

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

**MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE**

UNIVERSITE 20 Aout 1955 de Skikda

DEPARTEMENT : Informatique

OPTION : GLAA



Mémoire de fin d'étude

En vue de l'obtention du

DIPLÔME DE Master

THEME :

**MODELISATION ET SIMULATION D'UN PROCEDE
INDUSTRIEL**

Par :

Kozzi Roudjina

Souadi Dounia zed

Encadreur :

Mr. BENOUDINA

Année Universitaire : 2021/2022

Remerciement

*NOUS remerciant tout d'abord ALLAH qui nous aide et nous donne la force et
La volonté tout au long de nos travail de fin d'étude car sans lui rien
N'aurait pu être.*

*Un grand merci à nous encadreur de mémoire, Mr Benoudina
Pour ses précieux Conseils et son Orientation. Ainsi que
Son support, sa patience et ces efforts durant ce modeste travail que dieu le
Bénisse.*

*Nous remerciant également les membres du jury d'avoir accepté d'évaluer nous
travail.*

*Nous remerciant tous ceux qui ont contribué de près ou de loin à la
Réalisation de ce travail.*

*Enfin, Un grand merci à tous ceux qui nous ont accompagné au
Cours de toutes ces années, famille et amies.*

Merci...

Dédicace

Je dédie ce modeste travail A ceux qu'ont faits de moi
une femme mes très chers

Parents « mon père " et " ma mère ".

A mes sœurs SAMAH et IMANE

A Mes frères FATEH et ABDELBASSET ET
ABDELKADOUS

A ma chère amie :

RODJINA

Dounia Zed

Dédicace

J'ai l'immense honneur de dédier ce travail de mémoire
à
Mes chers parents qui m'ont toujours soutenue tout au
long de mes études

A mes très chères frères et surtout mon frère le docteur
R.KOZZI, que dieu les protèges et les gardes

Une spéciale dédicace pour ma douce amie qui compte
déjà pour moi. A toi << Dndn >>.

RODJINA

RÉSUMÉ :

La simulation à base d'agents ou encore Multi-Agents s'impose de plus en plus pour son efficacité et sa simplicité. En effet, ce type de simulation offre, au contraire des autres types, la possibilité de représenter directement les entités simulées, leurs comportements et leurs interactions sans avoir recours aux équations mathématiques. Ce travail est une contribution dans ce sens, le but est de construire un simulateur à base d'agents d'un processus industriel.

Ce processus comme presque tous les processus industriels présentent le problème de la complexité, qui peut être maîtrisé avec l'approche Multi-Agents. Nous avons utilisé l'outil logiciel MADKit comme plate-forme de développement et d'exécution de notre simulateur, avec des résultats satisfaisants.

MOTS-CLÉS : Simulation, Systèmes Multi-Agents (SMA), MADKit, Processus industriel

ABSTRACT:

Simulation based on agents or multi agents is more and more essential for its effectiveness and its simplicity. Indeed, this type of simulation offers, on the contrary of other types, the possibility of directly representing the simulated entities, their behaviors and their interactions without having recourse to the mathematical equations. This work is a contribution in this way; the goal is to build a simulator based on agents of an industrial process. This process as almost all the industrial processes present the problem of the complexity, which can be controlled with the approach multi-Agents. We used the software tool MADKit t as platform of development and execution of our simulator, with satisfactory results

Table de Matières

Chapitre1 : Modélisation et la simulation informatique

Introduction Générale

1.Simulation	4
1.1Introduction	4
2.Notions de système.....	5
2.1 Définition de système.....	5
2.2 Types de systèmes.....	5
2.2.1 Système déterministe.....	5
2.2.2 Système probabiliste(stochastique).....	5
2.2.3 Système continu.....	5
2.2.4 Système discret(discontinu).....	5
2.2.5 Système mixte.....	5
2.3Categorie de système.....	6
3 Notion de modèle.....	6
3.1 Définition.....	6
3.2 Types de modèles.....	6
3.2.1 Les modèles iconiques ou physiques	6
3.2.2 Les modèles symboliques ou abstraite.....	7
3.3 Processus de modélisation.....	7
3.4 Les outils de modélisation informatique.....	8
3.4.1 Les files d'attente.....	8
3.4.2 Les réseaux de pétri.....	9
3.4.3 Modélisation orientée objet	9
3.4.4 Modélisation orientée agent	10
4.Notion de simulation	10
4.1 Définition de la simulation	11
4.2 La simulation informatique.....	11
4.3 Catégories de simulation informatique.....	11

4.3.1 La simulation continue	12
4.3.2 La simulation discrète.....	12
4.3.3 La simulation par agents.....	12
4.4 Domaine d'application de la simulation	13

Conclusion

Chapitre 2 : Les systèmes Multi-Agents

1.Introduction

2.L'intelligence artificielle distribuée (IAD).....	15
2.1 Problématique de l'IAD.....	15
3.Definition d'un agent	16
3.1 Caractéristique d'un agent.....	17
3.2 Agent et objet.....	18
3.3Type d'agent.....	19
4.Les systèmes Multi-agents.....	20
4.1 Architecture d'un SMA.....	21
4.1.1 SMA à contrôle distribué.....	22
4.1.2 SMA à contrôle centralisé.....	23
4.2 Interaction entre agents.....	24
4.2.1 La communication.....	25
4.2.2 La coopération.....	25
4.2.3 La coordination.....	25
4.2.4 La négociation.....	25
4.2.5 Les protocoles d'interaction.....	25
4.3 Les domaines d'application des systèmes multi-agents.....	26
5 Les plateformes multi-agents.....	26
5.1 Jack	26
5.2 Zeus.....	27
5.3 Agent Builder	27
5.4 MADKit.....	27

5.5 JADE.....	27
6.Les méthodes de conception des systèmes multi-agents.....	28
6.1 Mase.....	28
6.2 GALA.....	28
6.3 AALAADIN.....	28
6.4 ADELFE.....	29
6.5 La méthodologie ingénias.....	29
6.6 La méthodologie Tropos.....	29

Conclusion

Chapitre 3 : Analyse et conception

Partie 1 :

1.Introduction

2.Presentation du complexe GL1K.....	33
3.Descreption générale de l'activité.....	34
4.Description globale des unités de liquéfaction du GN.....	34
4.1 Section (traitment du gaz)	35
4.2 Section(liquéfaction).....	35
4.3 Section (Production de vapeur)	35
4.4 Section (Fractionnement).....	36
5 Description du système de déshydratation.....	38
5.1 Pres refroidissement et séparations.....	39
5.2 Déshydratation.....	40
5.3 Protection anti-incendie	41
6.Attachements.....	42
6.1. Attachement 1.....	42
6.2 Attachement 2.....	42
7.Attachements.....	43
7.1 Attachement : liste des équipements.....	43
7.2 Attachement : Consommation des utilités.....	43

8.Attachements.....	44
8.1 Attachement : Liste des équipements.....	44
8.2 Attachement : Consommation des utilités.....	44
9.Attachements.....	45
9.1 Attachements 1 : liste des équipements.....	45
9.2 Attachement : consommation des utilités.....	46

Partie 2

1.Introduction

2.Le modèle AALAADIN.....	47
2.1 Le niveau descriptif.....	47
2.1.1 Agent.....	48
2.1.2 Groupe.....	49
2.1.3 Rôle.....	49
2.2 Le niveau Méthodologique.....	49
2.2.1 Structure de groupe.....	49
2.2.2 Structure organisationnelle.....	49
3.2 Les groupes et les roles.....	49
3.2.1 Le groupe perturbation.....	49
3.2.1.1 Agent GUS.....	49
3.2.1.2 Agent APM.....	49
3.2.2 Le groupe Régénération.....	50
3.2.2.1 Agent APM.....	50
3.2.2.2 Agent fv-107.....	50
3.2.2.3 Agent PV-102.....	50
3.2.2.4 Agent LV-101.....	50
3.2.2.5 Agent LV-116.....	50
3.2.2.6 Agent LV-105.....	51

3.2.3 Le groupe Lavage.....	52
3.2.3.1 Agent FV-103.....	52
3.2.3.2 Agent FV-101.....	52
3.2.3.3 Agent PV-101.....	52
3.2.3.4 Agent FV-109.....	52
3.2.3.5 Agent APM	52

Conclusion

Chapitre 4 : l'implémentation

1.Introduction

2.Outils de programmation.....	55
3.Presentation des outils.....	55
3.1 Le Language de programmation java.....	56
3.2 JDK	56
4.La plateforme MADKIT.....	56
4.1 Caractéristique.....	57
4.2 Motivation.....	57
4.3 Architecture de la plateforme MADKIT	57
4.4 Programmer les agents	58
4.4.1 Cycle de vie.....	58
4.4.2 Communication.....	59
4.4.3 Gestion des groupes et rôles.....	59
5.Implementation du système	60
5.1 Présentation des interfaces graphique.....	60

Conclusion

Table d'abréviations :

Abréviation	Signification
GL1/K	Complexe de Liquéfaction du Gaz Naturel SKIKDA
CP1/K	Complexe Pétrochimique SKIKDA
GN	Gaz Naturel
GNL	Gaz Naturel liquéfié
GPL	Unité de séparation Propane/ Butane
SMA	Système Multi-agents
IAD	Intelligence Artificiel Distribué
TIC	Température Indicator Controller
PC	Partie Commande
SDC	Salle de contrôle/commande
DCS	Distributed Control Système

Figure	Nom de figure	Numéro de page
Figure (1.1)	Un système	5
Figure (1.2)	Processus de Modélisation	8
Figure (1.3.)	Représentation graphique d'une file d'attente primaire	9
Figure (1.4)	Typologie de la modélisation graphique des réseaux de Pétri	10
Figure (1.5)	La simulation informatique	12
Figure (2.1)	Agent réactif	20
Figure (2.2)	Agent cognitif	21
Figure (2.3)	Modèle d'agent hybride	22
Figure (2.4)	SMA à contrôle distribué	24
Figure (2.5)	SMA à contrôle centralisé	25
Figure (3.1)	GL1K	34
Figure (3.2)	Les principes étapes de la liquéfaction du gaz naturel	37
Figure (3.3)	Schéma fonctionnelle	38
Figure (3.4)	Les sécheurs	39
Figure (3.5)	La déshydratation du gaz naturel	39

Figure (3.6)	Séparateur 13MD02A	40
Figure (3.7)	Schéma Système de déshydratation	41
Figure (3.8)	Déshydratation et filtration	42
Figure (3.9)	Concepts (Agent/Groupe/Rôle) du modèle Aalaadin	49
Figure (10)	Structure générale d'un agent dans le modèle Aalaadin	49
Figure (3.12)	Groupe perturbation	51
Figure (3.13)	Groupe Régénération	52
Figure (3.14)	Groupe lavage	53
Figure (3.15)	Le modèle organisationnel du simulateur	54
Figure (4.1)	Logo de java	56
Figure (4.1)	Architecture	60
Figure (4.2)	Desktop MADKIT	62
Figure (4.3)	Interface de la simulation (Groupe régénération)	62
Figure (4.4)	Interface de la simulation (Groupe lavage)	63
Figure (4.5)	Interface de la simulation (Groupe perturbation)	63
Figure (4.6)	L'interface du simulateur Dans état d'arrêt	64

Liste des tableaux

Tableau (1. 1)	Domaine d'application de la simulation
Tableau (2.1)	Différences entre agents cognitifs et agents réactifs
Tableau (3.1)	Equipements anti-incendie du système de déshydratation
Tableau (3.2)	Des équipements du système de déshydratation
Tableau (3.3)	Moteurs électriques du système de déshydratation
Tableau (3.4)	Des équipements du système de déshydratation
Tableau (3.5)	Moteurs électriques du système de déshydratation.
Tableau (3.6)	Des équipements du système de déshydratation
Tableau (3.7)	Moteurs électriques du système de déshydratation
Tableau (3.8)	Description fonctionnelle du simulateur
Tableau (4.1)	Gestion des groupes des rôles

Introduction générale

L'évolution rapide des technologies industrielles, le développement l'automatisation et l'introduction de l'informatique ont provoqué des changements importants dans le contenu et les méthodes de travail humain.

Dans les salles de contrôle de systèmes industriels complexes et ceci dans des domaines aussi variés que l'énergie ou la chimie, l'activité confiée à l'opérateur humain s'est déplacée d'une activité de contrôle manuel vers une activité essentiellement automatique de supervision et de contrôle.

L'activité de supervision et de contrôle est complexe en raison des caractéristiques dynamiques du processus ainsi que le plus grand nombre de composants qui le composent et par conséquent ça va engendrer plus d'interactions et plus de complexité. À la différence des situations statiques où l'environnement ne change que sous l'effet des actions de l'opérateur, la dynamique d'un environnement évolutif impose des situations dynamiques que l'opérateur ne contrôle que partiellement.

La modélisation et la simulation d'un système réel permettent de manipuler, d'observer et d'améliorer la compréhension des phénomènes complexes. Ces derniers sont étudiés à différentes échelles comme pour la simulation des changements climatiques, des phénomènes physiques et des systèmes de trafic urbain...etc. La plupart des phénomènes modélisés atteignent aujourd'hui des complexités et des degrés de finesse élevés qui imposent l'utilisation de modèles et d'outils informatiques de plus en plus performants et flexibles.

La simulation informatique est l'un des outils permettant de simuler des phénomènes réels, elle désigne un procédé selon lequel on exécute un programme informatique sur un ordinateur qui peut représenter son comportement, son interaction et tout se qu'il faire en réalité, elle peut être utilisé comme moyen qui permet de maîtriser la complexité des systèmes.

Notre travail se situe dans ce cadre, il permet de présenter la conception et l'implémentation d'un simulateur d'un processus industriel, son contrôle et son suivi, ce processus concerne le système de déshydratation (unité13) au niveau du complexe de liquéfaction du gaz naturel GL1K groupe SONATRACH SKIKDA.

Grace aux systèmes multi-agent, on peut modéliser des systèmes réels dans lesquels des comportements très complexes émergent d'interactions relativement simples et locales entre de nombreux individus différents, donc un système multi-agents est plus naturel pour décrire et simuler un système composé d'entités, il sera particulièrement bien adapté à la description d'un système du point de vue de l'activité de ses constituants, c'est-à-dire lorsque le comportement des individus est complexe (donc difficile à décrire avec des équations).

La simulation multi-agents nous apparaît être la démarche la plus adéquate à la modélisation de ce type de système (les systèmes complexes tels que le procédé du système de déshydratation du gaz). La simulation multi-agents est un domaine de recherche récent, qui se base sur la notion d'agent.

Ainsi, notre mémoire se compose de six chapitres organisés de la façon suivante :

Modélisation et simulation des systèmes : Ce chapitre constitue une introduction à notre travail par la présentation des notions sur la modélisation et la simulation d'une façon générale.

Les systèmes Multi-Agents : Ce chapitre contient les systèmes Multi-Agent et leur caractéristique.

Le processus de déshydratation du gaz : dans ce chapitre nous avons exposé les différents composants entrant dans le processus étudié, et son fonctionnement.

La conception : Dans ce chapitre, nous présentons notre modèle par l'utilisation des principes de la méthode AGR.

L'implémentation : Ce dernier chapitre présente la réalisation et l'implémentation du simulateur modèle par l'utilisation de la plate-forme MADKIT

CHAPITRE I :

MODILISATION ET

SIMULATION INFORMATIQUE

1 Simulation

1.1 Introduction

La simulation est aujourd'hui largement reconnue comme une technique puissante pour l'analyse et la conception des systèmes. Elle peut être appliquée dans divers domaines tels que l'analyse des systèmes de services (Banques, téléphonie, ...), les systèmes de production (ou de fabrication), les systèmes naturels (biologiques, écologiques, ...), les systèmes informatiques (d'exploitation, de communications, etc.)

L'étude scientifique d'un système nous amène dans la plupart des cas à faire un ensemble de suppositions sur le fonctionnement du système. Ces suppositions, prennent habituellement la forme de relations mathématiques ou logiques, constituent un modèle qui est utilisé pour essayer de comprendre comment le comportement du système étudié. Si les relations qui composent le modèle sont assez simples il peut être possible d'utiliser des méthodes mathématiques (telle que l'algèbre, la théorie des probabilités) pour obtenir des réponses exactes aux questions qui nous intéressent. Dans une simulation on utilise l'ordinateur pour évaluer numériquement un modèle et des données sont collectées dans le but d'estimer les caractéristiques souhaitées du modèle.

Ce chapitre constitue une introduction à notre travail par la présentation de quelques notions, qui nous semblent essentielles, sur la modélisation et la simulation d'une façon générale.

2 NOTIONS DE SYSTEME :

Les trois notions de système, Modèle et Simulation sont intrinsèquement liées. Nous allons définir de manière plus ou moins précise chacune d'elles avant de présenter les concepts de base des techniques de simulation.

2.1 Définition de système :

Le mot "système" couvre un champ d'application immense et de nombreuses définitions lui ont été attribuées dans la littérature. En voici une liste non exhaustive :

Ensemble de composants reliés entre eux

Ensemble organisé d'éléments fonctionnel

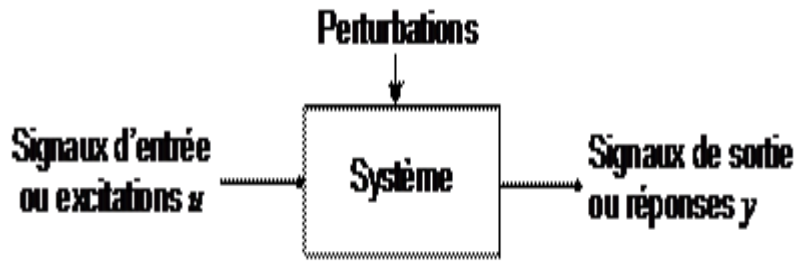


Figure 1.1 : Un système

Combinaison de parties qui se coordonnent, pour concourir à un résultat de manière à former un ensemble (Larousse)

Combinaison d'hommes, de machines, de matériaux et d'informations destinée à satisfaire un objectif donné.

2.2 Types de systèmes :

2.2.1 Système déterministe

Il fonctionne de manière prévisible, l'interaction entre les différentes parties est connue avec exactitude. Si l'on possède une description de l'état du système à un moment donné, le prochain état du système peut être donné exactement et sans erreur.

2.2.2 Système probabiliste (stochastique)

Ce modèle est basé sur l'élaboration d'équations précises, le système est alors totalement prévisible.

