

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

Université de Guelma

Faculté des Sciences et de L'ingénierie

Mémoire de Master



Département d'Informatique

Spécialité : Ingénierie des média

Automate cellulaire pour la simulation orientée agents

Présenté par :

TOUMI Mehdi
DERABLIA Hichem

Sous la direction de :

Mme. BENHAMZA Karima

- Juin 2011 -

Remerciement

Au nom de Dieu clément et miséricordieux, le grand merci lui revient, celui qui a nous créé, formé, et nous données la forme qu'il a voulu.

Ce mémoire représente l'achèvement de plusieurs années d'un long travail qui n'aurait pu voir le jour sans la participation, l'aide, les conseils, ou encore la présence de nombreuses personnes.

Nos remerciements les plus vifs, vont en premier lieu à notre encadreur Mme Benhamza Karima, qui nous a dirigé, orienté, conseillé et aidé dans l'élaboration de notre projet.

Nous tenons également à remercier le président du jury ainsi que l'examineur qui ont accepté de juger notre travail et évaluer sa valeur scientifique.

Nous exprimons également nos gratitudees à tous les enseignants qui ont corroboré à notre formation aussi bien primaire qu'universitaire.

C'est un énorme remerciement que nous adressons maintenant à nos parents. A tous ceux et celles qui directement ou indirectement de prés ou de loin nous ont aidé à réaliser ce modeste travail.

Résumé

Les systèmes multi-agents (SMA) proposent une approche nouvelle pour le développement à plusieurs composantes autonomes pouvant coopérer entre elles. Un défi permanent est d'appliquer ces SMA à la modélisation et la simulation orientée agents dont l'objectif est de comprendre les liens individus/collectifs et faire émerger un comportement dit « intelligent » pour que collectivement il y ait résolution de problèmes complexes.

Dans ce travail, on s'intéresse à l'un des outils de modélisation et simulation orientée agents qui est l'automate cellulaire. Les automates cellulaires sont des structures abstraites aux propriétés très complexes et dont l'abord n'est pas immédiat et qui permettent d'étudier des univers virtuels. Nous présentons un algorithme de segmentation d'images par les automates cellulaires et basé sur l'intelligence collective. Le modèle proposé et implémenté sur une plate-forme multi-agents, donne des résultats très satisfaisants et montre l'efficacité de notre algorithme.

Mots clés : Agent, système multi-agent (SMA), auto-organisation, modélisation, simulation, automate cellulaire, segmentation contour/région

SOMMAIRE

Sommaire	1
Table des figures	4
Introduction générale	6

Chapitre 01 : Les systèmes Complexes

I. Introduction.....	9
II. Définition	9
III. Caractéristiques.....	10
IV. Comportement	11
V Exemples de systèmes complexes	11
VI. Méthodes de résolution	12
1. Les méthodes mathématiques	13
2. Les méthodes par construction.....	13
3. Les méthodes par décomposition.....	13
4. Les méthodes par voisinage ou de recherche locale.....	13
5. Les méthodes évolutives.....	14
VIII. Conclusion.....	14

Chapitre 02 : La modélisation et la simulation orientée agent

I. Introduction.....	15
II Agent.....	15
1. Définition Agent	15
2. Architectures d'agents.....	17
2.1. Agents cognitifs	17
2.2. Agents réactifs	18
2.3 Agents hybrides	19

2.4 Agents cognitifs VS Agents réactifs.....	20
III. Système multi-agents	20
1. Définition d'un SMA (Système Multi-Agents).....	21
2. Caractéristiques d'un SMA :.....	22
IV. Auto-organisation dans les SMA réactifs.....	22
1. Définition.....	22
2. Émergence	23
3. La modélisation et simulation orientée agent.....	24
a. Modélisation orientée agent.....	24
b. Simulation Orienté agent	25
4. Simulation orientée agent	26
5. La modélisation orientée agent pour la résolution de problème complexe.....	28
VI. Conclusion.....	28

Chapitre 3 : Les automates cellulaires

I. Introduction.....	29
II Définition	29
III. Principe de création d'un automate cellulaire	31
IV. Les différentes architectures d'AC	31
1. Architecteurs à une dimension (1D)	31
2. Architectures à deux dimensions	32
3. Architectures à trois dimensions.....	33
V. Voisinage avec un schéma pour chaque type	33
VI. Caractéristiques	34
VII. Comportement	35

VIII. Application d'automate cellulaire.....	37
IX. Conclusion.....	37
Chapitre 4 : Conception et Implémentation	
I. Introduction.....	38
II. Description et objectifs de l'application	38
II.1. Choix de l'application	38
II.2. Présentation du problème « Segmentation d'image contour et région »	39
II.3. Objectifs de l'application	39
II.4. Travaux dans le domaine.....	40
III. Conception.....	41
III. 1. Modélisation du problème	41
III.1.1. Modélisation de l'Environnement.....	41
III.1.2. Modélisation de l'agent « Automate Cellulaire »	42
III.2. Comportement des agents « Automate cellulaire de segmentation ».....	42
III.3. Algorithme principal	42
III 3.1. Principe.....	42
III.3.2 Architecture de l'application	43
IV. Réalisation	44
IV.1. Environnement de programmation	45
IV.1.1. Programmation orientée agents	45
IV.1.2 Plate-forme d'implémentation (NetLogo)	46
IV. Présentation de l'application	47
V. Résultats expérimentaux.....	51
V.1. Discussion	51

1.1 Segmentation contour.....	51
1.2 Segmentation région.....	53
1.3 Segmentation mixte.....	55
VI. Conclusion	58
Conclusion Générale	59

Table des figures

Chapitre 02 : La modélisation et la simulation orientée agent

<i>Figure 2.1</i> : les agents interagissent avec leurs environnements au moyen de capteur et d'effecteurs [Russell et al., 2006]	16
<i>Figure 2.2</i> : Schéma d'un agent [Ferber, 1999]	17
<i>Figure 2.3</i> : Architecture fonctionnelle d'un agent cognitif [Hadj et al.,2006]	18
<i>Figure 2.4</i> : Fonctionnement d'un agent réactif [Hadj et al.,2006]	19
<i>Figure 2.5</i> : Architectures d'agents en couches [Jennings et al.,1998]	19
<i>Figure 2.6</i> : Positionnement des SMA	20
<i>Figure 2.7</i> : Système multi-agents [Ferber, 1995].	21
<i>Figure 2.8</i> Le désordre organisateur	22
<i>Figure 2.9</i> : Principe de la simulation	25
<i>Figure 2.10</i> Schéma de la Modélisation	26
<i>Figure 2.11</i> Le processus de simulation[Ralambondrainy, 2009].	27

Chapitre 03 : La modélisation et la simulation orientée agent

<i>Figure 3.1</i> Représentations d'automate cellulaire	30
<i>Figure 3.2</i> Représentations des cellules de l'automate unidimensionnel	32
<i>Figure 3.3</i> Représentations des cellules d'automate bidimensionnel. [Kramm, 2005]	32
<i>Figure 3.4</i> Représentations des cellules d'automate tridimensionnel	33
<i>Figure 3.5</i> représentation des différents types de voisinages; a) type de réseau hexagonal, b) environnement de Von Neumann, c) environnement de Moore.	34
<i>Figure 3.6</i> : états initiale d'automates	36
<i>Figure 3.7</i> : jeu de la vie	37

Chapitre 4 : Conception et implémentation du système

<i>Figure 4.1 : Voisinage de Moore</i>	42
<i>Figure 4.2 : Organigramme de l'algorithme proposé</i>	44
<i>Figure 4.3 : interface Net logo</i>	46
<i>Figure 4.4 : interface graphique de l'application</i>	47
<i>Figure 4.5 : Importation d'image</i>	48
<i>Figure 4.6 : Segmentation Région</i>	48
<i>Figure 4.7 : Réaction du pixel dans l'espace de couleur Net Logo</i>	49
<i>Figure 4.8 : Segmentation Contour</i>	50
<i>Figure 4.9 : Segmentation Contour au ralenti, Visualisation 3D</i>	50
<i>Figure 4.10 : (a) Images originaux, (b) Application du filtre Sobel, (c) Nos résultats</i>	52
<i>Figure 4.11 résultats comparative de segmentation région</i>	55
<i>Figure 4.12 : Segmentation Mixte</i>	56
<i>Figure 4.13 : résultats de Segmentation Mixte</i>	57

Introduction générale

L'Intelligence Artificielle (IA) est reconnue comme étant une discipline informatique qui a pour objectif de reproduire des comportements humains dits intelligents tels que la perception, la prise de décision, la compréhension, l'apprentissage, etc.

Donc une création de programme informatique, capable d'effectuer des tâches complexe en s'appuyant sur une centralisation et une concentration de l'intelligence au sein d'un système. Ce qui mène que l'IA a vite rencontré un certain nombre de difficultés, dues pour la plupart à la nécessité d'intégrer, au sein d'une même base de connaissances, l'expertise, les compétences et les connaissances d'individus différents qui, reproduisent la réalité, communiquent et collaborent à la réalisation d'un but commun.

Au début des années 80, de la volonté de remédier aux insuffisances et d'enrichir l'approche classique de l'IA en proposant la distribution de l'expertise sur un groupe d'agents, alors une Intelligence Artificielle Distribuée (IAD) est née.

L'inspiration de la vie biologique pour la conception de comportement intelligent conduit vers une nouvelle voie de recherche, une nouvelle discipline qui a vite fait sa place parmi les sciences et reconnue sous le nom de « La vie artificielle ».

La vie Artificielle engage de nombreuses disciplines telles que la biologie, les sciences cognitives, l'informatique, les mathématiques, l'intelligence artificielle, etc., Dans les années 1940, Von Neumann (1903-1957) a initié l'étude du vivant au travers de modèles informatiques. De ce fait une modélisation et une simulation des comportements intelligents, qui sont le produit de l'activité coopérative entre plusieurs agents « système multi-agents », est nécessaire pour résoudre des problèmes éventuellement complexes. Ce principe s'appuie sur l'approche reconnue sous le nom de « simulation orienté agents ».

