

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Abderrahmane Mira de Bejaia



Faculté des Sciences Exactes

Département d'Informatique

Mémoire fin de cycle

En vue de l'obtention du diplôme de Master Recherche en

Informatique

Option : Intelligence Artificielle

Thème

Conception et réalisation d'un simulateur à base de SMA pour activités humaines

Réalisé par :

Oughlis Nadia

Évalué par :

- M^r AMROUN Kamel Université de Béjaïa
- M^{me} KHALED Hayat Université de Béjaïa
- M^{me} YAICI Malika Université de Béjaïa
- M^{me} ALOUI Soraya Université de Béjaïa

Promotion : 2020-2021

Remerciements

Je remercie Dieu tout Puissant de m'avoir donné la force, la santé, le courage et la patience de pouvoir accomplir ce travail qui est pour moi le point de départ d'une merveilleuse aventure, celle de la recherche, source de remise en cause permanent et de perfectionnement perpétuelle.

Un grand merci à ma famille surtout mes parents pour leur encouragement et leur suivi avec patience le déroulement de mon projet.

Je tiens à remercier également mon encadreur M^{me} YAJEJ Malika et mon Co-encadreur M^{me} ALOUJ Soraya d'avoir accepté de me guider tout au long de ce travail.

Enfin, je tiens aussi à remercier également tous les membres de jury pour avoir accepté d'évaluer mon travail.

Dédicace

Je dédie ce modeste travail à:

*A mes parents pour tous leurs amour et leurs soutien pour réaliser ce
travail;*

A mes frères;

A ma famille sans exception;

A mes enseignants;

*A tous mes amis et camarades et tous ceux qui m'a aidé de près
et de loin;*

Table des matières

Table des matières.....	i
Liste des figures.....	iv
Liste des tableaux	v
Liste des Acronymes	vi
Introduction générale.....	1
1 Généralités sur les agents et systèmes multi-agents (SMA)	
1.1 Introduction	3
1.2 Agent.....	3
1.2.1 Définition agent.....	3
1.2.2 Propriétés des agents	4
1.2.3 Types d'agents.....	5
1.2.3.1 Agents réactifs	5
1.2.3.2 Agents cognitifs	7
1.2.3.3 Agents hybrides.....	8
1.2.4 Avantages de la technologie agent.....	9
1.3 Système multi-agents.....	10
1.3.1 Définitions	10
1.3.2 Caractéristiques d'un SMA.....	12
1.3.3 Environnement multi-agent	12
1.3.4 Plates-formes multi-agents	14
1.3.4.1 Définition	14
1.3.4.2 Catégorisation	14
1.3.4.3 Différentes plates-formes et leurs caractéristiques.....	15
1.3.5 Avantages	19
1.3.6 Interaction entre agents	20
1.3.7 Organisation des agents	21
1.3.8 Communication entre agents	22

1.3.8.1	Modes de communication	23
1.3.8.2	Langages de communication	23
1.3.9	Coordination	24
1.3.9.1	Coopération	25
1.3.9.1.1	Planification entre agents coopératifs	26
1.3.9.2	Compétition	27
1.3.9.2.1	Négociation entre agents compétitifs	27
1.4	Conclusion.....	28
2	Reconnaissance d'activités humaines	
2.1	Introduction	29
2.2	Définition de la RAH	29
2.3	Objectifs de la RAH.....	29
2.4	Chaine de reconnaissance d'activité	29
2.5	Caractéristiques des systèmes de RAH	31
2.6	Classification des activités humaines.....	33
2.7	Types de capteurs	34
2.7.1	Capteurs portés par le corps humain	34
2.7.2	Capteurs ambiant.....	35
2.7.2.1	Caméras vidéo et microphones	35
2.7.2.2	Capteurs sans fils à sortie binaire	36
2.7.2.3	Identification par radiofréquence (RFID)	37
2.8	Challenges en reconnaissance d'activités complexes	38
2.9	Domaines d'applications	39
2.10	Conclusion	40
3	Conception et réalisation du simulateur	
3.1	Introduction	41
3.2	Etude de l'existant.....	41
3.2.1	Simulateur UbiREAL	41
3.2.1.1	Définition	41

3.2.1.2	Objectifs.....	41
3.2.1.3	Structure UbiREAL	42
3.2.2	Simulateur SIMACT.....	43
3.2.2.1	Architecture générale.....	43
3.2.2.2	Modèle d'activité de la vie quotidienne utilisée pour les tests.....	44
3.2.3	Plateforme SMACH	46
3.2.3.1	Présentation générale de l'architecture.....	46
3.2.3.2	Génération de populations	48
3.2.3.3	Modèle agent	50
3.3.	Analyse et synthèse.....	51
3.4	Outils de développement	51
3.5	Conception du simulateur	52
3.6	Manuel d'utilisation du simulateur	55
3.7	Conclusion	59
	Conclusion générale et perspectives	60
	Bibliographie	61
	Résumé	66

Liste des figures :

Figure 1.1 : Fonctionnement d'un agent réactif	7
Figure 1.2 : Architecture d'un agent cognitif	8
Figure 1.3 : Architectures d'agents en couches.....	9
Figure 1.4 : Fonctionnement du Système multi-agents	11
Figure 1.5 : Les différents types de coordination entre agents	25
Figure 2.1 : Chaînes de reconnaissance d'activités	30
Figure 3. 1 : Architecture UbiREAL.....	42
Figure 3.2 : Spécification de l'itinéraire de l'avatar	42
Figure 3.3 : L'outil de simulation d'habitat intelligent SIMACT	44
Figure 3.4 : Modèle des activités choisies : faire du café et préparer des toasts	45
Figure 3.5 : Les différents composants de SMACH	47
Figure 3.6 : Diagramme d'activité pour trois occupants d'un foyer implémenté dans SMACH.....	50
Figure 3.7 : Architecture général de l'application	52
Figure 3.8 : Diagramme de cas d'utilisation.....	53
Figure 3.9 : Schéma de diagramme d'activité du simulateur	54
Figure 3.10 : Première interface	55
Figure 3.11 : Le scénario téléchargé de l'activité dormir.....	56
Figure 3.12 : Les capteurs de l'activité dormir.....	56
Figure 3.13 : Choisir le nom d'agent humain.....	57
Figure 3.14 : Résultat de la simulation.....	57
Figure 3.15 : Nouvelle simulation.....	58
Figure 3.16 : Les capteurs de la nouvelle simulation.....	59

Liste des tableaux :

Table 1.1 : Avantages de la technologie agent	9
Table 1.2 : Plateformes à objectif général (encore utilisées)	15
Table 1.3 : Plateformes de simulation et de modélisation (encore utilisées).....	17
Table 1.4 : Plateformes à objectif spécial (encore utilisées)	18
Table 1.5: Les différentes topologies d'organisation	21
Table 2.1 : Caractéristiques principales des systèmes de RAH	31

Liste des Acronymes :

ACL : Agent Communication Language.

API : Application Programming Interface.

BDI : Beliefs-Desires-Intentions (Croyances-Désirs-Engagement).

EDF : Electricité De France.

FIPA: Foundation for Intelligent Physical Agents.

GPS : Global Positioning System.

IAD : Intelligence Artificielle Distribuée.

INSEE : Institut National de la Statistique et des Etudes Economiques.

KQML : Knowledge Query and Manipulation Language.

NIKE : National Indian Knitting Enterprise.

RAH : Reconnaissance d'Activité Humaine.

RFID : Radio Frequency Identification.

SMA : Système Multi-Agents.

SMACH : Simulation Multi-Agents des Comportements Humains.

UbiREAL: Ubiquitous Application Simulator with REAListic Environments.

TUS : Time Use Surveys.

Introduction générale :

Le domaine des systèmes multi-agents se situe à l'intersection de l'intelligence artificielle, branche de l'informatique qui vise la recherche de moyens susceptibles de doter les systèmes informatiques de capacités de raisonnement comparables à celles des êtres humains et des systèmes distribués, connectés à l'aide d'un réseau de communication.

Les systèmes multi-agents naissent d'une volonté de distribuer l'intelligence d'un système dans le but de faire résoudre plus efficacement par plusieurs entités (les agents), des problèmes qui, par leur nature, ne sont pas adaptés à une résolution centralisée (trop complexes, distribués, etc).

D'une manière générale, un système multi-agents est un ensemble d'agents situés dans un certain environnement et interagissant selon une certaine organisation. Ces derniers doivent être capables de réaliser leurs objectifs d'une manière autonome, c'est-à-dire qu'une fois le système et les agents créés, les agents doivent être capables d'évoluer sans que l'intervention de leur concepteur ne soit nécessaire.

La reconnaissance des activités humaines (RAH) est un domaine important : reconnaître les activités d'un patient aide à diagnostiquer sa maladie, les patients en coma pourraient être surveillés par ces systèmes afin de détecter des activités anormales.

L'objectif général de ce travail est de concevoir et réaliser un simulateur pour activités humaines. Et pour cela on s'est basé sur les systèmes multi-agents.

Ce mémoire est organisé en trois chapitres comme suit :

Dans le premier chapitre on a présenté des généralités sur les agents et systèmes multi-agents (SMA).

Dans le deuxième chapitre on a donné des généralités sur la reconnaissance d'activités humaines.

Dans le troisième chapitre on a étudié le fonctionnement du simulateur SMACH et on a décrit la manière de conception et de réalisation de notre simulateur. On termine par une conclusion générale et quelques perspectives.

1

Généralités sur les agents et systèmes multi-agents (SMA)

1.1 Introduction :

Les agents autonomes et les Systèmes Multi-Agents (SMA) ont donné lieu à une nouvelle façon de voir, d'analyser, de modéliser et de concevoir les systèmes informatiques complexe. Pour certains chercheurs, il s'agit d'une nouvelle étape dans la modélisation des applications, qui suit celle du paradigme objet. Les systèmes multi-agents (SMA) sont des systèmes informatiques distribués. Comme la plupart des systèmes distribués, ils sont composés d'entités informatiques qui interagissent entre elles.

A la différence des systèmes distribués classiques, les entités qui les constituent sont " intelligentes". En effet, le domaine des systèmes multi-agents est issu du domaine de l'Intelligence Artificielle Distribuée (IAD).

Dans ce chapitre, nous allons décrire les concepts du SMA. L'intérêt premier est porté d'abord sur le concept de "l'Agent" qui constitue la brique fondamentale du domaine. Ensuite nous précisons la notion de " systèmes multi-agents" en définissant les concepts clés tels que l'interaction, l'organisation et la communication.

1.2 Agent :

De nos jours, la notion d'agent est utilisée dans plusieurs domaines : sociologie, biologie, psychologie cognitive, psychologie sociale, informatique, et plusieurs significations sont attachés car de nombreux chercheurs ont défini ce terme de manières différentes.

1.2.1 Définition Agent :

Il n'existe pas actuellement une définition du concept agent acceptée universellement par les chercheurs concernés [1].

L'une des premières définitions de l'agent a été fournie par J. Ferber [2]:

Un agent est une entité autonome, réelle ou virtuelle, évoluant dans un environnement, capable de percevoir et d'agir dessus, qui dans un univers multi-agents, peut communiquer avec d'autres agents, et dont le comportement

est la conséquence de ses observations, de ses connaissances et des interactions avec les autres agents.

Jennings, Sycara et Wooldridge [3] proposent la définition suivante :

Un agent est un système informatique, situé dans un environnement, et qui agit d'une façon autonome et flexible pour atteindre les objectifs (buts) pour lesquels il a été conçu. Les notions « situé », « autonomie » et « flexible » sont définies comme suit :

-**situé** : l'agent est capable d'agir sur un environnement à partir des entrées sensorielles qu'il reçoit de même environnement. Exemples : systèmes de contrôle de processus, systèmes embarqués.

- **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 se caractérise par la réactivité, la pro activité, et l'aspect social.

Plus tard, Russell et Norvig [4] ont proposé la définition suivante pour un agent :

On appelle agent toute entité qui peut être considérée comme percevant son environnement grâce à des capteurs et qui agit sur cet environnement via des effecteurs.

1.2.2 Propriétés des agents:

A partir des différentes définitions du concept "Agent", on peut extraire les propriétés clés d'un agent : l'agent doit être autonome, flexible, situé dans un environnement, il agit, il perçoit et il communique. Nous essayons maintenant de clarifier ces notions.

1. Autonomie : Le système peut agir sans une intervention directe d'humains ou d'autres agents, et que le système possède le contrôle total sur ses propres actions et sur son état interne [2].

2. Flexibilité : L'agent doit également être flexible, la flexibilité peut être vue comme une forme de l'intelligence. Etre flexible signifie que l'agent est [5]:

(a) **Réactif** : L'agent doit être capable de percevoir son environnement et de répondre, en temps aux changements qui peuvent affecter cet environnement [5].

(b) **Proactif** : L'agent n'agit pas seulement d'une manière réactive (en fonction de son environnement) mais il doit avoir un comportement orienté objectifs et que l'agent peut prendre des initiatives [5].

(c) **Social** : L'agent est capable d'interagir avec d'autres agents intelligents et humains pour qu'il puisse atteindre ses propres objectifs et aider les autres dans leurs activités [5].

3. Situation dans un environnement : L'agent est implanté dans l'environnement, sur lequel il agit à partir des entrées sensorielles qu'il reçoit de ce même environnement [5].

4. Perception : La perception est un moyen de recevoir de l'information d'un environnement qui est souvent complexe et évolutif. L'agent doit percevoir directement son environnement par ses propres capteurs [5].

5. Action : L'agent est capable d'agir, et non pas seulement de raisonner comme dans les systèmes d'IA classique. L'action, repose sur le fait que les agents accomplissent des actions qui vont modifier l'environnement des agents et donc leurs prises de décision futures [5].

6. Communication : Un agent fait un acte délibéré de transfert d'informations vers un ou plusieurs autres agents. La communication entre agents est l'un des modes principaux d'interaction existant entre les agents [5].

1.2.3 Types d'agents :

On distingue trois types d'agents : Agents réactifs, Agents cognitifs et Agents Hybrides.

1.2.3.1 Agents réactifs : les agents réactifs sont basés sur l'intelligence artificielle réactive. Cette école de l'intelligence artificielle est basée sur la possibilité de concevoir des comportements intelligents à partir de comportements simples. L'exemple de fourmis est un exemple typique dans ce

cas. En fait, le comportement des fourmis est un comportement simple qui n'est pas doté d'une vraie intelligence. Cependant, l'interaction des fourmis produit des comportements intelligents. En conséquence, les agents réactifs sont des agents qui ne possèdent pas une représentation de leur environnement et des autres agents.

Ils sont incapables de prévoir ce qui va se passer et donc d'anticiper en planifiant les actions à accomplir [2]. Leur principe de réaction est liés directement à leur perception d'environnement, ou ils réagiront et interagiront de façon dynamique selon les données fourni en temps réel sans mémorisation.

Les principales caractéristiques d'un agent réactif sont les suivantes :

- ✓ Pas de mémoire de passé.
- ✓ L'absence du mécanisme du raisonnement.
- ✓ Pas de représentation explicite de l'environnement.
- ✓ La forme de prise de décision se fait sous «stimuli-réponse» qui produit des actions à partir de perceptions.
- ✓ L'organisation entre les agents inspirés du phénomène biologique.
- ✓ L'implémentation du système multi agents nécessite un grand nombre d'agents homogènes.

Ces caractéristiques permettent des avantages tels que la fiabilité grâce à leur autonomie (pas d'influence sur le travail générale en cas de perte d'agent), et la grande flexibilité vis-à-vis du changement dynamique de son environnement. En effet ce mécanisme permet de ne pas concevoir un système rigide et facilite la réalisation d'ajustement en cours d'exécution, donc gère la flexibilité de système de manière à le rendre plus robuste [6].

On présente ci-dessous l'architecture d'un agent réactif :

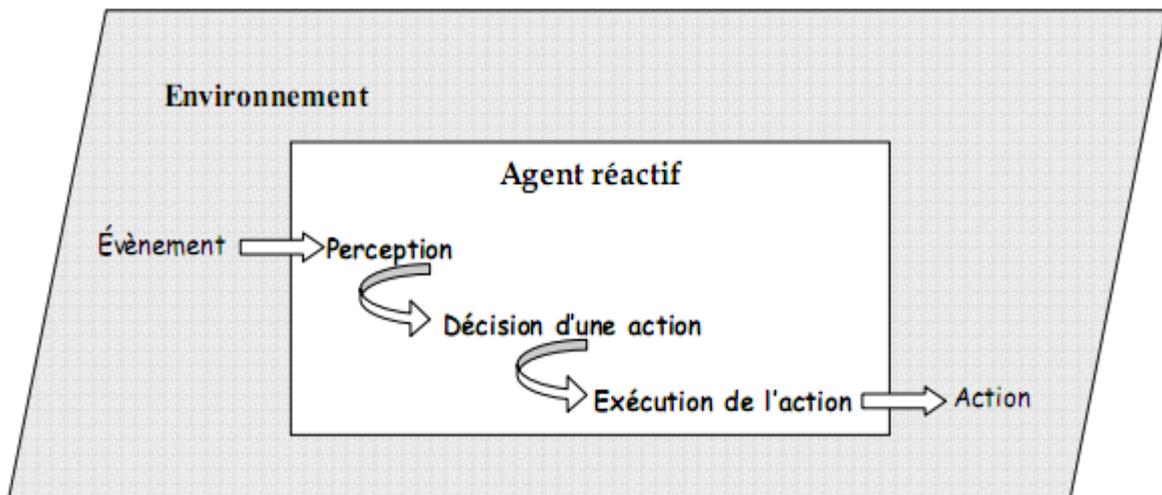


Figure 1.1 : Fonctionnement d'un agent réactif [7].