2.2.3 Système continu

Les changements de l'état du système se font permanences avec le temps (dépend d'une variable liée au temps).

2.2.4 Système discret (discontinu)

Le système est caractérisé par des événements qui surviennent à des instants non fixes et causent un changement de l'état jusqu'au prochain événement.

2.2.5 Système mixte

La séparation entre système continu et discret est en quelque sorte artificielle, car en réalité la plupart des systèmes possèdent des composantes continues et discrètes à la fois. De tels systèmes sont dits mixtes.

2.3 Catégorie de système :

On distingue deux grands ensembles de systèmes : les systèmes artificiels et les systèmes naturels.

Exemples de systèmes naturels

- Système nerveux,
- Système solaire

Exemples de systèmes artificiels

Système industriel,

- Système informatique.

3 Notion de modèle :

3.1 Définition :

D'une façon générale un modèle est une représentation d'un système réel (physique, économique, humain, etc.) réalisé dans le but de mieux étudier ce système et expliquer certains de son comportement. de façon un peu plus particulière.

Parfois, il est impossible d'étudier le système directement du fait qu'il soit inaccessible (système solaire), trop coûteux, il change trop rapidement (tir nucléaire) ou lentement (mouvement d'une comète), dans ce cas, l'étude est faite sur un modèle, c'est-à-dire un deuxième système original et aussi parfait que l'étude l'exige.

3.2 Types de modèles :

On distingue plusieurs catégories de modèles On selon la nature du système étudié :

- Iconiques ou physiques.
- Symboliques ou abstraits.

3.2.1 Les modèles iconiques ou physiques

Sont une représentation du système réel par une maquette à l'échelle réduite qui est encore très utilisée par exemple (construction automobile, architecture...).

3.2.2 Les modèles symboliques ou abstraits

Sont une représentation mathématique et logique d'un problème pouvant manipuler de façon expérimentale sur un ordinateur. Exemple : le modèle mathématique du système solaire. L'ensemble des symboles et des équations mathématiques décrivent le comportement dynamique du soleil et de ses planètes à un niveau purement abstrait. Seuls les deux derniers modèles concernent la simulation.

3.3 Processus de modélisation

Le processus de construction d'un modèle de simulation peut être schématisé comme suit :

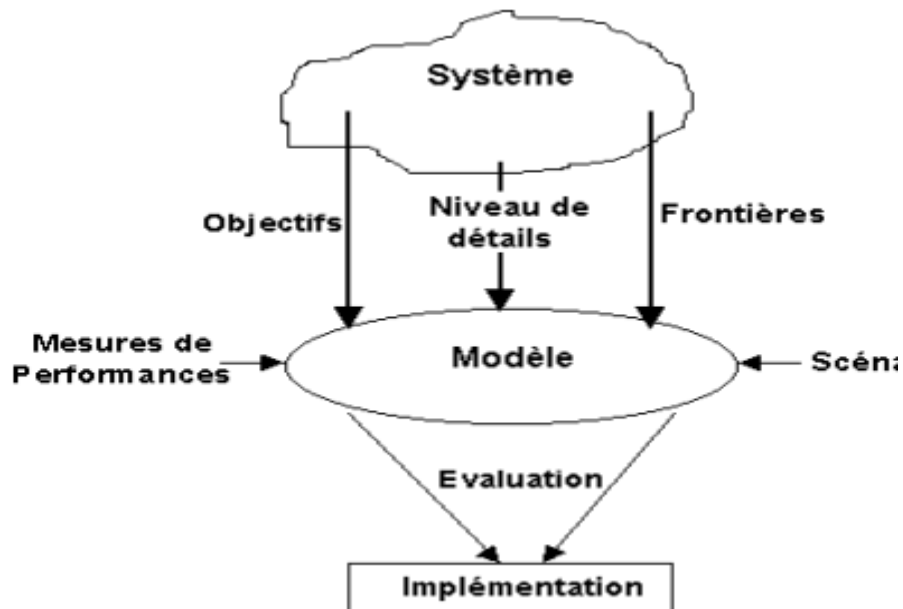


Figure 1.2 : Processus de Modélisation

La modélisation informatique des données est en réalité un processus de description de la structure, des associations, des relations et des impératifs liés à des datas disponibles. Elle permet de fixer des normes, tout en codant des modèles de gestion des données dans une organisation.

Le processus de modélisation vise à obtenir une solution acceptable du système informatique. La solution finalement retenue n'est pas obtenue en une seule itération. Plusieurs étapes sont

nécessaires ; ces étapes successives permettent de raffiner le niveau de détails du système à réaliser La modélisation des systèmes a deux objectifs :

Simuler leur comportement,

Communiquer des descriptions.

3.4 Les outils de modélisation informatique :

Il existe plusieurs outils pour la modélisation informatique : les systèmes à file d'attente, les réseaux de pétri, etc. Ces outils sont à la base de différents logiciels de simulation.

3.4.1. Les files d'attente

Une file d'attente est un outil important utilisé pour modéliser certains types de systèmes stochastiques, elle est constituée d'un ensemble d'individus (ou objets) qu'on appelle clients qui viennent acquérir un service auprès d'un ou plusieurs serveurs.

Les réseaux de files d'attente sont fréquemment utilisés pour modéliser des systèmes à événements discrets. Elle permet la modélisation du partage des ressources.

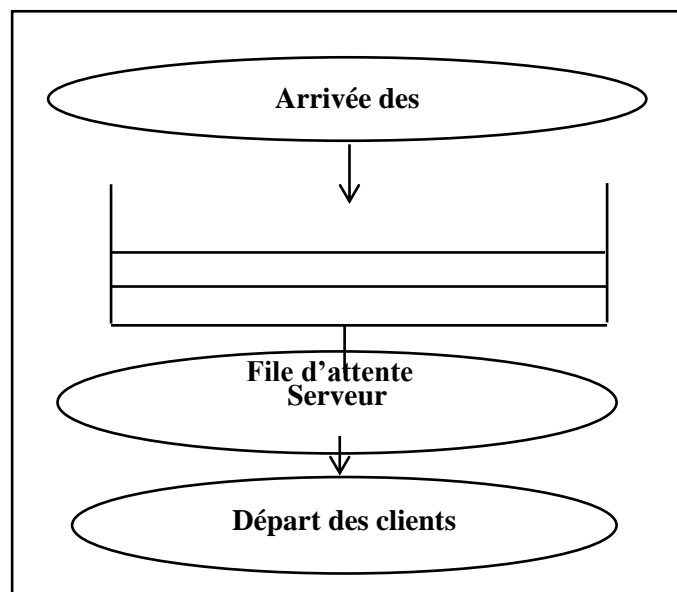


Figure 1.3. Représentation graphique d'une file d'attente primaire.

La représentation ci-dessus (Figure 1.3) peut être étoffée par l'ajout de plusieurs utilisateurs (serveurs multiples) au bout de la file d'attente et d'une capacité limitée de la file d'attente.

Le système entré-sortie des clients peut être contrôlé (FIFO, LIFO, Random...). Les clients eux-mêmes peuvent être classés, puis ensuite répartis dans la file selon ce classement. Finalement le réseau de files d'attente est créé par la connexion d'un ensemble de fils simples.

3.4.2 Les réseaux de pétri

Le réseau de pétri est un modèle mathématique à variables discrètes, créés en 1962 par Carl Adam Pétri. Il se représente par un graphe biparti (composé de deux types de nœuds) orienté (composé d'arc(s)) reliant des places et des transitions (les nœuds). Deux places ne peuvent pas être reliées entre elles, ni deux transitions. Les places peuvent contenir des jetons (pouvant par la suite prendre différentes valeurs), représentant généralement des ressources disponibles.

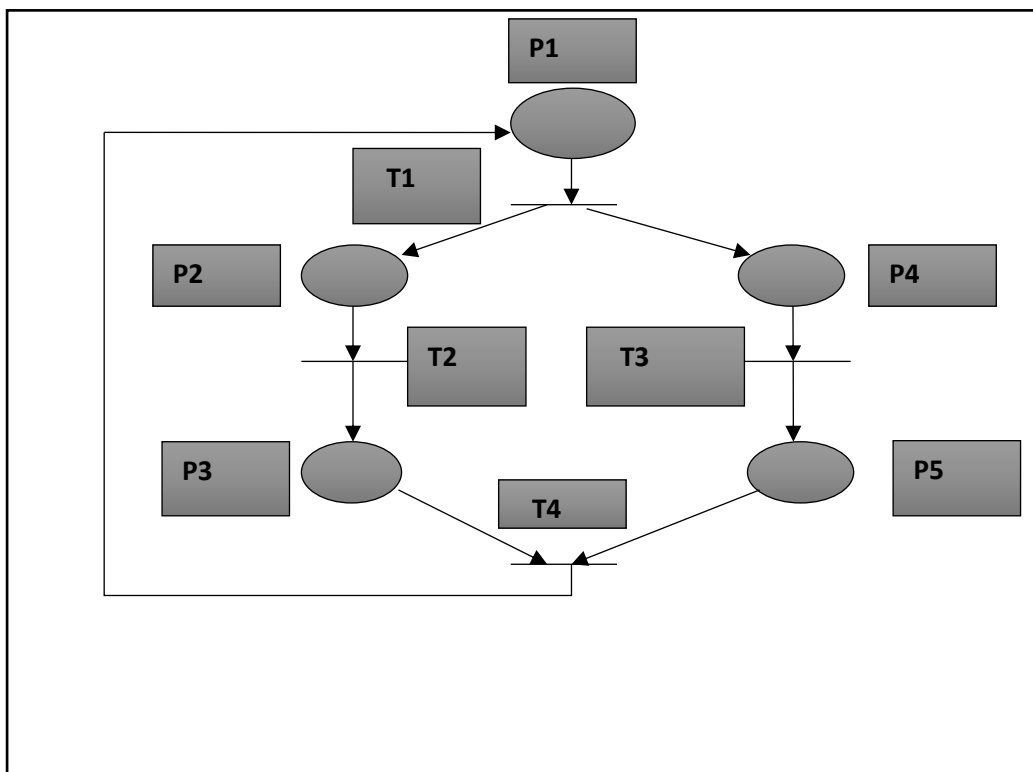


Figure 1.4 : Typologie de la modélisation graphique des réseaux de Pétri.

3.4.3. Modélisation orientée objet

Dans la Modélisation orientée-objet, les systèmes sont uniquement constitués d'entités appelées objets. Ces objets sont définis par des types qui définissent de façon syntaxique et

sémantique les propriétés que présenteront les objets du type, ces propriétés permettent de délimiter, de qualifier les objets et de savoir comment on peut communiquer avec eux.

L'implémentation de ces propriétés est ensuite fournie par une classe, structure de données qui lie à une propriété une implémentation possible.

L'implémentation des propriétés par une classe peut être représentée soit sous forme d'emplacement mémoire (champs de données) appelé attribut, soit sous forme de calcul (opération) appelé méthode.

En orientée-objet, si un type définit donc l'interface de l'objet, la classe lui fournit une implémentation. A ce titre, la classe est le moule par lequel est construit un objet. On dit alors d'un objet qu'il est une instance de telle classe. Les objets interagissent via des envois de messages.

3.4.4. Modélisation orientée agent

La modélisation OA permet une factorisation (décomposition en sous-systèmes) et une délégation naturelle des tâches, buts et/ou rôles à sous-systèmes multi-agent et leurs agents.

Cette décomposition permet de concevoir plus facilement des systèmes complexes en décomposant le problème en sous-problèmes. Les sous-systèmes sont séparés sur différentes machines et les communications et interactions s'effectuent via des langages de communication qui sont la plupart du temps basés sur la théorie des actes de langages.

4. NOTION DE SIMULATION

Etymologiquement, le terme "simulation" est dérivé du mot latin "SIMULARE" qui veut dire : copier, feindre, faire paraître comme réelle une chose qui ne l'est point.

On peut donc très bien dire que simuler le fonctionnement d'un système c'est imiter son fonctionnement au cours du temps en manipulant un modèle. Ceci équivaudrait à la génération d'un historique artificiel des changements d'état du système et l'observation de cet historique pour faire des déductions sur ses caractéristiques de fonctionnement. C'est donc une méthodologie essentiellement pratique qui permet de modéliser aussi bien des systèmes conceptuels (qu'on veut concevoir) que des systèmes existants déjà. Elle peut être utilisée pour :

Décrire et analyser la dynamique d'un système

Répondre aux questions de type « What If ? » sur le système réel

4.1 Définition de la simulation

La simulation est définie comme étant la représentation d'un phénomène physique à l'aide de modèles mathématiques simples permettant de décrire son comportement.

Autrement dit, la simulation permet de représenter par des modèles mathématiques les différents phénomènes de transfert de masse, d'énergie et de quantité de mouvement qui se produisent dans les différentes opérations unitaires.

4.2. La simulation informatique :

La simulation informatique est l'un des outils permettant de simuler des phénomènes réels, le but principal de la simulation est de :

- Étudier un système réel de manière à comprendre son fonctionnement interne et/ou à en prévoir son évolution sous certaines conditions.
- Cette étude se fait nécessairement à travers un modèle du système réel qui est utilisé pour réaliser les expérimentations.

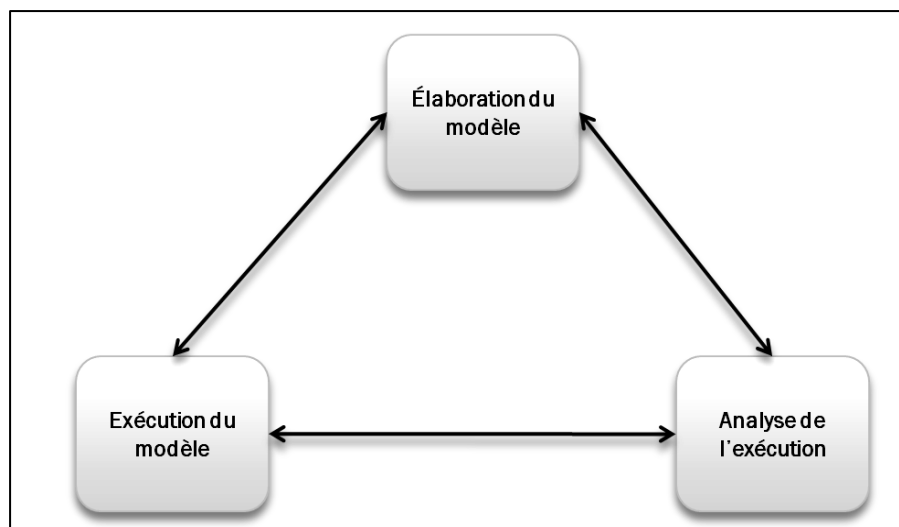


Figure 1.5 : La simulation informatique.

4.3 Catégories de simulation informatique :

On peut distinguer trois catégories de simulations :

4.3.1 La simulation continue

S'applique à la classe des systèmes dynamiques, caractérisés par des changements qui se font en permanence avec le temps. Ces types de changements peuvent être présentés par des

équations différentielles qui vont théoriquement permettre aux variables d’être calculées à tout instant.

4.3.2 La simulation discrète

Dans laquelle le système est soumis à une succession d'évènements qui le modifient. Ces simulations ont vocation à appliquer des principes simples à des systèmes de grande taille. La simulation discrète se divise en deux grandes catégories :

- **Asynchrone ou time-slicing** : on simule à chaque fois le passage d'une unité de temps sur tout le système.
- **Synchrone ou event-sequencing** : on calcule l'arrivée du prochain événement, et on ne simule qu'événement par événement, ce qui permet souvent des simulations rapides, bien qu'un peu plus complexes à programmer.

4.3.3 La simulation par agents

Où la simulation est segmentée en différentes entités qui interagissent entre elles. Elle est surtout utilisée dans les simulations économiques et sociales, où chaque agent représente un individu ou un groupe d'individus. Par nature, son fonctionnement est asynchrone.

4.4 Domaine d’application de la simulation :

Le tableau ci-dessous présente quelques domaines d’application de la simulation :

Région	Domaines d’application de la simulation
<p>Systèmes informatiques</p>	<p>Les composants « hardwares », les logiciels, le réseau du « Hardware », les bases de données et la gestion, le processus D’information, la fiabilité des « hardwares » et des logiciels, etc.</p>
<p>Domaines manufacturiers</p>	<p>Systèmes de manutention, les lignes d’assemblage, les installations De production automatisées, les installations de stockage, les Systèmes de contrôle d’inventaire, l’étude de fiabilité et de</p>

	Maintenance, le plan d'aménagement, le design des machines, etc.
Les affaires	Analyse des stocks et des commodités, la politique des prix, Stratégies de marketing, les études d'acquisition, les « cash-flow », Les prévisions, les alternatives de transport, la planification de la Main d'œuvre, etc.
Gouvernement	Les armes militaires et leurs utilisations, les stratégies militaires, Planification de la population, l'utilisation des terres, la distribution Des soins médicaux, la protection contre les feux, services de Polices, etc.
Biosciences	Les analyses des performances du sport, le contrôle des maladies, Les cycles de vie biologiques, les études biomédicales, etc.

Tableau 1.1 : Domaine d'application de la simulation

Conclusion

Le modèle peut être caractérisé par quelques paramètres physiques, et il s'adapte aux données expérimentales.

La simulation nous permet de visualiser le comportement d'un système et de l'analyser. Ceci dit, il ne faudrait pas oublier que les résultats obtenus par simulation, dépendent des hypothèses retenues pour construire les modèl

Chapitre 2 :
Les systèmes Multi-Agents

1.Introduction

Les systèmes multi-agents (SMA) sont actuellement très largement utilisés pour la modélisation des systèmes industriels complexes et conduise la plupart du temps à des séries de simulation exploratoires, ce qui simplifie la construction, la maintenance et l'exécution de ces systèmes complexes.

Ce chapitre présent la problématique de l'intelligence artificielle distribuée (IAD), les concepts principaux de la technologie d'agent et des systèmes multi- agent, quelques outils pour le développement de tels systèmes.

2. L'intelligence artificielle distribuée (IAD)

L'expression Intelligence Artificielle (IA) est employée pour la première fois (1955-1970) par John McCarthy. Il fonde l'Intelligence Artificielle sur le postulat mécaniste qui veut que toute activité intelligente soit modélisable et reproductible par une machine.

