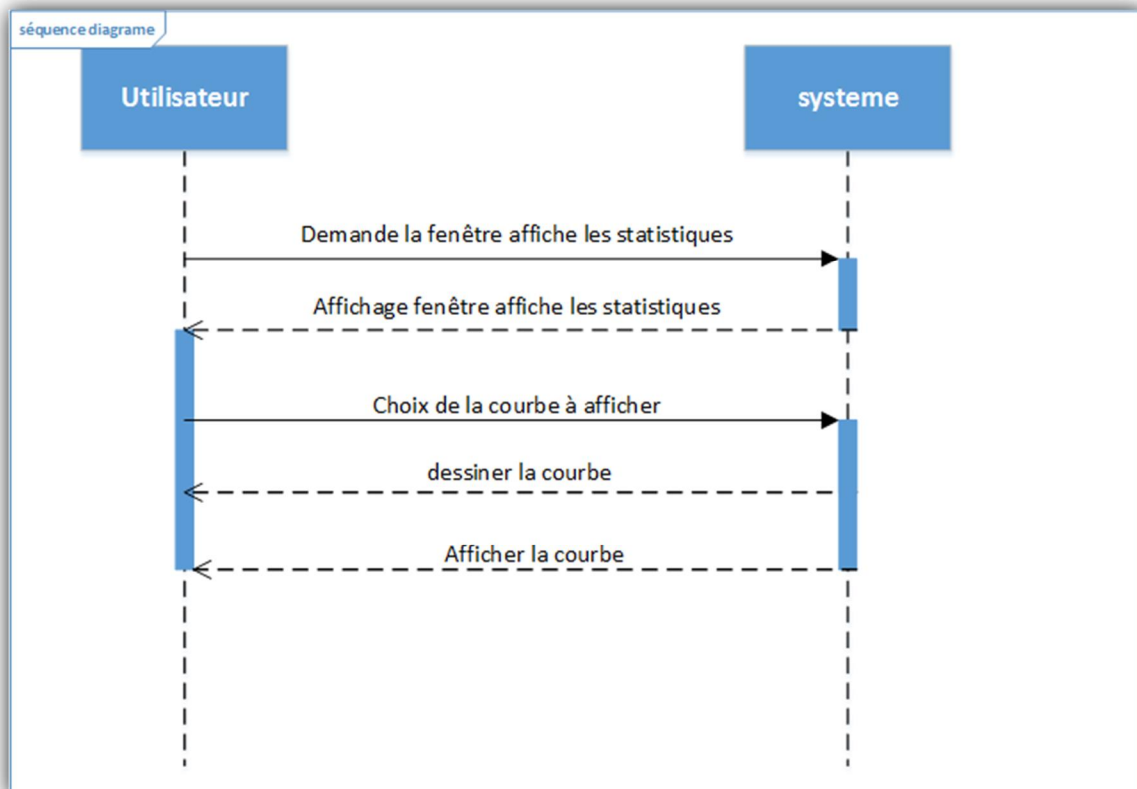


Figure 8 : Diagramme de séquence- afficher les statistiques.

b) Afficher les statistiques

- ❖ Le diagramme précédent montre les étapes à suivre pour la fonction « afficher les statistiques »:
 1. l'utilisateur demande la fenêtre afficher les statistiques
 2. le système affiche la fenêtre afficher les statistiques
 3. l'utilisateur choisi la courbe qu'il veut afficher
 4. le système dessine et affiche la courbe

c) La matrice de codage

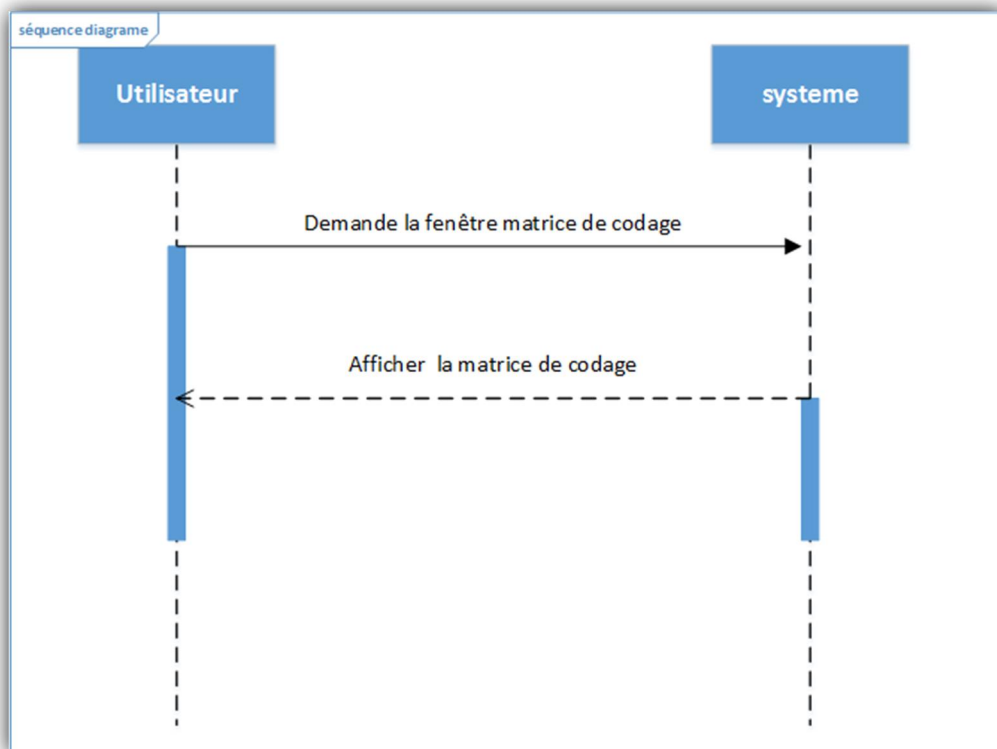


Figure 9: Diagramme de séquence- matrice de codage.