1.2.3.2 Agents cognitifs : cette classe des agents fait partie de l'intelligence artificielle symbolique. Ainsi, l'agent possède une représentation symbolique de son environnement, des autres agents, des actions et des états internes des agents. Sachant que les buts de l'agent sont modélisés comme des états internes, le raisonnement de l'agent consiste à trouver l'ensemble des actions possibles pour satisfaire ses buts. Ainsi, le problème essentiel dans cette catégorie des agents devient un problème de planification. Bien entendu, il est possible de prendre en compte plusieurs attributs dans le processus de raisonnement comme les croyances de l'agent sur les croyances des autres agents ou l'historique de l'exécution de l'agent. Cependant, l'intégration de divers attributs dans le processus de raisonnement peut compliquer ce dernier. En conséquence, la performance devient une question pertinente dans cette classe d'agents [2].

A partir de cette définition, on peut faire ressortir les caractéristiques les plus importantes pour ce type d'agent :

- ✓ Représentation explicite (formellement expliqué).
- ✓ Mémoire locale, pour mémoriser des données sur l'état précédent
- ✓ Réaction basé sur la planification.
- ✓ Organisation sociale.
- ✓ Nombre réduit d'agents hétérogènes.

Ces caractéristiques offrent des avantages tels que une performance fiable pour des actions en temps réel, mais aussi des inconvénients majeurs tels que : la complexité de communication, un temps important de réalisation des tâches [6].

La figure 1.2 ci-dessous explicite le fonctionnement d'un agent cognitif :

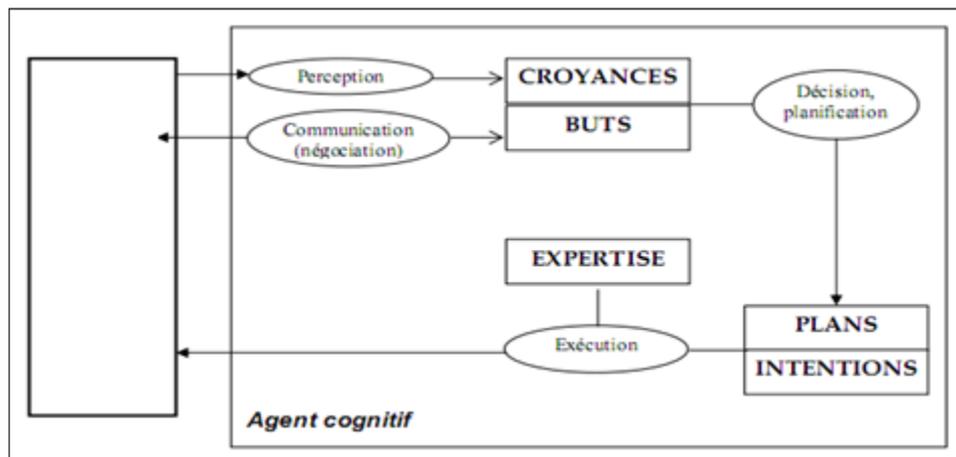


Figure 1.2 : Architecture d'un agent cognitif [7].

1.2.3.3 Agents hybrides : cette architecture vient pour résoudre des problèmes posés précédemment par les deux architectures cognitive et réactive. C'est la fusion ou le couplage entre l'agent cognitif et réactif dans le même environnement, pour tenter de bénéficier des avantages de chacune d'elles. On parle également d'architectures multicouches, qui se basent sur la hiérarchie de niveaux [6].

Un agent hybride est un agent conçu en couches. Les couches inférieures assurent des comportements réactifs. Par contre, les couches supérieures sont responsables de capacités cognitives complexes comme les aspects sociaux. Bien entendu, les couches supérieures manipulent des connaissances alors que les couches inférieures manipulent directement des données. En conséquence, des couches intermédiaires sont responsables de transformer des données en connaissances [8].

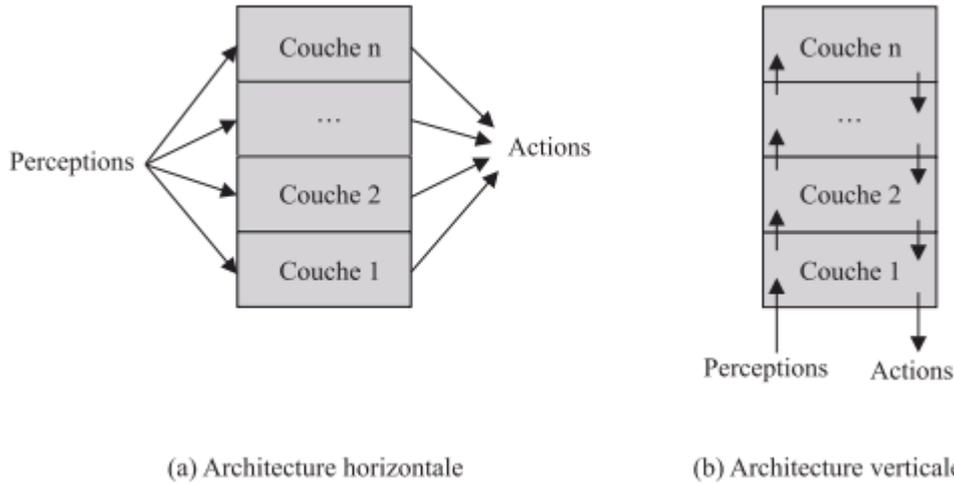


Figure 1.3 : Architectures d’agents en couches [3].

1.2.4 Avantages de la technologie agent :

La technologie agent présente plusieurs avantages, le tableau suivant les résume.

Caractéristiques	Avantages	Bénéfices
Automatisation	Le dispositif est adaptable facilement aux comportements répétitif d’un utilisateur ou similaire de plusieurs dans un groupe.	Augmenter la productivité individuelle ou d’un groupe.
Personnalisation	Modèle compatible avec l’utilisateur et l’équipement (selon le profil et les préférences d’utilisateur).	Réduit la quantité d’information présentée.
Notification	Prévenir l’utilisateur des événements importants.	Réduit la charge du travail.
Apprentissage	Apprend à reconnaître le comportement des utilisateurs : cible les	Architecture interne simple

	taches d'automatisation pour une meilleure personnalisation.	
Tutorat	Suit et dirige l'utilisateur dans un contexte particulier.	Réduit le temps de formation.
Communication	Réalise les tâches à distance.	Travail local en mode déconnecté

Table 1.1 : Avantages de la technologie agent [9].

1.3 Système multi-agents :

Dans cette section, nous présentons ce qui appelé système multi-agents en donnant certaines de ses caractéristiques et quelques avantages.

1.3.1 Définitions :

Définition 1.3.1. On appelle un système multi-agents (ou SMA), un système composé des éléments suivants :

- **Un environnement (E)** : Représente un espace disposant généralement d'une métrique.
- **Un ensemble d'objets (O)** : ces objets sont situés, c'est-à-dire qu'il est possible, à un moment donné, d'associer une position dans E. Ces objets sont passifs, c'est à dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents.
- **Un ensemble d'agents (A)** : ce sont des objets particuliers ($A \subseteq O$), lesquels représentent les entités actives du système.
- **Un ensemble de relations (R)** : permet de réunir des objets (et donc des agents) entre eux.

- **Un ensemble d'opérations (Op)** : permettant aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O.
- Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification [2].

La Figure 1.4 explique le fonctionnement du SMA :

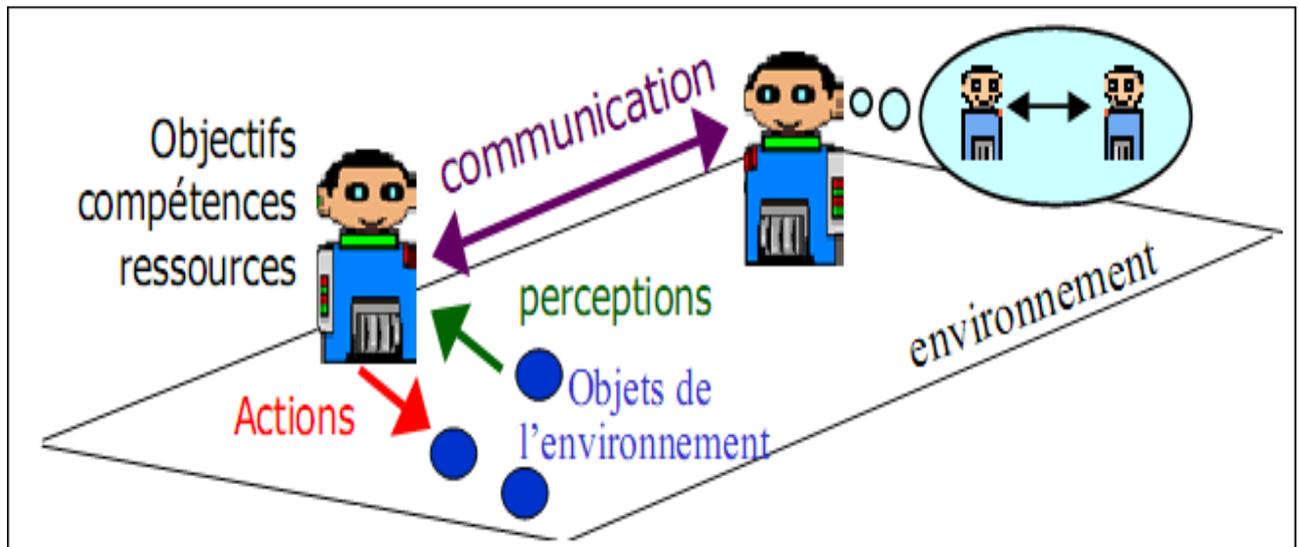


Figure 1.4 : Fonctionnement du Système multi-agents [2].

Définition 1.3.2. Un système multi-agents est un système distribué composé d'un ensemble d'agents. L'interaction entre ces agents se présente sous forme de coopération, coordination, compétition, négociation, concurrence, etc. Chaque agent du système possède généralement un certain point de vue sur son environnement et les autres agents qui l'entourent.

Les agents ont des capacités limitées et ne possèdent généralement que des informations incomplètes et partielles. La gestion d'un système multi-agents est totalement distribuée : il n'existe pas de système central de contrôle, les données et les informations sont décentralisées et le fonctionnement du système est entièrement asynchrone [10].

Selon la nature de l'application, un système multi-agents peut être :

- ✓ **Ouvert** : Les agents y entrent et sortent librement.

- ✓ **Fermé** : L'ensemble des agents constituant le système reste inchangé.
- ✓ **Homogène** : Tous les agents composants le système sont construits sur le même modèle d'organisation.
- ✓ **Hétérogène** : Les agents composants le système ont des modèles différents [5].

1.3.2 Caractéristiques d'un SMA :

Un système multi-agents possède plusieurs caractéristiques dont les principales sont :

- ✓ L'hétérogénéité des agents où les messages doivent être mutuellement compréhensibles.
- ✓ L'échange de savoir où chaque agent coopératif au sein du SMA doit pouvoir exprimer ses différents types de connaissances.
- ✓ Les agents doivent être autonomes, c'est-à-dire, leur comportement ne doit dépendre ni d'un planificateur central ni d'interactions prédéfinies.
- ✓ Chaque agent a des informations ou des capacités de résolution de problèmes limitées, ainsi chaque agent a un point de vue partiel.
- ✓ Les données sont décentralisées.
- ✓ Le calcul et les traitements sont asynchrones.
- ✓ Il n'y a aucun contrôle global du système multi-agents (contrôle reparté).
- ✓ Les schémas d'interaction sophistiqués : ils incluent la coopération, la coordination et la négociation [9].

1.3.3 Environnement multi-Agent :

Les mondes dans lesquels les agents évoluent sont appelés des environnements. Ils peuvent être de nature très différente allant du plus simple au plus complexe.

L'environnement fournit une structure spécifiant les protocoles de communication et d'interaction. Il est ouvert, non centralisé et contient des agents autonomes et distribués qui peuvent agir soit pour leur intérêt personnel,

soit en coopération avec les autres agents de l'environnement. L'environnement peut être considéré selon deux points de vue :

Selon le point de vue du système multi-agent : l'environnement est tout ce qui est extérieur au système multi-agent. L'environnement du système multi-agent contient donc les autres systèmes multi-agents, les objets, les variables globales, les lois et les moyens techniques permettant la simulation du système multi-agent.

Selon le point de vue de l'agent : cet environnement est tout ce qui est extérieur à l'agent. L'environnement de l'agent contient alors tout ce que contient l'environnement du système multi-agent ainsi que les autres agents [11].

Les principales propriétés de cet environnement sont :

- **Accessible/ inaccessible :** l'agent peut percevoir l'état complet, valide et à jour de l'environnement alors cet environnement est dit accessible. La plupart des environnements sont inaccessibles.
- **Déterministe/non-déterministe ou stochastique :** dans un environnement déterministe, l'action d'un agent a un et un seul effet possible sur l'environnement. L'état de l'environnement est donc prévisible en fonction des actions des agents du système.
- **Épisodique / non-épisodique:** On parle d'environnement épisodique (on dit aussi que le temps est discrétisé) si l'expérience de l'agent peuvent être divisée en "épisodes". Chaque épisode consiste en une phase de perception puis en une phase d'action, le résultat de l'action ne dépendant que de la période courante.
- **Statique/ dynamique :** un environnement est dit dynamique s'il peut changer pendant la prise de décision de l'agent. Il est dit statique dans le cas contraire.

- **Discret/ continu** : si le nombre de perceptions et d'actions distinctes possibles est limité, on parlera d'environnement discret. Sinon, l'environnement est continu [12].

1.3.4 Plates-formes multi-agents :

1.3.4.1 Définition : une plate-forme est un ensemble d'outils nécessaire à la construction et à la mise en service d'agents au sein d'un environnement spécifique. Ces outils peuvent servir également à l'analyse et au test du SMA ainsi créé, ils peuvent être sous la forme d'environnement de programmation (API) et d'applications permettant d'aider le développeur à la programmation d'un SMA [13].

1.3.4.2 Catégorisation :

Depuis plusieurs années, on a vu se multiplier les plateformes de développement agents. Ces plates-formes sont dédiées soit à un certain type d'agents (cognitifs, réactifs) soit à un certain type d'applications (simulation, mobilité). Cependant il n'existe pas à notre connaissance de plate-forme générique qui permette de réaliser des applications dans tous ces domaines. Les plateformes peuvent ainsi être classées en plusieurs catégories :

- **Plates-formes de simulation (Cormas, Swarm, etc.)**: reproduisent l'environnement ou le comportement d'un système complexe afin d'en étudier la dynamique. Elles permettent aux développeurs de concevoir et réaliser leurs modèles sans perdre de temps à réaliser des fonctions de base pour la création et l'interaction entre agents et éliminent, dans la plupart des cas, la nécessité d'être familier avec les différents concepts théoriques des systèmes multi-agents.
- **Plates-formes de développement (AgentBuilder, Madkit, Zeus, Magique, Dima, ...)** : servent de support à une méthode en fournissant des outils pour assister une démarche de conception. Elles sont des

infrastructures de logiciels utilisées comme environnement pour le déploiement et l'exécution d'un ensemble d'agents. Elles devraient permettre donc de créer, exécuter et supprimer des agents.

- **Plates-formes d'exécution (Jade, Zeus, FIPA-OS, JATLite ...)** : proposent des outils d'implémentation à partir de modèles particuliers. Elles permettent seulement d'exécuter un ensemble d'agents. Les développeurs doivent donc coder leurs agents dans un environnement de développement puis les exporter pour les exécuter. Toutefois la plupart de ces plates-formes offrent des outils (Classes, Packages, ..., etc.) pour faciliter le développement des agents [14].

1.3.4.3 Différentes plates-formes et leurs caractéristiques :

Les plateformes peuvent être à objectif général (Table 1.2) ou dédiées simulation et modélisation (Table 1.3) ou à d'autres objectifs spécialisés (Table 1.4).

Outils	Langage	Caractéristiques
AGLOBE, AglobeX Simulation	Java	Conçu pour le test (et aussi pour la modélisation et le développement) de systèmes multi-agent décentralisés
Cybele	Similaire à Java	Commercialisé et utilisé par le gouvernement, l'industrie et le secteur académique en robotique, planification, data mining(extraction de données), modélisation et simulation et contrôle de systèmes de transport aériens et urbains, communication réseaux.
JaCaMo	AgentSpeak (Jason)	Un framework de programmation Multi-agent combinant trois technologies séparées: Jason pour la programmation d'agent autonomes; CartAgO (Common ARTifact infrastructure for AGents Open environments) pour la programmation des artefacts

		d'environnement et Moise pour la programmation des organisations multi-agent.
JACK	JACK Agent Language, a super-set of Java	Un framework de développement de systèmes multi-agent, utilisant (parmi peu d'autres) le modèle BDI et fournissant son propre Langage de plan basé-Java et des outils de planification graphiques.
JADE (Java Agent DEvelopment)	Java, C#/.NET (JADE LEAP)	Framework implémenté en Java simplifiant l'implémentation de systèmes multi-agent à travers un middleware compilé avec les spécifications FIPA et des outils graphiques supportant les phases de débogage et de déploiement.
JADEX (JADE extension)	Java	Une couche rationnelle au-dessus de JADE permettant un développement facile d'agents rationnels selon un paradigme de modélisation BDI (Belief, Desire, Intention). Les agents sont écrits en XML et les corps de plan en Java. ActiveComponents nouvelle version se focalisant sur les web services.
BDI4JADE	Java	Fourni une implémentation d'architecture BDI comme unecouche au-dessus de JADE. Le but est de fournir un environnement pour l'implémentation d'applications d'entreprise, ce qui implique une intégration totale des technologies standards.
Jason	AgentSpeak	Plateforme de développement de systèmes multi-agent. AgentSpeak est un des plus influant langages abstrait basé sur l'architecture BDI. Ses agents sont parfois mentionnés comme des systèmes de planification réactifs. Jason inclut aussi une communication inter-agent basée speech-act.