L'Intelligence Artificielle Distribuée (IAD) est un sous-domaine d'Intelligence Artificielle qui s'occupe des situations où plusieurs systèmes interagissent pour résoudre un problème commun.

L'IAD se divise en deux branches principales :

La Résolution Distribuée de Problèmes (RDP) qui étudie comment distribuer des compétences au niveau de chaque partie du système, de façon à ce qu'il soit globalement plus compétent que chacune de ses parties.

La Simulation des Systèmes Complexes (SSC), qui concerne plus particulièrement les Systèmes Multi-Agents (SMA). Les SMA traitent le comportement d'un ensemble d'agents autonomes qui essaient de résoudre un problème commun.

2.1 Problématique de l'IAD

Les problèmes que l'IAD s'attache à résoudre, sont les problèmes classiques de l'IA qui ont pris une nouvelle dimension dans le contexte multi-agents et les nouveaux problèmes proprement liés au thème de l'IAD, on peut citer :

- La modélisation de la connaissance et le problème de sa répartition sur les différents agents regroupés en des sociétés : comment formuler, décomposer, allouer des problèmes et synthétiser les résultats d'un groupe d'agents ?

- Les problèmes de génération de plans d'actions où il faut prendre en considération la présence d'autres agents. Ces problèmes sont liés au comportement d'un agent au sein d'un groupe. On s'intéresse alors à ses capacités sociales : la répartition des tâches, le partage des ressources, le raisonnement sur les autres agents (pouvoir modéliser leurs connaissances et être en mesure de connaître leurs plans d'actions et de raisonner en fonction de ces plans).
- La gestion des conflits entre les agents et le maintien de la cohérence des décisions et des plans d'actions.
- Le problème de la communication : comment permettre la communication et l'interaction entre les agents ? Quel langage et quel protocole faut-il employer ? Une communication dans les univers multi-agents n'est plus une simple tâche d'entrée-sortie, mais doit être modélisée comme un acte pouvant influencer sur l'état des autres agents.
- Les problèmes spécifiques au groupe d'agents, qui portent sur l'organisation, l'architecture de l'ensemble des agents et les paradigmes de coopération et d'action ;
- D'autres thèmes de recherche sont présents dans le contexte multi-agents, à savoir, le raisonnement temporel, le raisonnement hypothétique, la représentation de la connaissance imprécise, etc.

3. Définition d'un agent

Plusieurs chercheurs sont données différentes définitions du terme agent.

Woolridge

Un Agent intelligent est une entité située dans un environnement, capable de réaliser des actions flexibles et autonomes dans cet environnement dans le but d'atteindre ses objectifs.

Maes1

Un agent autonome est un système calculatoire qui, placé dans un environnement complexe et dynamique, perçoit et agit de manière autonome dans cet environnement et, se faisant, réaliser les objectifs ou les tâches pour lesquels il est conçu.

Ferber

Jacques Ferber, propose cette définition

« On appelle agent une entité physique ou virtuelle :

- ❖ Qui est capable d'agir dans un environnement.

- ❖ Qui peut communiquer directement avec d'autres agents.
- ❖ Qui est mue par un ensemble de tendances.
- ❖ Qui possède des ressources propres.
- ❖ Qui est capable de percevoir (mais de manière limitée) son environnement.
- ❖ Qui ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune).
- ❖ Qui possède des compétences et offre des services.
- ❖ Qui peut éventuellement se reproduire, mourir et changer d'état.

3.1 Caractéristiques d'un agent

- **Autonomie** : capacité à agir sans aucune sollicitation externe.
- **Réactivité** : capacité à percevoir des changements dans son environnement et d'y réagir.
- **Pro-active** : capacité à prendre des initiatives sans intervention extérieure.
- **Sociabilité** : capacité d'interaction avec d'autres agents (éventuellement humains) si nécessaire.

3.2. Agent et Objet

Il existe une similarité superficielle entre objet et agent, la terminologie objet n'est pas adaptée aux systèmes agents :

- les objets sont généralement passifs alors que les agents sont permanents actifs.
- les agents sont autonomes et responsables de leurs actions alors que les objets n'en sont toujours pas.
- on ne peut prévoir tous les comportements des agents dans les systèmes.
- certains chercheurs définissent un agent comme un objet actif ayant une autonomie et un objectif.
- l'approche orientée-objet ne fournit pas un ensemble adéquat de concepts et de mécanismes pour modéliser les systèmes complexes dans lesquels les rapports évoluent dynamiquement.

3.3 Type d'agents

Les agents peuvent être classés en deux catégories principales selon leur comportement et leur granularité. Cette notion de granularité est bien sûr très subjective, elle exprime la complexité de « raisonnement » d'un agent afin de séparer les agents dits "intelligents" des agents moins "intelligents". On parle d'agents réactifs et d'agents cognitifs.

➤ **Agents réactifs :**

Selon Ferber, un agent réactif est tropique s'il n'est pas pulsionnel, c'est-à-dire qu'il ne répond qu'à des stimuli de l'environnement, son comportement étant guidé intégralement par l'état local du monde dans lequel il se trouve plongé. Il agit de manière totalement réflexe aux états du monde, il n'a aucun but ni aucun état interne.

Les caractéristiques essentielles des Agents réactifs sont :

Pas de représentation explicite.

Pas de mémoire de son historique.

Fonctionnement stimulus/réponse.

Grand nombre d'agents.

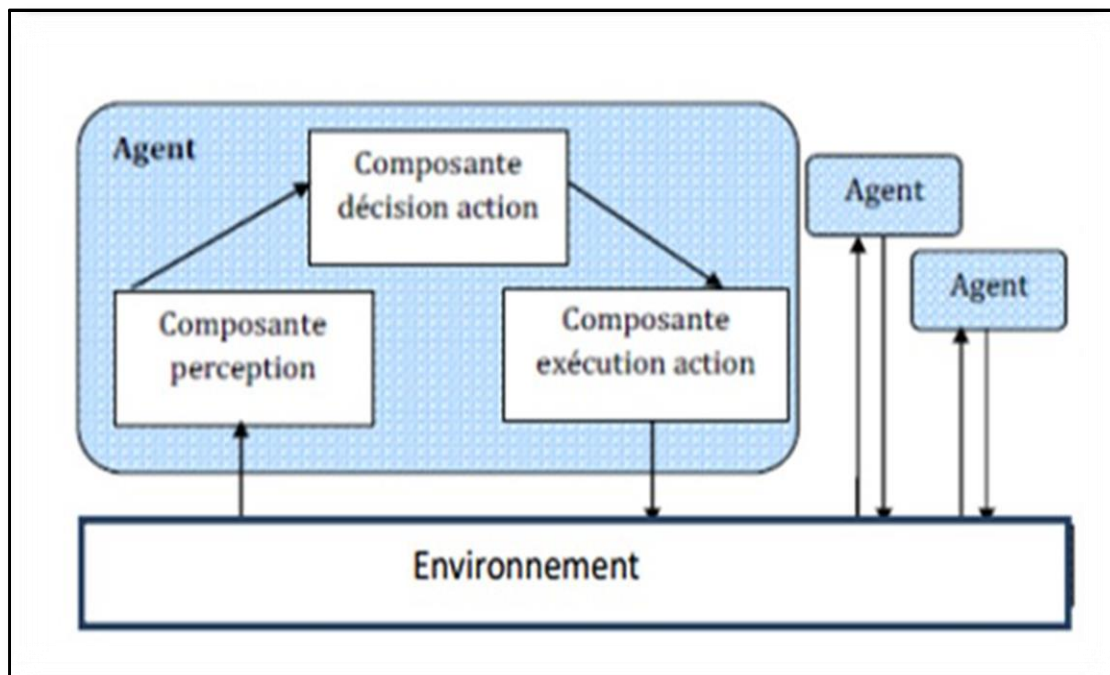


Figure 2.1 : Agent réactif.

➤ **Agents cognitifs :**

Un agent est cognitif s'il est "intelligent" :il dispose d'une base de connaissance comprenant l'ensemble des informations et des savoir-faire nécessaires à la réalisation de sa tâche et à la gestion des interactions avec les autres agents et avec son environnement.

Les Agents réactifs caractérisés par :

- ❖ Représentation explicite de l'environnement.
- ❖ Peut tenir compte de son passé.
- ❖ Agents complexes (difficile à implémenter).
- ❖ Petit nombre d'agents.

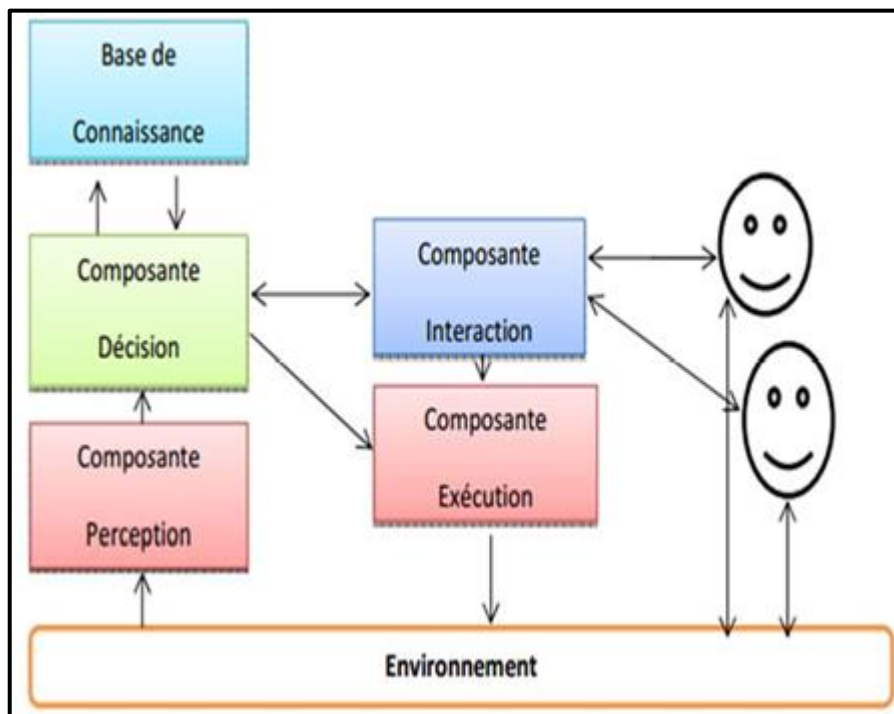


Figure 2.2 : Agent cognitif.

Le tableau suivant résume les différences entre les agents cognitifs et les agents réactifs :

AGENTS COGNITIFS	AGENTS REACTIFS
Representation explicite de environnement	Pas de representation explicite
Peut tenir compte de son passé	Pas de mémoire locale
Agents' complexes	Fonctionnement stimulus/action
Nombre d'agents Reduit	Nombre d'agents élevé

Tableau 2.1 : Différences entre agents cognitifs et agents réactifs.

➤ **Agents hybrides :**

Les agents hybrides sont des agents ayant des capacités cognitives et réactives. Ils conjuguent en effet la rapidité de réponse des agents réactifs ainsi que les capacités de raisonnement des agents cognitifs. Cette famille regroupe donc des agents dont le modèle est un compromis autonomie/coopération et efficacité/complexité.

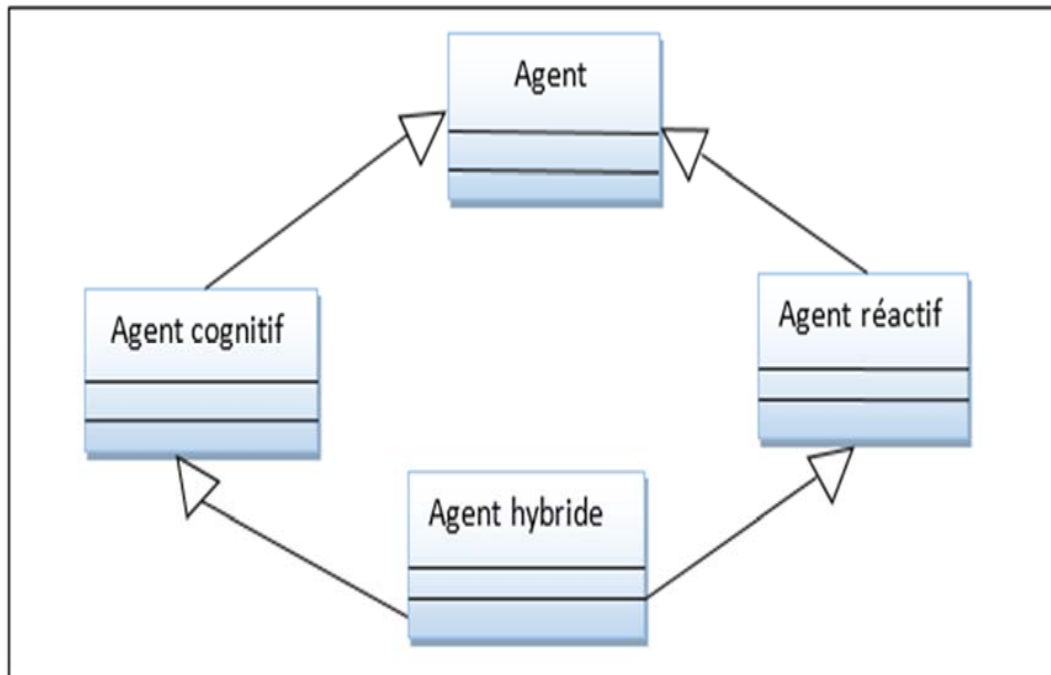


Figure 2.3 Modèle d'agent hybride.

4. Les systèmes Multi-Agents

Le thème des systèmes multi-agents est à la base de la connexion de plusieurs domaines en particulier de l'intelligence artificielle, des systèmes informatiques distribués et du génie logiciel. C'est une discipline qui s'intéresse aux comportements collectifs produits par les interactions de plusieurs entités autonomes et flexibles appelées agents, que ces interactions tournent autour de la coopération, de la concurrence ou de la coexistence entre ces agents.

Un système multi-agent est un système distribué composé d'un ensemble d'agents. Les SMA sont conçus et implantés idéalement comme un ensemble d'agents interagissant, le plus souvent, selon des modes de coopération, de concurrence ou de coexistence.

Un SMA est généralement caractérisé par :

Chaque agent a 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 agent ;
- ❖ Les données sont décentralisées ;
- ❖ Le calcul est asynchrone.

D'autre part, **Ferber** donne la définition suivante :

Un système multi-agents est un système composé des éléments suivants :

- ❖ Un environnement identifié et muni d'un système, c'est-à-dire un espace disposant généralement d'une métrique.
- ❖ un ensemble d'objets passifs pouvant être perçus, créés, modifiés ou détruits par des agents.
- ❖ Un ensemble d'agents actifs.
- ❖ Un ensemble de relations, qui unissent des objets entre eux.
- ❖ Un ensemble d'opérations offrant la possibilité aux agents de percevoir, produire, consommer, transformer et manipuler des objets.
- ❖ Un ensemble de lois universelles qui sont des opérateurs chargés de représenter l'application des actions des agents sur le monde et la réaction du monde à ces actions.

4.1. Architecture d'un SMA

On distingue deux types d'architecture pour les systèmes multi-agents.

4.1.1 SMA à contrôle distribué :

Dans ce type d'architecture les agents peuvent communiquer directement entre eux par le mécanisme d'envoi de messages. Le langage d'acteurs est la technique la plus utilisée pour la mise en œuvre de ce type d'architecture, qu'il est caractérisé essentiellement par :

- ❖ Distribution totale des connaissances et du contrôle.
- ❖ Communication par envoi de message.
- ❖ Traitement local.

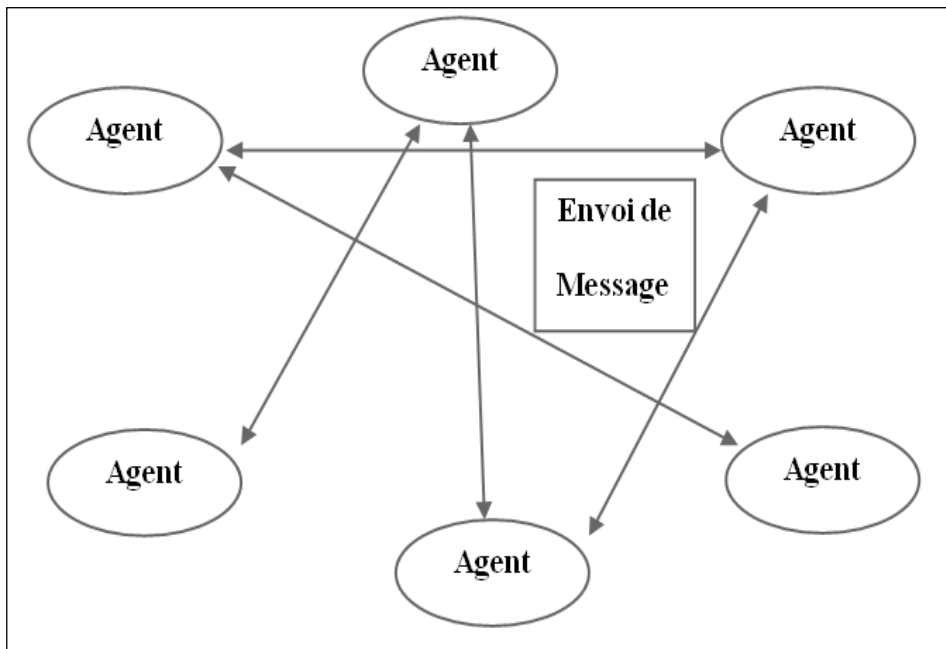


Figure 2.4 : SMA à contrôle distribué

4.1.2 SMA à contrôle centralisé :

Les systèmes centralisés se basent sur un concept assez simple et puissant : le partage des données. Dans ces systèmes, les agents ne se communiquent pas directement les données entre eux. Ils envoient et obtiennent les données du tableau (board). Cette structure permet aux agents d'échanger les données utiles à l'exécution du programme et de les garder dans une structure accessible à tous. Cette architecture possède trois sous-systèmes : KS (Knowledge Source) qui sont en fait les agents du système, le tableau qui garde et partage les données pour les agents ainsi qu'un contrôleur qui gère les conflits d'accès aux ressources entre les agents.