La simulation orientée agent a démontré l'efficacité de son paradigme auprès des autres sciences, Aujourd'hui, elle apparaît comme une science à part entière et non plus comme un simple outil. Ceci est dû pour une part, son utilisation dans un nombre croissant de secteurs, où elle remplace progressivement les différentes techniques de micro-simulation, de simulation orientée objets ou individu-centrée utilisées jusqu'à présent.

Parmi les modèles de simulation orientés agent, les automates cellulaires (AC) détiennent une place importante comme modèle mathématique qui émerge vers une résolution de système complexe. Ceci est dû à son formalisme, sa simplicité, sa souplesse qui permet une parfaite maniabilité.

Objectif du travail

L'objectif de notre travail est la mise en application de la technique de la simulation orientée agent via les automates cellulaires pour la résolution du problème complexe. Vu notre spécialité « Ingénierie des Media », nous avons choisi le problème de la segmentation d'image (contour et région). Ce problème est reconnu comme complexe et n'étant pas encore résolu avec un niveau de satisfaction élevé. Ce travail a donné des résultats satisfaisants en appliquant les Automates cellulaires pour:

- une segmentation de différentes catégories d'image (couleur et niveau de gris).
- une segmentation d'images d'extension standard (BMP, JPG, GIF, et PNG)

Consistance du mémoire

Ce mémoire est organisé comme suit :

Dans le Chapitre 1, nous présentons une étude sur les systèmes complexes. Puis nous exposons dans le chapitre 2 les méthodes de résolution de ces systèmes via la modélisation et la simulation orientée agent. Le chapitre 3 présente les automates cellulaires comme outil de modélisation et simulation orientée agent, puis la conception et l'implémentation du système proposé sont exposées dans le chapitre 4. Nous terminons ce mémoire avec une conclusion générale, quelques perspectives et les références bibliographiques.

Chapitre 1 :

Les systèmes Complexes

I. Introduction :

Le monde qui nous entoure est complexe et cette complexité ne cesse de s'amplifier en regard de l'explosion des échanges d'information, des flux de données et de la révolution de l'internet.

L'observation du monde réel est une source d'acquisition de connaissances. Cependant de nombreuses barrières viennent limiter cette observation : grandeurs du système (temporelles ou spatiales), coût, lois, éloignement géographique, précision des outils de mesure, perturbations induites au moment de la mesure enfin complexité du système.

Dans ce chapitre, la notion de complexité est introduite avec caractéristiques, et comportement. Des exemples de systèmes complexes sont présentés et on termine avec les méthodes de résolutions.

II. Définition :

« *Le tout est plus que la somme de ses parties* » [Aristote, 2002]. Souvent utilisée à tort et à travers, cette proposition devient de plus en plus d'actualité à propos de ce qui est appelé « système complexe ».

Un système complexe est composé d'un grand nombre d'éléments ou d'entités qui interagissent simultanément, tout en empêchant l'observateur de prévoir sa rétroaction.

Plusieurs définitions des systèmes complexes existent dans la littérature, celles de :

[Simon, 1962] qui définit un système complexe comme un système constitué d'un nombre important de parties qui interagissent d'une manière non simple. Et aussi de

[Moigne, 1990] qui caractérise la complexité des systèmes par l'imprévisibilité possible et l'émergence du nouveau et du sens plausible.

Ces deux définitions sont incomplètes mais complémentaires, une qui évoque le nombre important des entités en interactions et l'autre qui parle d'imprévisibilité du comportement.

Une définition plus complète d'un système complexe est donnée par :

[Ralambondrainy, 2009] : « La complexité pouvant provenir du nombre important de parties du système qui interagissent d'une manière non simple, et conduisant à de l'imprévisibilité possible, de l'émergence du nouveau et du sens plausible ».

Des recherches plus approfondies conduisent à lier la complexité à différentes notions [Rouquier, 2008 ; Moujahed, 2007]:

- La complexité est tout d'abord liée au hasard ou au désordre.
- La complexité est liée à ce que nous appellerons « la multiplicité des unités ».
- La complexité est par conséquent liée aux relations, à l'organisation qui résulte de l'ordre ou désordre.
- La complexité est liée également à l'interdépendance des phénomènes; en relation avec leurs irréversibilités. Il existe trois formes d'interdépendance, celle du système avec son environnement, celle du système avec ses propres sous-systèmes, et enfin l'interdépendance entre les éléments d'un système.
- La complexité est enfin liée à la récursivité des phénomènes et à la notion de rétroaction.

En nous fondant sur ces notions, on peut dégager les caractéristiques les plus importantes d'un système complexe :

III. Caractéristiques :

On peut dire qu'un système est complexe si:

- Il est composé d'un grand nombre fini d'entités.

- Un comportement global structuré jaillit.
- Les interactions sont locales, de même que la plupart des informations, il y a peu d'organisation centrale.
- Le graphe d'interaction est non trivial.
- Il y a des boucles de rétroaction (feedback) : l'état d'une entité a une influence sur l'état d'autres entités, qui en retour ont une influence sur la première.

De plus, les entités d'un système complexe peuvent être elles-mêmes des systèmes complexes, faisant ainsi apparaître une hiérarchie.

Sur la base de ces caractéristiques, le comportement d'un système complexe peut être déduit.

IV. Comportement :

Un système complexe présente la plupart des comportements suivants :

- Il y a organisation autonome et apparition de propriétés ou de structures cohérentes, ou de motifs ; Le système a une mémoire de son évolution. Son état présent influence les états futurs.
- Si un élément est affecté par un événement extérieur, ses voisins le seront aussi.
- La connaissance d'une partie du système ne permet pas d'affirmer la connaissance du système global.
- Plusieurs comportements possibles sont en compétition, certains sont simples, d'autres chaotiques.

Pour mieux connaître les systèmes complexes on propose de voir quelques exemples.

V. Exemples de systèmes complexes :

- Colonie de fourmis dans son activité de fourragement (qui consiste à aller chercher des ressources dans l'environnement et les ramener dans la fourmilière) : Les fourmis sont des organismes qui ont des comportements relativement simples au niveau individuel, mais dont le groupe montre des comportements complexes jaillissants d'une stupéfiante organisation.

- La ville est considérée comme un système complexe, spatialement organisé (trafic routier, évolution d'infrastructure,...). Sa détermination doit prendre en compte à la fois les facteurs socio-économiques, politiques ou géographiques agissant à des échelles d'observation variées et selon des temporalités très différentes.

D'autres systèmes paraissent complexes mais ne le sont pas, en voici quelques exemples :

- Réactions chimiques : malgré le grand nombre de molécules et d'interactions en jeu, on connaît toujours le résultat ou on peut le prédire. La taille du système n'est donc pas un critère de complexité.
- Physique statistique étudie des particules extrêmement simples, comme les atomes d'un gaz. Elle s'intéresse au comportement limite lorsque le nombre de particules tend vers l'infini, alors qu'un système complexe a toujours un nombre fini d'éléments.

L'entropie n'est donc n'est pas non plus un critère de complexité : un gaz a une forte entropie car un nombre gigantesque de configurations possibles pour les atomes le constituant, et pourtant il n'est pas complexe : son comportement est simple à décrire, à l'aide d'outils statistiques.

Les systèmes complexes nécessitent des modèles et des méthodes adaptés aux problèmes pour leurs résolutions.

VI. Méthodes de résolution :

Plusieurs méthodes ont contribué à la formulation et la résolution des systèmes complexes, on peut citer :

Les modèles mathématiques, les méthodes par construction, les méthodes par décomposition, les méthodes par voisinage ou de recherche locale, les méthodes évolutives :

1. Les méthodes mathématiques :

- La description formelle des systèmes est généralement exprimée sous forme mathématique.

La mise en œuvre d'outils mathématiques pour la modélisation ou l'analyse de systèmes complexes permet de prévoir leur fonctionnement leur évolution afin de les résoudre. Si le nombre de paramètres de ces systèmes augmente la modélisation et la résolution mathématique devient très difficile.

2. Les méthodes par construction :

- Un algorithme glouton, est un algorithme qui permet généralement de résoudre d'une manière optimale différents problèmes, mais il existe des problèmes pour lesquels une petite augmentation de la quantité de données à traiter mène rapidement la résolution du problème hors des capacités des machines, tant le nombre de possibilités devient important.

3. Les méthodes par décomposition :

- Elles se basent sur la division du problème initial en plusieurs sous-problèmes de taille réduite, facilitant ainsi la résolution de ces sous-problèmes et permettant de construire une solution globale à partir des solutions partielles obtenues par chaque sous-problème. Il existe plusieurs techniques de décomposition selon que le type est : *hiérarchique, structurel, spatial, temporel*, ou *de l'ensemble des solutions* [Portmann, 1988]. Le problème est de savoir comment décomposer et aussi certains problèmes ne sont pas décomposables.

4. Les méthodes par voisinage ou de recherche locale :

- Les approches par voisinage, appelées également les méthodes de recherche locale, sont des algorithmes itératifs initialisés par une solution réalisable (obtenue soit de manière aléatoire, soit à l'aide d'une autre heuristique, par exemple une heuristique constructive). Ils cherchent à améliorer, à chaque itération, la solution courante par des modifications locales. La recherche consiste donc à se déplacer successivement dans l'ensemble des solutions en passant, à chaque étape, d'une solution à une autre voisine, jusqu'à la vérification d'une condition d'arrêt. Les modalités des déplacements conduisent à de nombreuses méthodes telles que : *La méthode de descente, La méthode de recherche tabou, La méthode de recherche*

à voisinage variable...etc. Le problème de ces méthodes est de tomber dans un minimum local.

5. Les méthodes évolutives :

- Contrairement aux méthodes par construction et par voisinage qui font intervenir une solution unique, les méthodes évolutives, dites aussi à base de population, considèrent un ensemble de solutions, appelé population. Ils font évoluer une population de solutions à chaque étape du processus de recherche, permettant d'identifier et d'explorer les caractéristiques que les solutions ont en commun. Parmi les méthodes de cette catégorie, on peut citer les algorithmes évolutionnaires, les algorithmes de colonies de fourmis, les méthodes par essais particuliers et les algorithmes à estimation de distribution et enfin les automates cellulaires. Toutes ces méthodes se basent sur la modélisation et la simulation orientée agent qui est nouvelle voie plus prometteuses pour la résolution des problèmes complexes.