MADKIT	Java	Une bibliothèque Java légère pour la conception et la simulation de systèmes multi-agent. MADKIT est conçu pour construire facilement des applications et des simulations distribuées et suit une approche centrée organisation (OCMAS) au lieu d'une centrée agent (ACMAS) et donc n'impose pas un modèle prédéfini d'agent. Madkit est construite sur le modèle organisationnel AGR: l'Agent joue des Rôles dans des Groupes et crée donc des sociétés artificielles.
Soar	Soar	Soar est une architecture cognitive générale pour développer les systèmes qui exhibent un comportement intelligent. SoarMarkuplanguage permet aux agents la communication avec des environnements externes et même avec des agents écrits en d'autres langages.

Table 1.2 : Plateformes à objectif général (encore utilisées) [15].

Outils	Langage	Caractéristiques
Gama	Java	Plateforme de simulation fournissant aux experts de domaines, aux informaticiens et aux modeleurs un environnement de développement complet de modélisation et de simulation pour construire des simulations multi-agents spatiales explicites.
NetLogo	Logo dialect	Environnement programmable de modélisation pour la simulation de phénomènes naturels et sociaux. Privilégié pour la modélisation de systèmes complexes se développant dans le temps.
PRESAGE2	Java	Plateforme de simulation de prototypage rapide de

		sociétés d'agents.
Repast	Java, Python, C#/.NET, C++, ReLogo, Groovy	Conçu pour des applications de science sociale, supporte le développement de modèles très flexibles d'agents en interaction pour utilisation en stations de travail.

Table 1.3 : Plateformes de simulation et de modélisation (encore utilisées) [15].

Outils	Langage	Caractéristiques
AgentSheets	Programmation conversationnelle	Création de jeux et simulation, démonstration interactive, applications informatiques pour les multimédias.
Framsticks	FramScript (similar à JavaScript, Java), C++	Simulation 3D de vie (modélisation de structures mécanique "corps" et systèmes de contrôle "cerveaux").
MATSim	Java pour développeurs. Interface graphique, fichiers de configuration pour les utilisateurs.	Implémente des simulations de transport basées-agent à grande échelle.
Mobile-C	JACK Agent Language, a super-set of Java	Un framework de développement de systèmes multi-agent supportant des agents mobiles, utilisant (parmi peu

		d'autres) le modèle BDI et fournissant son propre Langage de plan basé-Java et des outils de planification graphiques.
Orleans	C#	Construire des applications de calcul distribué à grande échelle, sans avoir besoin d'apprendre des techniques de programmation complexes pour gérer la concurrence, tolérance aux pannes et gestion de ressources. Déployable en cloud.
StarLogo	Graphicaluser interface, modèles construits avec "blocks"	Environnement de programmation permettant aux étudiants et aux enseignants de créer des jeux et simulations 3D pour comprendre les systèmes complexes, jeux vidéo éducationnels.
TuCSon (Tuple Centres Spread over theNetwork)	ReSpecT, a logicbased Coordination language	Bibliothèque Java, c'est un modèle de coordination de processus distribués d'agents autonomes, intelligents et mobiles.

Table 1.4 : Plateformes à objectif spécial (encore utilisées) [15].

1.3.5 Avantages :

L'utilisation des Systèmes Multi-Agents (SMA) présente une série d'avantages, qui sont résumés comme suit :

- **Système dynamique** : contrairement aux approches traditionnelles de l'Intelligence Artificielle qui simulent, dans une certaine mesure, les

capacités du comportement humain, les SMA permettent de modéliser un ensemble d'agents qui interagissent.

Les agents sont structurés afin d'exercer une influence sur chacun pour faire évoluer le système dans son ensemble (système dynamique). On rencontre de nombreuses interactions entre agents telles que la coordination (organiser la résolution d'un problème de telle sorte que les interactions nuisibles soient évitées, ou que les interactions bénéfiques soient exploitées), la négociation (parvenir à un accord acceptable pour toutes les parties concernées), la coopération (travailler ensemble à la résolution d'un but commun).

- **Robustesse et sûreté de fonctionnement** : la mise hors fonctionnement de quelques agents ne modifie pas sensiblement le comportement global du système.
- **Souplesse de l'outil informatique** : qui permet de modifier le comportement des agents, ajouter ou supprimer des actions possibles, étendre les informations disponibles à l'ensemble des agents.
- **Flexibilité et traitement des systèmes à grandes échelles** : on peut toujours augmenter le nombre d'agents pour traiter des systèmes de plus en plus gros, sans pour autant perturber le travail des agents existants.
- **Résolution distribuée de problèmes** : il est possible de décomposer un problème en sous-parties et de résoudre chacune de façon indépendante pour aboutir à une solution stable.
- **Schémas d'interaction sophistiqués** : en utilisant la coopération, la coordination et la négociation [16].

1.3.6. Interaction entre agents :

Un système multi-agents 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 [9].

Chaque agent peut-être caractérisé par trois dimensions : ses buts, ses capacités à réaliser certaines tâches et les ressources dont il dispose. Les interactions des

agents d'un SMA sont motivées par leurs interdépendances selon ces trois dimensions : Leurs buts peuvent être compatibles ou non ; les agents peuvent désirer des ressources que les autres agents possèdent ; un agent peut disposer d'une capacité nécessaire à un autre agent pour achever votre action [17].

En général, les interactions sont mises en œuvre par un transfert d'informations entre agents ou entre l'environnement et les agents, soit par perception, soit par communication.

L'interaction peut être décomposée en trois phases non nécessairement séquentielles:

- La réception d'informations ou la perception d'un changement,
- Le raisonnement sur les autres agents à partir des informations acquises,
- Une émission de message(s) ou plusieurs actions (plan d'actions) modifiant l'environnement. Cette phase est le résultat d'un raisonnement de l'agent sur son propre savoir-faire et celui des autres agents [18].

Les situations sont diverses et variées: l'aide d'un robot par un autre, l'échange de données entre serveurs informatiques, l'utilisation d'une imprimante par deux programmes simultanément, la répartition de charge sur plusieurs processeurs, etc. les interactions sont non seulement la conséquence d'actions effectuées par plusieurs agents en même temps, mais aussi l'élément nécessaire à la constitution d'organisations sociales [2].

1.3.7 Organisation des agents :

Le besoin d'organisation est nécessaire lorsque des objectifs communs à l'ensemble des entités doivent être atteints. Une organisation est un schéma de prise de décision et de communication appliqué à un ensemble d'acteurs qui réalisent un ensemble de tâches fin de satisfaire des buts tout en assurant un état global cohérent [19] .

Parmi les modèles organisationnels des systèmes multi-agents les plus répandus, on cite :

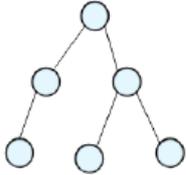
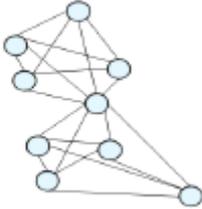
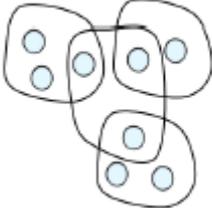
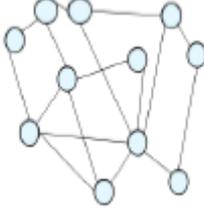
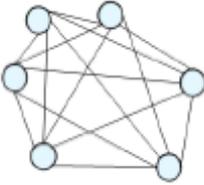
Nom Architecture	Architecture
Le modèle hiérarchique.	
Le modèle holonique	
Le modèle orienté groupe.	
Le modèle peer-to-peer.	
Le modèle de diffusion (broad-cast)	

Table 1.5: Les différentes topologies d'organisation [19].

1.3.8 Communication entre agents :

Les communications dans les systèmes multi-agents, comme chez les humains, sont à la base des interactions et de l'organisation des agents. La

communication est donc un outil permettant l'interaction (coopérative ou antagoniste) [2].

En communiquant, les agents peuvent échanger des informations et coordonner leurs activités.

1.3.8.1 Modes de communication :

On peut distinguer deux modes de communication ; la communication directe et la communication indirecte.

- Communication indirecte : dans ce mode de communication les agents se communiquent par signaux via l'environnement. Ces signaux, une fois interprétés, vont produire des effets sur les agents⁹. Ce type de communication est spécifique aux agents réactifs.
- Communication directe : l'échange direct est réalisé volontairement en direction d'un individu ou groupe d'individus soit :
 - Par partage d'informations : les agents se communiquent en utilisant la technique du tableau noir ou « Blackboard » [4] [20] [21].

Dans cette technique, on suppose l'existence d'un espace où les agents peuvent partager et échanger les informations relatives à la résolution de problème.

Par envoi de messages : la communication se fait par échange de messages entre les agents, ce mode de communication provient initialement du domaine des acteurs [22]. Cette approche s'est inspirée de l'interaction sociale telle que nous la trouvons dans d'autres contextes comme la communication entre les humains.

1.3.8.2 Langages de communication :

L'intérêt des langages de communication est de faciliter l'échange et l'interprétation des messages et l'interopérabilité entre les agents. Ces langages se focalisent essentiellement sur la manière de décrire exhaustivement des actes de communication d'un point de vue syntaxique et sémantique. Deux langages de communication entre agent, FIPA-ACL et KQML, ont émergé des efforts de

standardisation de la communauté des SMA, portés par des besoins d'interopérabilité logicielle.

Le premier langage qui a été introduit est KQML (*KnowledgeQuery and Manipulation Language*). A l'origine, KQML a été développé pour échanger des informations et des connaissances entre des systèmes à base de connaissances. Il a été ensuite repris pour décrire les messages échangés entre les agents.

FIPA-ACL (*FIPA Agent Communication Language*), est proposé dans le cadre d'un travail de standardisation mené au sein l'organisation FIPA (*Foundation of Intelligent Physical Agents*). Ce langage est fondé sur vingt et un actes communicatifs, exprimés par des performatifs, qui peuvent être groupés selon leurs fonctionnalités de la façon suivante:

- Passage d'information : Inform, Inform-if, Inform-ref, Confirm, Disconfirm,
- Réquisition d'information : Query-if, Query-ref, Subscribe,
- Négociation : Accept-proposal, Cfp, Propose, Reject-proposal,
- Distribution de tâches (ou exécution d'une action) : Request, Request-when, Requestwhenever, Agree, Cancel, Refuse.
- Manipulation des erreurs : Failure, Not-understood [24] [25].

1.3.9 Coordination:

De nombreux exemples de coordination existent dans la vie quotidienne: deux déménageurs déplaçant un meuble lourd, deux jongleurs échangeant des balles avec lesquelles ils jonglent, des personnes qui parlent à tour de rôle en se passant un micro, etc. Les composantes fondamentales de la coordination entre agents sont l'allocation de ressources rares et la communication de résultats intermédiaires [1].

Dans ce contexte, les agents doivent être capables de communiquer entre eux de façon à pouvoir échanger les résultats intermédiaires. Pour l'allocation des

ressources partagées, les agents doivent être capables de faire des transferts de ressources.

Bien entendu, la coordination est une question centrale pour les SMA et la résolution de systèmes distribués [26].

Il existe plusieurs types de coordination, suivant la nature des agents en jeu. On parlera de coopération entre des agents non-antagonistes et de négociation entre des agents compétitifs ou simplement indépendants (Figure 1.5). Une bonne coopération nécessite que chaque agent maintienne une représentation de ses collègues et des futures interactions entre eux. Cela présuppose des comportements qui intègrent les autres (sociabilité).

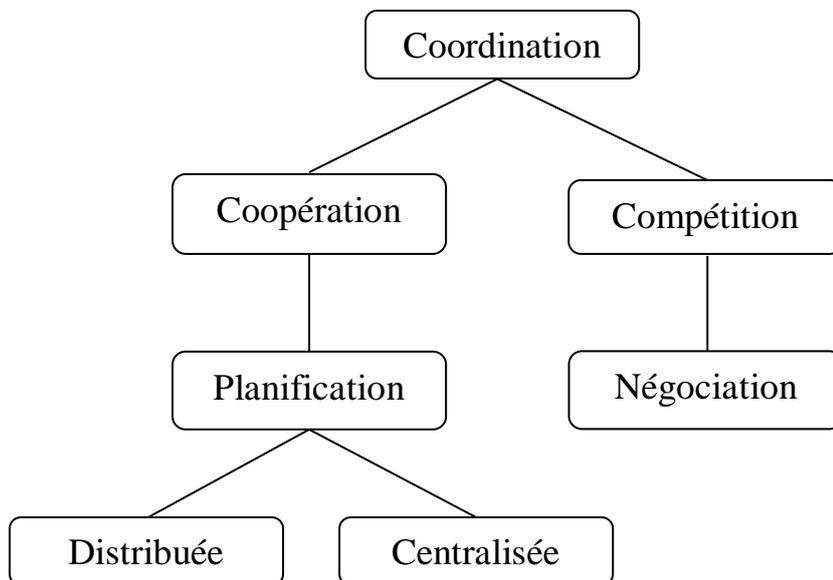


Figure 1.5 : Les différents types de coordination entre agents [26].

1.3.9.1 Coopération :

La coopération est une caractéristique très importante des SMA. En effet, une résolution distribuée d'un problème ou le partage des tâches et de résultats et les résultats d'interactions coopératives entre les différents agents qui disposent d'un ensemble des caractéristiques qui leur rendent coopératifs.

Ferber a donné une définition plus précise de la coopération : « On dira que deux agents coopèrent, ou encore qu'ils sont en situation de coopération, si l'une des deux conditions est vérifiée » :

1. L'ajout d'un nouvel agent permet d'accroître d'une manière différentielle les performances du système.
2. L'action des agents sert à éviter ou à résoudre des conflits potentiels ou actuels.

C'est surtout le dernier point qui rend la négociation une interaction coopérative dans le cas où elle interviendrait pour résoudre les conflits [27].

1.3.9.1.1 Planification entre agents coopératifs :

Les techniques de planification centralisée pour des groupes d'agents, de conciliation de plans, de planification distribuée, d'analyse organisationnelle, sont toutes des façons d'aider les agents à aligner leurs activités en assignant les tâches après avoir raisonné sur les conséquences de réaliser ces tâches dans des ordres particuliers. Dans une approche de planification multi-agents, un plan multi-agents est un plan qui est créé afin que plusieurs agents puissent l'exécuter. La création d'un tel plan peut être faite par un seul ou par plusieurs agents.

En planification multi-agents centralisée, un agent est responsable de la création du plan qui spécifie les actions planifiées pour tous les agents concernés. Dans une autre façon d'implémenter la planification multi-agent centralisée, les plans des agents sont d'abord créés de façon individuelle; ensuite un agent centralisateur rassemble ces plans et les analyse pour identifier les conflits. L'agent centralisateur en question essaye de résoudre les conflits en modifiant les plans locaux des autres agents et en introduisant des commandes de communication afin que les agents se synchronisent de façon appropriée.

Par ailleurs, dans une approche de planification distribuée, les activités de planification sont réparties au sein d'un groupe d'agents. Cette approche est utilisée quand un seul agent ne peut pas avoir une vue globale des activités du groupe.

En général, la planification multi-agents nécessite une forme ou une autre de synchronisation de plans qui peut être réalisée à divers moments: pendant la

décomposition de plan, pendant la construction de plan ou après celle-ci. Les plans des agents peuvent être en conflits en raison d'incompatibilités d'états des systèmes, de l'ordre des activités ou de l'usage des ressources. De tels conflits peuvent être résolus par un agent en particulier (coordonnateur ou médiateur) ou une solution peut être obtenue par négociation.

Dans un SMA, les agents doivent gérer des ressources distribuées qui peuvent être physiques (capacités de communication, matières premières, argent) ou informationnelles (telles que les informations au sujet de la décomposition du problème). Les agents doivent adapter leurs plans pour tenir compte de la disponibilité des ressources. Bien entendu, la planification contribue à la coordination, dans la mesure où lorsque les agents adoptent un plan " bien fait ", ils agissent généralement de manière coordonnée [28].

1.3.9.2 Compétition :

C'est une interaction entre des agents égocentrés se basent principalement sur la négociation qui est détaillée dans ce qui suit [28].

1.3.9.2.1 Négociation entre agents compétitifs :

La négociation joue un rôle fondamental dans les activités de compétition en permettant aux agents de résoudre des conflits qui pourraient mettre en péril des comportements coopératifs. La négociation est définie comme étant le processus d'améliorer les accords (en réduisant les inconsistances et l'incertitude) sur des points de vue communs ou des plans d'action grâce à l'échange structuré d'informations pertinentes. En général, les chercheurs en Intelligence Artificielle distribuée utilisent la négociation comme un mécanisme pour coordonner un groupe d'agents.

Un des protocoles les plus étudiés pour la négociation s'appuie sur une métaphore organisationnelle. Le protocole du réseau contractuel ("Contract-Net") a été une des approches les plus utilisées pour les SMA. Les agents coordonnent leurs activités grâce à l'établissement de contrats pour atteindre des buts spécifiques. Un agent, agissant comme un gestionnaire ("manager"),

décompose son contrat (une tâche ou un problème) en sous-contrats qui pourront être traités par des agents contractants potentiels. Le gestionnaire annonce chaque sous-contrat sur un réseau d'agents. Les agents reçoivent et évaluent l'annonce.

Les agents, qui ont les ressources appropriées, l'expertise ou l'information requise, envoient au gestionnaire des soumissions qui indiquent leurs capacités à réaliser la tâche annoncée. Le gestionnaire évalue les soumissions et accorde les tâches aux agents les mieux appropriés. Ces agents sont appelés des contractants ("contractors"). Enfin, gestionnaires et contractants échangent les informations nécessaires durant l'accomplissement des tâches. Par exemple, ce protocole a été utilisé pour développer un système de contrôle de production.

- Négociation centrée environnement :

Dans ce cas, l'idée est de voir comment on peut agir sur l'environnement, en décrivant les règles qui régissent son fonctionnement, pour faciliter le bon fonctionnement des agents en particulier dans les résolutions de conflits via les négociations.