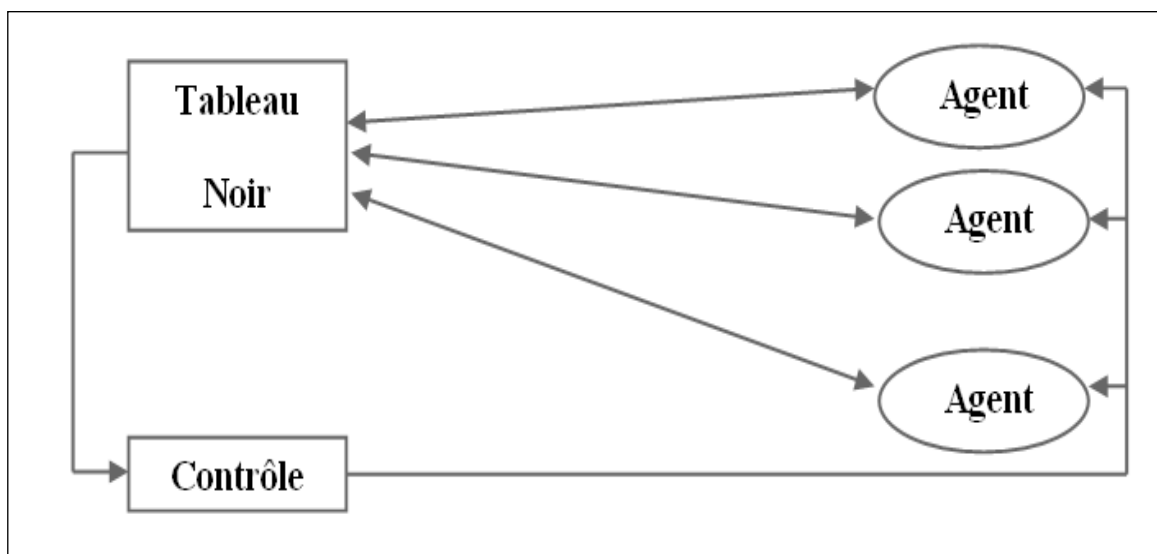


Figure 2.5 : SMA à contrôle centralisé.

4.2. Interaction entre agents

Un système multi-agent (SMA) se distingue d'une collection d'agents indépendants par le fait que les agents interagissent en vue de réaliser conjointement une tâche ou d'atteindre conjointement un but particulier. Les agents peuvent interagir en communiquant directement entre eux ou par l'intermédiaire d'un autre agent ou même en agissant sur leur environnement. Chaque agent peut être caractérisé par trois dimensions : ses buts, ses capacités à réaliser certaines tâches et les ressources dont il dispose.

L'interaction a comme objectif de mettre en relation dynamique deux ou plusieurs agents par le biais d'une ou plusieurs actions réciproques. Elle peut se faire selon plusieurs manières suivant la situation à travers laquelle interagissent nos agents.

On distingue différents types d'interaction que les agents peuvent adopter comme : la communication, la coopération, la négociation et la coordination. L'interaction se base sur des protocoles qui permettront aux agents d'échanger des messages structurés.

4.2.1. La communication

La communication est un des éléments importants du système multi-agents. Elle permet l'échange des informations, la coopération et la coordination, dans la communication agent, l'intention de communiquer est de produire un effet sur les destinataires, ils exécuteront probablement une action demandée par l'émetteur ou répondront à une question de communication directe ou indirecte.

La communication indirecte est adoptée par les agents réactifs elle se fait à travers l'environnement ou bien par le biais d'un tableau noir. Dans une communication par environnement, les agents laissent des traces ou des signaux qui seront perçus par les autres agents. Dans ce genre de communication il n'y a pas de destinataire bien défini. La communication directe, quant à elle, est spécifique aux agents cognitifs. elle est intentionnelle et se fait par envoi de messages à un ou plusieurs destinataires, la communication directe se base sur trois éléments essentiels :

- **les langages de communication** : ils permettent de structurer les messages échangés entre les agents (ACL FIPA).
- **l'ontologie** : elle sert à fournir un vocabulaire et une terminologie compréhensible par tous les agents. Cette sémantique sera régie par des règles et des contraintes qui permettront de définir un consensus sur le sens des termes. En effet, l'ontologie concernera les symboles non-logiques du contenu du message.
- **les mécanismes de communication** : ils permettent de stocker, rechercher et adresser les messages aux agents. Ces mécanismes sont présents dans les plates-formes multi-agents comme Jade ou MadKit...

4.2.2. La coopération

La coopération se traduit par le fait qu'un ensemble d'agents travaillent ensemble pour satisfaire un but commun ou individuel. L'ajout ou la suppression d'un agent influe considérablement sur la performance du groupe. Le besoin de faire coopérer des agents, vient essentiellement du fait qu'un agent ne peut atteindre son objectif individuellement et a, par conséquence besoin de l'aide des autres agents du système. La coopération a comme devise

"diviser pour régner", c'est-à-dire que la tâche va être décomposée entre plusieurs agents et ceci peut être réalisé par plusieurs mécanismes.

4.2.3. La coordination

La coordination est présente lorsqu'il existe une interdépendance dans les actions des agents, leurs buts ou même les ressources qu'ils utilisent. A ce stade, coordonner les activités des agents devient un aspect essentiel afin d'éviter des problèmes dans le comportement global du système. En effet, la coordination met de l'ordre dans le processus global effectué par des agents. Elle consiste à synchroniser leurs activités ou à régler les conflits qui existent entre eux.

4.2.4. La négociation

Les activités des agents dans un système distribué sont souvent interdépendantes et entraînent des conflits. Pour les résoudre, il faut considérer les points de vue des agents, les négocier et utiliser des mécanismes de décision concernant les buts sur lesquels le système doit se focaliser. La négociation est caractérisée par :

- Un nombre faible d'agents impliqués dans le processus.
- Un protocole minimal d'actions : proposer, évaluer, modifier et accepter ou refuser une solution.

Le processus de négociation ne consiste pas forcément à trouver un compromis mais peut s'étendre à la modification des croyances d'autres agents pour faire prévaloir un point de vue, pour le mener à bien, il est nécessaire d'utiliser un protocole qui facilite la convergence vers une solution.

4.2.5. Les protocoles d'interaction

Afin de structurer la communication entre les agents, des protocoles d'interaction sont élaborés. Les protocoles d'interaction permettent de décrire explicitement les échanges de messages entre les agents. Ils représentent un schéma commun de conversation utilisé pour exécuter une tâche ou appliquer une stratégie de haut niveau. Un protocole précis qui peut

Dire quoi à qui et les réactions possibles à ce qui est dit. Il décrit également les enchaînements de messages qui peuvent être possibles entre les agents. Il existe différents types de protocoles d'interaction, on cite particulièrement :

- Les protocoles de coordination.
- Les protocoles de coopération.

- Les protocoles de négociation.

4.3. Les domaines d'application des systèmes multi-agents

La technologie multi-agent a trouvé sa place dans les systèmes manufacturiers, le commerce électronique, le contrôle de processus et la robotique collective en informatique industrielle. Ainsi les systèmes financiers, les loisirs, les télécommunications, le contrôle-commande, les systèmes embarqués, et pas mal d'autres applications. D'autres domaines d'application se concrétisent, tels que la simulation des phénomènes complexes, l'étude de contrôle multi-agents des lignes de production, etc.

5. Les plates-formes Multi-Agents

Il existe une multitude de plates-formes multi-agents dédiées à différents modèles d'agent. Les plates-formes fournissent une couche d'abstraction permettant de facilement implémenter les concepts des systèmes multi-agents. D'un autre côté, elle permet aussi le déploiement de ces systèmes. Ainsi, elles constituent un réceptacle au sein duquel les agents peuvent s'exécuter et évoluer. En effet, les plates-formes sont un environnement permettant de gérer le cycle de vie des agents et dans lequel les agents ont accès à certains services.

On peut regrouper les plates-formes de systèmes multi-agents en cinq catégories :

1. Les outils pour la simulation qui permettent de fournir un ensemble d'outils et de bibliothèques pour faciliter le développement de simulation multi-agents.
2. Les outils pour l'implémentation d'architectures d'agents.
3. Les outils pour la conception fondée sur un modèle de composants.
4. Les outils pour la conception et l'implémentation offrant un ensemble d'utilitaires pour définir un groupe d'agent.
5. Les outils pour la conception, l'implémentation, et la validation

5.1. Jack

Jack est décrit comme étant un environnement pour construire, exécuter et intégrer Des systèmes multi-agents commerciaux, écrite en Java et utilisant une approche orientée composants. Elle est développée par la société australienne Agent Oriented Software Pty. Les agents sont basés sur le modèle BDI (Belief, Desire, Intention) développés à l'Australian Artificial Intelligence Institute (AAII) .[16]

5.2. ZEUS

Est un environnement intégré développé par l'Agent Research Program du British Telecom Intelligent System Research Laboratoire. Cette plateforme s'organise à partir des notions suivantes : les agents, leurs buts, leurs tâches (ce que les agents doivent réaliser pour atteindre leurs buts) et les faits (ce que les agents considèrent comme étant vrai). Une version de plateforme est disponible sous licence open source.

5.3. Agent Builder

La plate-forme Agent Builder : Est un environnement de développement qui a été développé par Reticular System Inc. en JAVA. Cet outil commercial permet de construire des agents intelligents à partir du modèle BDI qui utilisent langage AGENT-0. KQML pour communiquer.

5.4. MADKIT

MADKit est un environnement basé sur la méthodologie Aalaadin ou AGR (agent / groupe/ rôle). L'outil fournit un éditeur permettant le déploiement et la gestion des SMA (G-box).

La gestion faite via cet éditeur offre plusieurs possibilités intéressantes. L'outil offre aussi un utilitaire pour effectuer des simulations.

5.5. JADE

Jade est une plate-forme Java pour les systèmes multi-agents respectant le standard FIPA, JADE a été développée par l'université de Parme et C-SELT – centre de recherche télécom Italien.

Le but de JADE est de simplifier le développement des systèmes multi-agents en assurant la conformité des standards par un ensemble complet de services et agents. En se conformant aux standards FIPA : service de nom, service de pages jaunes, messages transportés et service d'analyse, et une bibliothèque de protocole d'interactions de FIPA, Jade possède trois modules principaux nécessaires aux normes FIPA, Ils sont lancés à chaque démarrage de la plate-forme.

- DF « Directory Facilitator » fournit un service de « pages jaunes » à la plate-forme.
- ACC « Agent Communication Channel » gère la communication entre les agents.
- AMS « Agent Management System » supervise l'enregistrement des agents, leur Authentification, leur accès et l'utilisation du système.

6. Les Méthodes de conception des Systèmes Multi-Agents

Les systèmes multi-agent dispose plusieurs modèles de conception : GAIA, MASE, ALLAADIN...etc.

6.1. MaSE

MaSE repose sur une extension des méthodologies orientées objet comme OMT (Rumbaugh's Object Modeling Technique) et UML auxquelles sont ajoutées des caractéristiques comme les buts, les agents étant considérés ici comme des objets actifs ayant un but. Les notations orientées objets sont aussi modifiées pour prendre en compte la sémantique liée aux agents. MaSE est similaire à Gaia au niveau du respect de la généralité et du domaine d'application supporté mais propose en plus de la génération automatique de code grâce à l'outil « MaSE Tool ». Les restrictions de MaSE sont celles de Gaia avec en plus des interactions entre agents 1 à 1 et ne propose pas le multicast. Cette méthode est basée sur un concept de rôle, également utilisé dans la plupart des méthodologies aujourd'hui car les rôles sont dirigés par un ou des buts. Cela est donc particulièrement adapté aux SMA. [21]

6.2. GAIA

La méthodologie Gaia utilise une approche centrée sur l'organisation pour analyser et concevoir un système multi-agents. Elle est considérée comme une méthodologie générique permettant le développement de tous types de systèmes. Gaia prend en considération deux niveaux : un macro-niveau (qui modélise une société d'agents) et un micro-niveau (qui se focalise sur l'agent). [22]

6.3. AALAADIN

Cette méthode est basée sur l'association de trois concepts clés : l'agent, le groupe et le rôle, utilisés simultanément pour décrire des organisations concrètes d'agents.

On le résumera par la formule suivante :

Un agent peut intervenir dans plusieurs communautés (que nous appellerons 'groupes' dans notre modèle) en parallèle. Il peut jouer dans chacun de ces groupes un ou plusieurs rôles correspondant à ses activités ou interactions. Ces rôles peuvent être portés par un nombre d'agents arbitraire, dépendant de la situation et des normes de l'organisation, c'est le modèle de conception qu'on va utiliser lors de la conception de notre travail, donc on va le détailler dans les prochains chapitres.

6.4 ADELFE

ADELFE (Atelier pour le Développement de Logiciels à Fonctionnalité Emergente) est une méthodologie dédiée à la conception des systèmes adaptatifs. Ces systèmes sont capables de s'adapter à un environnement fortement dynamique. Leur conception impose une méthode rigoureuse qui se distingue de l'approche globale descendante. ADELFE est une méthodologie qui a pour but de guider les développeurs lors du développement de ces systèmes en appliquant les théories adaptatives des systèmes multi agents. Elle propose un processus, des notations et des outils.

6.5. La méthodologie Ingénias

Dans Ingénias, l'approche générale pour spécifier un système multi-agents consiste à diviser le problème en plusieurs aspects plus concrets qui forment les différentes vues du système. Le but global d'Ingénias est de fournir un ensemble de méthodes et d'outils de développement génériques permettant le développement des systèmes multi-agents. Ingénias est fondée sur la définition de méta-modèles décrivant les différents aspects d'un système multi-agents ainsi que leurs relations. Sur cet aspect, Ingénias s'inscrit dans la continuité de la méthodologie MESSAGE [24] qui a appliqué les techniques de méta-modélisation dans le domaine des systèmes multi-agents. Ingénias a fait évoluer les méta-modèles en incorporant d'autres concepts. Les modèles sont ensuite construits avec des instances des entités méta-modèles.

Ainsi, le méta-modèle est considéré comme la définition des langages de modélisation pour la description des systèmes multi-agents

6.6. La méthodologie Tropos

Tropos est une méthodologie de développement fondée sur les concepts utilisés en ingénierie des besoins. Elle adopte une approche transformationnelle, dans le sens où, à chaque étape, les modèles vont être raffinés de manière itérative par ajout ou suppression d'éléments ou de relations dans les modèles.

Conclusion

Les systèmes multi-agents permettent de mieux décomposer un problème complexe en un ensemble d'entités autonomes et interagissant entre eux afin d'atteindre un objectif donné. Ainsi, on peut maîtriser la complexité des problèmes complexes en élaborant des modèles basés

SMA qui seront mis en œuvre sous forme de simulateurs logiciels permettant de faire évoluer un groupe d'agents dans le temps.

Actuellement le domaine informatique peut être le domaine le plus actif. Parmi les nouveautés, la technologie Multi-agent c'est ouvrir la voie des changements radicaux dans une large échelle des sciences qui sont liées aux plusieurs domaines tel que l'intelligence artificielle distribuée, l'économie, les systèmes distribués...etc.

Chapitre 3 :
Analyse et conception

❖ Partie I : Le procédé industriel de déshydratation (Cas d'études) :

1 : Introduction

Le groupe Sonatrach créé en 1963 est la compagnie algérienne de recherche, d'exploitation, de transport par canalisation, de transformation et de commercialisation des hydrocarbures et de leurs dérivés. Elle intervient également dans d'autres secteurs tels que la génération électrique, les énergies nouvelles et renouvelables et le dessalement d'eau de mer. Elle exerce ses métiers en Algérie et partout dans le monde où des opportunités se présentent.

Notre application concerne la simulation d'un système de déshydratation au niveau de ce groupe (Complexe GNL1/K), donc, dans ce chapitre nous allons présenter l'architecture et le fonctionnement de ce procédé tout en présentant les différents composants matériels entrant dans le cadre de notre étude.

2. Présentation du Complexe GL1.K

Le complexe GL1.K, l'un des principaux pôles d'hydrocarbures d'Algérie, est situé à environ quatre kilomètres à l'est de la ville de Skikda, wilaya de Skikda située au nord-est de l'Algérie. Il est implanté dans la zone industrielle de Skikda située dans le golfe de Stora.

Sa capacité annuelle de production est de 13.2 millions de M3 de gaz naturel liquéfié et une capacité de stockage de 196 000 M3 de GNL.



Figure 3.1 :GL1K

3 : Description générale de l'activité

Le complexe GL1K comprend les installations de réception du gaz naturel produit par les champs gaziers d'HASSI R'MEL qui se situent à 550 km au sud d'Alger.

Après traitement dans les trains de liquéfaction, le gaz liquéfié est stocké à la température d'environ - 162°C dans trois bacs cryogéniques d'une capacité totale de 196 000 m³ avant d'être exporté.

Le complexe de liquéfaction du gaz naturel comprend actuellement :

3 unités de liquéfaction :

- Une unité de stockage et de chargement de GNL ;
- Une unité de traitement, de stockage et d'expéditions de GPL ;
- Une unité de production d'isobutane ;

Les unités annexes / utilités (stockage gazoline, réseau de torche...).

A sa sortie du gisement de HASSI R'MEL, le gaz naturel (GN) est un mélange d'hydrocarbures dont la teneur en méthane est supérieure à 80%. Il contient également de l'éthane (C2), du propane (C3), du butane (C4), du pentane (C5) et des traces d'hydrocarbures lourds (C6+).

Le GN peut aussi contenir d'autres constituants tels que de l'hydrogène (H2), de l'azote (N2), du gaz carbonique (CO2), des vapeurs d'eau, ainsi que des impuretés sous forme de poussières.

Les objectifs principaux du complexe de Skikda sont de :

traiter et liquéfier le gaz naturel, qui sera destiné à l'exportation.

d'extraire les sous-produits « nobles » présents dans le GN, à savoir l'éthane, le propane, le butane et le pentane.

4. Description globale des unités de liquéfaction du GN

Chaque unité de liquéfaction se compose de 4 sections.

4.1. Section « Traitement du gaz »

Le traitement du gaz naturel consiste à éliminer le gaz carbonique, les vapeurs d'eau et le mercure (Hg) en 3 étapes :

- ✓ Décarbonatation : cette étape est destinée à abaisser la teneur en CO2 à une valeur inférieure à 100 ppm par lavage avec une solution aqueuse de monoéthanolamine .
- ✓ Déshydratation : cette étape est destinée à abaisser la teneur en eau à une valeur inférieure à 1 ppm.
- ✓ Démercurisation : cette section est destinée à éliminer toute trace de mercure.