VII. Conclusion :

Dans ce chapitre, on a abordé les systèmes complexes, leurs caractéristiques et leurs comportements. On a exposé quelques méthodes de résolutions et on a souligné l'importance de la modélisation et la simulation orientée agent qui fera l'objectif du chapitre.

Chapitre 2 :

La modélisation et la simulation orientée agent

I. Introduction :

Comme méthode de résolution de système complexe, la modélisation et la simulation orientée agent a été énoncée dans le chapitre précédent. Afin de bien comprendre cette méthode on commencera dans ce chapitre à définir tout d'abord l'élément essentiel qui est l'agent.

Dans la suite de ce chapitre on mentionnera en premier temps les différentes définitions d'un agent, leurs architectures et les concepts liés : système multi-agents, l'auto-organisation, émergence de système, on terminera avec une conclusion où on soulignera l'importance de la simulation orientée agent.

II. Agent :

De nos jours, le mot « agent » est utilisé dans plusieurs domaines, et plusieurs sens lui sont attachés car de nombreux chercheurs ont défini ce terme de manières différentes.

1. Définition Agent :

Les définitions les plus communes sont celles de Russell, Jennings et Ferber.

- «Un agent est un système informatique, situé dans un environnement, qui agit d'une façon autonome et flexible pour atteindre les objectifs pour lesquels il a été conçu.» [Jennings et al., 1998].
- «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 » [Russell et al., 2006].

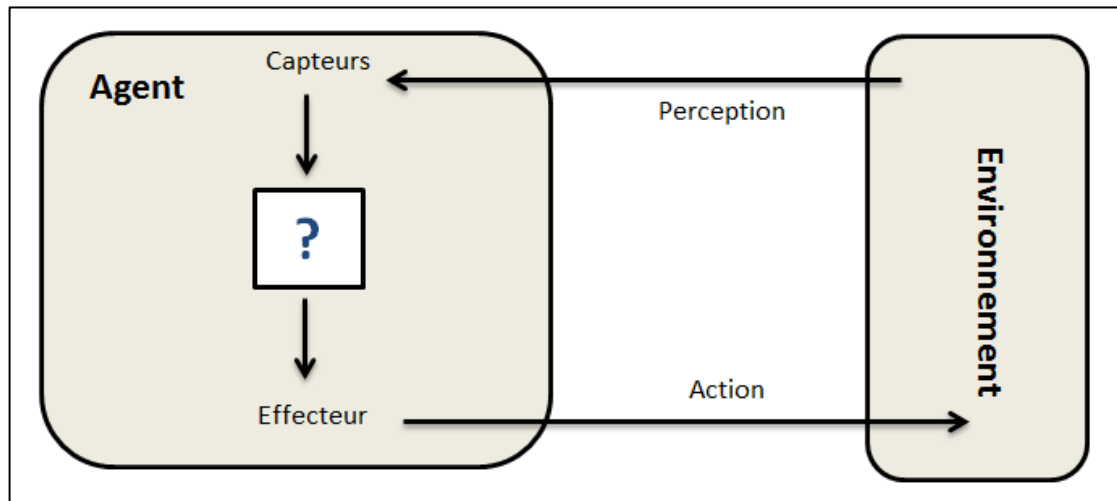


Figure 2.1 : les agents interagissent avec leurs environnements au moyen de capteur et d'effecteurs [Russell et al., 2006]

La définition de Ferber : « un agent est un ensemble d'entités logicielles, éventuellement hétérogènes, qui possèdent des capacités propres de résolution de problèmes et qui sont capables d'interagir pour atteindre leurs buts. [Ferber,1995].

Cette notion a été élargie par le même auteur [Ferber, 1999]: « On appelle agent, une entité physique ou virtuelle :

- qui est capable d'agir dans un environnement.
- qui peut communiquer directement avec d'autres agents.
- qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser).
- qui possède des ressources propres.
- qui est capable de percevoir (mais de manière limitée) son environnement.
- qui ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune).
- qui possède des compétences et offre des services.
- qui peut éventuellement se reproduire.
- dont le comportement tend à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit.».

La figure ci-dessous explicite l'architecture externe d'un agent :

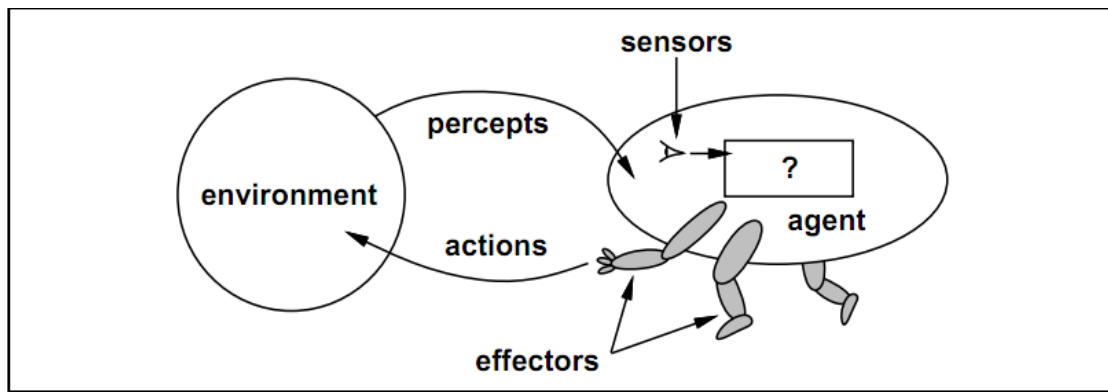


Figure 2.2 : Schéma d'un agent [Ferber, 1999].

2. Architectures d'agents :

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

2.1 Agents cognitifs : les agents cognitifs sont définis par leur capacité de raisonner sur des représentations du monde. Ils sont capables, à la fois de mémoriser des situations, de les analyser, de prévoir des réactions possibles à leurs actions, d'en tirer des conduites pour les événements futurs et donc de planifier leur propre comportement. [Ferber, 1995], ils sont connus aussi par «agents fondés sur les connaissances» [Russell et al., 2006].

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 pour implémenter un système basé sur plusieurs agents.

Ces caractéristiques offrent des avantages tels que une performance fiable pour des actions en temps réel, mais aussi des inconvénients majeurs tels que : complexité de communication, temps important de réalisation des tâches et une impossibilité d'adaptation pour un environnement dynamique.

La figure 2.3 ci-dessous explicite le fonctionnement d'un agent cognitif :

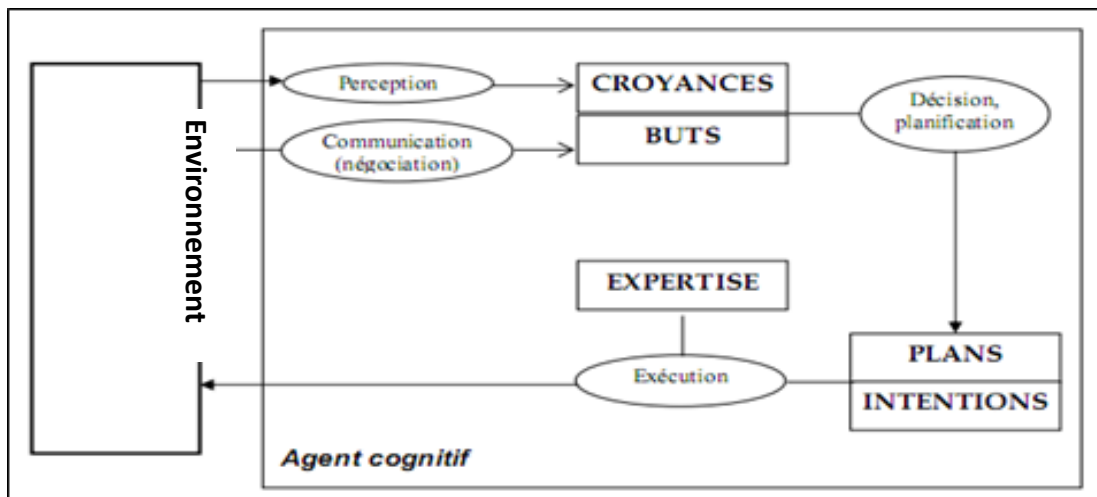


Figure 2.3 : Architecture d'un agent cognitif [Hadj et al., 2006].

2.2 Agents réactifs : Les agents réactifs sont définis, par le fait même qu'ils n'ont pas de représentation de leur environnement et des autres agents, sont incapables de prévoir ce qui va se passer et donc d'anticiper en planifiant les actions à accomplir. [Ferber et al., 1995].

Leur principe de réaction est lié directement à leur perception d'environnement, ou ils réagiront et interagiront de façon dynamique selon les données fournies en temps réel sans mémorisation.

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

- ✓ Pas de mémoire de passé.
- ✓ Pas de représentation explicite.
- ✓ La forme de prise de décision se fait sous «stimuli-réponse».
- ✓ L'organisation entre les agents inspirés du phénomène biologique.
- ✓ L'implémentation du système compte beaucoup d'agent.

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.

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

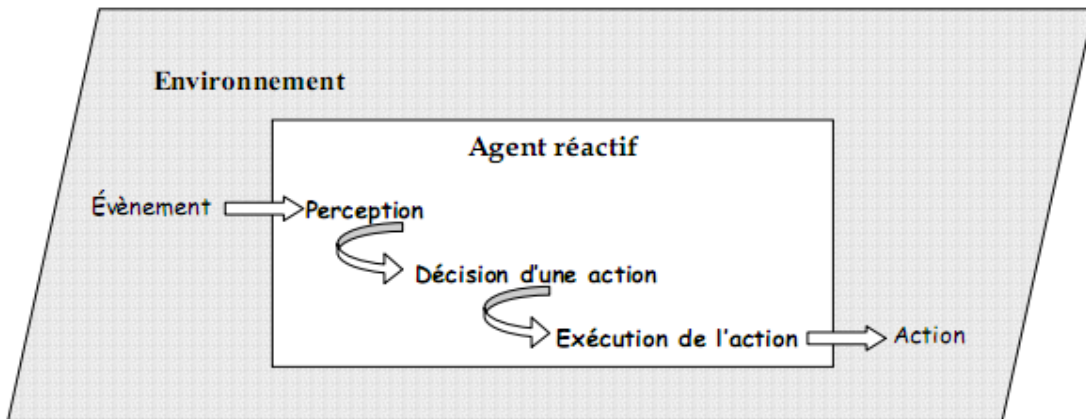


Figure 2.4 : Fonctionnement d'un agent réactif [Hadj et al.,2006] .