- Négociation centrée agent :

Ici le problème n'est plus d'adapter le contexte à la négociation, mais le comportement de l'agent compte-tenu des propriétés du contexte donné. La plupart des stratégies de négociation de ce type ont été faites pour des problèmes spécifiques, et par conséquent peu de principes généraux ont été proposés [28].

1.4 Conclusion :

Dans ce chapitre, nous avons présenté les concepts agent et le système multi-agents, nous avons défini leurs caractéristiques, et les notions fondamentales de ce domaine.

Dans le chapitre suivant, nous allons présenter une introduction de la reconnaissance des activités humaines.

2

Reconnaissance d'activités humaines

2.1 Introduction :

La reconnaissance d'activités humaines (RAH) (en anglais : *Human Activity Recognition*) est un axe de recherche très important et très actif au cours des deux dernières décennies notamment pour ses applications dans divers domaines tels que la santé, la télésurveillance, les jeux, la sécurité, la surveillance, et l'interaction homme-machine.

Dans ce chapitre nous présentons des généralités sur la RAH ainsi que les étapes suivies afin de reconnaître les activités.

2.2 Définition de la RAH :

La reconnaissance d'activité peut être définie comme la capacité à reconnaître/détecter et interpréter les mouvements et les actions des personnes en cours, sur la base des informations reçues de différents capteurs et donner une idée sur leurs comportements voire leurs intentions.

Ces capteurs peuvent être des caméras, capteurs portables ou des capteurs fixés aux objets du quotidien ou déployées dans l'environnement [29].

2.3 Objectifs de RAH :

Les principaux objectifs de la RAH sont :

- Décrire, analyser, reconnaître, comprendre et suivre les activités et les mouvements de personnes, d'animaux ou d'objets. Elle peut être utile pour détecter tôt les comportements anormaux de certaines personnes : difficultés dues à l'âge ou à une maladie [30].
- Aider les humains dans leurs tâches après avoir fourni des informations sur leur comportement [31].
- Faciliter la vie aux êtres humains [31].

2.4 Chaîne de reconnaissance d'activité :

La reconnaissance des activités humaines se base sur l'extraction des données à partir des différents capteurs, tel que les bracelets et les téléphones intelligents, permettant ainsi l'extraction de nouvelle information et l'apprentissage automatique à partir des données d'entrée.

Tout le processus partant de l'extraction des données à partir des capteurs et allant jusqu'à l'entraînement du modèle de reconnaissance d'activité est appelé la Chaîne de Reconnaissance d'Activité. Ce processus est classiquement divisé en différentes étapes qui sont toutes d'une importance cruciale : acquisition des données, prétraitement, segmentation, extraction de caractéristiques et classification comme représentés sur la Figure 2.1 [32] [33].

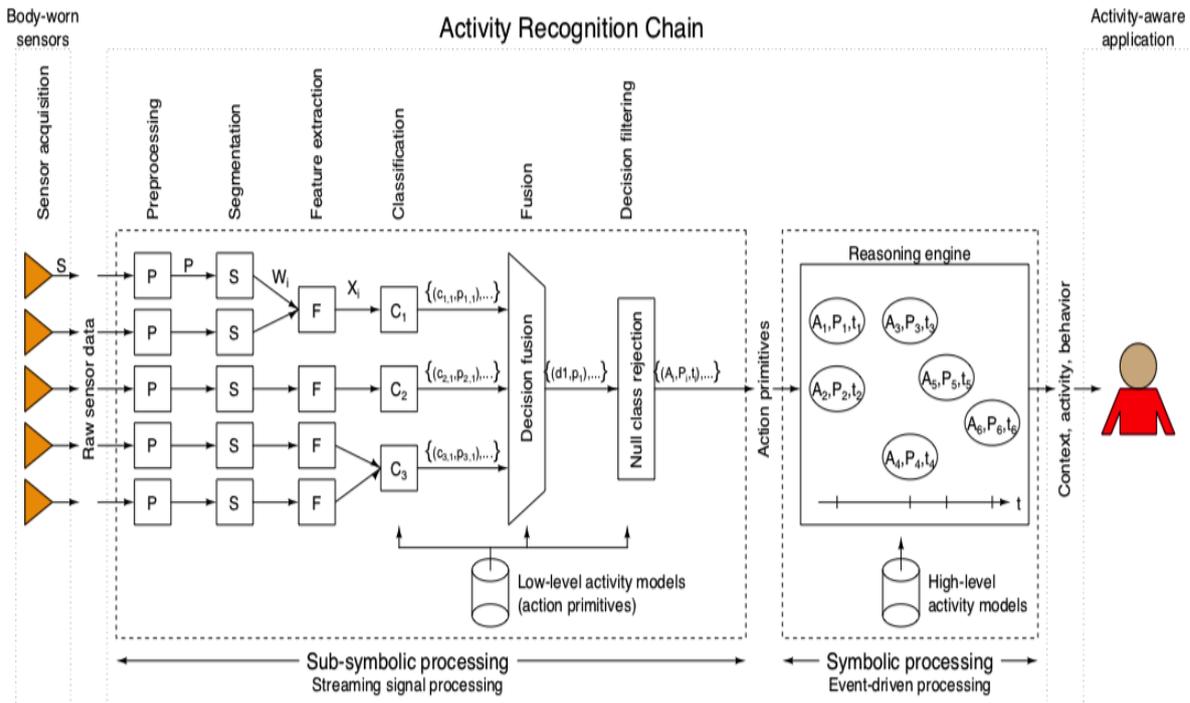


Figure 2.1 : Chaîne de reconnaissance d'activités [33].

Phase de reconnaissance d'activités :

- 1. Acquisition des données (Data acquisition) :** Permet de recueillir les données d'entrée à partir de tous les capteurs ;
- 2. Prétraitement (Preprocessing):** Permet le nettoyage des données brutes afin de réduire les bruits des données de capteurs ;
- 3. Segmentation (Segmentation) :** Permet l'extraction des sous-ensembles de données qui contiennent une activité, sur lequel le processus de

reconnaissance est appliqué. Les segments sont identifiés par leur instant de début et leur instant de fin dans le flux de données.

Cela peut être réalisé à l'aide de deux techniques : la technique de fenêtre glissante ou l'analyse de changement d'activité ;

- 4. Extraction de caractéristiques (Feature extraction) :** Les données contenues dans chaque segment sont transformées en un vecteur de caractéristiques ;
- 5. Classification (Classification) :** Un classifieur utilise les caractéristiques communes comme données d'entrée pour pouvoir reconnaître des classes (les activités de l'utilisateur).
- 6. Décision de fusion (Decision fusion):** elle combine les informations pour avoir une décision sur l'activité qui a eu lieu [30].

2.5 Caractéristiques des systèmes de RAH :

Les caractéristiques des systèmes de RAH sont résumées dans le tableau suivant:

Type	Caractéristiques	Description
Exécution	Hors ligne	Le système enregistre d'abord les données du capteur puis la reconnaissance sera effectuée après. Il est généralement utilisé pour des applications non interactives telles que la surveillance de l'état.
	En ligne	Le système acquiert les données du capteur et les traite en temps réel. Il est généralement utilisé pour l'informatique basée sur l'activité et les applications interactives en interaction homme-machine.
Généralisation	Indépendante de l'utilisateur	Le système est optimisé pour travailler avec un grand nombre d'utilisateurs.

	Spécifique à l'utilisateur	Le système est adapté à un utilisateur spécifique. Les performances sont généralement plus élevées que dans le cas indépendant de l'utilisateur mais ne se généralisent pas aussi bien aux autres utilisateurs.
	Temporel	Le système doit être robuste aux variations temporelles causées par des conditions extérieures (déplacement du capteur, réponse du capteur à la dérive, telle que baromètres ou gyroscopes).
Reconnaissance	Continu	Le système <<repère>> automatiquement l'occurrence d'activité ou de geste dans les données du capteur en flux continu.
	Isolé (segmenté)	Le système suppose que les flux de données du capteur est segmenté au début et à la fin d'un geste par un oracle. IL ne classe que les données de capteur de chaque segmenté dans l'une des classes d'activité. L'oracle peut être un système externe (par exemple, une segmentation inter-modalité) ou l'expérimentateur lors de l'évaluation des performances de classification lors de la phrase de conception.
Activités	Périodique	Activités ou gestes présentent une périodicité, tels que la marche, courir, ramer, faire du vélo, etc. La classification se fonde généralement sur la segmentation de

		la fenêtre coulissante et les caractéristiques du domaine fréquentiel.
	Sporadique	L'activité ou le geste se produit sporadiquement, entre coupé d'autres activités ou gestes. La segmentation joue un rôle clé pour isoler le sous-ensemble de données contenant le geste.
	Statique	Le système gère la détection des postures statique ou des gestes de pointage statique.
Modèle du système	Sans état	Le système de reconnaissance ne modélise pas l'état du monde. Les activités sont reconnues en détectant des signaux de capteurs spécifiques. C'est actuellement l'approche dominante en ce qui concerne la reconnaissance des primitives (par exemple, atteindre, saisir).
	Avec état	Le système utilise un modèle de l'environnement, tel que le contexte de l'utilisateur ou une carte d'environnement avec la localisation des objets. Cela améliore les performances de reconnaissance des activités, ou détriment d'avantage de connaissance ou moment de la conception et d'un système de reconnaissance plus complexe.

Table 2.1 : Caractéristiques principales des systèmes de RAH [31].

2.6 Classification des activités humaines :

Les activités humaines sont classées selon leur complexité : les gestes, les actions, les activités, les interactions et les activités de groupe.

- **Gestes** : ce sont les mouvements élémentaires du corps humain engendré par le déplacement d'un membre humain tel que, lever la main, lever le pied, tourner la tête, etc [34].
- **Actions** : c'est des mouvements complexes composé de plusieurs mouvements élémentaires (plusieurs gestes). On peut citer quelques exemples d'actions comme "marcher", "parler", "boire", "s'asseoir", etc [34].
- **Activités** : c'est des mouvements de plus en plus complexes composé de plusieurs actions, par exemple joué au football, boire du thé, etc. L'activité est aussi reconnue comme un évènement [35].
- **Interactions** : sont les activités humaines qui impliquent plusieurs personnes et/ou plusieurs objets. Par exemple : deux personnes qui se serrent la main est une interaction qui implique deux personnes, et un voleur qui vole le sac d'une femme est une interaction qui implique deux personnes et un objet [34].
- **Activités de groupe** : une activité de groupe est l'ensemble des activités réalisées par des groupes de plusieurs personnes et/ou objets. Par exemple : un groupe de personnes qui se battent entre elles ou un groupe de personnes qui font une réunion sont des exemples de telles activités [34].

2.7 Types de capteurs :

Les capteurs sont classés en fonction de l'emplacement par rapport à l'utilisateur en : portables/portables, quand ils sont portés ou fixés sur le corps de l'utilisateur et ambiants, lorsqu'ils sont à des emplacements fixes à l'environnement.

Ces deux catégories sont détaillées comme suit :

2.7.1 Capteurs portés par le corps humain :

Les capteurs portables sont utilisés pour collecter directement des signaux d'utilisateurs. Ils sont généralement attachés aux différentes parties du corps telle que la taille, le poignet, la poitrine, les jambes et la tête, mais aussi adaptés aux vêtements et intégrés dans d'autres accessoires à usage régulier tels que les montres, les lunettes ou les téléphones mobiles. La température cutanée, le rythme cardiaque, le flux thermique, la conductivité, le système GPS (*Global Positioning System*) et le mouvement du corps sont quelques exemples de variables qui peuvent être mesurées avec des technologies de capteurs portables actuels.

2.7.2 Capteurs ambiants :

Les capteurs ambiants, également connu sous le nom des capteurs environnementaux, sont la collection des dispositifs trouvés dans l'environnement qui mesure des propriétés physique de ceci, des utilisateurs qui lui font partie, et leur interaction. Il existe une large gamme de capteurs ambiants tels que les caméras vidéo, les microphones et les capteurs binaires simples qui peuvent être installés sur les meubles, les murs, les fenêtres et les portes, etc.

2.7.2.1 Caméras vidéo et microphones :

Les caméras vidéo peuvent fournir des informations très détaillées et un contexte riche sur les actions humaines et l'état de l'environnement. Une séquence d'images est directement utilisée pour détecter l'activité humaine dans leurs zones surveillées. Une information vidéo peut fournir des informations directes et claires sur les objets dans la maison intelligente, par exemple, le nombre de personnes. Ainsi, les caméras vidéo ont des avantages importants sur le contrôle et la reconnaissance d'activité de plusieurs utilisateurs. Cependant, ils font face à des difficultés, parmi lesquelles le problème de confidentialité (les gens sont généralement mal à l'aise d'être surveillé en permanence), le coût de calcul élevé et la dépendance à l'environnement.

Les microphones ont la capacité de fournir des informations précises sur les communications et les sons des utilisateurs dans des endroits spécifiques à l'intérieur de la maison intelligente. Cependant, ils souffrent de difficultés d'implémentation et le coût de calcul élevés associés aux algorithmes de traitement audio nécessaires pour distinguer les différents sons, surtout quand il y a plusieurs résidents à l'intérieur de la maison.

2.7.2.2 Capteurs sans fils à sortie binaire :

Les capteurs binaires détectent l'état d'un objet ou d'un mouvement avec un seul chiffre '0' ou '1'. Différents types de capteurs binaires ont été utilisés dans les maisons intelligentes, notamment les capteurs de changement d'état, les détecteurs de mouvement, les capteurs de pression, et les commutateurs à contrat.

Un capteur de changement d'état peut être utilisé pour détecter tout changement de l'état d'un objet qui peut être utilisée pour refléter les interactions utilisateur-objet. Par exemple, un capteur de changement d'état attaché au combiné d'un téléphone détecte si le combiné a été décroché.

Les détecteurs de mouvement sont généralement utilisés pour détecter la présence et l'emplacement des occupants dans toute la maison. Le détecteur de présence à infrarouge est le type de détecteur de mouvement le plus couramment utilisé dans les environnements intelligents.

Les capteurs de pression peuvent être discrètement installés sur des objets tels que les canapés, les lits, les chaises et les planchers.

Les commutateurs à contrat sont généralement installés sur les portes telles que la porte d'entrée et les portes d'armoires et les appareils pour fournir des informations sur l'interaction spécifique que l'occupant effectue avec les objets et les appareils.

En outre, contrairement aux caméras et microphones, les capteurs binaires simples sont invisibles, à faible coût, à faible entretien, facile à installer et à remplacer, et nécessitent un minimale de calcul. Cependant, ils fournissent des

informations limitées pour la reconnaissance des activités complexes et multi-utilisateur.

Une autre variété de capteurs tels que les capteurs de lumières, capteurs de température, et capteurs d'humidité peut être également déployés dans une maison intelligente pour aider au contrôle d'activités humaines ainsi qu'au déclenchement des services automatiques. Les capteurs de lumière sont utilisés pour mesurer l'intensité de la lumière dans un endroit précis. Les capteurs de température sont utilisés pour mesurer la température de l'objet et son environnement. Les capteurs d'humidité sont utilisés pour détecter l'humidité de l'air dans une zone spécifique.

2.7.2.3 Identification par radiofréquence (RFID : *Radio Frequency IDentification*) :

RFID fonctionne comme une combinaison des technologies de capteur environnemental et de capteur portable. C'est un terme générique pour les technologies qui utilisent les ondes radio pour identifier automatiquement les personnes ou les objets. Il existe plusieurs méthodes d'identification, mais le plus commun consiste à stocker un numéro de série qui identifie une personne ou un objet, et éventuellement d'autres informations sur une puce qui est attaché à une antenne (la puce et l'antenne ensemble sont appelés une étiquette RFID). L'antenne permet à la puce de transmettre les informations d'identification à un lecteur. Le lecteur convertit les ondes radio réfléchies à partir de l'étiquette RFID en informations numériques qui peuvent ensuite être transmis à des ordinateurs qui peuvent en faire usage. Il y a généralement deux types d'étiquettes RFID, à savoir, les étiquettes passives qui ont aucune source d'alimentation et sont souvent attachés à un objet pour détecter l'interaction utilisateur-objet, et les étiquettes actives qui contiennent une batterie et offrent une gamme plus large, et pourraient être portés par une personne pour l'identification personnelle.

La petite taille et le faible coût sont les avantages évidents des étiquettes RFID. En outre, en raison de l'identification unique, la RFID est bien adaptée pour le suivi d'emplacement et d'interactions de plusieurs utilisateurs afin d'aider à détecter les activités de plusieurs personnes. RFID représente aussi des inconvénients, tels que la collision entre les lecteurs RFID et la collision entre les étiquettes RFID. La collision entre les lecteurs est produite lorsque l'étiquette est simultanément lue par plusieurs lecteurs RFID, étant incapable de répondre à eux. La collision entre les étiquettes se produit habituellement quand un grand volume d'étiquettes est lu par un lecteur d'étiquette RFID et cela ne peut pas distinguer ces signaux. En outre, le problème de fiabilité et de stabilité, notamment lors de la lecture à travers le liquide et les métaux, constitue une autre limitation de la technologie RFID [36].

2.8 Challenges en reconnaissance d'activités complexes :

Les recherches existantes ont réussi à reconnaître les activités simples des personnes.

Mais reconnaître une activité complexe reste toujours un challenge. Les types de challenges en reconnaissance d'activités complexes sont :

- **Reconnaissance d'activités concurrentes :**

On peut dans certains cas avoir des activités qui se déroulent simultanément.

Une personne peut exécuter plusieurs activités en même temps, elle peut cuisiner pendant qu'elle parle à quelqu'un. Ce genre d'activité doit être reconnu avec une approche spécifique, autre que celle qui traite les activités séquentielles par exemple. Le traitement de ces activités parallèles peut engendrer des courses critiques donc du non déterminisme et des comportements de reconnaissance non souhaitée.