4.2. Section « Liquéfaction »

Dans une unité de GNL, la section liquéfaction est constituée d'échangeurs cryogéniques (à plaques ou bobinés) et d'un compresseur du fluide frigorigène.

Sur le complexe GNL de Skikda, 2 procédés de liquéfaction sont mis en œuvre :

- ✚ Le procédé PRICO pour les unités 5P et 6P (échangeurs cryogéniques à plaques).
- ✚ Le procédé TEAL pour l'unité 10 (échangeurs cryogéniques bobinés).

4.3. Section « Production de vapeur »

La chaudière est conçue pour transformer l'eau en vapeur surchauffée à une pression et une température bien déterminée, dont 80% de la quantité sert à entraîner le turbocompresseur de cycle.

Les 20% restants sont utilisés par les consommateurs suivants :

- les auxiliaires de la chaudière (turbopompes alimentaires, turbo-ventilateur).
- Le compresseur de gaz de combustible (pour les unités 5P et 6P).
- La section « Traitement du gaz ».
- La section « Fractionnement ».

4.4. Section « Fractionnement »

L'objet de cette section est d'extraire les hydrocarbures lourds du Gaz Naturel Traité (GNT) et d'y produire par distillation :

- ❖ De l'éthane qui est exporté à l'unité CP1K (complexe matières plastiques).
- ❖ Du bupro (mélange de butane et de propane) produit à partir des unités de liquéfaction et qui est ensuite fractionné dans l'unité GPL.
- ❖ du propane et du butane produits à partir des unités 5P et 6P et qui sont ensuite stockés dans l'unité GPL.
- ❖ De la gazoline qui est transférée à la raffinerie de Skikda (RA1K) où elle sert à fabriquer l'isopentane (composant du fluide frigorigène).

Voici l'image qui illustre ces principales étapes :

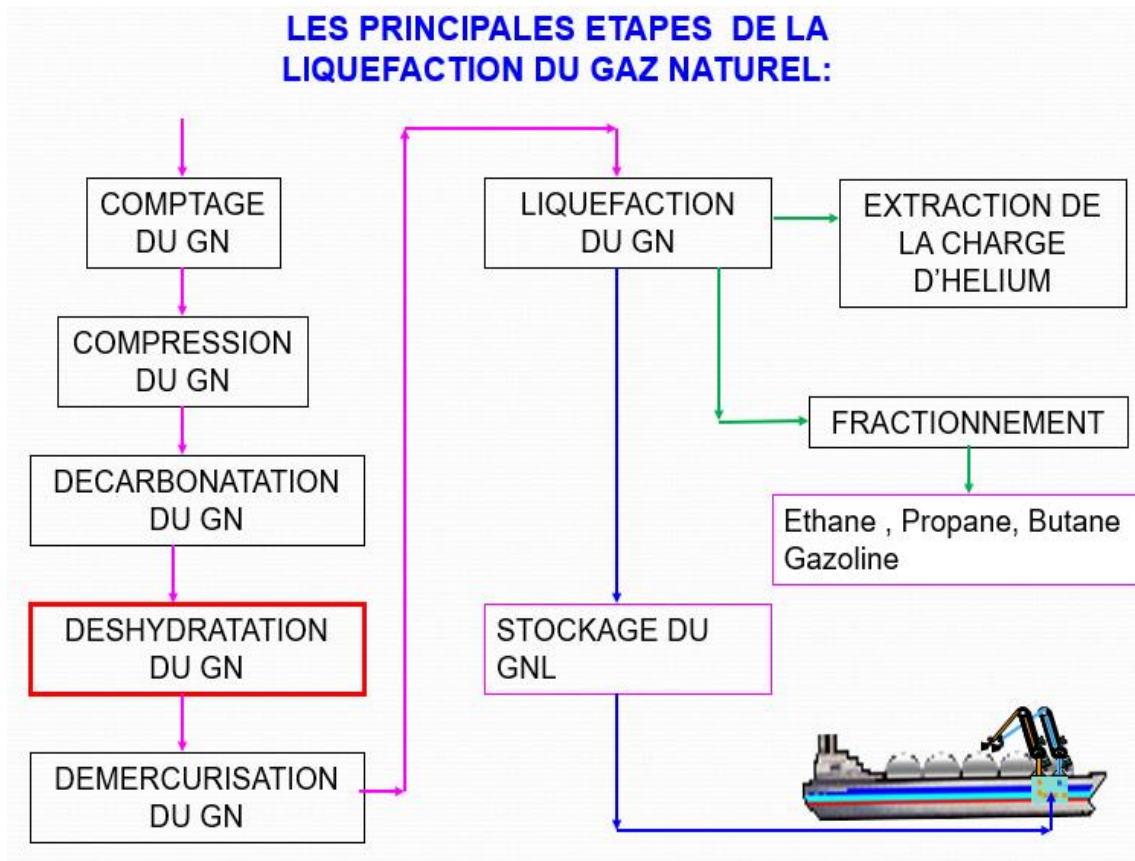


Figure 3.2 : Les principes étapes de la liquéfaction du gaz naturel

Les colonnes de fractionnement présentes au sein des unités de liquéfaction de Skikda sont – en fonction du procédé de liquéfaction mis en œuvre – les suivantes : Déméthaniseur, deethaniseur, debutaniseur, dépentaniser, Déispentaniseur et colonne de séparation C3/C4

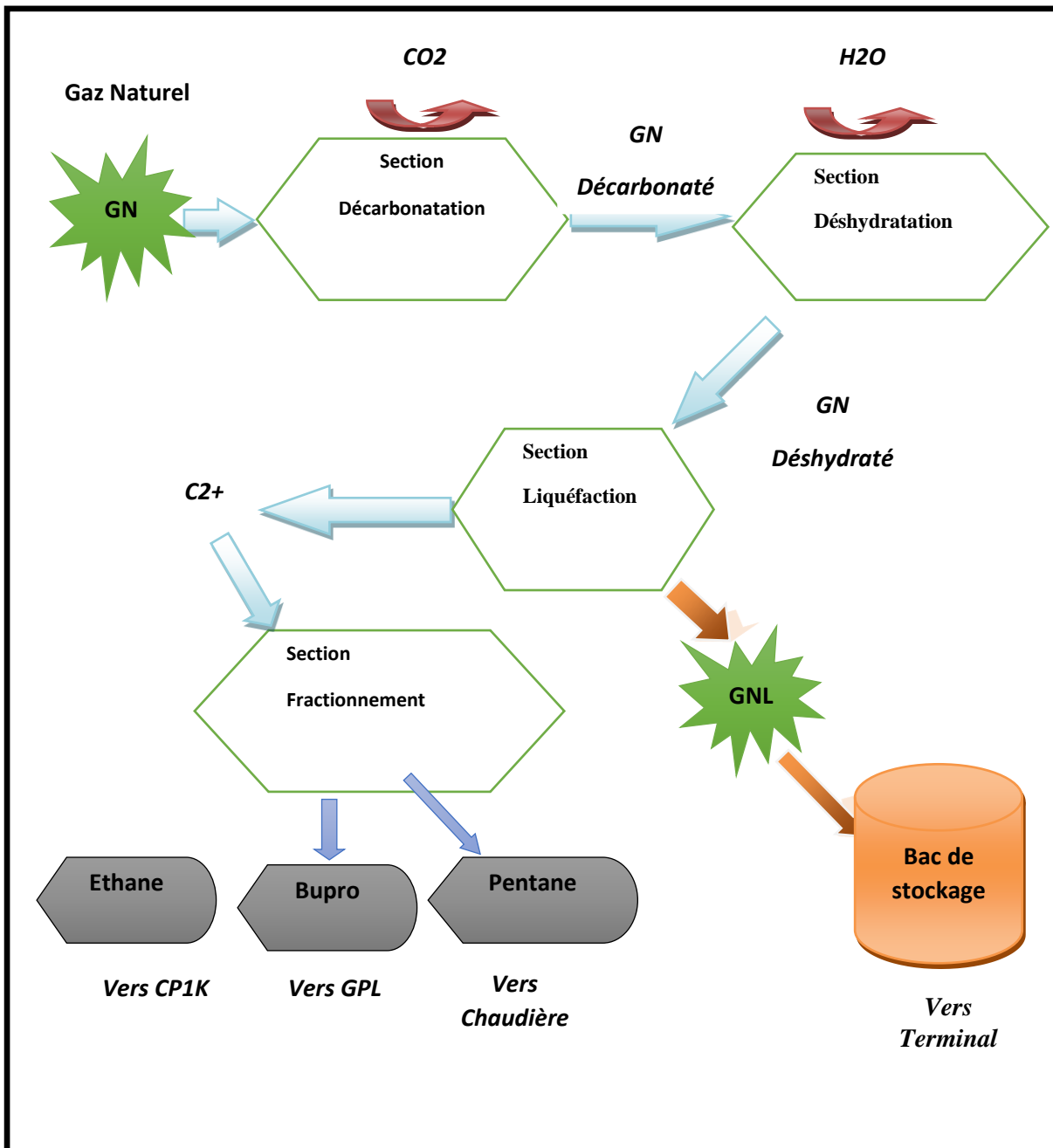


Figure 3.3 : schéma fonctionnelle

5. Description du Système de Déshydratation

Le principe de fonctionnement du système de déshydratation est basé sur l'utilisation de trois sécheurs lits à tamis moléculaire dont le premier est en mode adsorption, le deuxième est en stand-by (attente) et le troisième est en régénération. La figure .14 ci-dessous représente les différents sécheurs et leurs différents modes de fonctionnement 13-MD02-A en adsorption, 13-MD02-B en standby, et 13-MD02-C en régénération.

Le gaz sec sortant du sécheur est filtré par le filtre de gaz sec 13-MD04 situé en aval du sécheur, avant d'alimenter l'unité de démercurisation (unité 14).

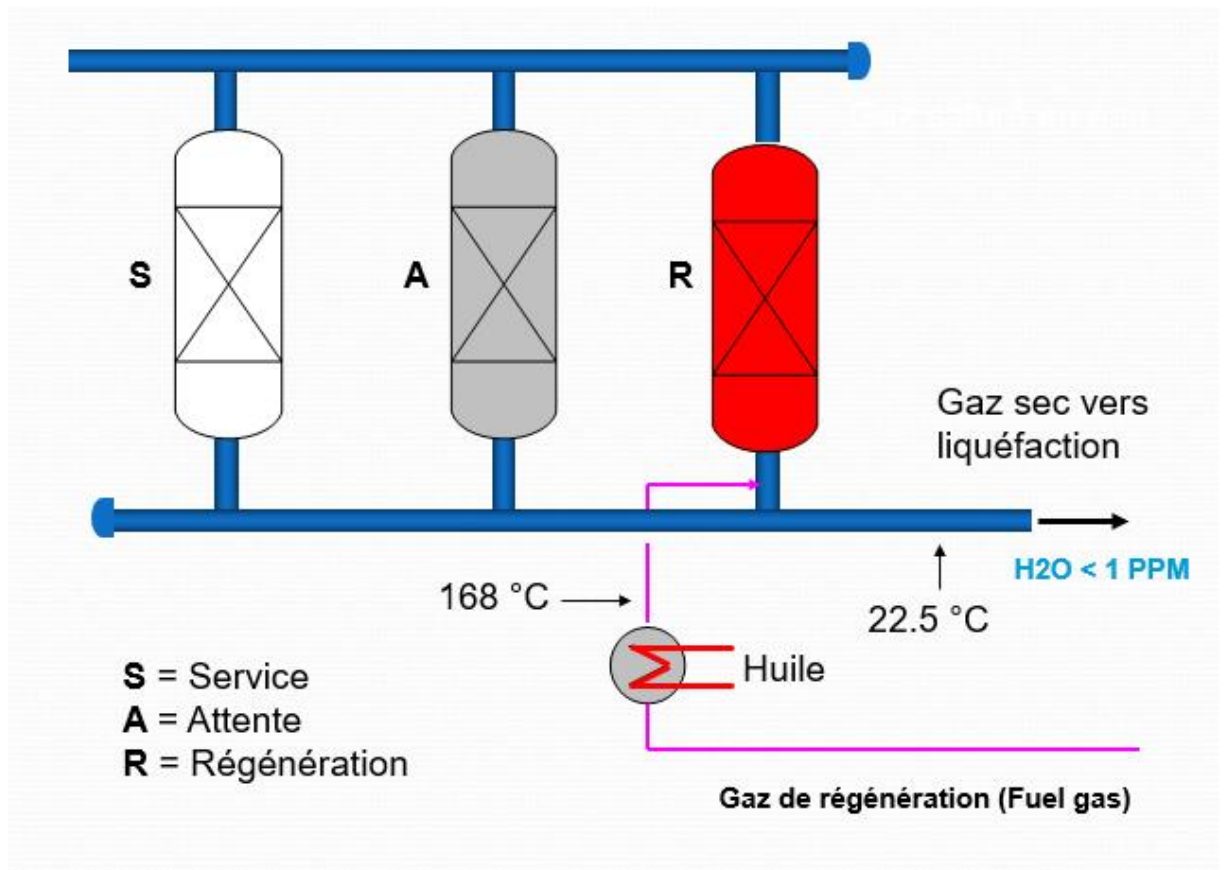


Figure 3.4 : Les sècheurs .

CYCLE DE FONCTIONNEMENT D'UN SÉCHEUR

Chaque sècheur fonctionne suivant le cycle suivant :

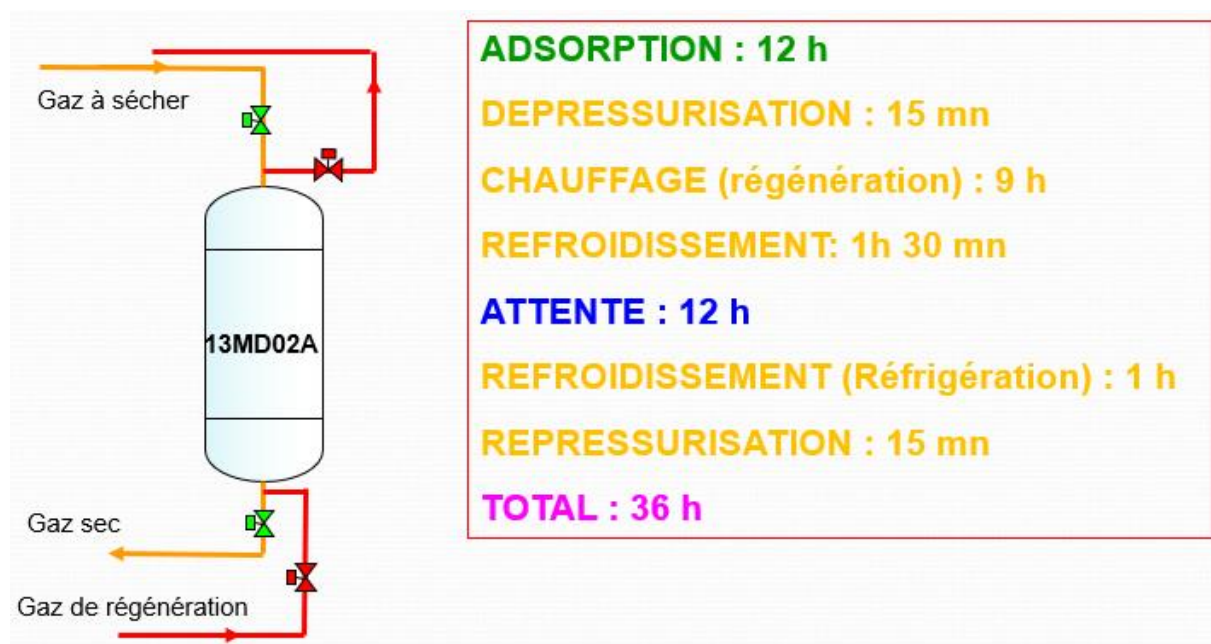


Figure 3.5 :(séparateur 13MD02A).

Le système de déshydratation peut être divisé en trois sous-systèmes :

Pré-refroidissement et séparation, Déshydratation, Filtration et régénération.

5.1 Pré-refroidissement et séparation

La fonction principale de ce système est de réduire la charge des sécheurs à tamis moléculaire 13-MD02-A/B/C.

Le gaz naturel décarbonaté venant de la colonne de décarbonatation (12-MD21) est refroidi par le propane liquide HHP côté calandre de l'échangeur 13-MC01.

Le refroidissement du GN réduit la charge du sécheur en diminuant la teneur en eau dans le GN entrant dans le ballon séparateur 13-MD01 avant d'alimenter le sécheur afin de séparer et récupérer toute eau et amine venant de l'unité de décarbonatation (unité12). Le GN en tête du séparateur se dirige vers le sécheur à tamis moléculaire disposé en mode d'adsorption à ce moment-là (13-MD02 A ou B ou C).

A partir du fond du 13-MD01, les liquides sont envoyés vers le ballon de flash d'eau du sécheur 13-MD09 pour y subir une autre séparation.