2.3 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.

Généralement on trouve deux modes de présentation pour cette architecture, horizontale et verticale, présenté dans la figure suivante :

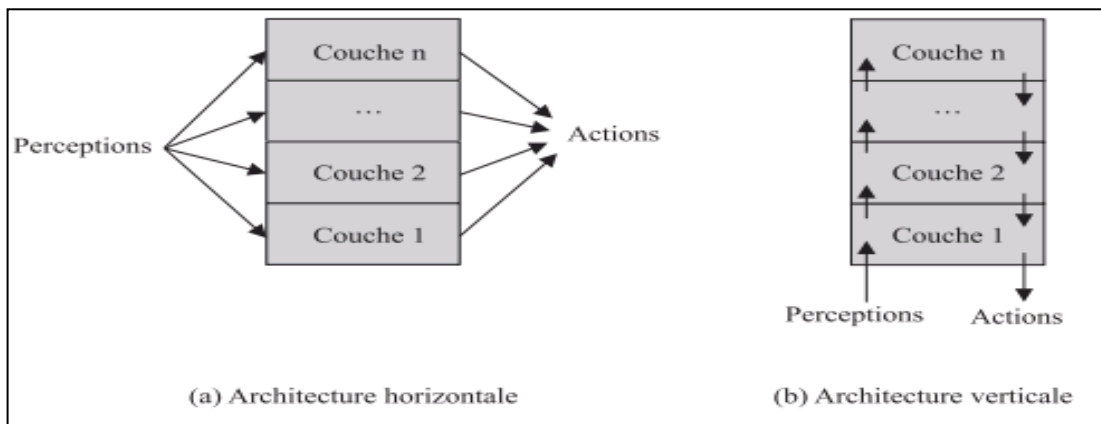


Figure 2.5 : Architectures d'agents en couches [Jennings et al., 1998].

2.4. Agents cognitifs VS Agents réactifs:

Chacune de ces architectures a ses avantages et ses inconvénients. Dans notre cas, on adoptera l'architecture d'agent réactif dans la construction d'un Système d'Agents pour la résolution de problèmes complexes, vu la similarité entre un système complexe et un système formé d'agents réactifs : imprévisibilité du comportement et grand nombre d'entités (agents) en interaction.

Les agents réactifs sont la solution aux problèmes de : la limite d'adaptation à son environnement, la complexité du système de communication, ajustement difficile des règles de comportement rencontrées par les agents cognitifs.

En fait, la plupart du temps un agent réactif n'est pas seul dans son environnement. Les agents doivent être capables d'interagir entre eux. Ils peuvent soit coexister, coopérer ou être en compétition entre eux. On parle alors d'un système multi-agents.

III. Système multi-agents :

L'utilisation des systèmes multi-agents (SMA) sont à l'intersection de plusieurs domaines de recherches, généralement en peut distinguer les quatre axes suivants : l'Intelligence Artificielle Distribuée basée sur la métaphore sociologique, le Génie Logiciel basé sur le paradigme objet, et les Systèmes Repartis basés sur les technologies réseaux et la Vie Artificielle, qui se base sur la métaphore biologique est inspirée des principes de bases sur la physique.

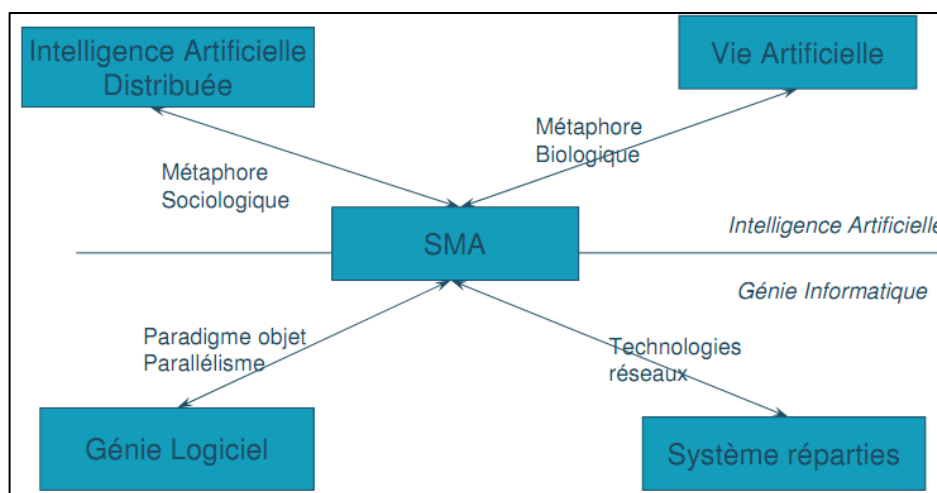


Figure 2.6 : Positionnement des SMA.

1. Définition d'un SMA (Système Multi-Agents) :

Plusieurs groupes de recherche proposent actuellement des définitions pour un SMA. On citera les plus connues

- [Ferber, 1995] : «Un SMA est défini comme:
 - ✓ Un ensemble B d'entités plongées dans un environnement E (E est caractérisé par l'ensemble des états de l'environnement S)
 - ✓ Un ensemble A d'agents avec $A \subseteq B$
 - ✓ Un système d'action (opérations) permettant à des agents d'agir dans E (une opération est une fonction de $S \Rightarrow S$)
 - ✓ Un système de communication entre Agents (envoi de messages, diffusion de signaux).
 - ✓ Une organisation O structurant l'ensemble des agents et définissant les fonctions remplies par les agents (notion de rôle et éventuellement de groupes)

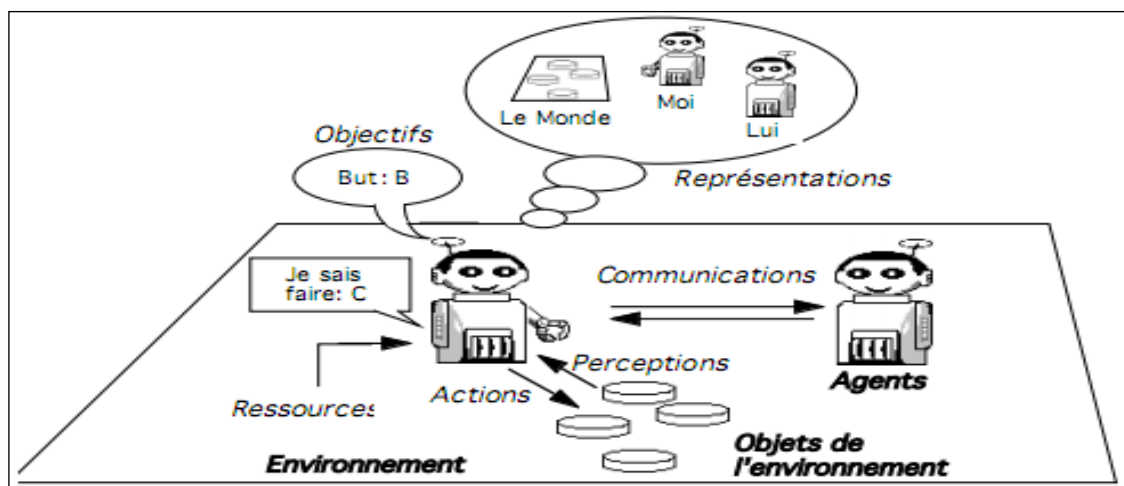


Figure 2.7: Système multi-agents [Ferber, 1995].

Dans notre cas, on s'intéresse aux systèmes multi-agents réactifs, les quelles sont défini par: [Simonin, 2001] : «Le système est composé d'agents autonomes interagissant localement, qui n'ont pas de mémoire individuelle, mais dont le système est lui-même une mémoire collective dynamique et qui constitue, à certains états, le résultat d'un problème ».

2. Caractéristiques d'un SMA :

Les systèmes Multi-agents réactifs sont caractérisés par les points suivant :

- ✓ Indépendance au niveau du comportement interne de chaque agent
- ✓ Autonomie
- ✓ Réaction asynchrone.

Une des caractéristiques les plus importantes dans les systèmes multi-agents réactif est le très forte dynamique du système c.-à-d. là, il devient très difficile de prévoir toutes les situations dès la conception, ou l'organisation globale du système cette caractéristique atteinte à long ou moyen terme sans supervision est appelée auto-organisation.

L'auto-organisation se réfère à ce qui est exactement suggéré par des systèmes qui s'organisent sans direction, manipulation ou contrôle externes. [Dempster, 1998]

IV. Auto-organisation dans les SMA réactifs:

a. Définition d'auto-organisation:

Pour Edgar Morin, l'auto-organisation c'est simplement et poétiquement:

« Le désordre organisateur ».

C'est aussi « une propriété d'un système rendant compte de sa capacité à transformer et se transformer, et produire et se produire, et relier et se relier, et maintenir et se maintenir ». [Morin, 1977].