- **Reconnaissance d'activités entrelacées :**

Certaines activités peuvent être suspendues ou entrelacées pour en faire une autre.

Par exemple, si une maman qui cuisine entend son enfant pleurer, elle va mettre en pause la tâche de préparation du plat et va voir son enfant, après elle retourne cuisiner.

- **Ambiguïté d'interprétation de certaines activités :**

Des interprétations d'activités similaires peuvent être différentes d'une situation à une autre. Par exemple : "prendre une cuillère" peut faire partie de plusieurs activités différentes comme "cuisiner", "préparer un café" ou "prendre un médicament".

- **Reconnaissance d'activités impliquant plusieurs personnes (activités de groupe) :**

On peut avoir plusieurs personnes présentes dans une activité et chacune de ces personnes peut faire une activité en parallèle (par exemple, dans une réunion une personne parle et une autre écrit) ou plusieurs personnes qui font la même activité (par exemple : "une chorégraphie") [34].

2.9 Domaines d'applications :

La reconnaissance d'activités est présente dans plusieurs domaines, nous citons :

- **Secteur du sport** : certains produits sportifs tels que les chaussures de course Philips Direct Life ou NIKE + intègrent des capteurs de mouvement et offrent aux athlètes amateurs et professionnels un retour d'informations sur leurs performances.
- **Divertissement**: à travers les consoles de jeux de la Nintendo Wii et le Microsoft Kinect qui s'appuient sur la reconnaissance des gestes ou même sur les mouvements du corps entier pour changer fondamentalement l'expérience de jeu.
- **Santé**: la reconnaissance d'activités permet aux médecins de surveiller le comportement et les activités des patients (par exemple, personnes âgées et/ou atteintes d'Alzheimer) même à distance et d'agir rapidement en cas d'urgence ou de danger. Elle leur permet aussi de mieux connaître le

patient, de diagnostiquer ses pathologies, et de développer de nouvelles stratégies d'intervention en matière de prévention et d'accompagnement.

- **Militaire:** Dans les scénarios tactiques et militaires, des informations précises sur les activités des soldats ainsi que leurs emplacements et leurs conditions de santé sont très bénéfiques pour leur performance et leur sécurité. De telles informations sont également utiles pour soutenir la prise de décision dans les environnements de combat et d'entraînement [29].
- **Secteur industriel :** bénéficie clairement de la reconnaissance des activités telles que le prototype de reconnaissance de tâches pour soutenir les activités de formation et de qualification sur les sites de production automobile de SKODA en république tchèque [32].

2.10 Conclusion :

Nous avons présenté dans ce chapitre un état de l'art sur la RAH, on a expliqué les différentes étapes suivies afin de reconnaître les activités humaines ainsi l'usage de ces systèmes dans notre vie.

3

Conception et réalisation d'un simulateur à base d'agents pour activités humaines

3.1 Introduction :

Ce chapitre présente la procédure de conception et de réalisation de notre simulateur d'activités humaines, après avoir étudié le fonctionnement de la plateforme de simulation SMACH qui est réalisé à base d'agents.

3.2 Etude de l'existant :

Ces dernières années, les technologies de la maison intelligente sont devenues un sujet de recherche très actif. Cependant, de nombreux scientifiques travaillant dans ce domaine ne possèdent pas d'infrastructures de maison intelligente leur permettant de mener des expériences satisfaisantes dans un environnement concret avec des données réelles. Pour cela plusieurs simulateurs ont été proposés dans la littérature pour améliorer les choses, tels que : UbiREAL [37], SIMACT [38] et SMATCH [39]. Une brève description de chacun de ces simulateurs est donnée comme suit :

3.2.1 Simulateur UbiREAL :

3.2.1.1 Définition :

Est un simulateur d'espace intelligent, dont l'objectif est de développer et de tester des applications ubiquitaires.

Ces applications peuvent contrôler de nombreuses informations sur les préférences de l'utilisateur, l'état de l'environnement externe, et autres. UbiREAL génère des changements de contexte basés sur le comportement de l'utilisateur et les communications des dispositifs et les représente dans un espace intelligent (par ex. interaction entre les dispositifs, changements d'humidité ou de température). Cette approche est plus intéressée à permettre aux développeurs de vérifier systématiquement l'influence des changements de contexte dans l'application lorsqu'elle est placée dans un espace virtuel 3D, que d'explorer l'expérience de l'utilisateur dans le lieu [37].

3.2.1.2 Objectifs :

- Offrir aux habitants (utilisateurs d'applications) un moyen de vérifier intuitivement le fonctionnement des dispositifs.

- Permettre la coopération entre les dispositifs virtuels et les dispositifs réels.
- Rendre les pilotes des périphériques (logiciels) pour les périphériques réels exécutables en tant que périphériques virtuels sans grande modification.
- Permettre aux testeurs de tester systématiquement l'ensemble du système d'application en générant automatiquement de nombreux contextes possibles [37].

3.2.1.3 Structure UbiREAL :

Le simulateur UbiREAL se compose des quatre parties suivantes comme indiqué dans la Figure 3.1 :

- (1) Visualiseur et interface graphique pour la conception du smartspace virtuel.
- (2) Simulateur de réseau.
- (3) Simulateurs de quantités physiques.
- (4) Testeur systématique.

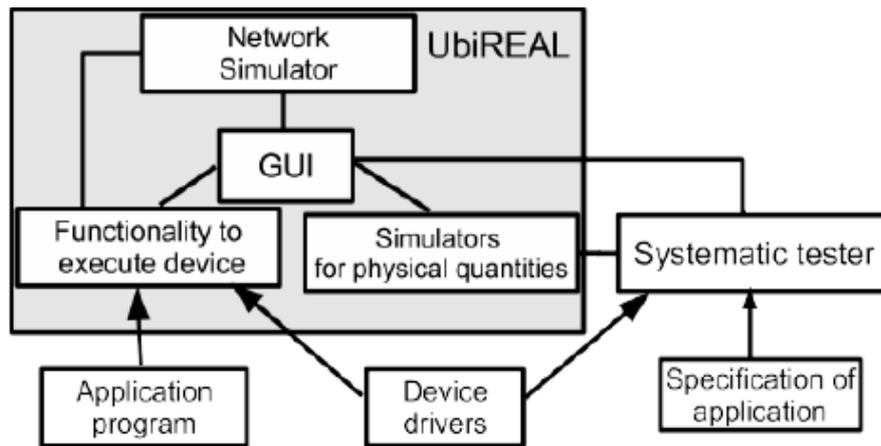


Figure 3.1 : Architecture UbiREAL [37].

Nous décrivons ci-après les quatre fonctions principales d'UbiREAL :



Figure 3.2 : Spécification de l'itinéraire de l'avatar [37].

-Visualiseur et interface graphique pour la conception du smartspace virtuel : cette fonction aide l'utilisateur à créer une maison virtuelle et des pièces dans lesquelles sont placés des dispositifs virtuels. De même cette fonction permet de concevoir des itinéraires que les utilisateurs d'applications (virtuelles) et d'autres dispositifs mobiles suivent.

-Simulateur de réseau : simule la communication entre les dispositifs virtuels ain qu'entres les dispositifs virtuels et réels, en tenant compte de leur position et des obstacles sur leur ligne de mire.

-Simulateurs de quantités physiques : afin de reproduire la variation temporelle de quantités physiques telles que la température, l'éclairage et l'intensité sonore dans une région spécifique du smartspace virtuel, un simulateur dédié est préparé pour chaque quantité physique.

-Testeur systématique : génère systématiquement de nombreux textes possible et teste si une application ubiquitaire donnée fonctionne comme prévu, en coopération avec des simulateurs de quantités physiques [37].

3.2.2 Simulateur SIMACT :

3.2.2.1 Architecture générale :

SIMACT Est un outil logiciel 3D flexible de maison intelligente développé en java. Ce simulateur est spécialement conçu pour aider les chercheurs travaillants dans le domaine de reconnaissance d'activité à mener des expériences avec des données réelles. L'architecture de ce simulateur permet à un agent de reconnaissance intelligent de se connecter facilement à une base de données afin de recevoir, en temps réel, les entrées des capteurs simulés. Des scénarios préenregistrés, réalisé à partir de donnés extraites d'essais clinique, sera inclus avec l'outil afin de fournir une base commune pour tester les algorithmes de reconnaissance d'activité. L'objectif est de publier le simulateur SIMACT en tant que composant open source gratuit sur internet qui profitera à l'ensemble de la communauté de recherche sur les maisons intelligentes [38]. La Figure 3.3 illustre l'interface utilisateur de l'outil SIMACT.

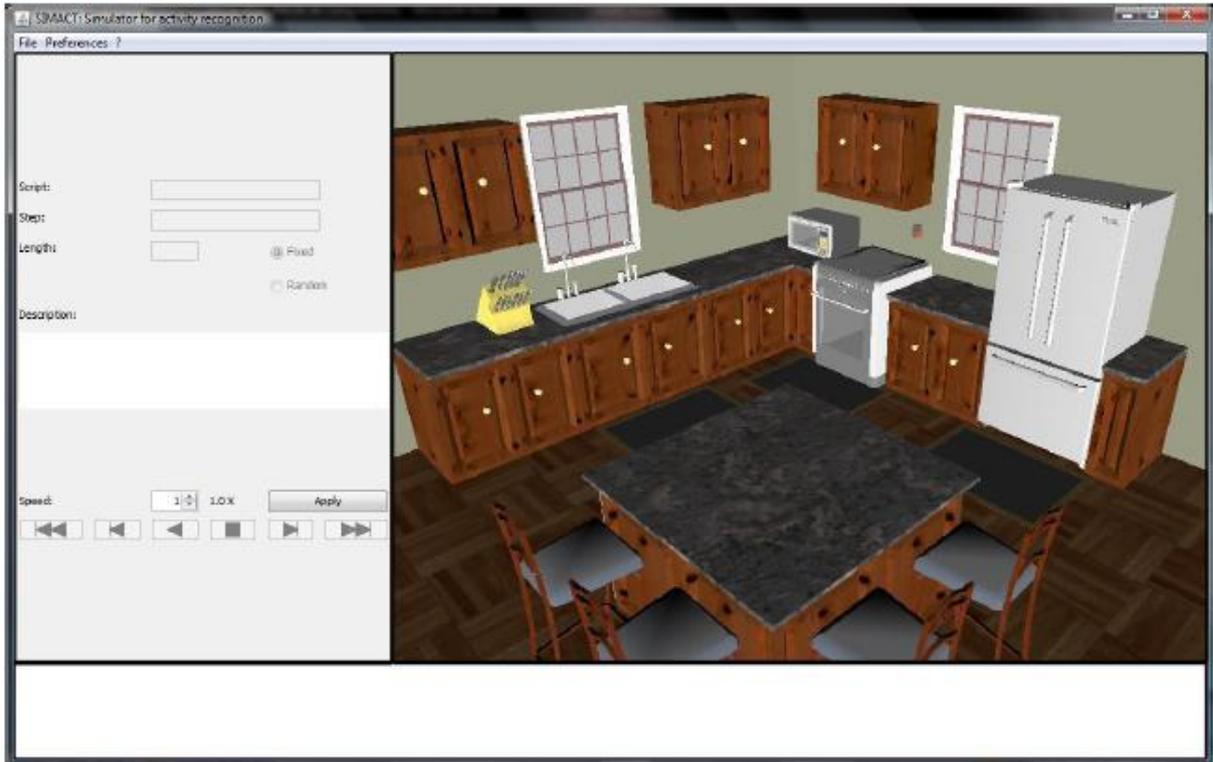


Figure 3.3 : L'outil de simulation d'habitat intelligent SIMACT [38].

Le principal bénéfice de ce type d'outil, est le fait qu'il est capable de simuler n'importe quelle activité effectuée dans un habitat intelligent, en reprenant les mêmes séquences d'actions et déclenchement de capteurs [38].

3.2.2.2 Modèle d'activité de la vie quotidienne utilisée pour les tests :

La littérature sur les habitats intelligents nous fournis plusieurs pistes intéressantes concernant le choix d'activités représentatives, comme le lavage des mains, la cuisson des pâtes et la préparation du thé, etc. Les modèles d'activités choisis pour les expérimentations doivent également être simples et faciles à expérimenter dans un contexte de recherche. En ce sens, ces activités doivent inclure un nombre suffisant d'étapes et limité. Ces activités devraient aussi être réalisables dans une durée de temps raisonnable dans l'éventualité de tests avec des sujets réels. Un temps d'environ 10 minutes maximum serait l'idéal. Finalement, il faut posséder des données réelles concernant les activités choisis. Pour cela, deux activités ont été sélectionnées : faire du café et préparer des rôtis. Ces activités sont intéressantes car elles se réalisent en moins de 10

minutes, comporte un nombre d'étapes suffisants mais limité, et on peut faire interagir les deux activités ensemble. Les objets impliqués dans ces activités sont : un carton de lait, un pot de sucre, une carafe d'eau chaude, un pot de café instantané, un sac de pain, un grille-pain, un couteau à beurre, un beurrier avec du beurre. La Figure 3.4 schématise, étape par étape, le modèle résumant les différentes façons de réalisées les deux activités choisies. Ce modèle a été conçu à la lumière des expérimentations réalisées avec des sujets réels normaux. Il est utilisé afin de déterminer si l'exécution de l'activité est correcte. Plus précisément, les informations recueillies lors de ces expérimentations antérieures permettront de vérifier si l'ordre des étapes est correcte, si la durée des étapes et les délais entre celles-ci est adéquate, etc [38].

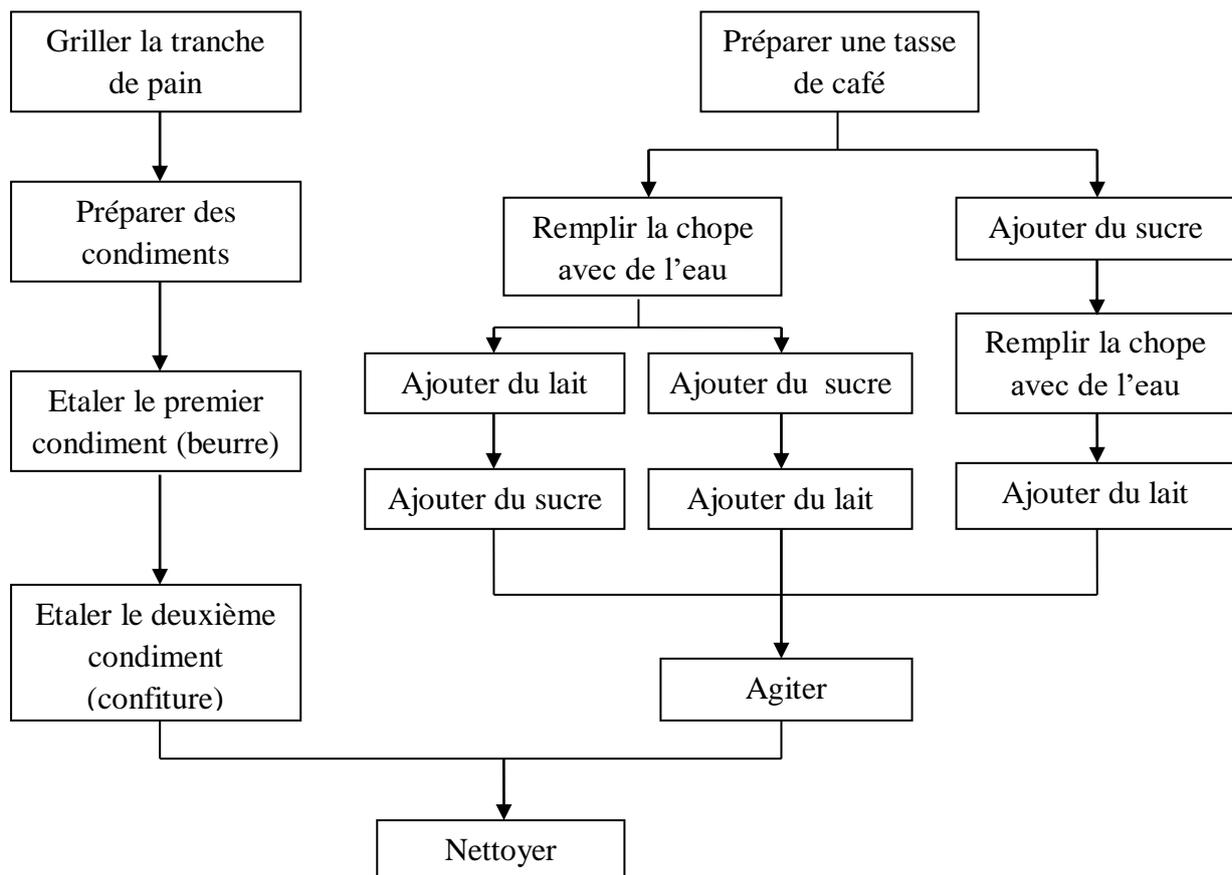


Figure 3.4 : Modèle des activités choisies : faire du café et préparer des toasts [38].

3.2.3 Plateforme SMACH :

3.2.3.1 Présentation générale de l'architecture:

SMACH (*Simulation Multi-Agents des Comportements Humains*) est une plateforme de simulation multi-agent de l'activité humaine au sein de l'habitat permettant de simuler la consommation électrique des ménages. Développée depuis 10 ans en collaboration entre EDF (Electricité de France) et plusieurs laboratoires de recherche académique, SMACH modélise la complexité du comportement humain (vie collective au sein du foyer, confort thermique des habitants, prise en compte du prix de l'énergie, etc.) et ses impacts sur la consommation électrique.