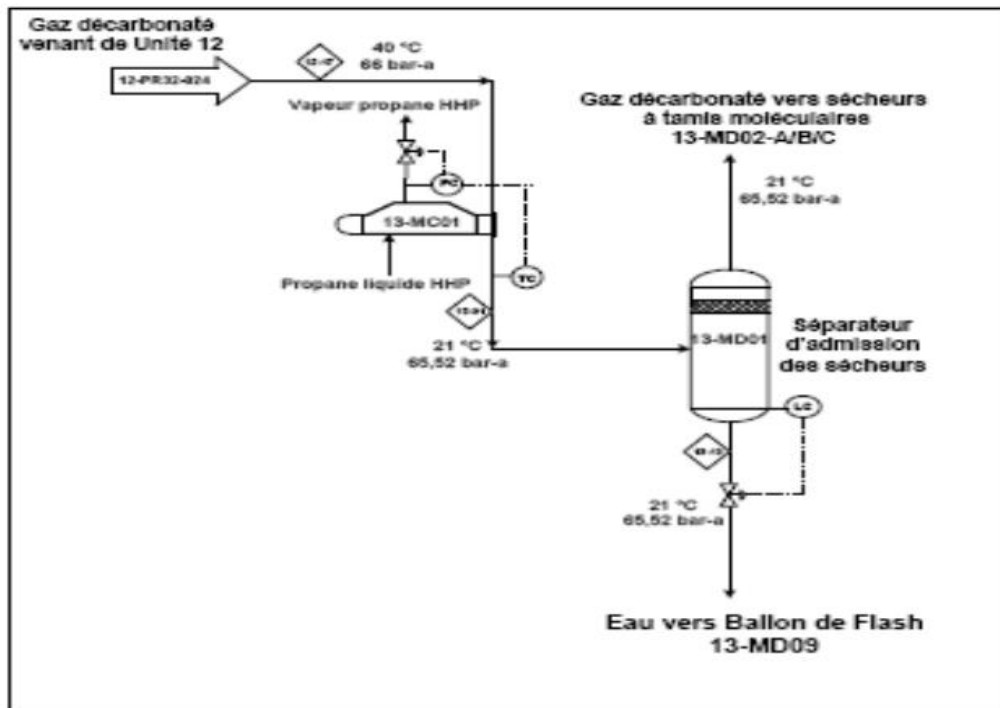


Figure 3.6 : Schéma Système de déshydratation

5.2 Déshydratation

Afin d'éviter le givrage, la solidification et la formation d'hydrates des systèmes cryogéniques situés en aval, la teneur en eau dans le GN est réduite à une valeur résiduelle maximum de 01 ppmv, ceci avant que le gaz séché ne soit envoyé vers le système de démercurisation.

Le système de déshydratation possède trois sécheurs à tamis moléculaire 13-MD02-A/B/C. Dans les conditions normales de fonctionnement, un sécheur fonctionne à une capacité de 100% en mode d'adsorption, le deuxième est en mode régénération et le troisième est en mode stand-by (attente).

Le gaz venant du séparateur d'alimentation de sécheur 13-MD01 entre par le haut du sécheur en adsorption. L'eau résiduaire dans le GN est adsorbée par les lits du sécheur.

Les vannes d'isolement des sécheurs sont contrôlées par une séquence sur le DCS.

Les séquences des sécheurs sont automatiquement programmées par un automate à partir du DCS.

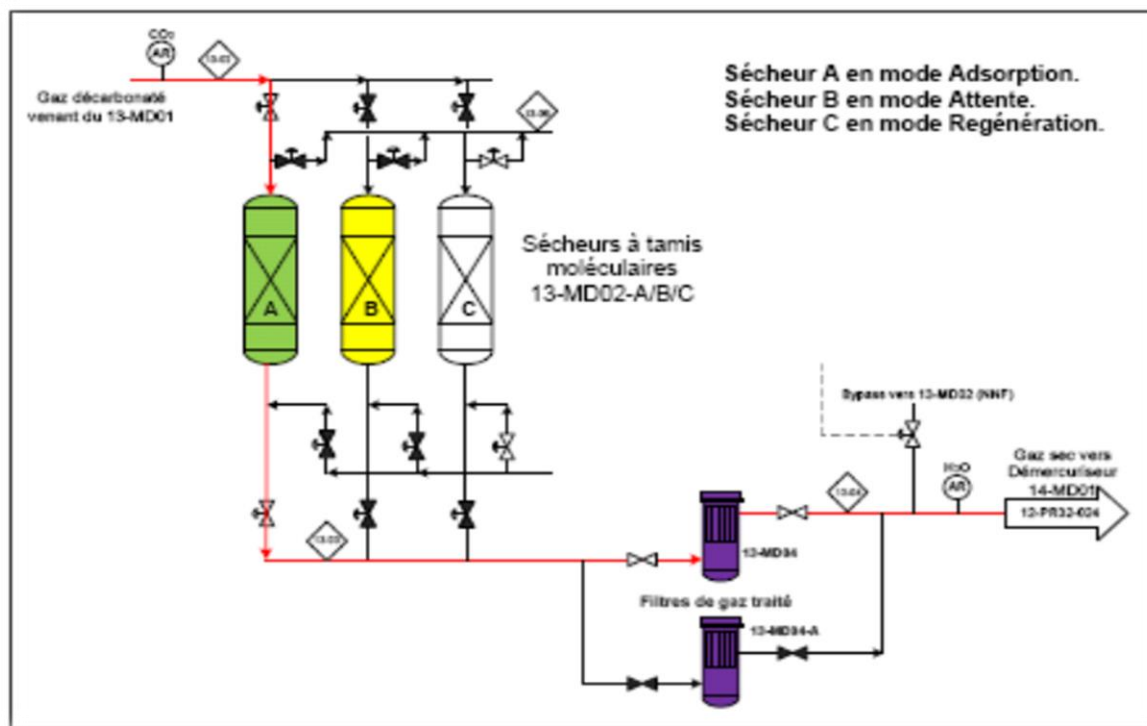


Figure 3.7 : Déshydratation et filtration.

5.3 Protection anti-incendie

Le tableau suivant représente la liste des équipements de protection anti-incendie

Associés au système déshydratation.

Description	Quantité
63-SP-008 Canon à eau anti-incendie (au sol)	1
63-SP-005 Couverture anti-feu	1
63-SP-010 Dévidoir à flexible anti-incendie	2
63-SP-011 Armoire d'incendie 4 directions	1
63-SP-024 Borne d'incendie 4 directions avec canon à eau anti-incendie	1
63-SP-050 Canon à eau anti-incendie (en hauteur)	1

Tableau 3.1 Equipements anti-incendie du système de déshydratation

6. ATTACHEMENTS

6.1 Attachement I : Liste des équipements

Equipement No.	Description	Pression De calcul (Bar eff)	Temp de Calcul (°C)
13-MC01	Pré-refroidisseur du sécheur	75	85
13-MC02	Réchauffeur du gaz de régénération	40	210
13-MC03	Refroidisseur du gaz de régénération	40	210
13-MD01	Séparateur d'admission du sécheur	75	85
13-MD02-A/B/C	Sécheur à tamis moléculaire	75	85
13-MD03	Ballon tampon du gaz de régénération	40	210

13-MD04/-A	Filtre du gaz sec	75	85
13-MD05	Filtre d'eau de régénération	40	210
13-MD09	Ballon de flash d'eau du sécheur	7	210

Tableau 3.2 : des équipements du système de déshydratation

6.2 Attachement : Consommation des utilités

Equipement No.	Description	Puissance disponible(kW)
13-MC03-MJ01	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ02	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ03	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ04	Ventilateur du refroidisseur de gaz de Régénération	15

Tableau 3.3: Moteurs électriques du système de déshydratation.

7. ATTACHEMENTS

7.1 Attachement : Liste des équipements

Equipement No.	Description	Pression De calcul (Bar eff)	Temp de Calcul (°C)
13-MC01	Pré-refroidisseur du sécheur	75	85
13-MC02	Réchauffeur du gaz de régénération	40	210
13-MC03	Refroidisseur du gaz de régénération	40	210
13-MD01	Séparateur d'admission du sécheur	75	85
13-MD02-A/B/C	Sécheur à tamis moléculaire	75	85

13-MD03	Ballon tampon du gaz de régénération	40	210
13-MD04/-A	Filtre du gaz sec	75	85
13-MD05	Filtre d'eau de régénération	40	210
13-MD09	Ballon de flash d'eau du sécheur	7	210

Tableau .5 : des équipements du système de déshydratation

7.2 Attachement II: Consommation des utilités

Equipement No.	Description	Puissance disponible(kW)
13-MC03-MJ01	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ02	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ03	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ04	Ventilateur du refroidisseur de gaz de Régénération	15

Tableau.6 : Moteurs électriques du système de déshydratation.

8. ATTACHEMENTS

8.1 Attachement I : Liste des équipements

Equipement No.	Description	Pression De calcul (Bar eff)	Temp de Calcul (°C)
13-MC01	Pré-refroidisseur du sécheur	75	85
13-MC02	Réchauffeur du gaz de régénération	40	210
13-MC03	Refroidisseur du gaz de régénération	40	210
13-MD01	Séparateur d'admission du sécheur	75	85

13-MD02-A/B/C	Sécheur à tamis moléculaire	75	85
13-MD03	Ballon tampon du gaz de régénération	40	210
13-MD04/-A	Filtre du gaz sec	75	85
13-MD05	Filtre d'eau de régénération	40	210
13-MD09	Ballon de flash d'eau du sécheur	7	210

Tableau.5 : des équipements du système de déshydratation

8.2 Attachement II: Consommation des utilités

Equipement No.	Description	Puissance disponible(kW)
13-MC03-MJ01	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ02	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ03	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ04	Ventilateur du refroidisseur de gaz de Régénération	15

Tableau.6 : Moteurs électriques du système de déshydratation.

9. ATTACHEMENTS

9.1 Attachement I : Liste des équipements

Equipement No.	Description	Pression De calcul (Bar eff)	Temp de Calcul (°C)
13-MC01	Pré-refroidisseur du sécheur	75	85
13-MC02	Réchauffeur du gaz de régénération	40	210
13-MC03	Refroidisseur du gaz de régénération	40	210

13-MD01	Séparateur d'admission du sécheur	75	85
13-MD02-A/B/C	Sécheur à tamis moléculaire	75	85
13-MD03	Ballon tampon du gaz de régénération	40	210
13-MD04/-A	Filtre du gaz sec	75	85
13-MD05	Filtre d'eau de régénération	40	210
13-MD09	Ballon de flash d'eau du sécheur	7	210

Tableau 3.4 : des équipements du système de déshydratation

9.2 Attachement : Consommation des utilités

Equipement No.	Description	Puissance disponible(kW)
13-MC03-MJ01	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ02	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ03	Ventilateur du refroidisseur de gaz de Régénération	15
13-MC03-MJ04	Ventilateur du refroidisseur de gaz de Régénération	15

Tableau3.5 : Moteurs électriques du système de déshydratation.

❖ Partie II : La conception et la modélisation :

1.Introduction

Toute application informatique nécessite une phase de conception afin d'aboutir à un produit final de qualité et dans les délais attendus.

Ce chapitre concerne la présentation du modèle Multi-Agents que nous avons adopté pour modéliser notre système.

Il existe plusieurs méthodologies de conception. Nous avons adopté l'approche simulation Multi-Agents du modèle AALAADIN, vue que de notre point de vue cette approche dispose d'assez d'outils performants qui permettent de bien représenter les systèmes industriels. De même, la mise en œuvre d'un modèle développé par cette méthode permet un passage quasi-automatique vers son implémentation en utilisant la plate-forme MADKit

2 Le modèle AALAADIN

Est une méthodologie de conception, le modèle AGR est basé sur trois concepts centraux : Agent, groupe, rôle. Ce modèle ne pose aucune contrainte sur l'architecture d'un agent.

Un Agent est simplement une entité interagissant autonome qui exerce des actions dans un contexte social. Cet agent appartient un ou plusieurs communautés appelées des groupes.

Un groupe est le terme générique utilisé pour désigner une communauté d'agents en relation. En plus, un agent peut jouer un rôle qui est une représentation abstraite d'une fonction du groupe.

Le modèle Aalaadin (appelé AGR : Agent-Groupe-Rôle) situe l'analyse des systèmes multi agents.

Ce modèle Analyse les systèmes multi agents a deux niveaux :

- Le niveau descriptif.
- Le niveau méthodologie

2.1. Le niveau descriptif :

Ce niveau correspond aux concepts suivants :

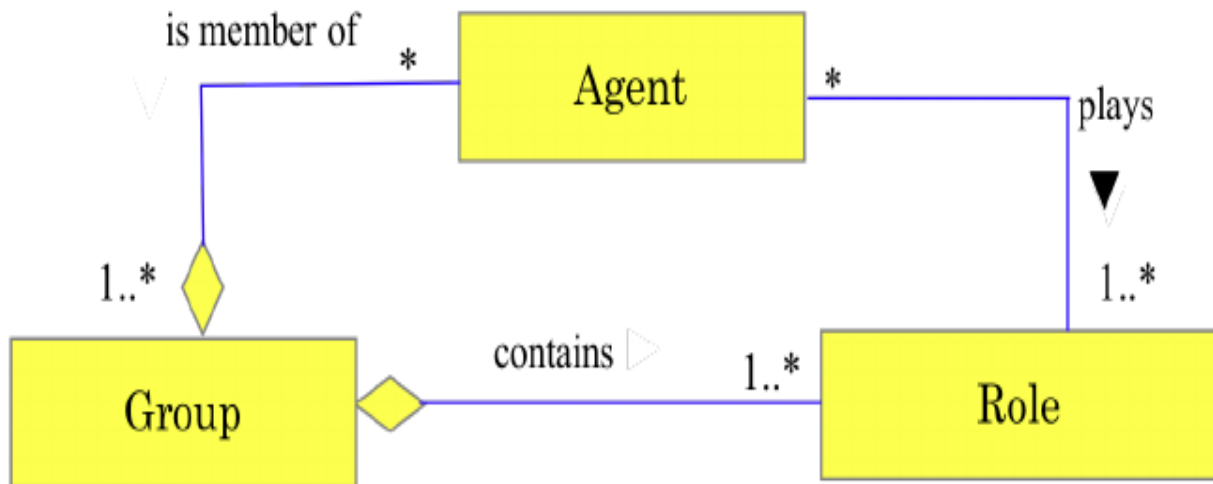


Figure 3.8: Concepts (Agent/Groupe/Rôle) du modèle Aalaadin

2.1.1 Agent :

Un agent est une entité autonome communicante qui joue des rôles au sein de groupes. Le modèle AALAADIN n'impose aucune architecture interne afin de laisser aux concepteurs toute liberté de choisir les architectures selon leurs besoins.

La figure 1 montre un agent dans un système Multi-Agent

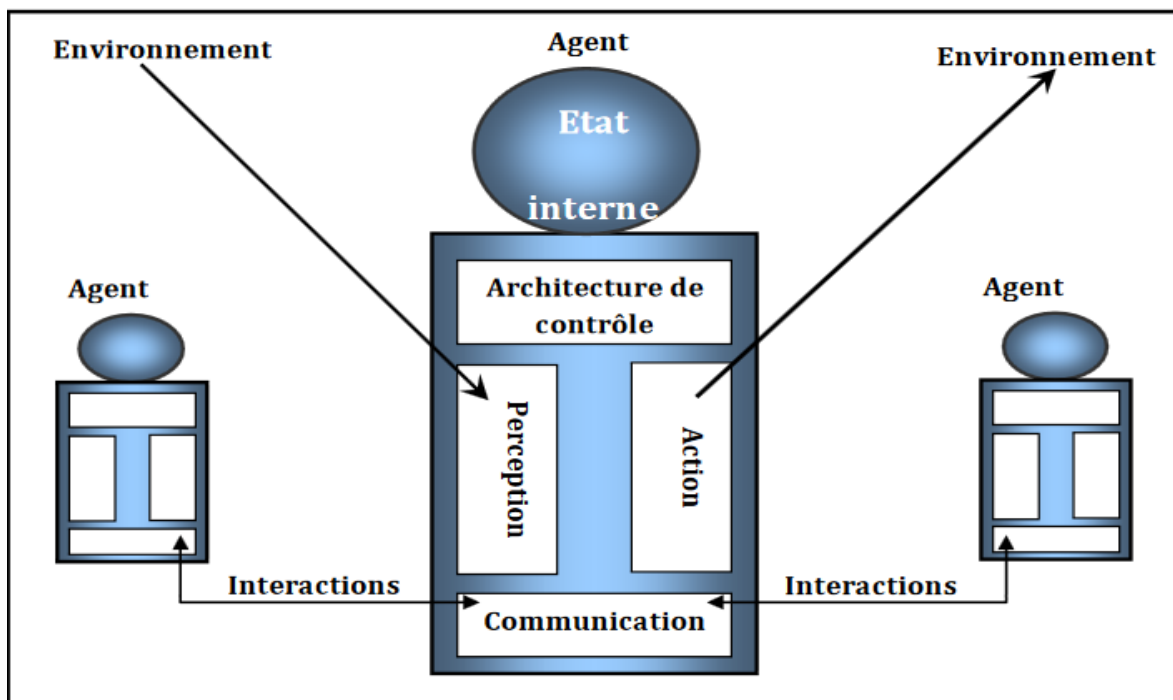


Figure 3.10 : Structure générale d'un agent dans le modèle Aalaadin

2.1.2 Groupe :

Le groupe est constitué d'un ensemble d'agents unifiés, regroupés pour accomplir une tâche donnée.

- Un agent peut appartenir à un ou plusieurs groupes simultanément.
- Les groupes peuvent se recouper librement.
- Tout agent peut créer un groupe et peut aussi rejoindre un groupe.
- Un groupe peut être local ou distribué.

2.1.3 Rôle :

Le rôle est une représentation abstraite d'une fonction, d'un service ou d'une identification d'un agent au sein d'un groupe particulier. Un agent peut avoir plusieurs rôles au sein de différents groupes, ainsi un même rôle peut être tenu par plusieurs agents.

2.2 Le niveau méthodologique :

En plus des trois concepts de base (Agent, Groupe, Rôle), le modèle Aalaadin définit à ce niveau l'ensemble des rôles possibles, spécifie les interactions et décrit les structures abstraites de groupe et d'organisation qui ne sont pas représentées directement dans une organisation multi agents, mais qui sont introduits comme étant des guides de conception.

Le but de ce niveau est de pouvoir spécifier une structure organisationnelle abstraite à partir de laquelle un système multi agents peut être développé et exprimé dans les concepts de base.

A ce niveau du modèle on se trouve le concept :

- Structure de groupe
- Structure organisationnelle

2.2.1 Structure de groupe

Cette structure identifie les rôles et les interactions qui peuvent survenir au sein d'un groupe.