- ❖ Le diagramme précédent montre les étapes à suivre pour la fonction « matrice de codage »:
 1. l'utilisateur demande la fenêtre matrice de codage
 2. le système affiche la matrice de codage

5. Diagramme d'activité

- ❖ Les diagrammes d'activités permettent de mettre l'accent sur les traitements. Ils sont donc particulièrement adaptés à la modélisation du cheminement de flots de contrôle et de flots de données. Ils permettent ainsi de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation.

a) Nouvelle simulation

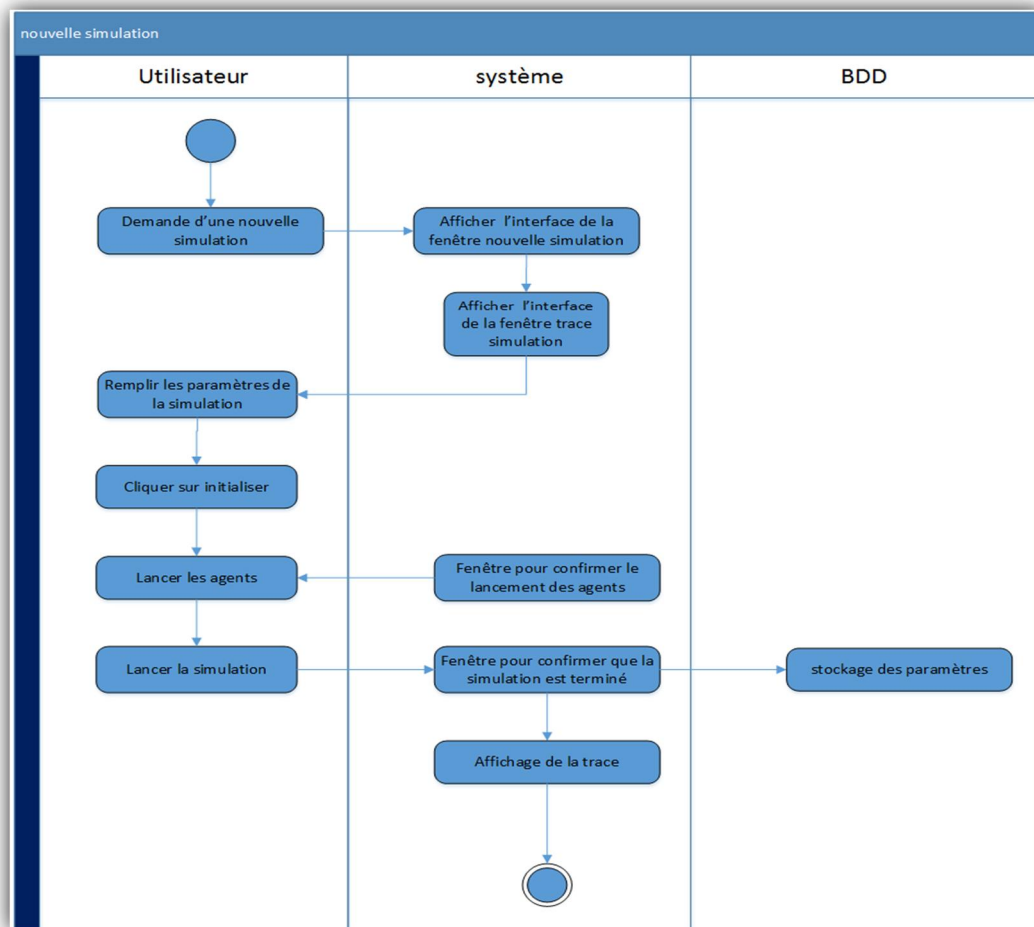


Figure 10: Diagramme d'activité-nouvelle simulation.

Le diagramme d'activité d'authentification nous permet de voir les comportements internes du système, il nous permet de voir qui est responsable de chaque activité.

Dans ce cas qui est « Nouvelle simulation » l'administrateur demande l'accès du système ce dernier lui affiche la fenêtre correspondante accompagné d'une autre fenêtre appelée trace simulation, il remplit les informations et confirme par le bouton initialiser.

L'utilisateur lance les agents un message de confirmation va apparaitre ensuite il lance la simulation, un autre pop-up message apparait pour informer que la simulation est terminée.

Les paramétré de ma simulation sont enregistrés dans la BDD et la trace est affichée

b) Afficher les statistiques

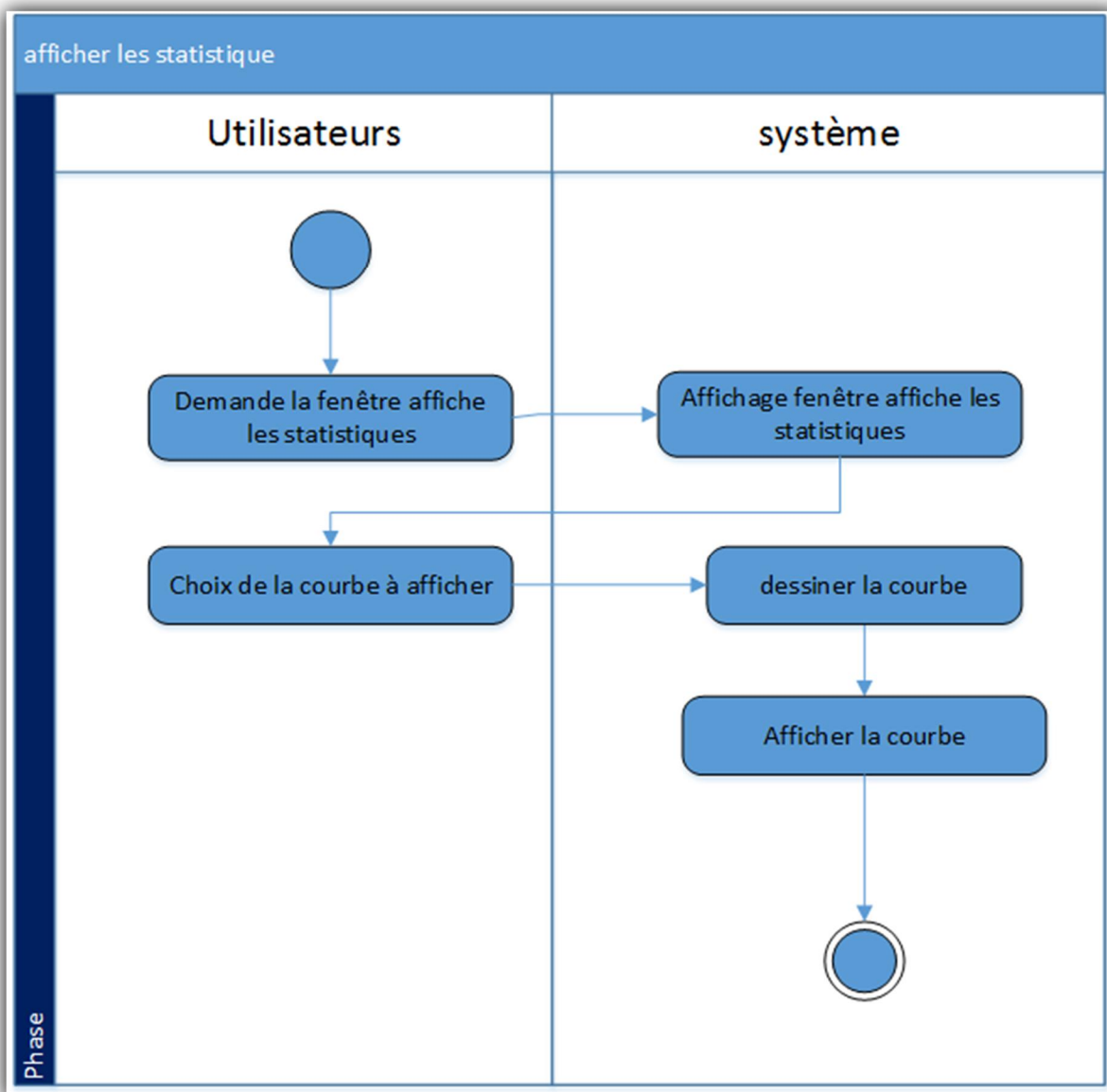


Figure 11: Diagramme d'activité - afficher les statistiques.