La plateforme de simulation SMACH est composée de plusieurs parties (voir schéma de la Figure 3.5). Le scénario (1) intègre différentes caractéristiques du foyer (nombre de personnes dans le foyer, âges, sexes, actions possibles, préférences de confort, équipements...) : il a un rôle de cadrage général de la simulation. Le moteur d'activité humaine (2) va créer la dynamique entre les agents. À chaque pas de temps (correspondant à une minute de simulation) les agents peuvent modifier leurs agissements et interactions avec les autres agents de la situation (les membres du foyer) ou avec l'environnement (l'habitat, le chauffage, etc.). Ainsi, un scénario ouvre sur un ensemble d'actions possibles tandis que le moteur d'activité humaine détermine ce qui sera réellement fait dans la situation simulée. Par exemple, un agent souhaite faire une promenade (action possible dans le scénario) mais un autre agent refuse car il est déjà engagé dans une action. Nous distinguons ainsi dans la simulation ce qui est de l'ordre du cadrage de la simulation (le scénario) et ce qui est de l'ordre de la dynamique des agents (le moteur d'activité humaine) [39].

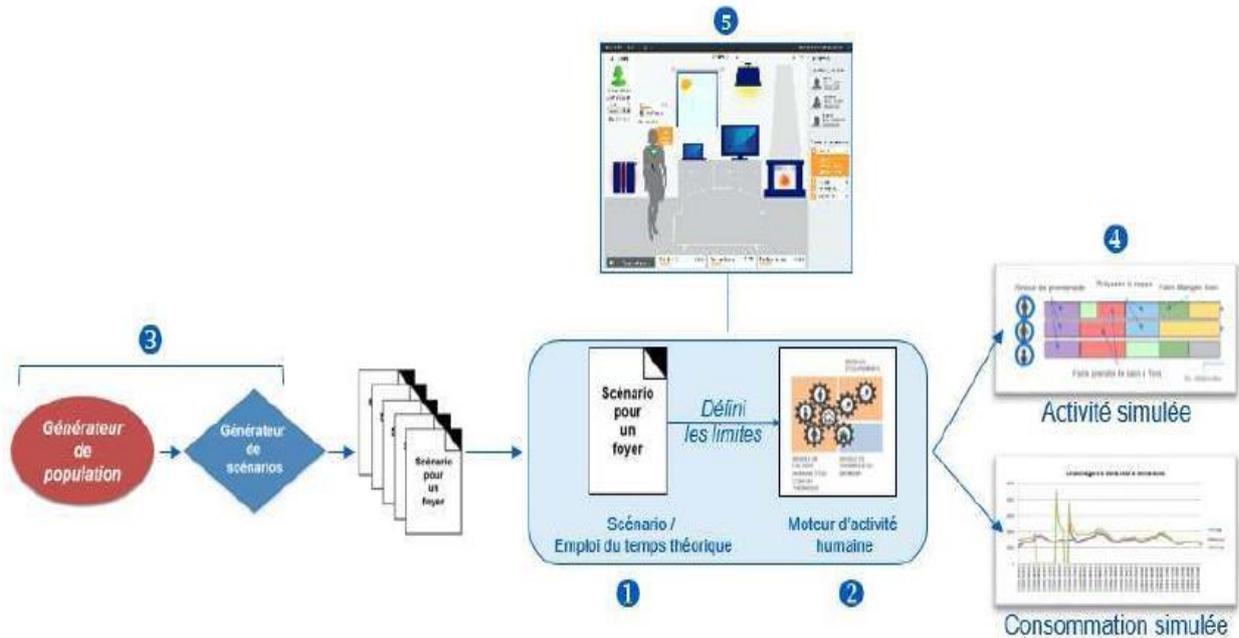


Figure 3.5 : Les différents composants de SMACH [39].

Le scénario et le moteur d'activité humaine correspondent à la brique de base de la simulation SMACH (la zone en bleue sur le schéma ci-dessus). Cet environnement technique est complété par des générateurs de population et de scénarios (3). En effet, il était nécessaire de pouvoir faire des simulations grande échelle pour répondre à des problématiques sur l'efficacité énergétique. Le générateur de population reconstitue une population de foyers (composition des foyers, âges, sexes, types d'habitat, etc.) sur la base des données de recensement de l'INSEE (Institut National de la Statistique et des Etudes Economique). Les scénarios sont créés en reconstituant des emplois du temps théoriques à partir de l'enquête "emploi du temps" de l'INSEE. Le modélisateur peut alors créer des simulations unitaires (quelques foyers), ou à l'échelle d'une population utiliser les générateurs de population et de scénarios. Il est possible de suivre le déroulé d'une simulation mais ce sont les résultats de la simulation qui sont utiles. Nous obtenons ainsi (4) le diagramme d'activité et la courbe de consommation pour chaque foyer simulé. Enfin, cet ensemble technique est complété par une

interface de simulation participative (5). Il était important, dans une logique de validation, qu'un acteur humain puisse interagir avec la simulation.

Ce dernier peut modifier les actions de son avatar, son niveau d'habillement, les consignes de température et peut allumer ou éteindre tous les équipements de l'habitat [39].

Le modèle général de l'activité humaine utilisé dans la plateforme SMACH se décompose en trois niveaux :

- L'autonomie et la dynamique de l'activité individuelle de l'agent. L'activité individuelle est portée par un agent (un individu) qui réalise de façon autonome des tâches. Des habitudes émergentes au cours de la simulation et elles sont influencées par des modes de vie liés à l'efficacité énergétique, au confort, au budget.
- La construction de l'activité collective d'un agent avec d'autres agents. L'activité collective se construit dans l'interaction entre agents (les individus) qui réalisent ensemble des tâches et forment pour l'occasion des groupes (ou collectifs) éphémères composés de tout ou partie des membres du foyer (adultes, enfants, proches...).
- Le couplage asymétrique de l'agent avec son environnement. Le couplage structurel avec l'environnement résulte des agents (les individus) qui, engagés dans leurs activités individuelles et collectives, sont en interaction avec un environnement composé d'appareils (chauffage, ballon d'eau chaude, réfrigérateur, etc.) et de pièces qui forment un logement ayant un niveau d'isolation [40].

3.2.3.2 Génération de populations :

Le premier objectif du simulateur SMACH est de générer des populations virtuelles représentatives des caractéristiques de la population étudiée. Pour cela, les auteurs ont utilisé différentes études provenant de l'INSEE (TUS - *Time Use Surveys*) afin de définir les caractéristiques de la population qu'ils ont souhaité étudier : la répartition des ménages (famille, couples sans enfant...), l'âge, le

sexe, la catégorie socio professionnelle, le contrat d'électricité, le type de production d'ECS (électrique ou non), le type de chauffage (électrique ou non), la surface du logement et sa performance thermique réglementaire. SMACH va également, à partir des données dont il dispose, générer les équipements présents au sein des différents foyers de la population.

La plateforme SMACH peut produire pour chacun des individus de sa population, un emploi du temps prévisionnel de la semaine. Pour cela, SMACH utilise les données issues des TUS réalisées en France entre 2009 et 2010.

Ainsi, SMACH est capable de construire, en fonction du profil de chaque habitant du foyer, un emploi du temps prévisionnel. Dans cet emploi du temps, les tâches sont caractérisées par :

- **Durée.** Chaque activité est définie par une durée minimum et une durée maximum. Lorsqu'un agent réalise une activité, son niveau de priorité augmente par rapport à celui des autres activités. Ce bonus représente le bonus d'engagement dans l'activité courante. Une fois que l'activité a été réalisée pendant une durée au moins égale à sa durée minimum, ce bonus d'engagement diminue, jusqu'à atteindre zéro lorsque l'activité a été réalisée pendant une durée égale à sa durée maximum.
- **Rythme.** A chaque activité est attribué un nombre de répétitions par jour ou par semaine. Par exemple l'activité de sommeil est généralement répétée une fois par jour.
- **Période préférentielle (PP).** A chaque activité est associée une période préférentielle (notée PP) indiquant les périodes de la journée (ou de la semaine) qui sont préférées pour la réalisation de l'activité. A l'intérieur de la PP, le niveau de priorité de l'activité augmente. En dehors de la PP, il diminue.
- **Un niveau de collectivité** qui indique si une tâche est réalisable seule ou en groupe.

Les trois premières caractéristiques sont directement issues des résultats des enquêtes emploi du temps. Le niveau de collectivité quant à lui a été défini par les développeurs de la plateforme pour chacune des tâches réalisables. Par exemple, la tâche « *dîner* » a un niveau de collectivité qui indique que cette dernière peut être réalisée à plusieurs [41].

3.2.3.3 Modèle agent :

L'objectif de la simulation multi-agent dans SMACH est de transformer la description statique du foyer en une simulation dynamique de l'activité humaine. Le résultat de cette simulation sera un diagramme d'activités représentant à chaque instant de la simulation ce que fait un habitant du foyer (Figure 3.6).

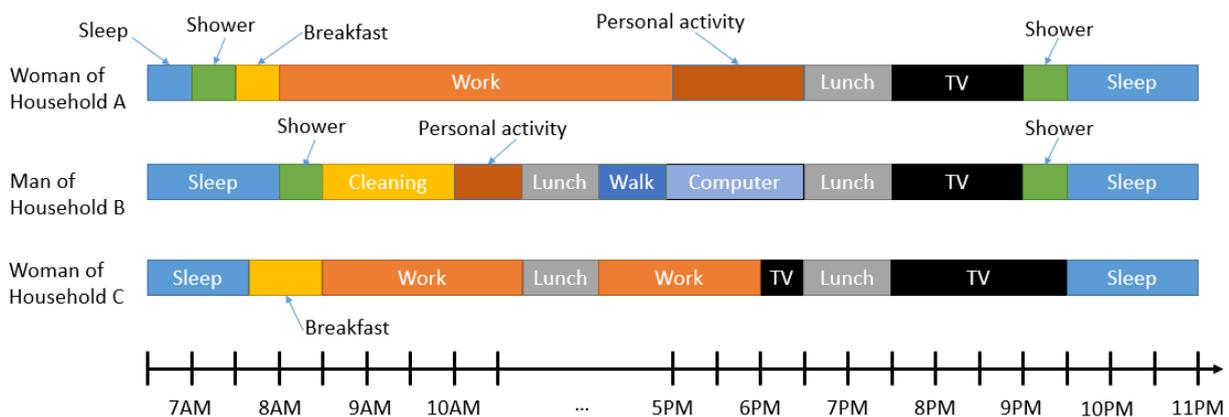


Figure 3.6 : Diagramme d'activité pour trois occupants d'un foyer implémenté dans SMACH [41].

Au sein de la plateforme SMACH, les agents sont modélisés avec des actions possibles à savoir leur emploi du temps prévisionnel, des connaissances sur leur environnement, l'activité courante des autres agents, ou en fonction d'événements et des préférences telles que la température de confort, l'attitude vis-à-vis du prix de l'énergie... Pour construire un diagramme d'activité, les agents sélectionnent une tâche à réaliser dans leur emploi du temps prévisionnel à chaque pas de temps de la simulation. Cette sélection de la tâche est basée sur un niveau dynamique de priorité. Chaque tâche disposera de son propre niveau

qui évoluera au cours de la simulation en fonction de différents paramètres. Par exemple, les caractéristiques d'une tâche comme sa durée, ses périodes préférentielles et son rythme influenceront ce niveau. Cependant, d'autres facteurs comme le prix de l'énergie, la disponibilité d'un appareil électrique ou encore les actions entreprises par les autres occupants du foyer sont pris en compte. Une tâche ainsi sélectionnée et effectuée par un agent devient alors une activité qui apparaîtra dans le diagramme [41].

3.3 Analyse et synthèse :

A travers l'étude de ces simulateurs d'activités humaines, on a choisit de se baser sur le fonctionnement du simulateur SMATCH à base d'agents pour la conception et la réalisation de notre simulateur d'activités humaines qui aussi à base d'agents.

3.4 Outils de développement :

Notre application est implémentée en utilisant le langage JAVA, JavaFX sous l'environnement de développement **NetBeans**. Pour le système multi agents nous avons utilisé la plateforme **JADE**.

3.4.1 Plateforme JADE (Java Agent Development Framework) :

JADE est une plate-forme de développement d'agents gratuite et Open Source développée par CSELT qui a comme but la construction des systèmes multi-agents et la réalisation d'applications conformes à la norme FIPA [42].

3.4.2 NetBeans :

NetBeans est un EDI (*Environnement de Développement Intégré*), un environnement libre et facile à utiliser. Il supporte plusieurs langages de programmation à savoir JAVA, C++, PHP, et bien d'autres. Il fournit plusieurs outils tels qu'un éditeur de texte doté d'un pré-compilateur avancé, un gestionnaire de projets. Ainsi que des outils de débogage et de test des programmes. C'est un outil qui facilite énormément la phase de développement et des tests [43].

3.4.3 JAVA :

C'est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de créer des logiciels compatibles avec de nombreux systèmes d'exploitation (Windows, Linux, Macintosh, Solaris). Java donne aussi la possibilité de développer des programmes pour téléphones portables. Enfin, ce langage peut être utilisé sur Internet pour des petites applications intégrées à la page web (applet) ou encore comme langage serveur (jsp) [42].

3.4.4 JavaFX :

JavaFX est un framework et une bibliothèque d'interface utilisateur issue du projet OpenJFX, qui permet aux développeurs Java de créer une interface graphique pour des applications de bureau, des applications internet riches et des applications smartphones et tablettes tactiles [44].

3.5 Conception du simulateur :

3.5.1 Architecture générale :

L'application implémentée se compose d'un Scénario, une interface utilisateur, moteur d'activités humaines et nous obtenons un diagramme d'activité et le tableau de résultat (voir le schéma de la Figure 3.7).

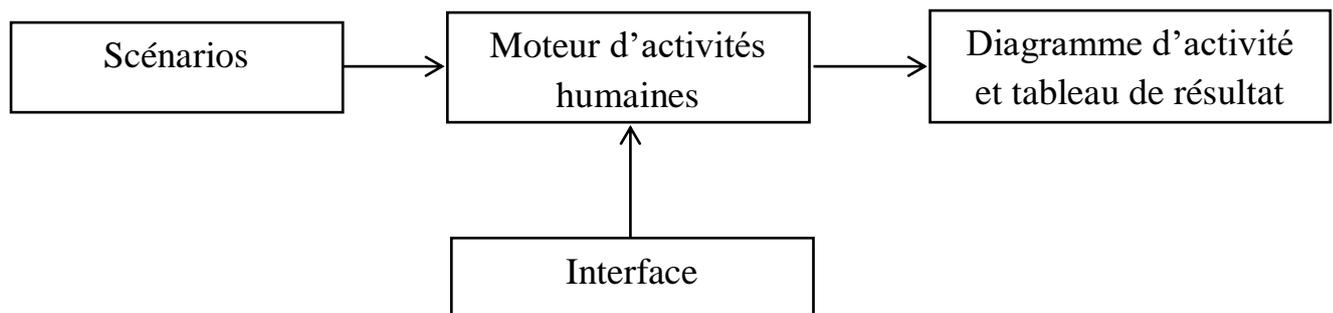


Figure 3.7 : Architecture général de l'application.

3.5.2 Système multi agents :

Le langage UML (Universal Modeling Language) fournit un moyen visuel standard pour spécifier, concevoir, et documenter les applications orientées objets, à l'aide d'éléments appelés diagrammes UML.

Notre SMA se compose de deux types d'agents : un agent capteur avec un comportement simple de type `cyclicBehaviour` qui exécute sa tâche d'une manière répétitive, et un agent humain avec comportement composé de type `FSMBehaviour` qui implémente un automate à états finis.

3.5.3 Diagramme de cas d'utilisation :

Un cas d'utilisation (use case) modélise une interaction entre le système informatique à développer et un utilisateur ou acteur interagissant avec le système. La Figure 3.8 représente les différents cas d'utilisation du simulateur.

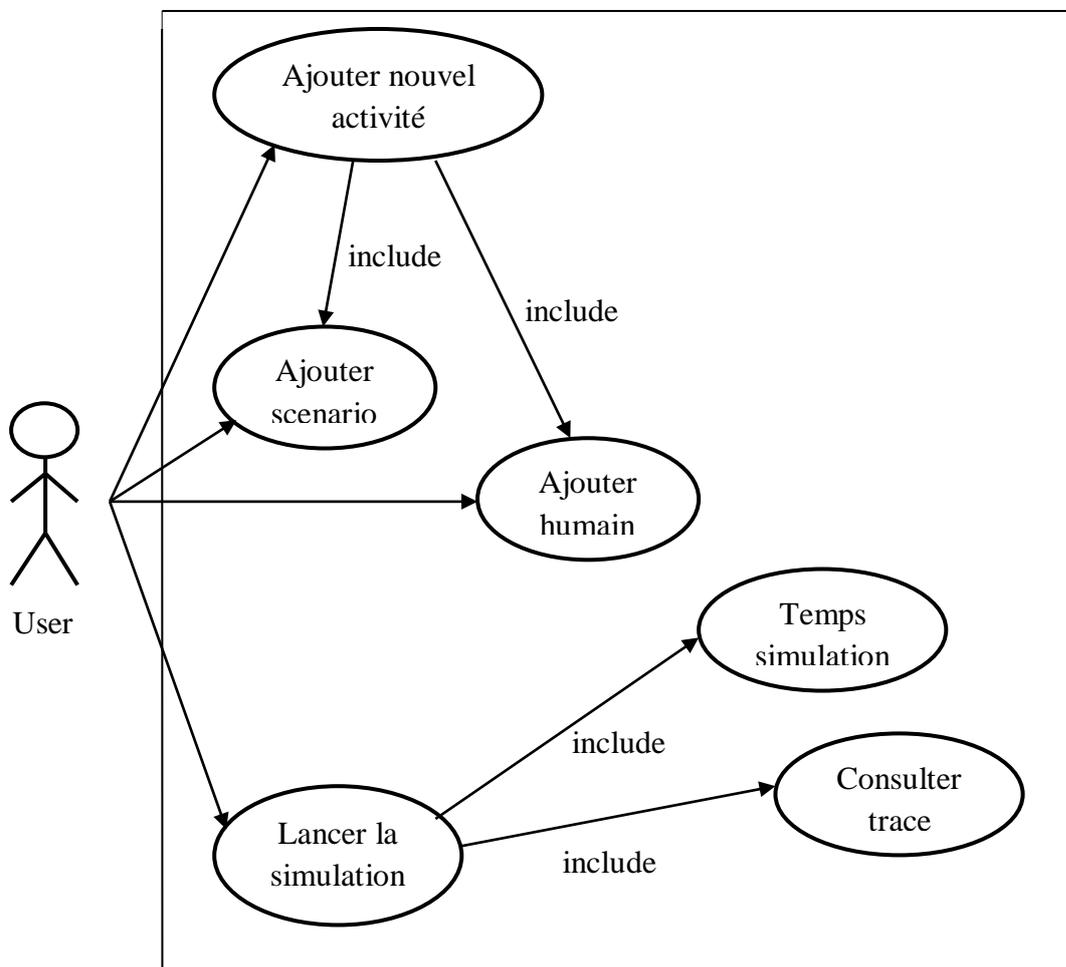


Figure 3.8 : Diagramme de cas d'utilisation.