2.2.2 Structure organisationnelle

Cette structure définie comme un ensemble de structure de groupe décrivant un modèle d'organisation Multi-Agents

Pour notre application a fait ressortir les trois groupes suivants :

1. Le groupe Régénération
2. Le groupe lavage
3. Le groupe perturbation

3.2 les groupes et les rôles :

3.2.1 le groupe Perturbation :

Le groupe perturbation se compose des agents suivants :

3.2.1.1 Agent GUS

Cet agent et interagit avec le système

3.2.1.2 Agent APM

Cet agent leur rôle et acquisition de consignes

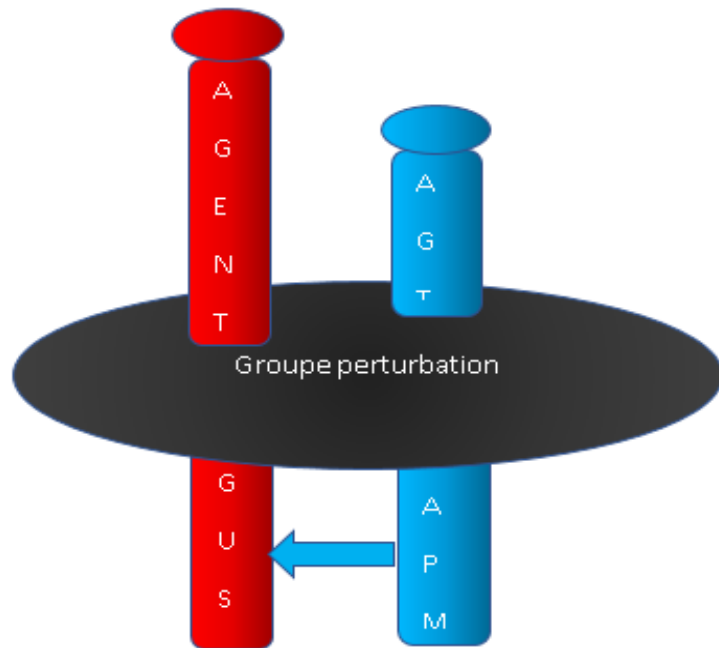


Figure 3.12 : Groupe perturbation

3.2.2 le groupe Régénération :

3.2.2.1 Agent APM :

Cet Agent leur rôle et Initialisation et régulation du groupe Régénération

3.2.2.2 Agent FV-107 :

Cet Agent leur rôle et la Vérification du flux vapeur E-104

3.2.2.3 Agent PV-102 :

Cet Agent leur rôle et la vérification de la pression T-102

3.2.2.4 Agent LV-101 :

Cet agent leur rôle et la vérification de niveau T-101

3.2.2.5 Agent LV-116 :

Cet agent leur rôle et la vérification de niveau D-116

3.2.2.5 Agent LV-105 :

Cet Agent leur rôle et la vérification de niveau T-102

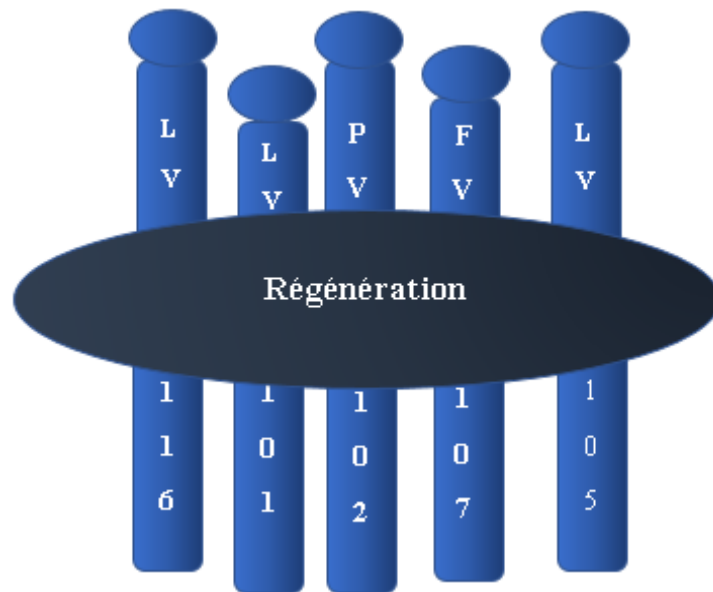


Figure 3.13 : Groupe Régénération

3.2.3 le groupe Lavage :

3.2.3.1 Agent FV-103

Cet agent leur rôle et la contrôle de flux d'Amine

3.2.3.2 Agent FV-101

Cet Agent leur rôle et la vérification du gaz naturel

3.2.3.3 Agent PV-101

Cet agent leur rôle et la vérification de la pression T-101

3.2.3.4 Agent FV-109

Cet agent leur rôle et la vérification du flux vapeur E-102

3.2.3.5 Agent APM

Cet agent leur rôle et initialisation et régulation du groupe « régénération »



Figure 3.14 : Groupe lavage.

ID Groupe	ID Agent	Rôle
Perturbation	GUS	Interagit avec le système
	APM	Acquisition de consignes
Lavage	APM	Initialisation et régulation du groupe « lavage »
	FV-103	Contrôle du flux d'amine
	FV-101	Vérification du gaz naturel
	PV-101	Vérification pression T-101
	FV-109	Vérification du flux vapeur E-102
Régénération	APM	Initialisation et régulation du groupe « Régénération »
	FV-107	Vérification du flux vapeur E-104
	PV-102	Vérification pression T-102
	Lv-101	Vérification de niveau T-101
	Lv-116	Vérification de niveau D-116
	Lv-105	Vérification de niveau T-102

Tableau 3.6 : Description fonctionnelle du simulateur

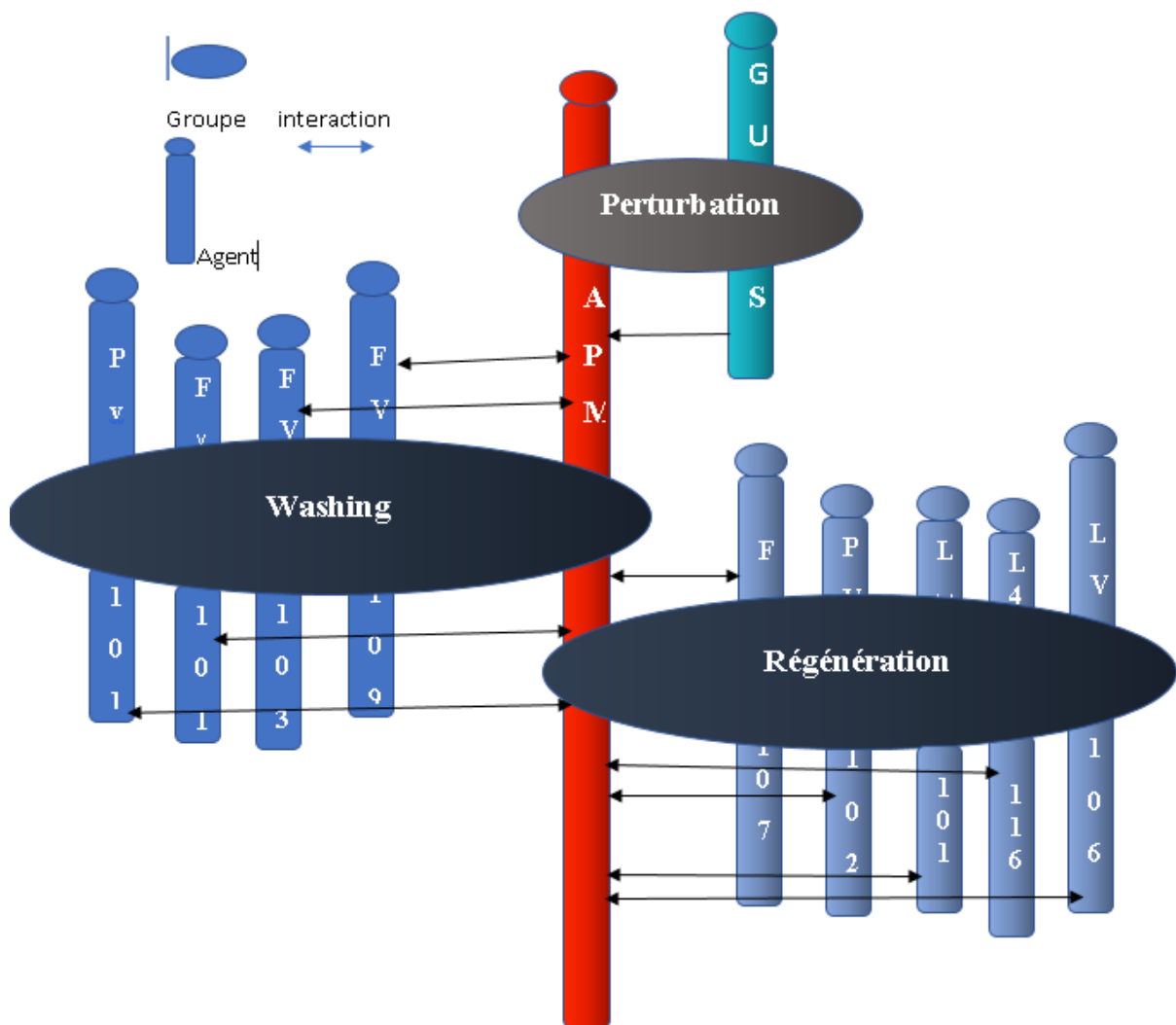


Figure 3.15 : le modèle organisationnel du simulateur

Conclusion

Afin de maîtriser la complexité de notre système, nous avons le décomposé en un ensemble d'entités autonomes qui interagissent entre eux afin de répondre aux besoins des clients.

Dans ce chapitre, on a présenter deux partie le cas d'étude «système déshydratation » et la conception et la modélisation notre système, et les outils nécessaires qui seront utilisés afin de réaliser l'application.

Chapitre 4 :
L'implémentation

1.Introduction :

Nous présentons, dans ce chapitre les principales étapes de notre réalisation. Cette implémentation traduit le passage du modèle conceptuel décrit dans le chapitre précédent vers un produit informatique Pour l'implémentation de notre logiciel, il est tout naturel d'utiliser la plateforme

MADKit, étant donné que cette plateforme repose sur le modèle Aalaadin. La mise en œuvre des différentes entités vues dans la phase de conception se fait en utilisant les outils correspondants dans la plateforme MADKit. Les signaux de commande sont représentés par des messages et les comportements agents sont implémentés en utilisant du code MADKit. Le système ainsi formé constitue le système Multi-Agents dont la dynamique de fonctionnement simule le processus industriel étudié.

Nous allons développer, dans ce chapitre, notre application en expliquant la réalisation des différents modules de programmes écrits en utilisant les concepts de MADKit. Des portions de codes extraits de notre logiciel permettront d'expliquer la façon avec laquelle nous avons implémenté notre application d'une part et de bénéficier des informations sur MadKit d'autre part.

2.Outils de programmation :

Pour réaliser notre application on a utilisé ces outils de développement :

- ❖ Le langage Java : notre travail est articulé sur ce langage.
- ❖ IDE Eclipse : est un environnement de développement intégré facilitant la tâche de la programmation.
- ❖ La plateforme MADKIT.

3. Présentation des outils :

3.1 Le langage de programmation Java :



Figure 4.1 : Logo de java

Java est à la fois un langage de programmation informatique orienté objet et un environnement d'exécution informatique portable créé par James Gosling et Patrick Naughton employés de Sun Microsystems avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld.

Il est caractérisé par sa portabilité, multiplateformes, orienté objet ainsi que la richesse de sa bibliothèque.

3.2 JDK : (Java Développement Kit)

Le Java Développement Kit, communément appelé JDK, est le kit de développement de base que propose gratuitement la société Oracle. Le Kit de développement comprend plusieurs outils, parmi lesquels :

Javac : le compilateur Java.

Java : un interpréteur d'applications (machine virtuelle).

Applet viewer : un interpréteur d'applets.

Jdb : un débogueur.

Javap : un décompilateur, pour revenir du byte code au code source.

Javadoc : un générateur de documentation.

Jar : l'éditeur d'archives Java.

4. La plateforme MADKit :

La plateforme MADKit (acronyme de Multi-Agents Développement Kit) a été conçue en 1996 par Jacques Ferber, Olivier Gut Knecht et Fabien Michel au laboratoire LIRMM de l'université de Montpellier (Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier), c'est un ensemble de packages écrits en Java qui implémente le micro noyau agent.

La plateforme MADKit est développée pour exploiter les avantages de la programmation Multi Agents, elle forme une plateforme générique de conception et d'exécution des systèmes Multi-Agents qui se basent sur le modèle organisationnel Aalaadin.

4.1 caractéristiques :

MADKit (Multi-Agent Développement Kit) est une plate-forme agent flexible. Parmi ces caractéristiques :

- S'appuie sur le modèle organisationnel AGR
- Pas de prérequis sur le modèle agent
- Capable de supporter plusieurs modèles de communication simultanément
- Mode distribué transparent
- Interfaces graphiques componentielles et flexibles

4.2 Motivations :

La caractéristique principale de la recherche et des applications agents est la grande hétérogénéité du domaine :

- Les agents sont spécifiés et construits avec différents modèles (BDI, réactifs, situés...) et formalismes (programmation objets, logiques diverses, réseaux de Pétri...).
- Les agents utilisent différents protocoles d'interaction (ContractNet, FIPARequest, FishMarket...) et de communication (ACL, KQML...).
- Les Systèmes Multi-Agents (SMA) sont utilisés pour des buts variés et dans divers domaines d'application.

MadKit fait ainsi le pari que tirer simultanément profit de cette diversité d'approches est important pour construire des systèmes complexes, ainsi que de garder cette hétérogénéité gérable à partir de modèles conceptuels et d'outils logiciels appropriés. La création de la boîte à outils MadKit fut motivé par le besoin de fournir une plate-forme agent hautement modulable et malléable. Le but fut de construire une couche de base pour divers modèles agent, et de faire que les services de bases soient extensibles et remplaçables.

4.3 Architecture de la plateforme MADKIT :

La plate-forme MadKit est construite autour du modèle AGR. En plus de ces trois concepts cœur, la plate-forme ajoute trois principes de design :

- Architecture du micro-kernel

- Agentification des services
- Interface graphique pour des modèles composants

MadKit lui-même est un ensemble de packages de classe Java qui implémentent le micro-kernel agent, ainsi que diverses bibliothèques de messages et d'agents.

Communautés

Les communautés sont des groupes de noyaux MadKit.

Modèle graphique

L'interface graphique MadKit est basée sur les spécifications Java Bean. Chaque agent est responsable de sa propre interface graphique dans chaque aspect (affichage, traitement des événements, actions...).

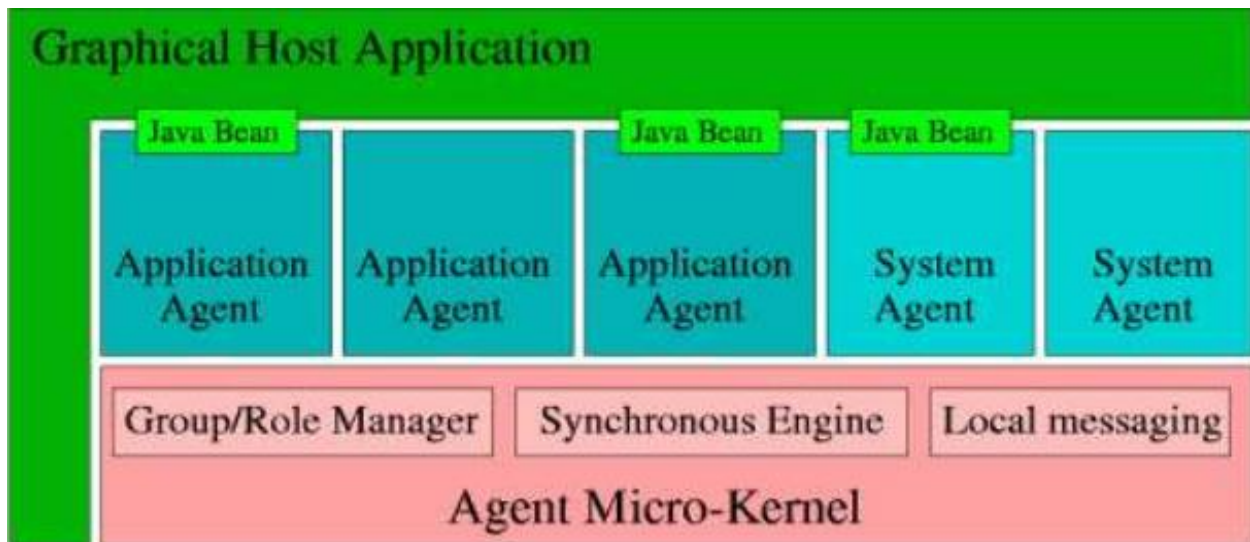


Figure 4.1 : Architecture

4.4 Programmer les agents (classe Abstract Agent) :

4.4.1 Cycle de vie

Activation

Quand un agent est créé le kernel appelle sa méthode **activate ()**.

Cette méthode peut être vue comme une sorte de constructeur dans le mode agent.

Toutes les initialisations doivent être faites dans la méthode activate () .

Vie (uniquement pour la classe thread Agent sous classe d'Abstract Agent)

La méthode live () est appelée par le kernel après la méthode activate (). Elle sert à gérer la vie de l'agent. A la sortie de la méthode, la méthode end () est appelée

Mort

A la fin de la vie de l'agent (kill Agent (), ou fin de live ()) la méthode end () est appelée.

Lancer un agent

Pour lancer un agent à partir d'un autre agent il faut en plus de l'instancier le faire enregistrer dans le contexte sociale (tables de groupes et rôles) gérer par le kernel en utilisant la méthode launchAgent ().

Tuer un agent

Un agent peut aussi être tuer en utilisant la méthode killAgent (). Seul le lanceur de l'agent, ou lui-même, peut le détruire.

4.4.2 Communications :

Les agents peuvent envoyer des messages (classe Message et classes filles XML Message, ActMessage, ACL Message...) à d'autres agents grâce à leur identifiant unique de type AgentAddress. Les agents communiquent de façon asynchrone par le biais des primitives :

- send Message (AgentAddress other, Message m)
- BroadcastMessage (Groupe g, Rôle r, Message m)
- BroadcastMessage (Community c, Groupe g, Rôle r, Message m) Pour accéder à ses message un agent utilise :
- isMessageBoxEmpty ()
- Next Message ()

4.1.3 gestions des groupes et rôles :

	Actions	Requêtes
Groupes	create Group(..), leave Group(..)	getMyGroups(..), getExistingGroups(..), getAvailableCommunities(..), is Community(..), isGroup(..)
Rôles	RequestRole(..), LeaveRole(..)	getMyRoles(..), getExistingRoles(..), getAgentsWithRole(..), getAgentsWithRole(..), is Role(..)