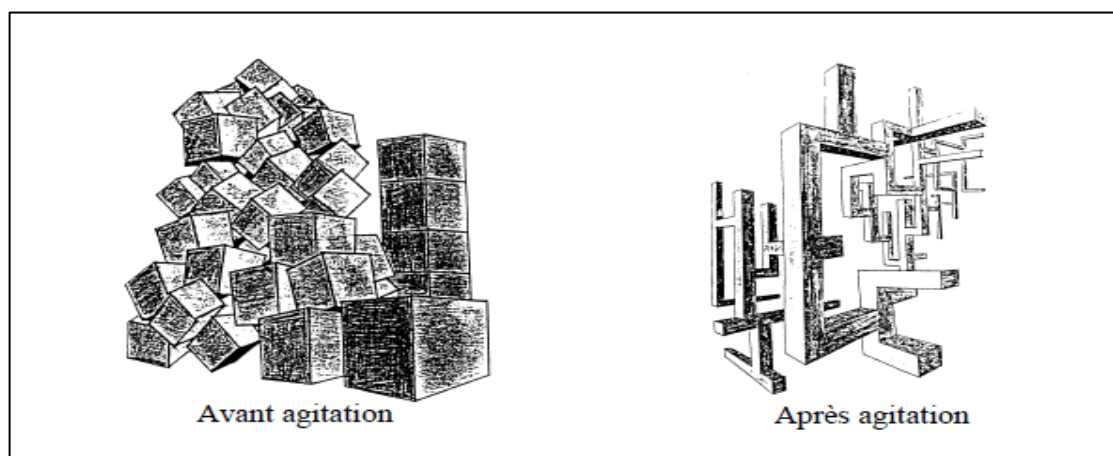


Figure 2.8 : Le désordre organisateur

Jacques Ferber dans son livre intitulé. “Les systèmes multi-agents - Vers une intelligence collective” consacre un chapitre entier aux organisations multi-agents et à leur analyse, [Ferber, 1995] «Les organisations constituent à la fois le support et la manière dont se passent les interrelations» et ajoute : «De plus, etc’est ce qui rend ce terme si difficile à cerner, l’organisation désigne donc à la fois le processus d’élaboration d’une structure et le résultat même de ce processus» .

Une définition minimaliste peut être donnée par Heylighen : « l’apparition de structures ou motifs sans aucune intervention externe »[Heylighen, 2002].

L’auto-organisation des agents dans leur environnement aboutit le plus souvent à l’apparition d’un phénomène imprévisible. Cette autre caractéristique est appelée émergence.

b. Émergence :

Beaucoup de travaux de recherche décrivent l’émergence et l’auto-organisation, incorrectement, comme des synonymes. Cette confusion fausse l’idée qu’on peut avoir quant à leurs significations respectives.

L’émergence est un concept pluri-défini. Elle est généralement vue comme un phénomène où le comportement global d’un système résulte des interactions entre les parties locales de ce même système.

En sciences, l’émergence sert souvent à décrire comment une multitude d’entités prises collectivement forme un tout cohérent. Cette cohérence signifie principalement qu’alors, le tout peut être étudié indépendamment de ses fragments, formant ainsi une nouvelle entité pour laquelle des lois, des propriétés et d’autres éléments d’une théorie peuvent être édictés. [Joris, 2008].

Dans ce qui suit, une définition plus poussée du concept d’émergence est présentée.

A system exhibits emergence when there are coherent emergents at the macro-level that dynamically arise from the interactions between the parts at the micro-level. Such emergents are novel with regard to the individual parts of the system. [De Wolf et al., 2004]

Cette définition utilise le concept d’« émergent » comme un terme général qui désigne le résultat du processus: propriétés, comportements, structures, modèles, etc. Le niveau (level) se rapporte au point de vue. Le niveau macro considère le système

dans son intégralité tandis que le niveau micro considère le système du point de vue des différentes entités qui le composent.

L'émergence et l'auto-organisation, soulignent des caractéristiques différentes du comportement d'un système. Les deux phénomènes peuvent exister séparément, comme ils peuvent coexister dans un système dynamique. Pour identifier ces deux phénomènes, une modélisation et simulation orientée agent est nécessaire. Est présenté ci-dessous

3. La modélisation et simulation orientée agent

L'observation du monde réel est une source d'acquisition de connaissances, qu'elles soient scientifiques ou non. Cependant de nombreuses barrières viennent limiter cette observation : grandeurs du système (temporelles ou spatiales), coût, éthique, lois, éloignement géographique, précision des outils de mesure, perturbations induites au moment de la mesure, complexité du système, etc. Ce qui conduit à l'utilisation de prototype ou de ce qu'on appelle « Modèle ».

3.1. Définition d'un model :

La définition de modèle la plus communément admise est celle donnée par Minsky. [Minsky, 1965].

Modèle : « pour un observateur B, un objet A* est un modèle d'un objet A dans la mesure où B peut utiliser A* pour répondre aux questions qui l'intéressent à propos de A»

La modélisation peut être donc une représentation informatique du système à étudier à partir des connaissances des spécialistes et des concepts informatiques dont disposent les modélisateurs.

Plusieurs modalisations existent. On s'intéresse à la modélisation orientée agents. Une fois le système complexe identifié et délimité dans le monde réel, il est nécessaire de lui donner corps dans le monde virtuel servant à la simulation : c'est l'étape de modélisation.

3.2. Définition de la simulation :

Le concept de simulation peut avoir différentes significations. Ainsi, comme l'annoncent Varenne [Varenne, 2001] et Pritsker [Pritsker, 1979], vingt et une (21) définitions différentes peuvent être répertoriées. Nous retiendrons la définition de Drogoul [Drogoul, 1993]: « On nomme simulation la démarche scientifique qui consiste à réaliser une reproduction artificielle, appelée modèle, d'un phénomène réel que l'on désire étudier, à observer le comportement de cette reproduction lorsqu'on en fait varier certains paramètres, et à induire ce qui se passerait dans la réalité sous l'influence de variations analogues ». Cette démarche est résumée par la Figure 2.9.

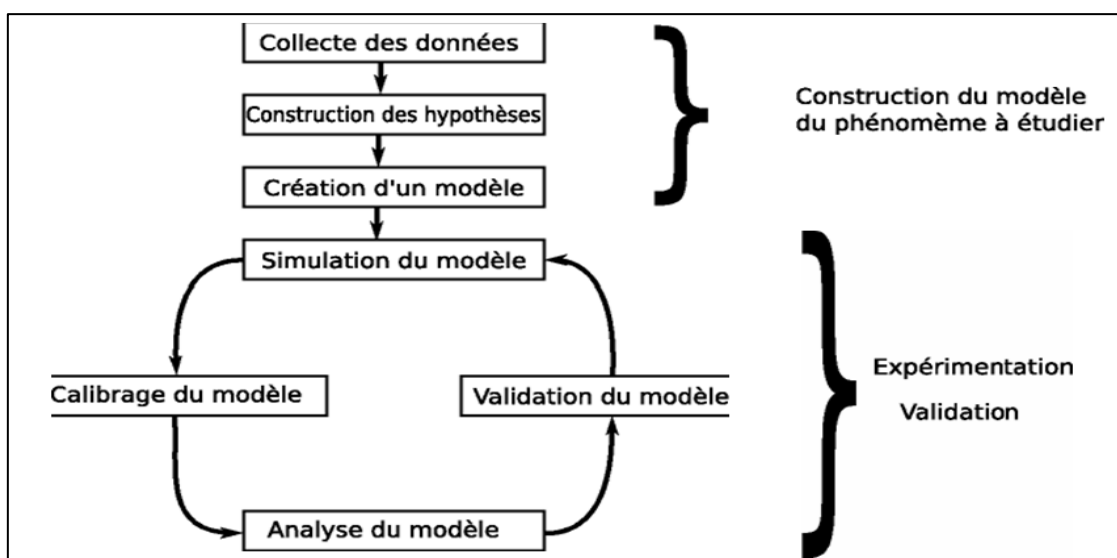


Figure 2.9 : Principe de la simulation.

La simulation consiste alors à reproduire le phénomène que l'on souhaite observer dans un environnement plus maîtrisable. Particulièrement, la simulation informatique est un outil intéressant car ce mode de simulation n'impose aucune contrainte sur le modèle à créer, ce qui permet de modéliser le système considéré en choisissant et définissant une à une les influences auxquelles il est soumis.

3.3. Modélisation et simulation Orientée agent :

Ce type de modélisation est plus souvent abrégé par ABMS (Agent Based Models and Simulation). Ce modèle a été bien détaillé par Ferber, [Ferber, 1995] :

- Modéliser et simuler un phénomène naturel, économique, social, ou éco-socio-naturel.
- Créer un monde artificiel composé d'agents en interaction.
- Chaque agent est décrit comme une entité autonome.
- Le comportement des agents est la conséquence de leurs observations, de leurs tendances internes, de leurs représentations (éventuellement) et de leurs interactions avec l'environnement et les autres agents (communications, stimuli, action directe, etc...).
- Les agents agissent et modifient l'état de leur environnement par leurs actions.
- On observe les résultats de leurs interactions comme si l'on était dans un laboratoire (notion de laboratoire virtuel).
- Simuler est reproduire un phénomène afin de :
 - Tester des hypothèses permettant d'expliquer le phénomène (définition d'un modèle).
 - Prévoir l'évolution du phénomène.

D'après la définition de Ferber, le modèle Orienté agent est la projection (modélisation+ implémentation) de l'environnement.

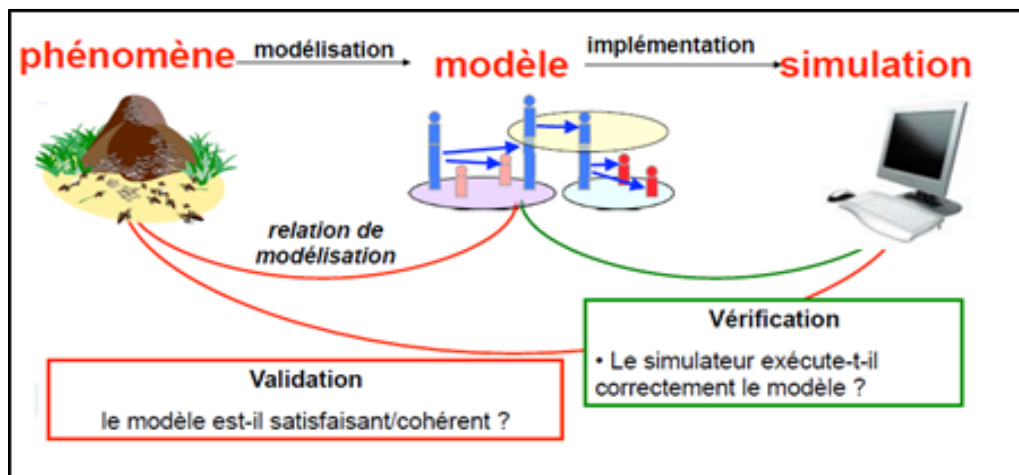


Figure 2 .10 : Schéma de la Modélisation.

La simulation consiste à faire évoluer les entités décrites dans le modèle suivant un temps virtuel. Ce temps virtuel n'est pas forcément explicité : par exemple, dans [Holcombe et al., 2006] « les agents interagissent entre eux au fil de leur exécution sans imposer de règles temporelles particulières ».