3.5.4 Diagramme d'activité :

Les diagrammes d'activités permettent de mettre l'accent sur les traitements. Ils permettent ainsi de représenter graphiquement le comportement

d'une méthode ou le déroulement d'un cas d'utilisation. La Figure 3.9 représente le déroulement complet du comportement du simulateur en interaction avec l'utilisateur.

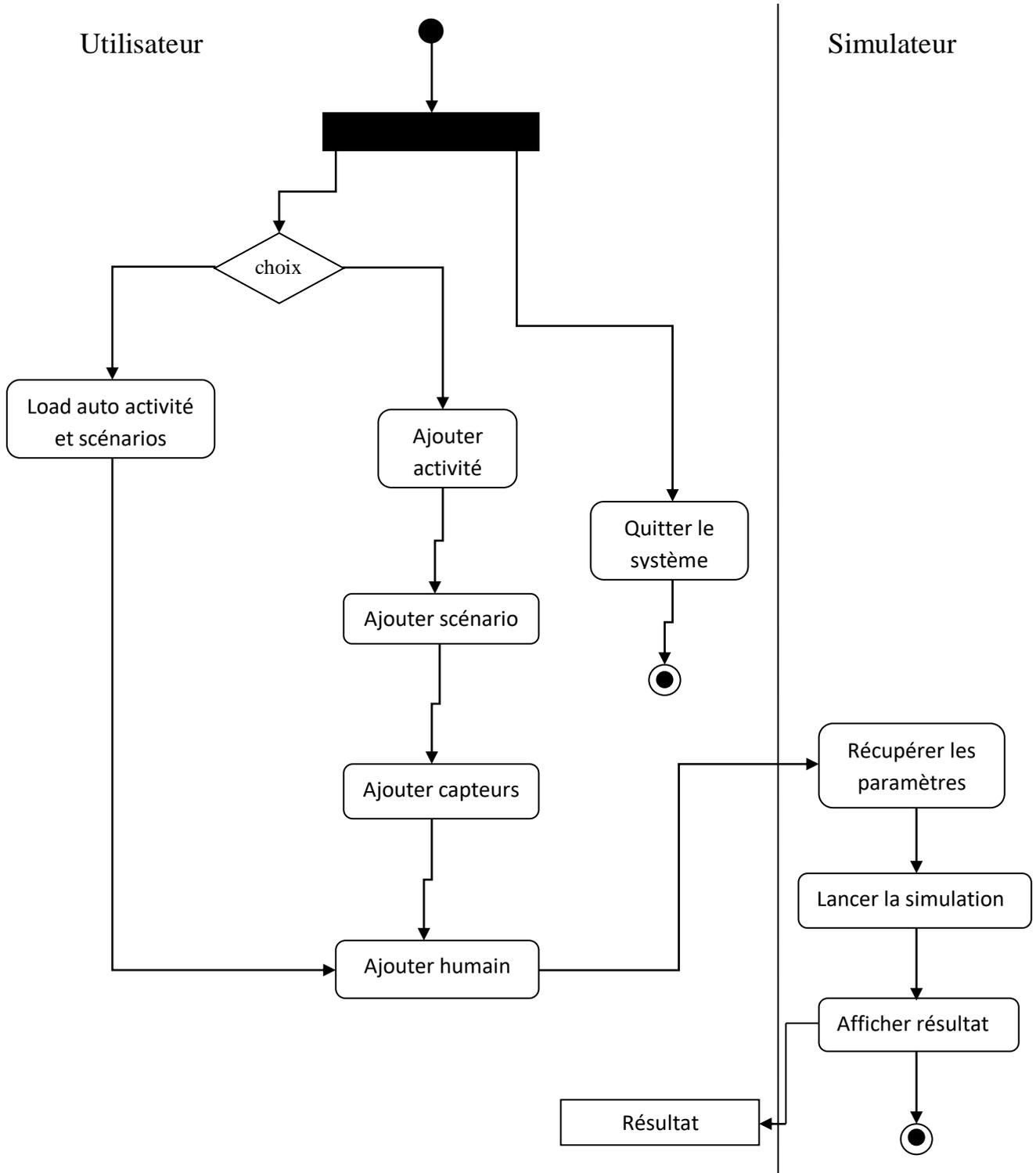


Figure 3.9 : Schéma de diagramme d'activité du simulateur.

3.6 Manuel d'utilisation du simulateur :

On a eu le temps d'implémenter une seule activité humaine, le reste du chapitre va concerner l'activité dormir.

La Figure 3.10 représente la première fenêtre de notre application.

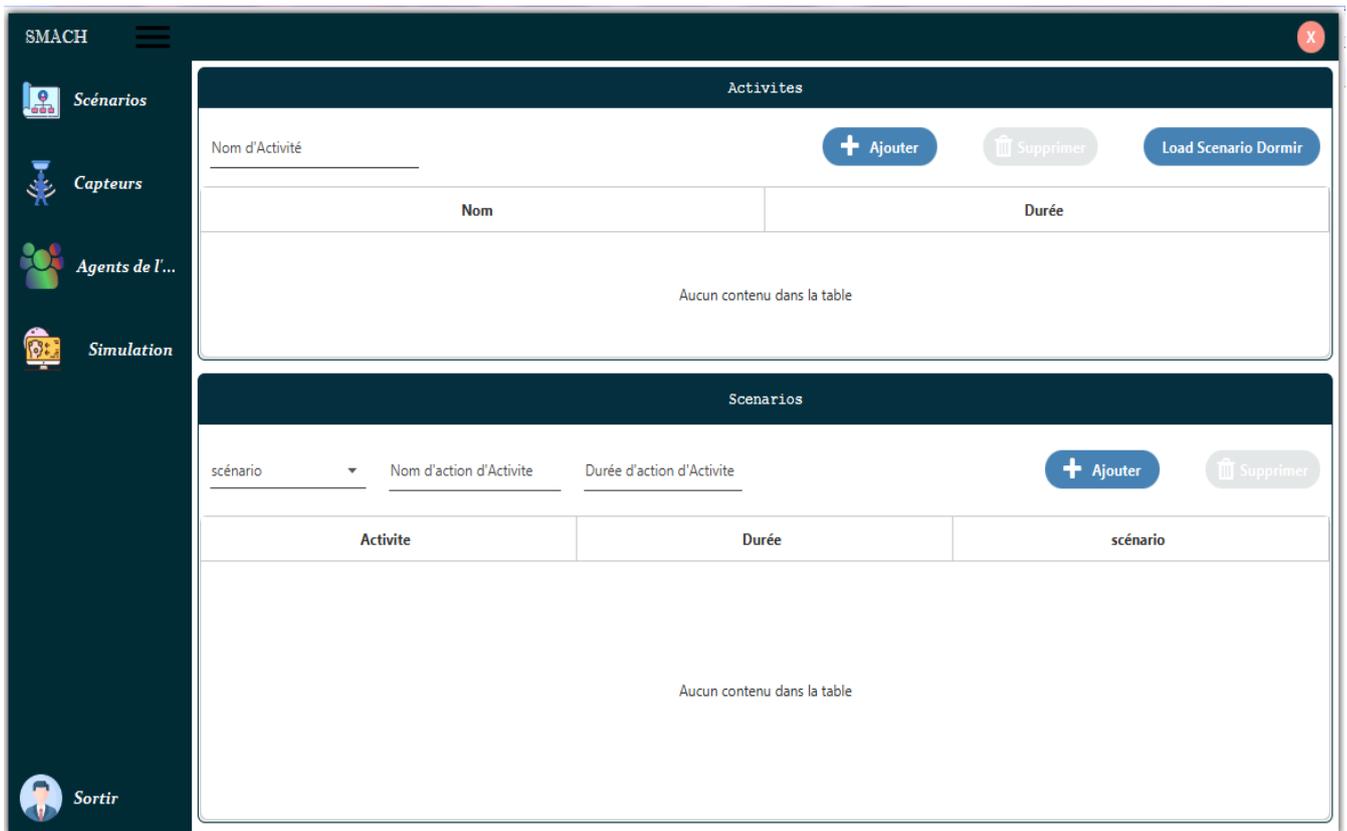


Figure 3.10 : Première interface.

Cette fenêtre permet à l'utilisateur soit, de télécharger le scénario de l'activité dormir, soit de créer une nouvelle simulation ou bien de sortir de l'application et cela comme suit :

3.6.1 Télécharger le scénario de l'activité :

Pour télécharger le scénario de l'activité dormir l'utilisateur doit aller au bouton « Load Scenario Dormir » à partir de la première fenêtre de l'application et la Figure 3.11 apparaît le scénario déjà créé avec des durées aléatoires:

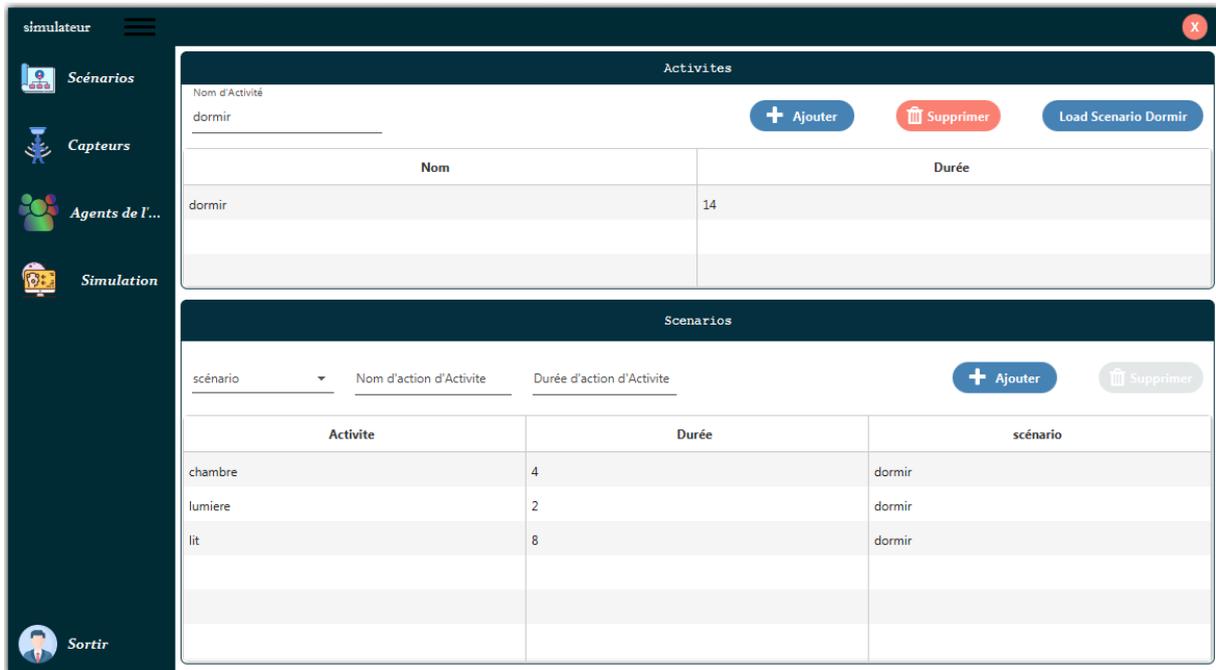


Figure 3.11 : Le scénario téléchargé de l'activité dormir.

Les capteurs de cette activité apparaissent dans la Figure 3.12 :

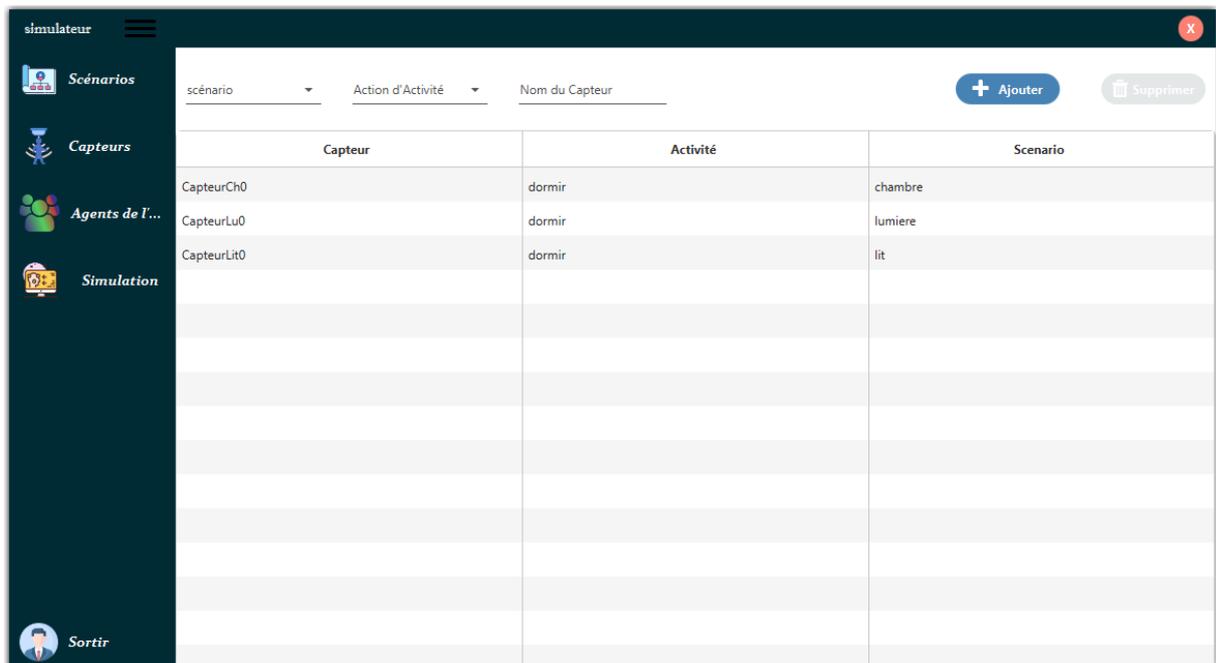


Figure 3.12 : Les capteurs de l'activité dormir.

L'utilisateur doit choisir le nom d'agent humain qui exécute cette activité comme dans la Figure 3.13.

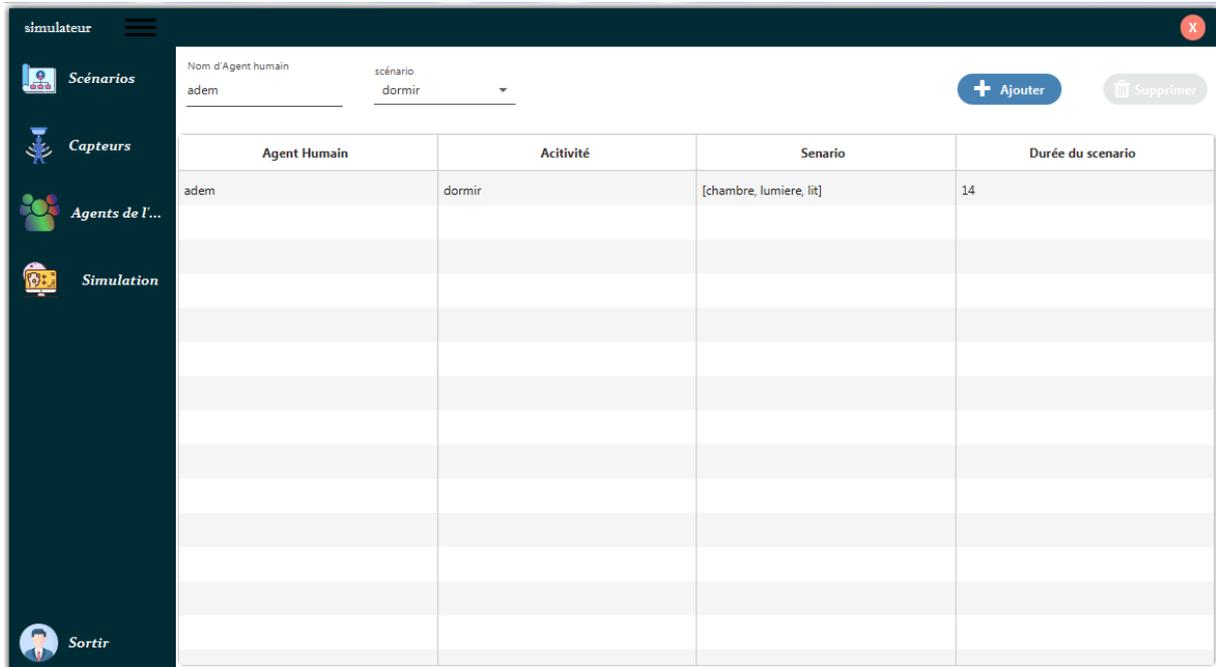


Figure 3.13 : Choisir le nom d'agent humain.