Tableau 4.1: Gestion des groupes et des rôles

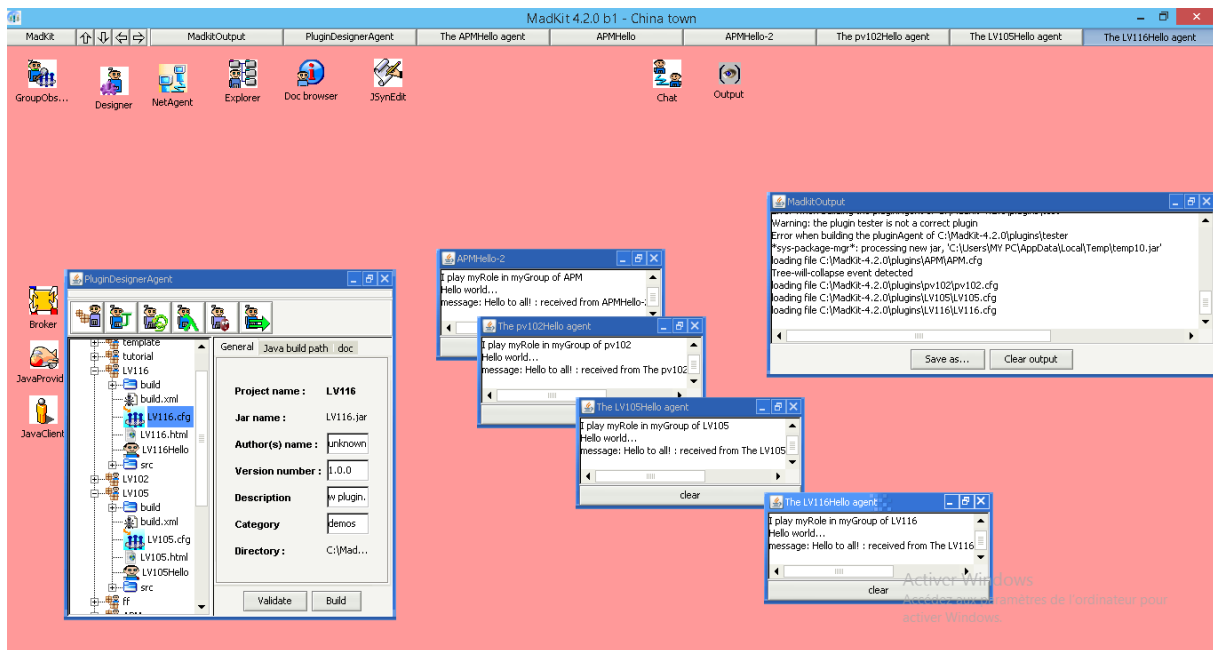


Figure 4.2: Desktop MADKit

5. Implémentation du système :

Pour illustrer la simulation multi-agents nous allons suivre un exemple pour fournir des captures d'écrans qui décrivent les différentes étapes de la simulation.

5.1 Présentation des interfaces graphique :

Exécution graphique :

Lancement du groupe Régénération :

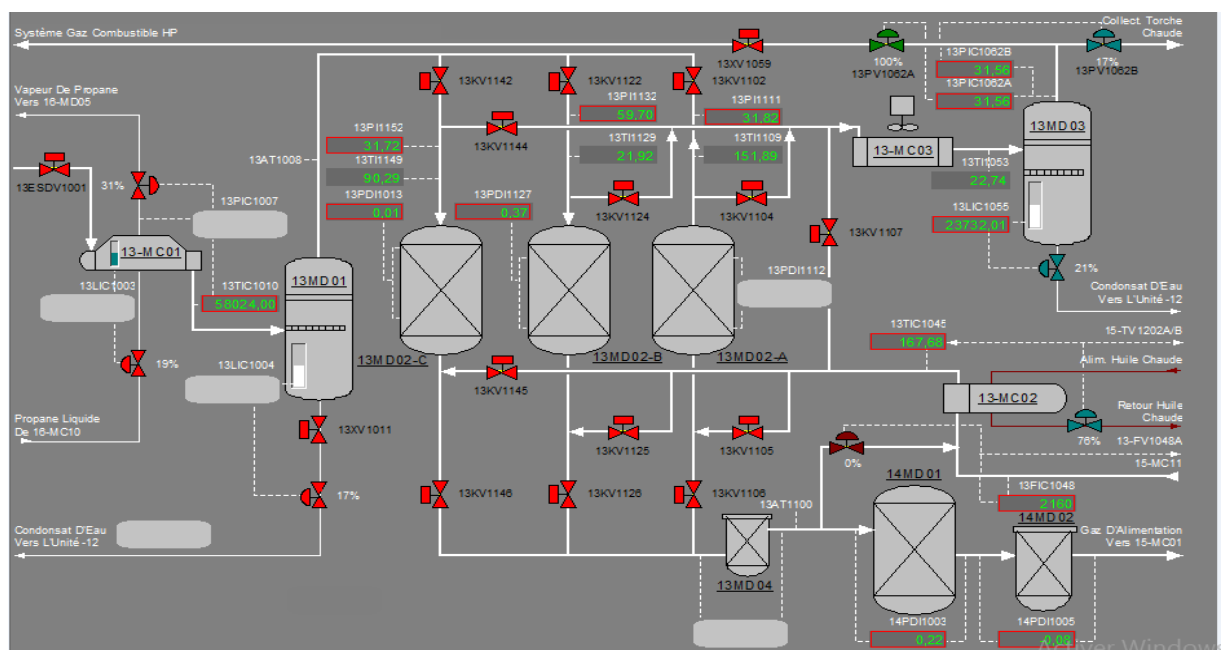


Figure 4.3 : Interface de la simulation (Groupe Régénération)

Lancement d'agent de groupe Lavage :

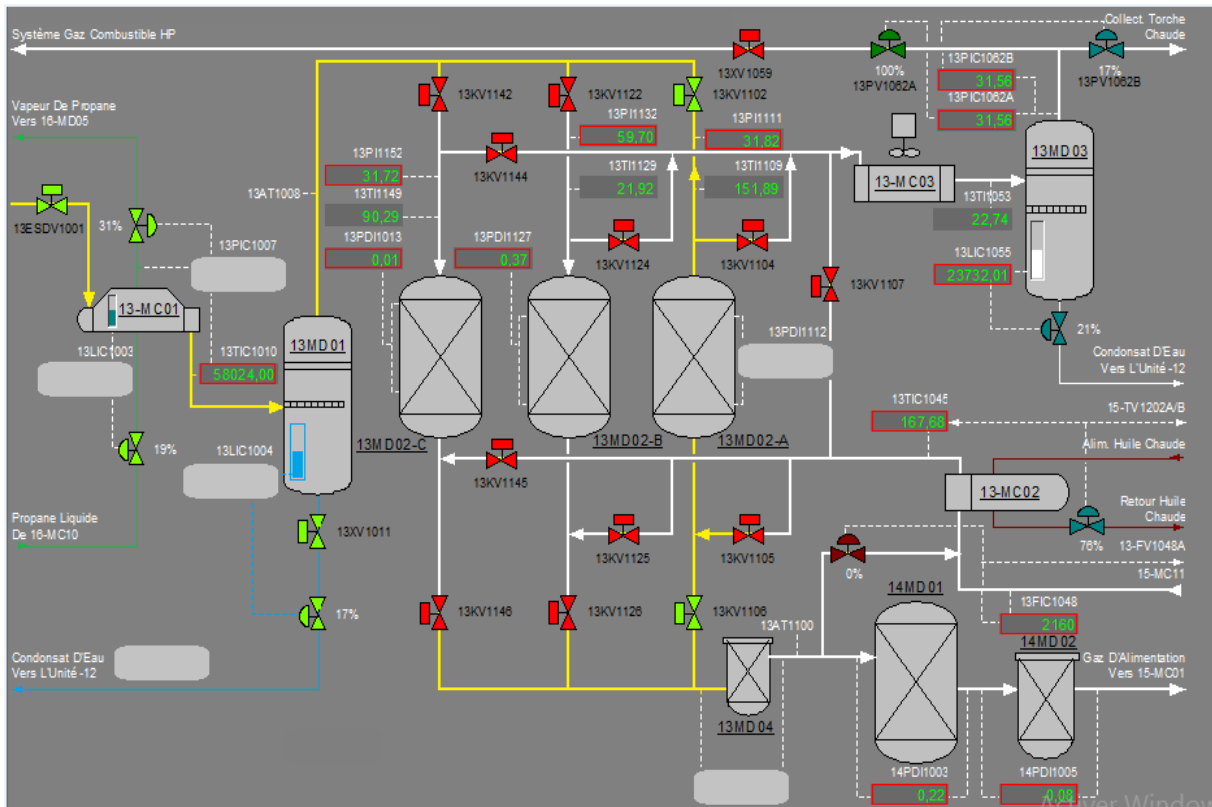


Figure 4.4 : Interface de la simulation (Groupe Lavage)

Lancement de l'agent du groupe Perturbation :

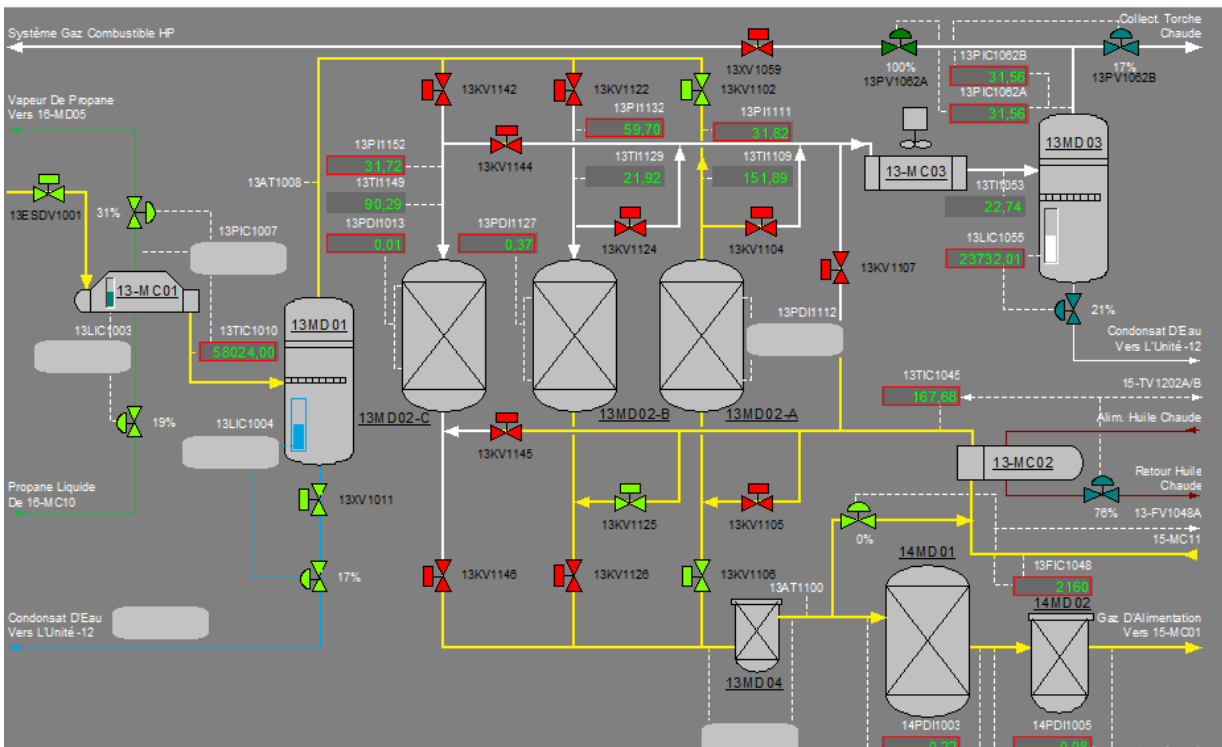


Figure 4.5 : interface de la simulation (Groupe Perturbation)

L'interface du simulateur dans état d'arrêt :

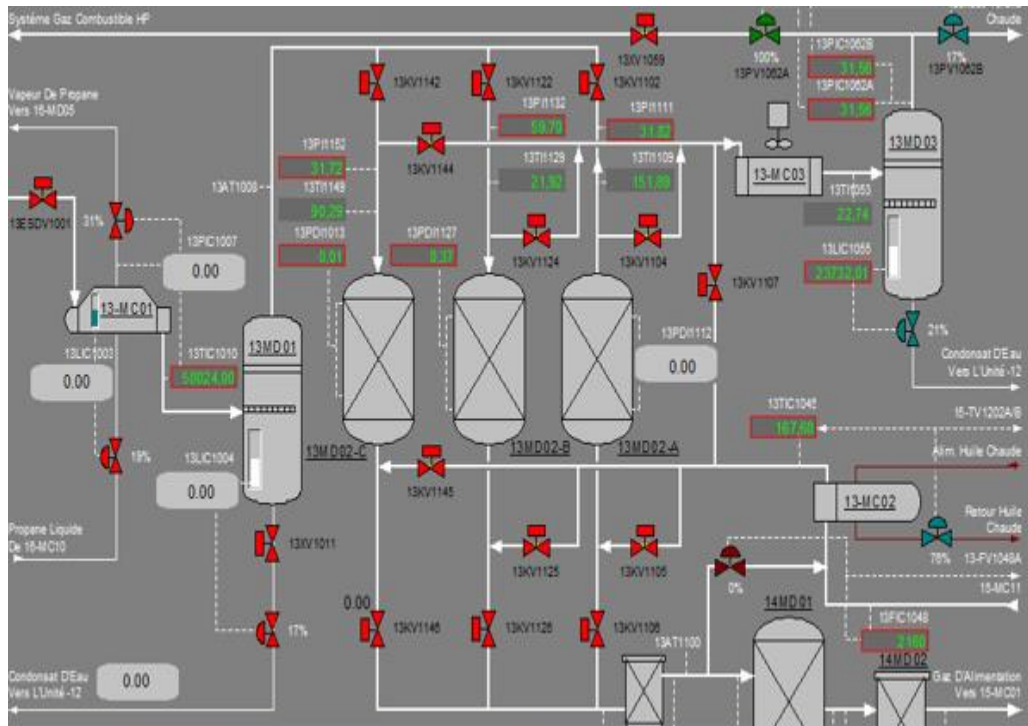


Figure 4.6 : L'interface du simulateur dans état d'arrêt.

Conclusion :

Nous avons utilisé la plateforme MADKIT afin de créer et d'implémenter notre système de déshydratation., nous avons constaté que cette plateforme est appropriée dans le domaine de simulation informatique et permet de faciliter la tâche aux développeurs débutants dans ce domaine.

Conclusion générale

A l'inverse des approches traditionnelles qui traitent de systèmes à composantes en nombres restreint, les systèmes complexes ne peuvent pas être approchés uniquement par l'analyse de leurs constituants. Les interactions entre les différentes parties et les propriétés qui en émergent sont aussi importantes. Le tout n'étant pas la somme des parties.

Dans ce travail, nous avons considéré que la dynamique globale d'un système complexe peut être modélisée par un ensemble d'entités autonomes qui interagissent dans un environnement commun. Cette dimension est prise en compte dans la modélisation et la simulation Multi-Agents qui offre aujourd'hui une alternative très séduisante aux représentations classiques des systèmes complexes. De plus, l'approche Multi-Agents n'est pas restreinte à un domaine particulier et le paradigme qu'elle définit peut-être utiliser pour modéliser toutes sortes de systèmes. C'est pourquoi, de par son aspect novateur et interdisciplinaire, la simulation Multi-Agents représente à n'en pas douter un paradigme de modélisation qui possède un énorme potentiel dans le cadre de l'étude des systèmes complexes.

Nous nous sommes efforcés d'appliquer et d'approfondir ces notions au cours de modeste travail. La considération principale est d'aboutir à effectuer un découpage (Agentification) optimal faisant passer de l'espace réel vers de l'espace virtuel (monde modélisé). Les résultats obtenus sont à nos yeux significatifs et prometteurs et qui se matérialisent par la mise en œuvre de l'un des paradigmes de programmation les plus récents qui est le paradigme agent.

L'utilisation du modèle organisationnel Aalaadin et de la plateforme MADKit sont l'aboutissement de la recherche de plateformes simples à mettre en œuvre et offrant des outils de développement facilement intégrables.

Les perspectives de travail sont nombreuses et ne concernent surtout pas uniquement l'informatique industrielle mais d'autres domaines tels que la conduite de grands projets, la gestion de trafics routiers, la gestion d'architectures urbaines ...

Bibliographie

Bibliographie

[1] https://fr.wikipedia.org/wiki/Simulation_informatique

[2] https://www.researchgate.net/publication/257936179_Du_modele_a_la_simulation_informatique

[3] https://www.researchgate.net/publication/32983688_Notions_de_syteme_et_de_modele_note

[4] <https://www.emse.fr/~grimaud/Simulation/CoursRapide/CoursRapide.htm>

[5] <https://www.univ-tlemcen.dz/~benmammar/IA3.pdf>

[6] <https://www.techno-science.net/glossaire-definition/Systeme-multi-agents.html>

[7] [Guichard, 1996] Guichard, F. (1996). La réorganisation dynamique dans les systèmes multi-Agent, Thèse de doctorat, Université de Savoie.

[8] [Gut Knecht, 2001] Gut Knecht, O. (2001) Proposition d'un modèle organisationnel générique de système multi-Agents Examen de ses conséquences

formelles, implantatoires et méthodologiques thèse de doctorat université de Montpellier II.

[9] [Jean, 1997] Jean, M. R. (1997). Emergence et SMA. In Quinqueton, J., Thomas, M.-C., and Trousse, B., editors, Journées Francophones sur l'Intelligence

[10] Artificielle Distribuée et Systèmes Multi-Agents (JFIADSMA'97), pages 323–342, Colle-Sur-Loup. Hermes.

[11] [Labidi et Lejouad, 1993] Labidi, S., Lejouad, W. (1993) ' De l'intelligence artificielle distribuée aux systèmes multi-agents' INRIA N° 2004.

[Le Moigne, 1990] Le Moigne, J.-L. (1990). La modélisation des systèmes complexes. Dunod. Paris 1990.

[Le Sturgeon et al., 1993] Le Sturgeon, E., Mandiau, R., and Libert, G. (1993). Proposition d'organisation dynamique d'un groupe d'agents en fonction de la tâche. In Actes des 1ères Journées Francophones sur l'Intelligence Artificielle Distribuée et les Systèmes Multi-Agents - JFIADSMA, Toulouse, France.

[12] [Giorgini Paolo] [et al.] The Tropos Methodology: An Overview [Revue]. – 2004