Dans ce cas qui est « l’affichage des statistiques » l’utilisateur demande l’accès du système ce dernier lui affiche la fenêtre correspondante il choisit la courbe à afficher et clique sur le bouton afficher la courbe, une autre fenêtre va apparaitre contenant la courbe.

c) Afficher la matrice

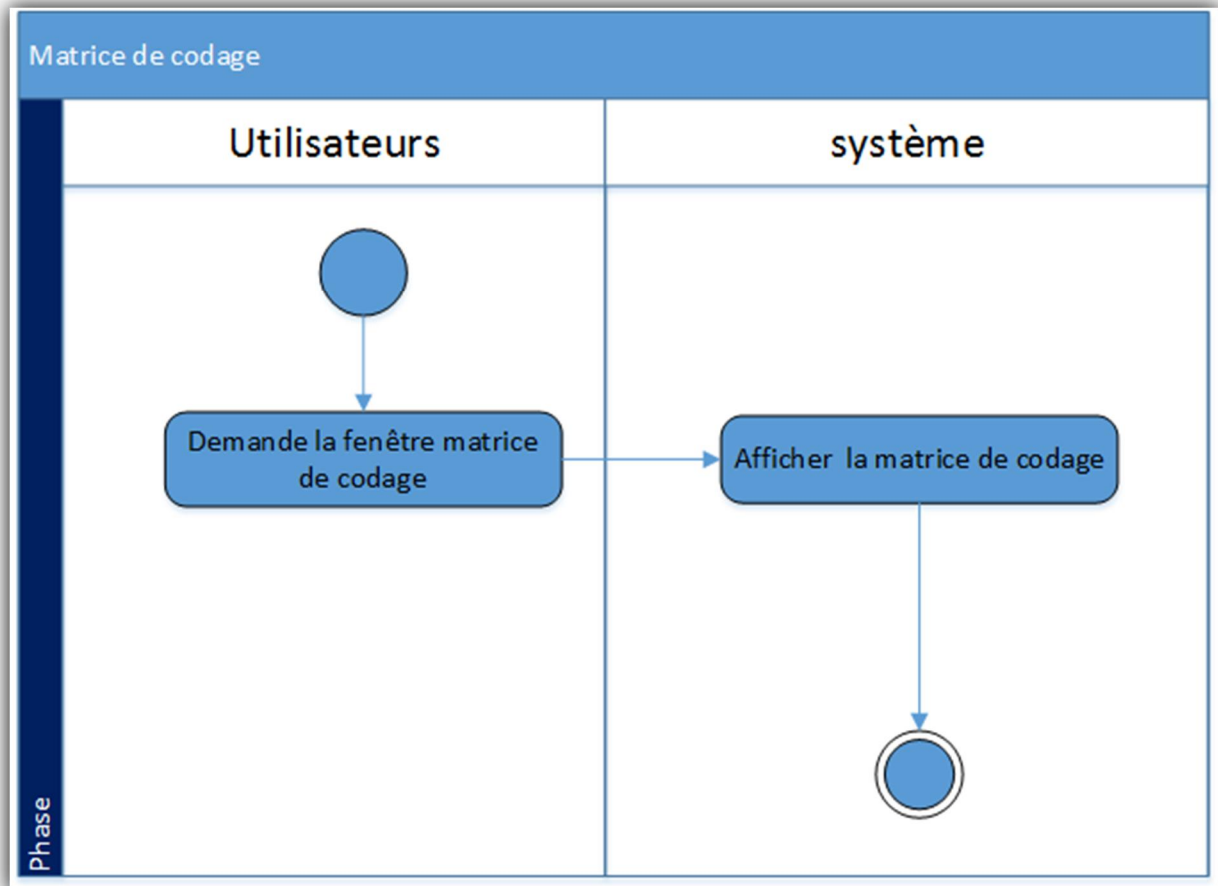


Figure 12: Diagramme d'activité-Matrice de codage.

Dans ce cas l’utilisateur demande l’accès à ma fenêtre matrice de codage qui se trouve dans le menu et l’onglet Résultats et le système l’affiche.