La description du processus de simulation n'est pas unique : selon les auteurs, ce processus peut comporter différentes étapes ([Drogoul et al., 2003, Ramat, 2006]). La différence se trouve en fait plus sur la granularité de la description que sur les opérations à réaliser en elles-mêmes. Ainsi, partageant la vision de [Ralambondrainy, 2009], nous considérons quatre étapes principales (voir figure 2.11) pour la simulation :

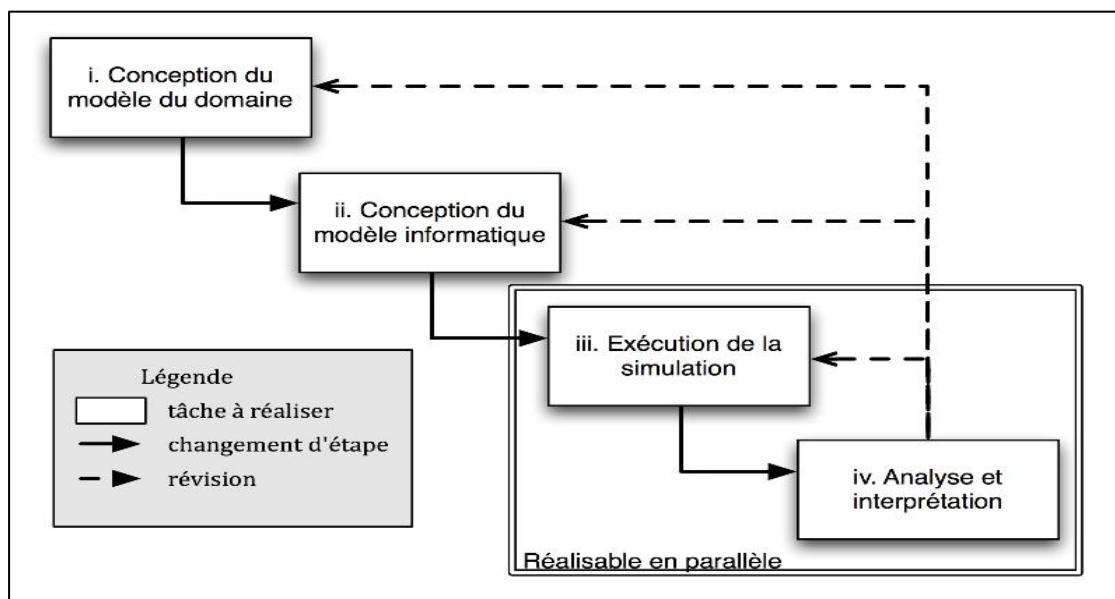


Figure 2.11 : Le processus de simulation [Ralambondrainy, 2009].

- Conception du modèle du domaine : délimitation de la partie du monde réel et modélisation de celui-ci sous la forme d'un modèle du domaine que l'on souhaite étudier.
- Conception du modèle informatique : transposition du modèle du domaine sous une forme exécutable. Les plates-formes de simulation proposent notamment une approche facilitant la réalisation de tels modèles.

- c. Exécution de la simulation : déploiement et exécution du modèle simulé. Les outils d'observation sont notamment configurés et sollicités pour extraire des données de la simulation.
- d. Analyse et interprétation des résultats produits par le modèle simulé : comparaison avec les données réelles et extraction de connaissances.

4. L'ABMS pour la résolution de problèmes complexes :

Les enjeux de ces simulations sont alors de comprendre les phénomènes émergents qui peuvent apparaître, analyser les interactions, ou mesurer l'impact d'une modification sur le système (introduction d'un nouvel élément).

Exemple et travaux :

- Une modélisation et simulation orientée agent de piéton : le problème de la sélection d'action. Un simulateur éducatif : jeu de déplacement dans un environnement urbain virtuel (Route empruntée en sécurité par le piéton-enfant confronté au trafic).
- Une modélisation et simulation orientée agent de pilotage : la conception de l'automobile fait face à des évolutions technologiques continues qu'il faut savoir intégrer et adapter.
- Modélisations et simulations multi-agents de systèmes complexes pour les Sciences de l'Homme et de la Société.

V. Conclusion :

Ce chapitre a été riche en définition et en concept. On a abordé tout d'abord la notion clé d'agent, ensuite on a souligné l'adaptation des systèmes multi-agents réactifs pour la résolution des problèmes complexes. Ces derniers ont été définis et on a fait ressortir l'importance de leurs caractéristiques « Auto-organisation et émergence ». La modélisation et la simulation orientée agent a été décrite et proposée enfin comme outil très adéquat pour la résolution des problèmes complexes. Dans le prochain chapitre, On abordera les automates cellulaires comme cas particulier de ces modèles et qui représente aussi l'objectif de notre travail.

Chapitre 3 :

Les automates cellulaires

I. Introduction :

Les automates cellulaires (AC) cités précédemment comme outils de modélisation et simulation, peuvent être considérés comme un moyen à générer une solution optimale d'un système complexe. Traités comme des produits des mathématiques discrètes, ils peuvent être représentés tout simplement comme une suite de matrices de nombres quand on veut les utiliser dans le but d'obtenir des effets plastiques, ils offrent un très grand nombre de représentations.

Dans ce chapitre, la notion d'automate cellulaire est abordée. Une définition rigoureuse sera donnée ultérieurement dans les sections suivantes avec une présentation des principes, des différentes architectures, de comportements, des caractéristiques et des exemples d'application. L'intérêt de ces structures sera souligné et on terminera avec une conclusion.

II. Définition :

Les automates cellulaires ont été introduits par Stanislaw Ulam (1909-1984) et John von Neumann (1903-1957) à la fin des années 40 au Los Alamos National Laboratory (Etats-Unis).

Un automate cellulaire peut être considéré comme un ensemble de composants élémentaires identiques appelées cellules et connectées les unes aux autres.

Les AC travaillent sur des grilles régulières ou chaque point, appelé cellule, possède une valeur dans un ensemble fini d'états à chaque itération toutes les cellules sont remises à jour simultanément grâce à une unique fonction locale dépendant des mêmes cellules voisines [Cori et al., 1993].

Ils permettaient d'idéaliser le processus de l'autoreproduction d'un système biologique [Montheillet, 1997]. A un instant donné chaque cellule se trouve dans un certain état, lequel varie au cours du temps de la façon suivante : les états de toutes les cellules sont modifiés en même temps à chaque top d'une horloge commune, en fonction de l'état de chacune de ses voisines qui lui sont connectées, et selon des règles prédéfinies.

« Automate cellulaire » est un terme mathématique qui fait référence à :

- Un réseau de cellules $\{i\}$ dans un espace à D dimensions
- Un ensemble d'états $\{s_i\}$ pour chaque cellule. On a habituellement le même nombre k d'états possibles pour toutes les cellules du réseau ; k est un nombre entier supérieur ou égal à
- Une règle F déterminant l'état d'une cellule à l'instant $t + 1$ en fonction de l'état de la même cellule et de son entourage à l'instant t :

$$S_i(t + 1) = F(\{S_j(t)\}) \text{ où } j \in v_i$$

Où v_i est un ensemble qui comprend la cellule elle-même et son entourage, encore désigné par voisinage tout au long de ce mémoire : le voisinage de la cellule compte très souvent parmi ses éléments la cellule elle-même [Kulakowski, 2000].

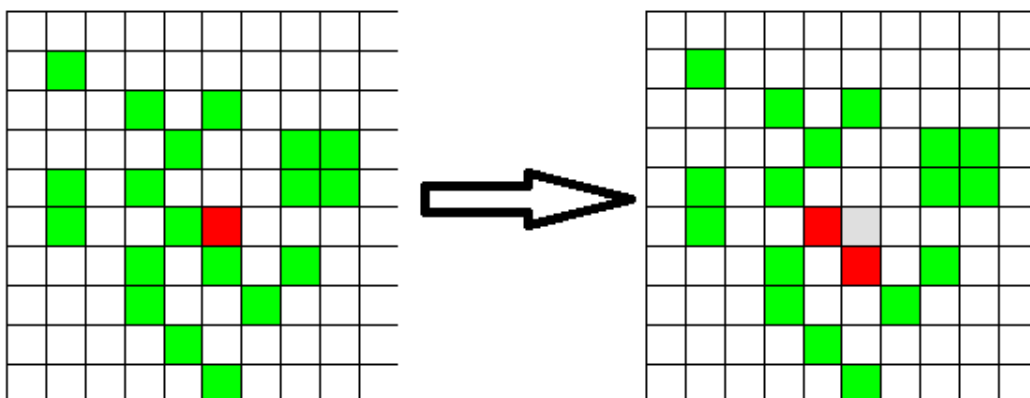


Figure 3.1 : Représentations d'automate cellulaire.

Finalement, un automate cellulaire complet sera défini par :

- sa topologie, son voisinage.
- son état initial, c'est à dire l'état initial de chaque cellule de l'automate.
- sa table de transition qui permet de calculer son état futur.

III. Principe de création d'un automate cellulaire :

D'une manière générale, le principe de création des AC consiste à:

- Diviser l'espace en un réseau de cellules, qui constituera un maillage à une dimension (1D), à deux dimensions (2D) ou à trois dimensions (3D) ;
- Définir une ou plusieurs variables d'état pour chaque cellules. Le plus souvent deux état sont défini pour une cellule : (vivante ou morte), (allumée, éteinte) ou encore (0 ou 1).
- Définir les règles de transitions (entre l'état initial t et l'état suivant $t+1$) pour une cellule (comportement) connaissant l'état de son voisinage (type de voisinage).

Les différents maillages offrent différentes architectures pour un AC défini ci-dessous.

IV. Les différentes architectures d'AC :

Il existe trois types d'architectures pour les AC :

1. Architecteurs à une dimension (1D) :

Placées sur une ligne ou un vecteur sous forme linéaire de dimension N (le nombre de cellules vaut N). Chaque cellule possède un voisinage de côté gauche et droit.