Et enfin pour voir les résultats de la simulation de l'activité dormir l'utilisateur doit cliquer sur le bouton « Lancer la simulation » comme dans la Figure 3.14:

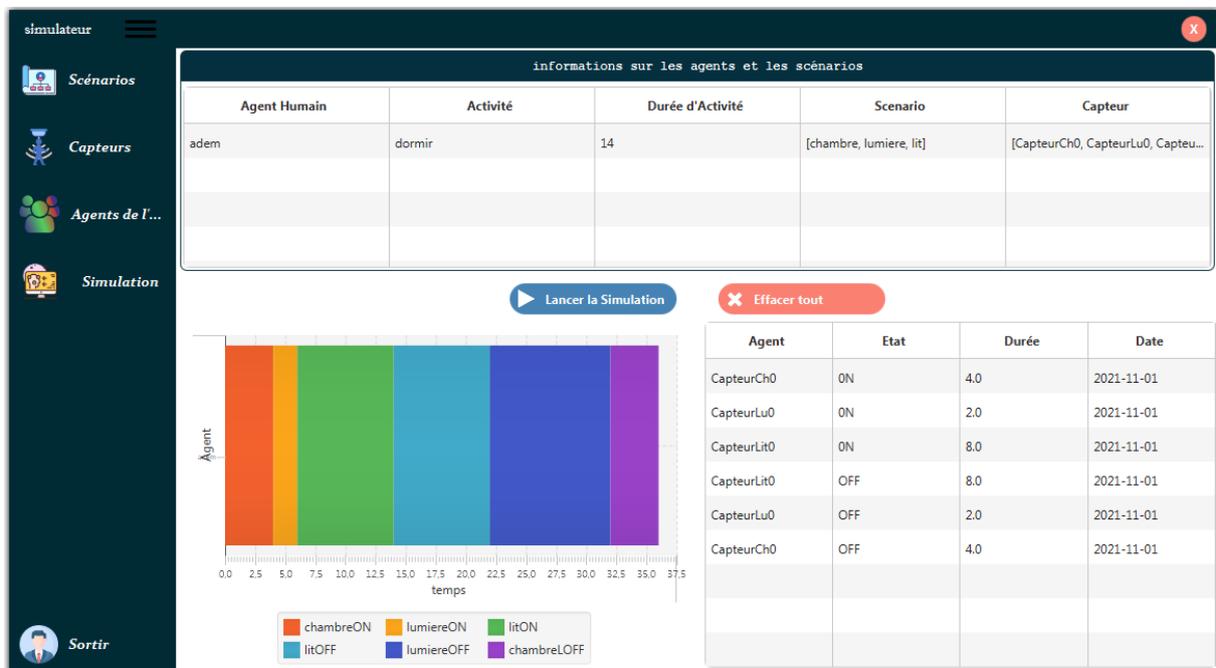


Figure 3.14 : Résultat de la simulation.

3.6.2 Nouvelle simulation :

Pour faire une nouvelle simulation l'utilisateur doit, écrire le nom de l'activité et cliquer sur le bouton « ajouter » pour ajouter le scénario de l'activité comme dans la Figure 3.15 :

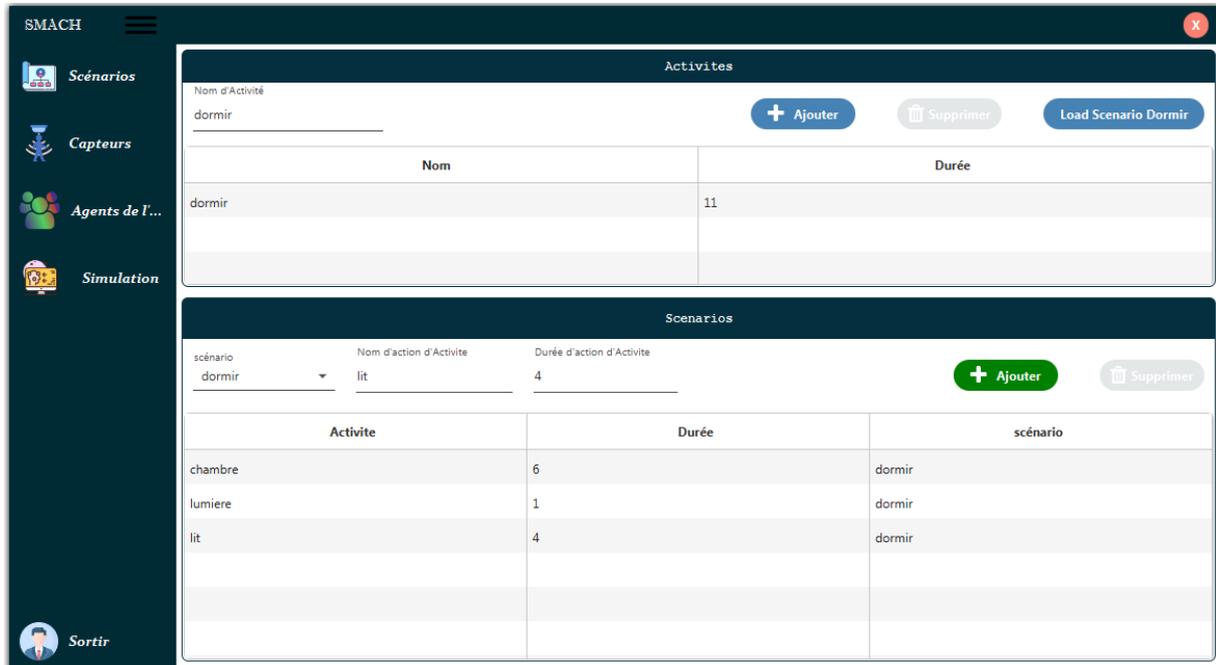


Figure 3.15 : Nouvelle simulation.

A travers cette fenêtre l'utilisateur ajoute le nom de l'activité à simuler et son scénario (scénario de qu'elle activité, noms d'actions d'activité, durée d'actions d'activité) après l'utilisateur doit cliquer sur le bouton « Capteurs » pour donner un nom du capteur de chaque action d'activité comme c'est montrer dans la Figure 3.16.

Conclusion générale et perspectives:

L'objectif de notre travail est de concevoir et réaliser un simulateur pour activités humaines à base d'agents. Pour cela nous avons présenté dans ce mémoire un état de l'art sur la reconnaissance d'activités humaines, après avoir étudié les systèmes multi-agents.

Nous avons aussi étudié le simulateur SMACH. A travers cette étude nous nous sommes inspirés de son fonctionnement pour la conception et la réalisation de notre simulateur.

Afin de concevoir et de réaliser notre simulateur, on a opté pour l'utilisation du langage JAVA sous la plate-forme JADE.

Notre simulateur réalisé permet de :

- Simuler une activité pour plusieurs personnes.
- Créer le scénario d'activité à simuler en choisissant : le nom d'activité à simuler, nom d'action d'activité et sa durée, le capteur de chaque action d'activité, le nom d'agent humain.
- Visualiser le résultat de la simulation.

En perspective, nous espérons que notre simulateur sera étendue pour :

- Fixer les actions de chaque activité.
- Fixer les capteurs sur chaque objet.
- Simuler plusieurs activités nécessaires pour une seule personne.
- Réaliser une évaluation de performance.

Bibliographie:

- [1] Thomas W. Malone, Kevin Crowston, 1994, The interdisciplinary study of coordination; ACM computer surveys, 26, n° 1, pp. 87-117.
- [2] J. Ferber. Les systèmes multi-agents. Vers une intelligence collective. Editions Inter Editions, 1995.
- [3] N. Jennings, K. Sycara, M. Wooldridge, A roadmap of agent research and development. Autonomous Agents and Multi-Agent Systems, 1: pp. 7–38, 1998.
- [4] Stuart Russell & Peter Norvig, P, "Artificial intelligence: A modern approach", second edition, Englewood Cliffs, NJ: Prentice Hall, 1995.
- [5] B. Houssein. Enchères multi-objets pour la négociation automatique et le commerce électronique. Mémoire de Master, Université Laval, 2001.
- [6] M. Toumi, H. Deralia, Université de Guelma, Faculté des Sciences et de L'ingénierie, Mémoire de Master, Département d'Informatique, Spécialité : Ingénierie des médias, Automate cellulaire pour la simulation orientée agents, 2011.
- [7] B. Hadj, K. M'raoui, (INI) Oued Smar, thèse magister en informatique, «Approche de modélisation des tests de logiciels complexes par un système multi-agents », 2006.
- [8] T. MARIR, Université Larbi Ben M'hidi – Oum El Bouaghi, Un support de cours destiné aux étudiants de 02ème année Master Option : Architectures Distribuées, Les Systèmes Multi-Agents, 2016/2017.
- [9] M. Seif Eddine. Conception d'une architecture basée agents pour la création d'un marché virtuel. Mémoire Magistère, Département d'Informatique, Université Mentouri de Constantine, 2008.
- [10] B. Chaib-draa, I. Jarras, and B. Moulin. Systèmes multi-agents : Principes généraux et applications. Hermès, 2001.
- [11] DAURES, Nicolas, Etude de la simulation de systèmes multi-agents pour la conception vivante d'agents dans la méthode ADELFE, 2006, 64 p, Rapport de Master 2 Recherche, Université Paul Sabatier-Toulouse.

- [12] BUFFET, Olivier. Une double approche modulaire de l'apprentissage par renforcement pour des agents intelligents adaptifs, 2003, 215p. Thèse de doctorat, Université Henri Poincaré-nancy 1.
- [13] Lars Braubach, Alexander Pokahr (DistubutedSystemsGroub), "Jadex User Guide", Université of Hambourg, Germany, 2007
- [14] Fegas Mounir, AbbouteHassane, Boularias Abdeslam, Ramdanilotfi. Plates-formes de développement des systèmes Multi-Agents, Mise à jour : 2005.
Disponible:http://www.limsi.fr/~jps/enseignement/examsma/2005/1.plateformes_1/plateformes.htm
- [15] Florin Leon, MarcinPaprzycki, and Maria Ganzha. 2015, A Review of Agent Platforms, Multi-Paradigm Modelling for Cyber-Physical Systems (MPM4CPS), ICT COST Action IC1404, pp. 1-15.
- [16] L.B. Marjorie. Un simulateur multi-agents pour l'aide à la décision d'un collectif: Application à la gestion d'une ressource limitée agro-environnementale. Thèse de Doctorat, Université Paris IX-Dauphine, 2003.
- [17] B. Moulin ,B. Chaib-draa. An overview of distributed artificial intelligence. In G. M. P. O'Hare and N. R. Jennings, editors, Foundations of Distributed AI, pp. 3-54. John Wiley&Sons: Chichester, England, 1996.
- [18] B. Hedjazi and Dellal. Workflow de décision coopérative à base de système multi-agents: Cas de processus de décision spatial. Mémoire de Magistère, Institut National de Formation en Informatique (I.N.I) Algérie, 2006.
- [19] C. Alexis. Mécanisme de coordination multi-agents fondé sur des jeux : application à la simulation comportementale de trafic routier en situation de carrefour. Thèse de Doctorat, Université de Valenciennes et du Hainaut-Cambrésis, 2003.
- [20] H. Penny, "Blackboard Systems: The Blackboard Model of Problem-solving and the Evolution of Blackboard Architectures", AI Magazine, 7 (2), pp. 38 - 53, 1986.

- [21] Daniel D. Corkill. "Collaborating Software: Blackboard and Multi-Agent Systems & the Future", In Proceedings of the International Lisp Conference, New York, New York, 2003.
- [22] C. Hewitt, "Viewing control structures as patterns of passing messages", *Artificial Intelligence*, 8: pp. 323–364, 1977
- [23] AUSTIN J.L, "How to do things with words", Clarendon Press, Oxford, 1962.
- [24] Y. Labrou and T. Finin, "A Proposal for a new KQML Specification", rapport technique TR CS-97-03, C.S. and E.E. dep., university of Maryland Baltimore County, 1997.
- [25] T. Finin, R. Fritzson, D. McKay, and R. McEntire, "KQML as an agent communication language", In Third international conference on information and knowledge management. ACM Press, 1994.
- [26] Jarras, I. et B.Chaib-draa, 2002, Aperçu sur les systèmes multi-agents. *Sériescientifique*. 2002s-67. Montréal.
- [27] HanineSahki, Université Mentouri de Constantine, Mémoire présenté en vue de l'obtention du Magister en Informatique Option : Systèmes d'Information & Intelligence Artificielle Distribués, Vers une Architecture d'un Système de Dialogue Multi Agents Basé sur l'Argumentation Application à la négociation dans le domaine de e-commerce, 2008.
- [28] ZEGHIDA Djamel, UNIVERSITE BADJI MOKHTAR-ANNABA, THESE de Doctorat en Sciences, Option : Intelligence Artificielle, Démarche de développement des systèmes logiciels biomorphiques, 2019.
- [29] T. Boultache, A. Lamzaoui, Reconnaissance d'activités humaines à l'aide de capteurs de Smartphone. Mémoire de master 2 : Réseaux, Mobilité et Systèmes Embarqués, Université Mouloud Mammeri de Tizi-Ouzou, 2019-2020.

- [30] N. Taouri, Clustering hiérarchique pour la classification et la reconnaissance des activités dans un système ubiquitaire. Mémoire de master 2 : Génie Logiciel, Université Abderrahmane Mira-Béjaia, 2020
- [31] S. Semchaoui, Approche système expert pour la reconnaissance des activités humaines dans un système ubiquitaire. Mémoire de master2 : Réseaux et Systèmes Distribués Université Abderrahmane Mira de Béjaia, 2018-2019.
- [32] M. Selmi ; Reconnaissance d'activités humaines à partir de séquence vidéo ; Réseaux et télécommunication ; Institut National des Télécommunication ; 2014.
- [33] D. Roggen, S. Magnenat, M. Waibel, and G. Troster ; Designing and sharing activity recognition systems across platforms ; IEEE Robotics et Automation Magazine ; 2011; pp. 5 - 6.
- [34] I. Sarray, Conception de systèmes de reconnaissance d'activités humaines. Thèse de doctorat, Université Côte d'Azur, France, 2020
- [35] A. Khelalef, Reconnaissance d'activités humaines en utilisant les descripteurs spatio-temporels 2D/3D. Thèse de doctorat : Traitement du signal, Université Batna 2- Mostefa Ben Boulaid, 2020.
- [36] N.YALA, Université des Sciences et de la Technologie Houari Boumedienne Alger, thèse de doctorat en sciences, Electronique spécialité : Traitement du Signal et des Images, Contribution aux méthodes de classification de signaux de capteurs dans un habitat intelligent, 2019.
- [37] Nishikawa H., Yamamoto S., Tamai M., Nishigaki K., Kitani T., Shibata N., Yasumoto K., and Ito M., 2006. UbiREAL: Realistic Smartspace Simulator for Systematic Testing. *In Dourish, P., Friday, A. (eds.) UbiComp'2006. Springer, Berlin, Heidelberg, LNCS, 4206: pp. 459-476.*
- [38] Bouchard K., Ajroud A ., Bouchard B ., Bouzouane., 2010. SIMACT: a 3D Open Source Smart Home Simulator for Activity Recognition with Open Database and Visual Editor. *Proc. of International Conference on Ubiquitous Computing and Multimedia Application (UCMA'10), June 23-25, Miyazaki, Japan, pp. 524-533.*

[39] Yvon Haradji, Julien Guibourdenche, Quentin Reynaud, Germain Poizat, Nicolas Sabouret, François Sempé, Thomas Huraux et Mariane Galbat, 2018., De la modélisation de l'activité humaine à la modélisation pour la simulation sociale : entre réalisme et fécondité technologique.

[40] Yvon Haradji, « Simulation multi-agent de l'activité humaine : une concrétisation en ergonomie du programme de recherche technologique « cours d'action » », Activités [En ligne], 18-1 | 2021, mis en ligne le 15 avril 2021, consulté le 17 avril 2021. URL : <http://journals.openedition.org/activites/6166> ; DOI : <https://doi.org/10.4000/activites.6166>.

[41] Jérémy ALBOUYS-PERROIS, UNIVERSITÉ DE LA ROCHELLE, thèse de doctorat, Discipline : Informatique et Applications, Simulation multi-agent de l'autoconsommation collective de l'énergie à l'échelle du quartier en lien avec l'activité humaine et la consommation énergétique des bâtiments, 2021.

[42] Younsi K., MALA L., 2009. Conception et réalisation d'un simulateur pour systèmes distribués. Mémoire du diplôme d'ingénieur d'état : Systèmes parallèles et Distribués. Université de Béjaia.

[43] Guermoud M., Benamar A., 2011. Conception et Implémentation d'un Système multi-agent Pour le test de primalité de nombre premier. Mémoire de Licence. Université Abou Bakr Belkaid– Tlemcen.

[44] <https://fr.wikipedia.org/wiki/JavaFX>.

Résumé:

Dans ce mémoire nous avons donné des généralités sur la reconnaissance d'activités humaines, après avoir présenté la notion d'agent et systèmes multi-agents. Nous avons étudié la plate-forme de simulation multi-agent de l'activité humaine au sein de l'habitat SMACH ainsi son fonctionnement et son architecture.

Finalement notre contribution consiste en la conception et la réalisation d'un simulateur d'activités humaines à base d'agents, en utilisant le langage JAVA sous la plate-forme JADE.

Mots clés : reconnaissance d'activités humaines, agent, système multi-agents, SMACH, simulateur, JADE.

Abstract:

In this report we gave generalities on the recognition of human activities, after having presented the concept of agent and multi-agent systems.

We studied the multi-agent simulation platform for human activity within the habitat SMACH, as well its operation and architecture.

Finally our contribution is the conception and realization of a simulator of a simulator of human activities based on agents, using the JAVA language under the JADE platform.

Keywords: recognition of human activities, agent, multi-agent system, SMACH simulator, JADE.