6. Diagramme de classe

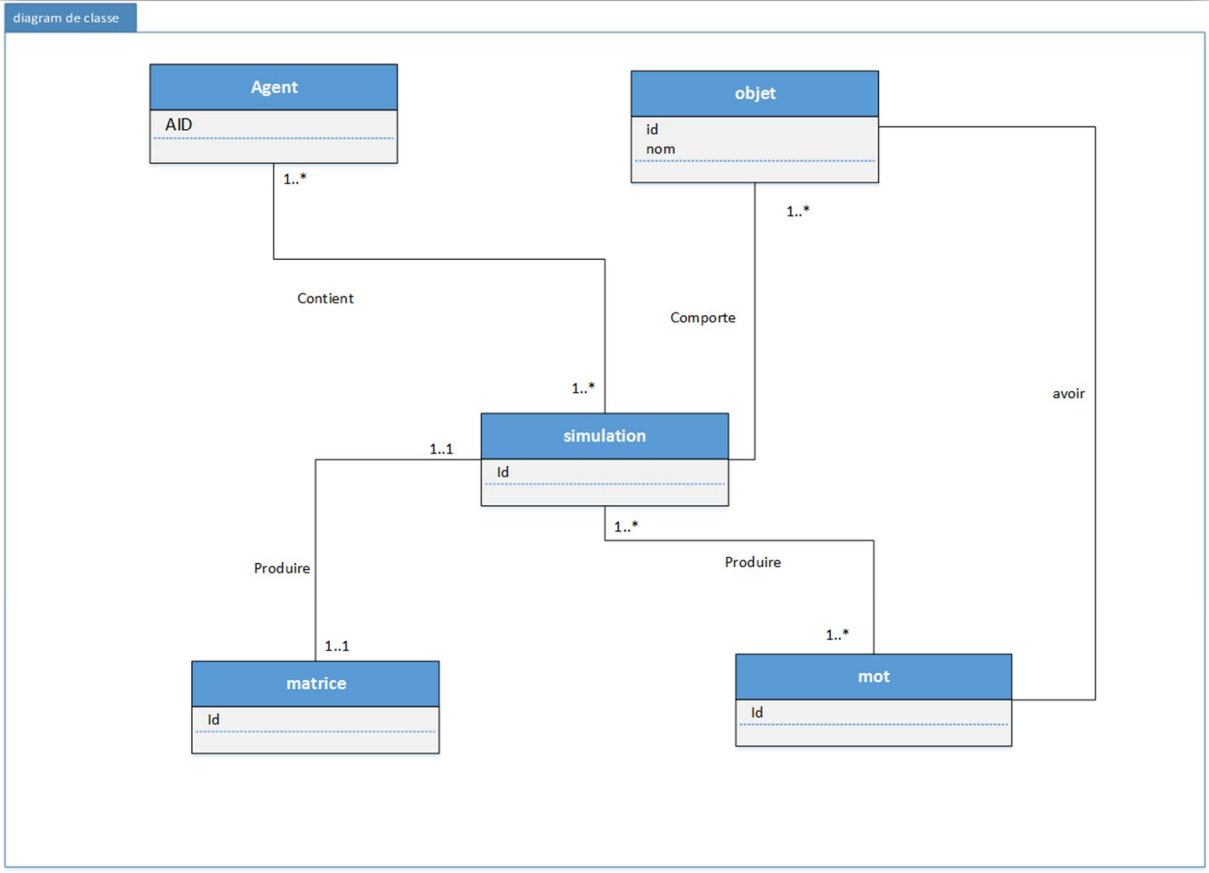


Figure 13: Diagramme de classe.

Chapitre IV:

Implémentation

Introduction

Après avoir faire une conception de notre système dans le chapitre précédent, il reste maintenant d'essayer à le réaliser en se basant sur cette conception. Ainsi dans ce chapitre nous présenterons les différentes étapes de sa réalisation, les différents outils exploités et les différentes interfaces de sa mise en œuvre.

1. Outils exploités

Afin de bien réaliser ce travail, plusieurs outils de développement peuvent nous aider.

- L'IDE (Environnement de Développement Intégré) ,Netbeans
- La plateforme JADE.

2. Définition de la plate forme JADE

JADE est une plateforme de programmation multi-agent implémentée en Java. Les agents qui tournent sous JADE communiquent via le langage Agent Communication Langage ou ACL[17].

2.1. FIPA (Fondation for Intellegent Physical Agents)

FIPA est une association internationale sans but lucratif, travaillant pour produire des spécifications pour des technologies d'agent générique.

Le premier document de la FIPA, et le FIPA97, établissant les règles normatives qui permettent à une société d'agents d'inter opérer. Les documents FIPA décrivent le modèle de référence d'une plate-forme multi-agents ou il identifient les rôles de quelques agents clés nécessaires pour la gestion de la plateforme, et spécifient le contenu du langage de gestion des agent et l'ontologie du langage. Trois rôles (agent) principaux sont identifiés dans une plate-forme d'agent[18] :

✓ **Le Système de gestion d'Agent (AMS- Agent Management System)**

Agent qui exerce le contrôle de supervision sur l'accès et l'usage de la plateforme ; il est responsable d'authentifier les agents résidents et de contrôler la les enregistrements.

✓ **Le Canal De communication (ACC- Agent Communication Canal)**

Agent qui fournit le chemin pour les interactions de base entre les agents dans et en d'hors de la plateforme ; c'est la méthode de communication implicite qui offre un service fiable et précis pour le routage des messages ; il (l'agent) doit aussi être compatible avec le protocole IIOP () pour assurer l'interopérabilité entre les différentes plateformes.

✓ **Le facilitateur d'Annuaire (DF- Directory Facilitator)**

Agent qui fournit un service de pages jaunes à la plateforme.

FIPA spécifie aussi le **Langage de Communication d'agents (ACL- Agent Communication Language)**. La communication entre agents ne se fait que par envoi de messages seulement.

FIPA ACL est le langage standard des messages et impose le codage, la sémantique et la pragmatique des messages. La norme n'impose pas de mécanisme spécifique pour le transport interne de messages. Plutôt, puisque les agents pourraient s'exécuter sur des plateformes différents et utilisent des technologies différentes 'interconnexion, FIPA spécifie que les messages transportés entre la différente plateforme devrait être codé sous forme textuelle[19].

2.2. L'interface graphique

La plate-forme offre une interface graphique utilisateur (GUID) pour la gestion à distance qui permet de contrôler et superviser les états des agents, par exemple arrêter et remettre en marche un agent. L'interface graphique permet aussi de créer et de commencer l'exécution d'un agent sur un hôte éloigné, à condition qu'un réceptacle d'agents s'exécute déjà sur cet hôte. L'interface elle-même a été implémentée comme un agent, appelé **RMA** (Remote Monitoring Agent). Toute la communication entre les agents et l'interface (GUI) et toute la communication entre cette interface et l'**AMS** est faite par **ACL** via une extension **ad hoc** de l'ontologie des agents de gestion FIPA[19].