La rangée des cellules peut être aussi bien représentée par des points, des segments de droite, des rectangles, des hexagones, des triangles tel que représenté par la figure 3.2

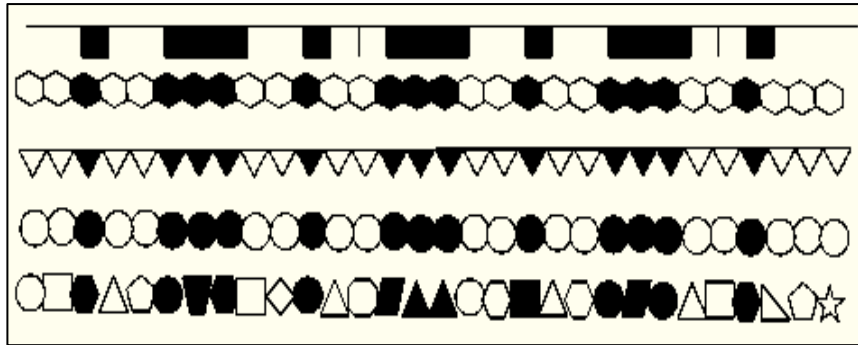


Figure 3.2 : Représentations des cellules de l'automate

Le choix de la forme du pavage ne change en rien l'évolution de l'automate, seul l'emplacement des cellules change et l'automate a un autre aspect.

Cette représentation peu intéressante a été rarement employée. Les automates unidimensionnels sont couramment représentés en deux dimensions.

2. Architectures à deux dimensions (2D) :

Placées sur une matrice ou grille de dimension $N \times N$ (le nombre de cellules vaut donc $le = N * N$). Chaque cellule possède un voisinage des côté gauche, droit, haut et en bas Figure 3.3.

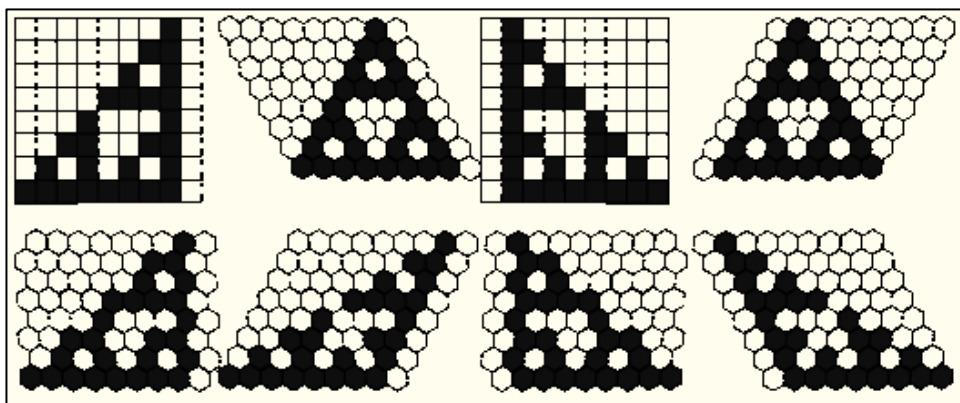


Figure 3.3 : Représentations des cellules d'automate bidimensionnel. [Kramm, 2005]

La grille 2D employée pour suivre le parcours d'un automate cellulaire 1D dans le temps est lue du haut vers le bas, les automates 2D peuvent aussi être représentés sur une grille 3D.

3. Architectures à trois dimensions (3D) :

Les automates en 3D sont d'habitude représentés sur un système cubique. Cependant il existe d'autres pavages de l'espace où la cellule aura un nombre stable de voisins.

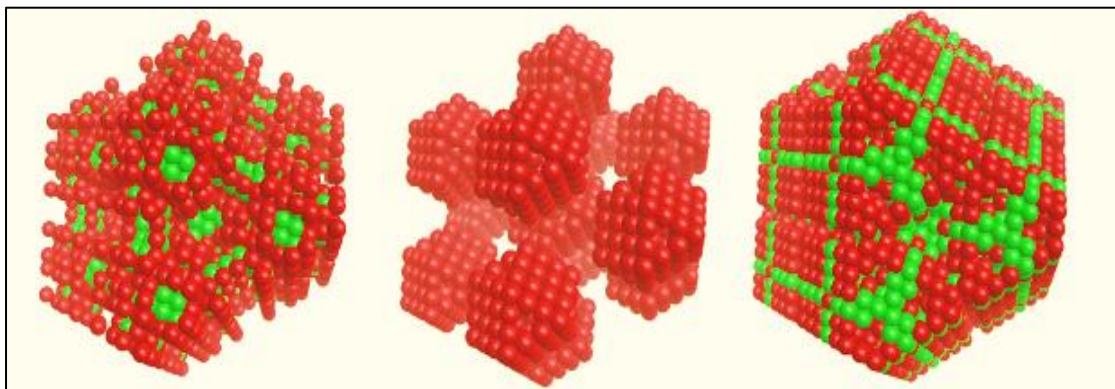


Figure 3.4 : Représentations des cellules d'automate tridimensionnel.

Les formes tridimensionnelles obtenues sont impressionnantes mais on se retrouve en face d'une grande difficulté pour montrer toutes les cellules.

Le choix de la grille n'affecte pas le comportement de l'automate, plusieurs type de voisinages peuvent aussi caractérisés un automate, les plus utilisés sont décrit dans le paragraphe suivant.

V. Voisinage :

Quel que soit la cellule, toute cellule possède un voisinage (un ensemble de taille finie de cellules) et le changement de la forme de cellule mène à un changement dans nombre de voisinage.

Pour un automate carré on a huit voisins, et pour une forme hexagonale on a six voisins etc., comme le montre ce schéma

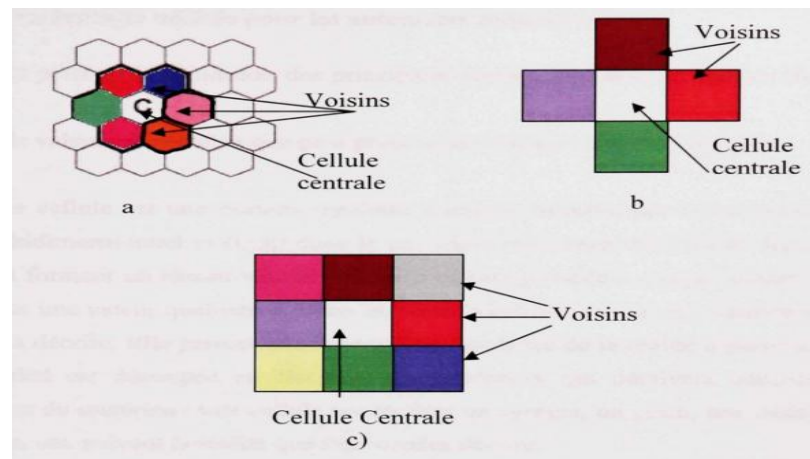


Figure 3.5 Représentation des différents types de voisinages; a) type de réseau hexagonal, b) environnement de Von Neumann, c) environnement de Moore.

Les voisinages les plus courants sont: Von Neumann (voisins Nord/Sud/Est/Ouest.) chaque automate a cinq entrées: lui-même plus ses quatre voisins (Figure 3.5-b), mais chez le voisinage de Moore, il en a neuf: lui-même plus ses huit plus proches voisins (Figure 3.5-c) (on ajoute les diagonales: cas du Jeu de la vie).qui sera présenté dans ce qui suit.

VI. Caractéristiques :

Un automate cellulaire (AC) peut-être caractérisé par les quatre points suivants :

1. Sa dimension :

Le plus généralement 1D, 2D dimensions ou 3D, il n'y a pas de limite à la dimension d'un automate, si ce n'est la puissance de calcul des machines sensées le reproduire.

2. Le voisinage d'une cellule :

Celui-ci définit l'ensemble des cellules qui auront une influence sur la cellule étudiée. En pratique, le voisinage est souvent limité à la cellule cible et aux cellules adjacentes.

3. Son espace d'états :

Cet espace correspond à l'ensemble des états que peut prendre une cellule. Le plus souvent limité à 2, il n'y a aucune limite théorique. Pour exemple, Von Neumann a étudié mathématiquement un automate à 29 états. Pratiquement, ces états sont représentés par des couleurs, qui permettent de suivre les évolutions de l'automate.

Lors de modélisation de systèmes, les états des cellules correspondent à des états physiques locaux. Par exemple, dans le Jeu de la Vie, une cellule est soit "morte" soit "vivante", mais on pourrait très bien imaginer des états transitoires de dégénérescence d'une cellule, en augmentant le nombre d'états de l'automate.

4. Sa fonction de transition :

C'est l'ensemble des règles qui permettent de déterminer le nouvel état d'une cellule en fonction des règles du son environnement. Pour un automate à n états et avec un voisinage de k cellules il peut y avoir nk configurations de voisinage différentes :

pour $n=2$ et $k=3$, $nk=8$ voisinages différents (AC élémentaire de wolfram) ;

pour $n=2$ et $k=9$, $nk=512$ voisinages différents (jeu de la vie).

VII. Comportement :

Un automate cellulaire est un procédé de simulation informatique fondé sur un découpage de l'espace en cases appelées « cellules ». On fait évoluer dans le temps le contenu des cases avec pour convention que la transformation d'une case ne dépend que de ses cases voisines. La dynamique du phénomène est régie par des influences locales et peuvent être simulés ainsi.

Dans la suite, on va illustrer un des exemples les plus connus « le jeu de la vie », automate cellulaire imaginé par John Horton Conway en 1970, pour mieux comprendre le comportement d'un automate cellulaire :

Exemple jeu de la vie :

À l'origine, le Jeu de la vie fut présenté comme un jeu mathématique. A l'instar des espaces cellulaires d'Ulam, le Jeu de la vie se présente sous la forme d'une grille constituée de cellules.

Entre deux instants t et $t+1$, une case est allumée, ou éteinte, en fonction de son état (allumée/éteinte), (Mort/vivante) et de ceux de ses huit plus proches voisins.