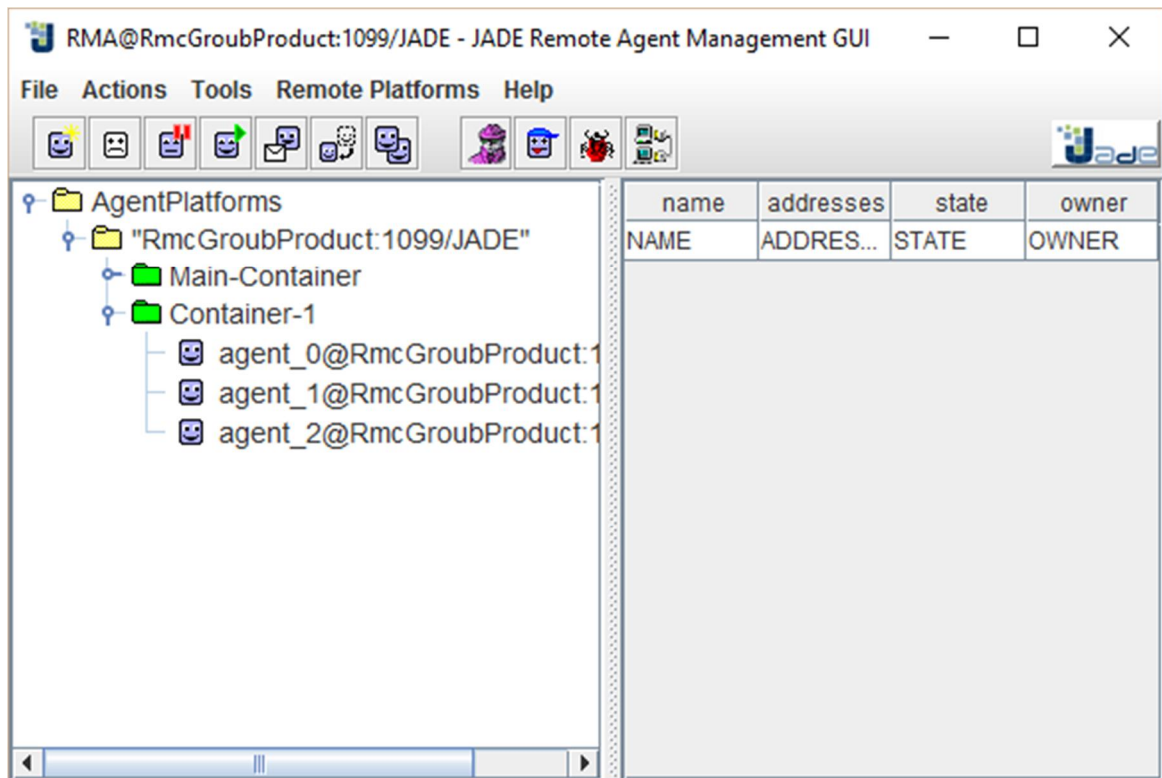


Figure 14 : L'interface utilisateur (GUI)

Parmi les outils graphique ou l'utilisateur de JADE peut déboguer c'est l'agent Sniffer. Il chasse l'échange de messages et donne une interface graphique pour afficher les échanges de messages entre les différents groupes d'agents on utilisant une notation proche d'UML[6].

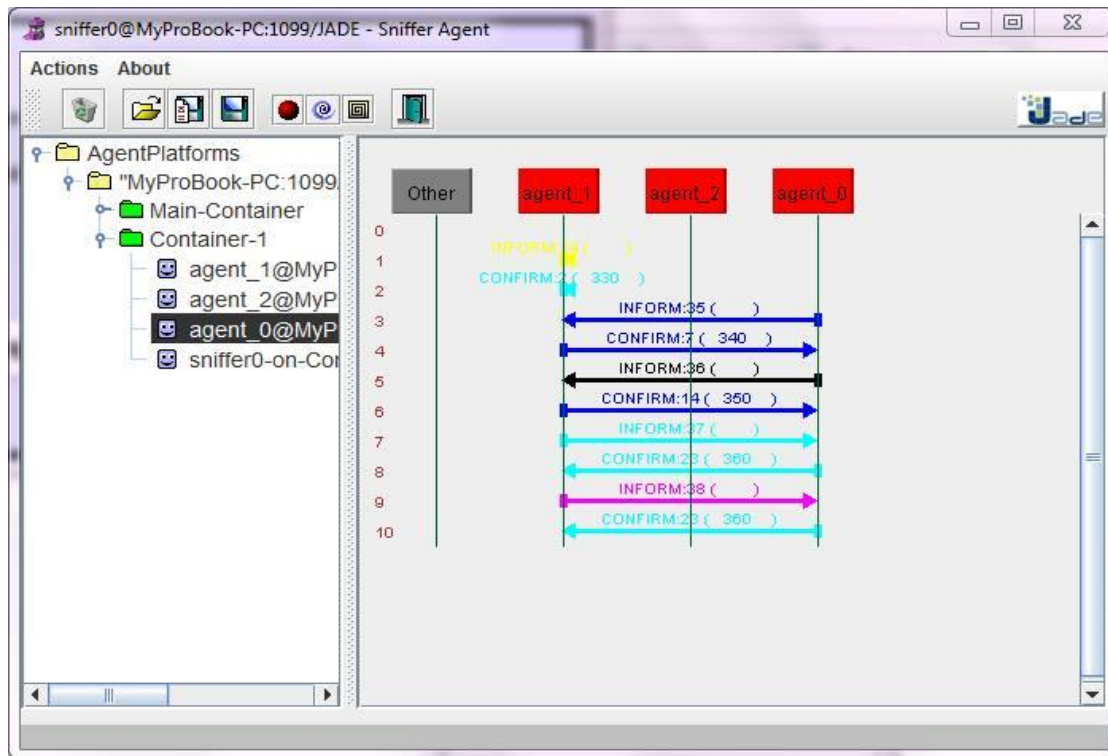


Figure 15 : Echange de message entre les agents (1, 2 ,3).

2.3. Pourquoi JADE [19]

Afin d'implémenter les agents de notre modèle, notre choix soit tombé sur la plate forme jade car elle répond mieux à nos attentes, et il a des caractéristiques importantes :

- il prend en compte les spécifications de la FIPA pour l'interopérabilité des Systèmes Multi Agents.
- il est entièrement écrit en java et propose une large gamme de bibliothèques de classes implémentant les fonctionnalités de base des agents (identification, comportements, communication, protocoles d'interaction, ontologies, mobilités etc...).
- il offre une interface graphique permettant la gestion des agents et facilitant leur débogage.
- il permet l'intégration efficace du moteur d'inférence Jess.
- il a la bonne documentation avec une liste d'expédition (mailing list) très actif.
- il est libre, et même open source (i.e le code source peut être facilement réutilisé pour construire de nouveaux systèmes).
- enfin, il est largement répandu pour le développement orienté agent, et il a été employé avec succès dans les différents milieux[17].

2.4. Un agent selon JADE [19]

- Conforme au standard FIPA.
- possède un Cycle de vie.
- Possède un ou plusieurs Comportements (Behaviours) qui définissent ses actions
- Communique et interagit avec les autres agents grace à des messages (ACLMessage)
- Rend des services

2.5. Langage de communication de la plate-forme JADE

Le langage de Communication de la plate-forme JADE est FIPA-ACL (Agent Communication language). La classe ACLMessage représente les messages qui peuvent être échangés par les agents. La communication de messages se fait en mode asynchrone. Lorsqu'un agent souhaite envoyer un message, il doit créer un nouvel objet ACLMessage, compléter ces champs avec des valeurs appropriées et enfin appeler la méthode send(). Lorsqu'un agent souhaite recevoir un message, il doit employer la méthode receive() ou la méthode blockingReceive()[19].

Un message est une instance de la classe ACLMessage appartenant au package `jade.lang.acl`

La définition d'un ACLMessage vérifie les spécifications FIPA.

Il est composé au minimum[20] :

- d'un performatif,
- d'un ensemble de destinataires,

Et contient de préférence :

- un contenu,
- un expéditeur,
- un protocole d'interaction,
- une id de conversation,...

Il peut spécifier un langage de contenu et/ou une ontologie.

Les valeurs des différents attributs d'un ACLMessage sont accessibles par les méthodes getXXX / setXXX / addXXX correspondantes

Performatif	getPerformative ()	setPerformative ()
Destinataire	addReceiver ()	removeReceiver ()
Contenu	getContent () ou getContentObject () ou getBytesSequenceContent ()	setContent () ou setContentObject () ou setByteSequenceContent ()
Expéditeur	getSender ()	setSender ()
Protocole	getProtocol ()	setProtocol ()
Id de conversation	getConversationId ()	setConversationId ()
Langage	getLanguage ()	setLanguage ()
Ontologie	getOntology ()	setOntology ()
...

Tableau 1: Structure d'un message ACL[20]

Les performatifs définissent le type d'action souhaité par l'agent qui expédie le message.

- Les performatifs sont des variables statiques de la classe ACLMessage [21]

Exemples :

- INFORM : envoi d'information/croyance,
- FAILURE : action non réalisable,
- REQUEST : demande d'information,
- SUBSCRIBE : souscription à une source d'information,...

Les performatifs disponibles sous Jade, sont ceux définis par la FIPA Le performatif est spécifié en argument du constructeur de l'ACLMessage

Exemple : `inform = new ACLMessage(ACLMessage.INFORM)`

3. Implémentation et simulation

Avant d'exécuter le programme on doit lancer le jade, une fenêtre JADE Remote Agent Management GUI va apparaître qui montre les échanges de communication entre les agents existants.

La figure suivante montre la première fenêtre qui apparaît lors du lancement du système et qui est la fenêtre principale.



Figure 16: Fenêtre principale

Dans l'onglet simulation on peut lancer une nouvelle simulation, les deux fenêtres montrées dans les figures 18 et 19 vont apparaître.

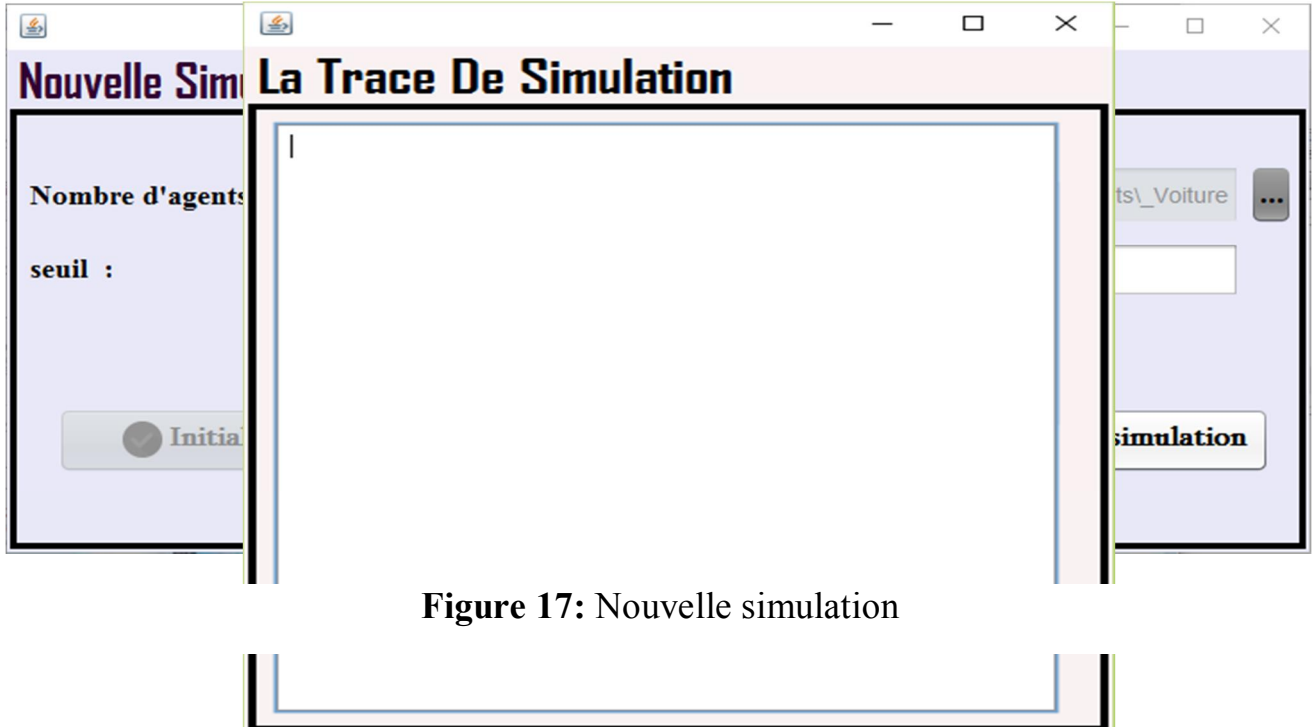


Figure 17: Nouvelle simulation

Figure 18:La trace de la simulation

L'utilisateur remplit les paramètres de la simulation et clique sur initialiser puis il lance les agents un message pop-up va apparaitre pour confirmer le lancement des agents ensuite il clique sur lancer la simulation un autre message pop-up indique si la simulation est terminée.

La trace de la simulation va apparaitre dans la figure.

En cours de l'exécution on peut consulter cet agent qui nous montre les interactions entre les agents de notre système.

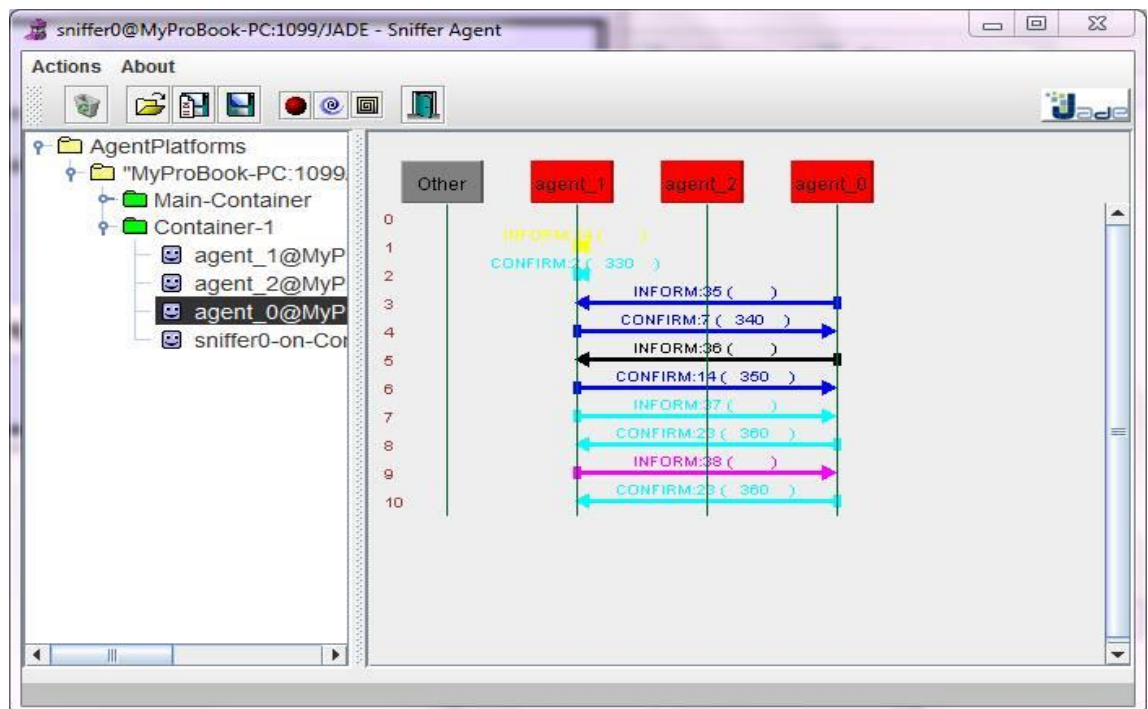


Figure 19: Les messages échangés entre les agents de système

Afin d'afficher la matrice de codage illustrer dans la figure l'utilisateur cliques sur l'onglet résultats ensuite Matrice de codage.

	kwcl	qqru	jmdd	oby	spmv	lgij	hswe
1288...	0	4	0	0	0	1	0
1339...	0	3	0	0	0	0	1
1c43f...	0	1	0	0	0	0	3
Accor...	0	2	2	0	0	2	0
band...	0	3	0	0	0	0	0
bugat...	0	4	0	1	0	0	0
ham...	0	2	0	2	0	1	0
hom...	0	4	0	0	0	2	0
indm...	0	2	0	4	0	1	0
lamb...	0	3	0	3	0	0	0
listin...	0	3	1	0	0	1	0
locati...	0	5	0	0	0	2	0
mygr...	0	3	0	3	0	0	0
peug...	0	4	0	0	0	0	0
PHO...	0	0	0	1	0	2	2
realis...	0	2	0	1	0	1	0
S1-El...	0	3	0	0	0	2	0
S1-S...	0	3	0	3	1	2	0
S1-Z...	0	2	0	0	0	0	0
S7-D...	0	3	0	0	0	1	0
sit lo...	0	1	2	0	0	0	0
télec...	0	2	0	1	0	2	0
vehic...	0	0	0	0	0	0	0
vehic...	0	1	0	2	0	2	0
vehic...	0	4	0	1	0	1	0
vehic...	0	4	1	3	0	2	0
vehic...	0	4	0	0	0	3	0
vehic...	0	4	1	0	0	0	0
vente...	0	3	0	0	0	1	0
visue...	0	3	0	2	0	0	0
voiture	0	4	0	2	0	1	0

Figure 20: Matrice de codage

Dans le même onglet Résultats, on peut afficher des différents graphes pour voir les statistiques des simulations comme montre la figure suivante.

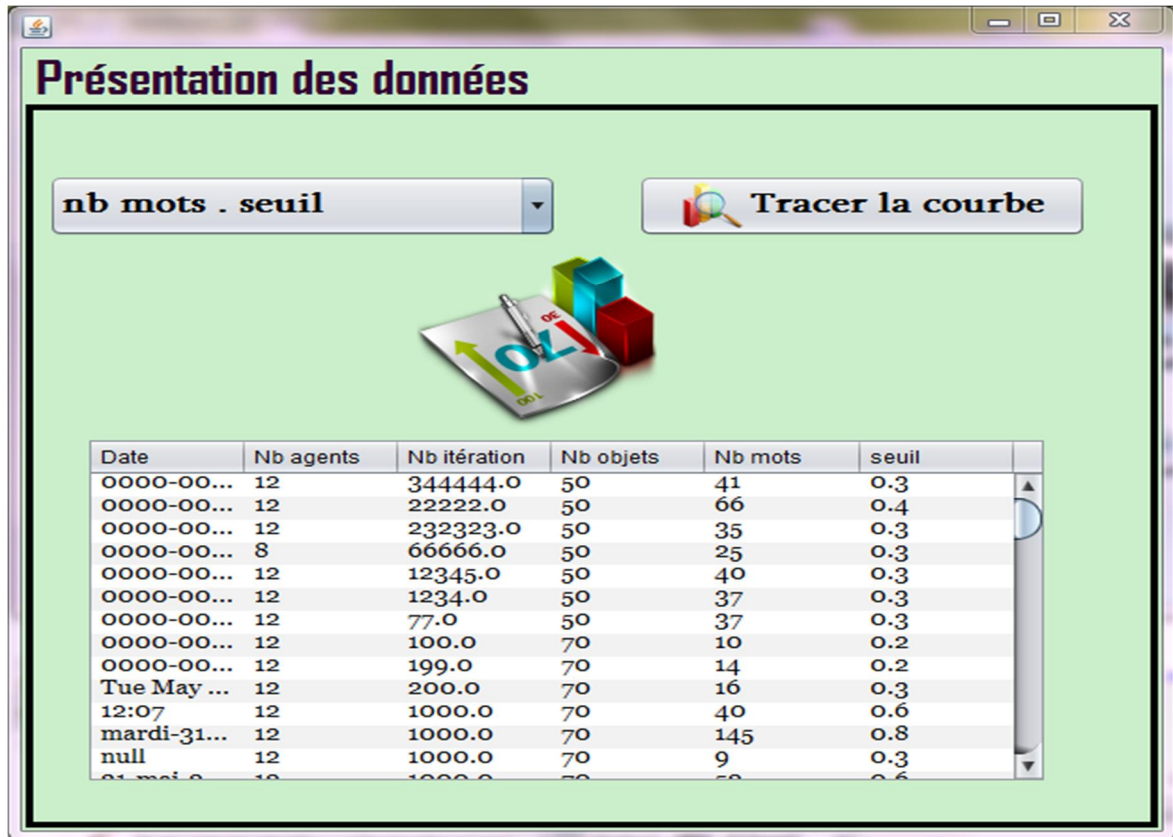


Figure 21: Présentation des statistiques

Il y'a plusieurs graphes à afficher, l'utilisateur fais son choix et clique sur Tracer

La courbe, la fenêtre montrée dans la figure va apparaitre.



Figure 22: La courbe

4. Résultats Expérimentaux

Nous désignons par cette étape l'application de notre système sur le jeu de données précédemment préparé et dont le résultat sera la définition des facteurs qui influent sur l'émergence du langage.

Notre objectif est d'évaluer la performance de notre programme, pour cela nous avons appliqués une série de tests sur une base de données.

Dans notre implémentation, nous utilisons la base de données WAMPSEVER.

Nous avons implémentés ce programme en Java en utilisant la plateforme jade sous le système Windows 7 sur un ordinateur équipé d'un microprocesseur i5 avec une vitesse de 2,4 GHz et une mémoire centrale de taille 4 G bytes.

Pour évaluer les résultats, nous utilisons des simulations en variant le seuil. Les résultats sont représenté sur l'axe des Y, Les résultats sont présentés graphiquement

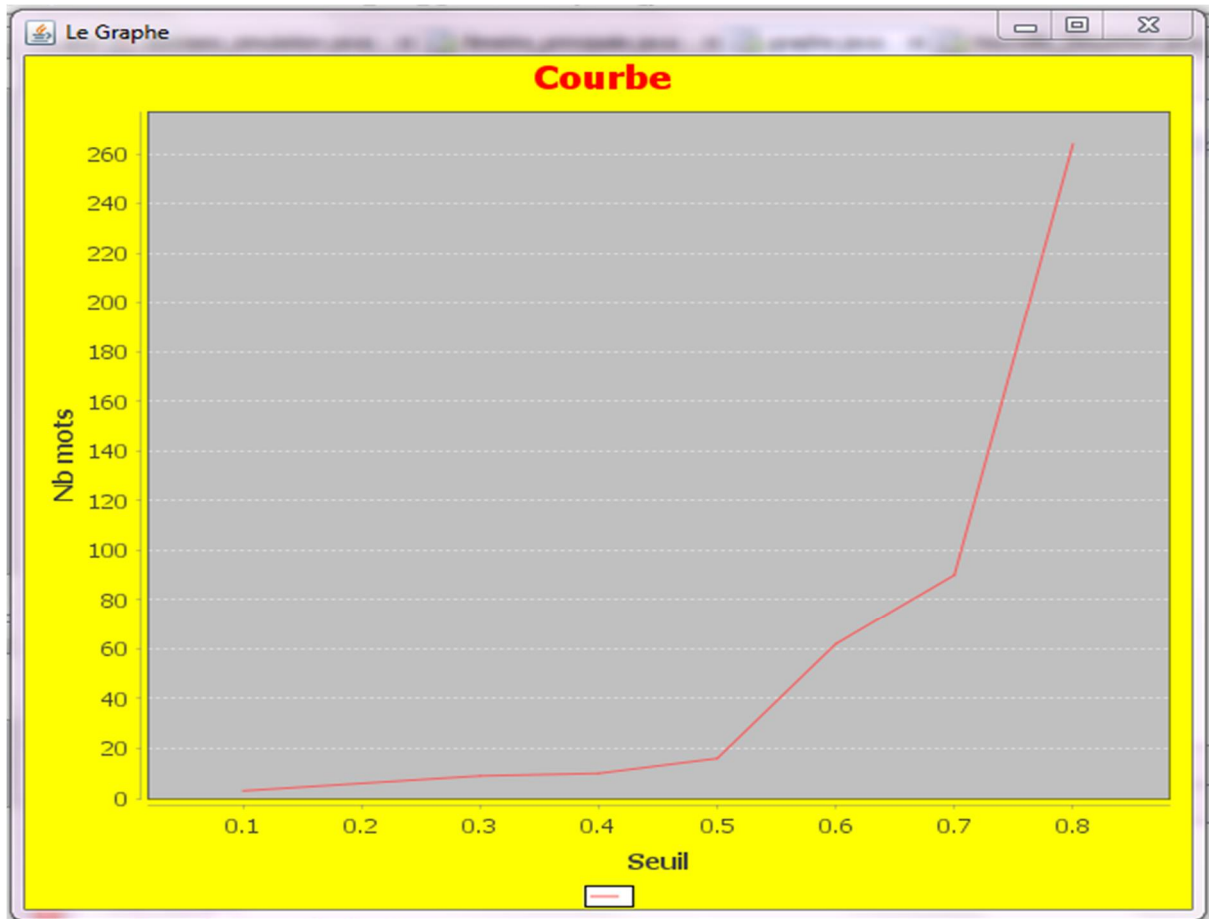


Figure 23 : graphe- nombre de mot en fonction du seuil

Selon les résultats présentés dans ce graphique, nous pouvons conclure que la variation du seuil influe sur le nombre de mots créés ce qui affecte directement le langage produit dans cette communauté d'agents.

Conclusion générale

Parmi les caractéristiques qui nous distinguent comme des êtres humains est sans doute notre faculté d'utiliser le langage dans la communication. Ce langage est utilisé pour différents buts comme l'expression des idées, la demande des informations et même pour penser symboliquement.

Pour étudier l'émergence du langage, on a inventé un ensemble de scénarios de communication permettant de modéliser un ensemble d'agents engagés dans une communication verbale, ces scénarios sont appelés les jeux du langage.

Nous avons montré comment un lexique pouvait émerger dans une population d'agents autonomes. Nous avons montré comment ce lexique pouvait être transmis culturellement de génération en génération, comment il se modifiait, s'affinait et devenait toujours plus adapté pour décrire l'environnement réel ou virtuel auquel les agents étaient confrontés. Nous avons montré comment la structure de ce lexique se régularisait et se simplifiait pour devenir plus facile à apprendre, plus facile à transmettre.

Finalement, ce que nous avons tiré de ce projet que les agents ont abouti un langage compréhensible par tous les agents existant dans l'environnement

Les références Bibliographies

1. Chaib-draa, I.J.e.B., Aperçu sur les systèmes multiagents 2002. p. 45.
2. Nicolle, A., Les systèmes multi-agents. 2002.
3. Développement normal du langage et ses troubles 2008.
4. Agents intelligents cour WEB interactif. 2002; Available from: <http://turing.cs.pub.ro/auf2/html/chapters/chapters.html>.
5. Coupé, C., De l'origine du langage a l'origine des langues 6 Janvier 2003.
6. Sapir, E., Le langage 1921.
7. Moulin, B., SUR LE LANGAGE, Avril 2009. p. 7.
8. Wékipedia. Langage humain. 6 février 2016; Available from: https://fr.wikipedia.org/wiki/Langage_humain.
9. Herslund, M., L'origine du langage – qu'en savons-nous?, 2000. p. 14.
10. wikipedia, Histoire évolutive des hominés. 27 mai 2016.
11. Fayard, Aux origines des langues et du langage, 2005.
12. M.Tiouidouine.S, introduction a la linguistique 2016.
13. moreau, m.l., l'acquisition du langage, ed. mardaga 1997.
14. Kaplan, F., L'émergence d'un lexique dans une population d'agents autonomes, 2000, PARIS 6. p. 284.
15. Sunier, P.-A. Formation en informatique (HES). Available from: <http://www.Lgl.isnetne.ch>.
16. memoireonline. Available from: <http://www.memoireonline.com>.
17. Plate-forme JADE: Java Agent Development Framework. 2000; Available from: <http://jade.cselt.it/>.
18. Bellifemine, F., JADE, A FIPA compliant agent framework.
19. FIPA: Foundation for Intelligent Physical Agents. Specifications. 1996.
20. Boissier, O., JADE Environnement pour la programmation multi-agent 2010.
21. Boissier, O., Systèmes Multi-Agents 2010. p. 23.