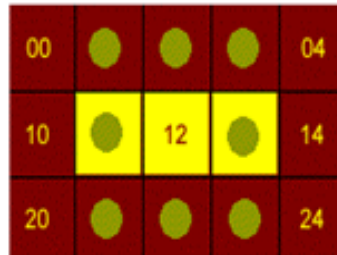


Figure 3.6 : Etat initial d'automate

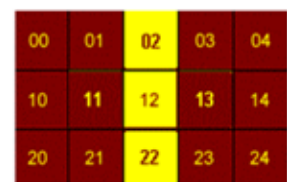
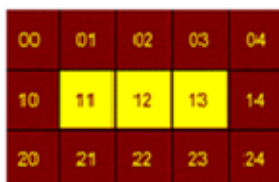
L'univers est limité ici à un rectangle de 5 par 3. Les cases sont numéroté de 0 à 4 en horizontal et de 0 à 2 en vertical. Les cellules claires sont actives. Dans le Jeu de la vie, est considérée comme voisine toute cellule contiguë, y compris les diagonales.

La figure ci-dessus montre le voisinage de la cellule 12. En l'occurrence, sur les huit voisins, deux sont actifs, les règles du Jeu de la vie sont simples : Figure 3.7.a

- Une cellule inactive entourée de 3 cellules actives devient active (« naît ») ;
- Une cellule active entourée de 2 ou 3 cellules actives reste active ;
- Dans tous les autres cas, la cellule « meurt » ou reste inactive.

À la génération suivante, seules les cellules 02, 12 et 22 seront donc actives. Figure 3.7.b.

Vu le comportement d'un AC, on peut faire ressortir les principales caractéristiques. Figure 3.7.b.



a : détermination du voisinage.

b : valeur de voisinage.

C : seconde génération.

Figure 3.7 : jeu de la vie

VIII. Application d'automate cellulaire :

Les automates cellulaires peuvent être utilisés pour simuler des formes de vie simplifiées [Gardner, 1970 ; Gardner, 1971]. Ils sont notamment utilisés pour prévoir l'évolution des feux de forêt [Drossel et al., 1992] ou pour générer des nombres aléatoires [Wolfram, 1986].

En biologie, les AC servent à étudier l'organisation d'éléments biologiques entre eux [Gardner et al., 1970], en informatique, les AC sont un outil de modélisation très important pour les architectes massivement parallèles ainsi que pour beaucoup de recherches théorique. Ils ont été utilisés pour le traitement d'image par [Sansonet, 2004] pour le lissage et par [Djemame, 2009] pour la segmentation contour et aussi dans le traitement du signal par [Emaszko, 2004].

En physique, ils sont utilisés, pour l'étude de la croissance de cristaux, ou l'apparition de structures d'interfaces [Victor, 1984]. Ils sont également utilisés comme alternative aux équations différentielles de l'hydrodynamique, car leur implémentation est plus aisée. [Maxwell, 1890] avait déjà évoqué l'emploi de tels modèles discrets pour étudier les gaz et les écoulements de fluide. Des modèles d'automates cellulaires permettent effectivement de retrouver des comportements prédits par les équations de Navier-Stokes [Chate et al., 1987 ; Frisch et al., 1986 ; Clavin et al., 1986 ; Toffoli, 1984].

De même, en sciences sociales, la représentation de populations par un grand nombre d'individus répartis dans l'espace, dont le comportement est codé par une valeur et régi par l'interaction avec les individus proches, permet d'avoir recours aux automates cellulaires [Louis, 2002].

IX. Conclusion :

Dans ce chapitre on a exposé avec beaucoup de détails l'outil choisi pour la modélisation et simulation orientée agent : « les automates cellulaires ». Vu les résultats obtenus dans les différents domaines et leur facilité de manipulation, on emploiera ce dernier dans la résolution d'un problème complexe dans le domaine du traitement d'image.

Chapitre 4 :

Conception et Implémentation

I. Introduction :

Dans ce chapitre, on présentera en premier temps le problème choisi, qui est la « Segmentation d'image ». On soulignera l'importance et la complexité de ce problème, et on citera quelques méthodes dans le but de valoriser notre application. Dans un deuxième temps on présentera en détail les phases de conception et de réalisation, on terminera par la discussion des résultats obtenus.

II. Description et objectifs de l'application :

II.1. Choix de l'application :

Parmi les challenges du traitement d'images, celui de la segmentation qui est l'opération la plus importante dans un système de traitement des images, car elle est située à l'articulation entre le traitement et l'analyse.

Dans ce travail, la résolution de ce problème complexe s'accomplisse par un modèle mathématique puissant qui est les automates cellulaires, un domaine d'application qui favorise la mise en valeur de cette technique.

Les raisons pour lesquelles on a opté pour ce choix sont :

- Le problème de segmentation est par sa nature complexe, cela est dû à la complexité de l'image elle-même (taille, ...).
- Le problème de non satisfaction des résultats de segmentation d'image, qui est encore un problème ouvert malgré le nombre important d'algorithmes. Ce qui va nous permettre d'appliquer un nouvel algorithme de segmentation d'image qui se base sur les AC.

- Le besoin en applications de traitement d'images se fait de plus en plus pressant à mesure que l'image numérique s'impose comme un support et une source d'information privilégiée.

II.2. Présentation du problème « Segmentation d'image contour et région » :

La segmentation d'image est un problème important dans les domaines de l'analyse d'image. Elle est par exemple utilisée en imagerie médicale afin d'analyser et de quantifier les différentes structures anatomiques présentes dans les images.

Les méthodes de segmentation existantes peuvent être classées selon le but à atteindre. Par exemple, il existe des méthodes qui déterminent des régions dans l'image alors que d'autres au contraire cherchent à déterminer les frontières des régions. Il est aussi possible de classer les méthodes selon l'approche utilisée pour résoudre le problème de la segmentation. On trouvera par exemple des méthodes basées sur des connaissances explicites de l'image, d'autres basées sur un ensemble de probabilités.

L'objectif de la segmentation est donc de fournir une description des objets contenus dans l'image en extrayant diverses indications visuelles telles que le contour des objets, et/ou l'homogénéité des régions.

Jusqu'à présent, il n'existe pas de méthodes universelles pour la segmentation d'images. Toute technique n'est efficace que pour un type d'image donné, pour un type d'application donnée, et dans un contexte donné. En raison de ces contraintes, de diverses stratégies de segmentation ont été proposées, affirme leurs incapacités et leurs limites. Il est donc nécessaire de chercher de nouveaux horizons pour trouver de nouvelles méthodes plus souples et plus efficaces.

II.3. Objectifs de l'application :

La méthode qu'on propose dans ce travail est d'un genre nouveau. Elle repose en effet sur un comportement d'un système multi-agents qui émerge vers une segmentation de l'image.

On s'intéresse à la segmentation frontalière et régionale. Le problème est de détecter donc les contours et les régions dans une image en utilisant les automates cellulaires. Dans ce contexte notre application vise les buts suivants :

- Remplacer les méthodes traditionnelles de segmentation d'image qui utilisent une approche séquentielle consistant à segmenter l'image pixel après pixel, par une approche qui consiste à utiliser les systèmes multi-agents (traitement parallèle).
- Représenter ce système par un réseau d'automate cellulaire. Chaque cellule représente un pixel, qui est capable de s'auto-organiser avec les autres pour faire émerger un comportement global complexe.
- Programmer ces AC pour émerger une segmentation idéale de l'image.
- Essayer de mettre en valeur les performances de l'algorithme en termes de qualité de la détection, souplesse de segmentation, et aussi temps d'exécution par rapport aux approches séquentielles.
- Coopérer ces deux techniques pour avoir une segmentation (contours - région). La segmentation par coopération régions-contours suscite un grand intérêt ces dernières années. Elle consiste en une coopération entre la segmentation par régions et la segmentation par contours. Elle exploite les avantages de ces deux types de segmentation pour aboutir à un résultat de segmentation plus précis et plus fidèle que celui obtenu à l'aide d'une seule technique.

II.4. Travaux dans le domaine :

Malgré l'efficacité et la rapidité d'exécution des automates cellulaires pour la segmentation des images, peu de travaux en été trouvé dans la littérature, et concernant seulement la segmentation contour.

Parmi les travaux, on retrouve ceux de [Popovici et al., 2002], [Slatnia, 2007], [Wongthanavas, 2007], [Melkemi, 2010] et [Djemame et al., 2011].

Le modèle proposé par A.Popovici [Popovici et al., 2002], utilise une règle dynamique pour l'automate cellulaire. Cette règle doit être telle que à partir d'une

configuration initiale (image), il est nécessaire que l'automate atteigne une configuration finale où les seules cellules actives correspondent à la frontière.

Les travaux de S. Slatnia [Slatnia, 2007] sont orientés vers l'hybridation d'un automate cellulaire et un algorithme génétique pour la segmentation d'image Contour. L'algorithme génétique a été utilisé pour la sélection des meilleures règles de comportement et réduisant ainsi leur nombre.

S.Wongthanavasou [Wongthanavasou, 2007] propose une méthode de segmentation contour par automate cellulaire sur des images binaires et niveau de gris. Il utilise aussi des traitements préalables avant d'appliquer son algorithme (élimination de bruit).

K.Melkemi [Melkemi, 2010] propose un projet de fin d'étude, où il s'intéresse à la segmentation d'image basée sur l'utilisation des automates cellulaires. Mais aucune publication n'a été trouvée sur ce travail.

S.Djemame [Djemame et al., 2011] a employé un type d'automate cellulaire dit « continu » : au lieu d'employer un nombre fini des états des cellules, elle a utilisé une valeur continue (nombre $\in [0,1]$). Et au lieu d'utiliser un nombre limité de règles, elle a employé une fonction qui prend comme paramètres les valeurs (états) des cellules voisines. Plusieurs fonctions de traitement ont été rajoutées pour obtenir un bon résultat de segmentation (lissage,...).

III. Conception :

III. 1. Modélisation du problème :

III.1.1. Modélisation de l'Environnement :

On considère un ensemble de N pixels $\{P_1 \dots P_n\}$ qui constituent l'image. L'image en niveaux de gris ou couleur représentant l'environnement sur lequel les automates vont évoluer. Cet ensemble qu'on désire regrouper en des classes aussi homogènes que possible. L'environnement de nos agents possède un ensemble des propriétés. Ce dernier est composé de cellules qui représentent les pixels de l'image.

III.1.2. Modélisation de l'agent « Automate Cellulaire » :

Dans ce travail, on considère pour chaque pixel un automate cellulaire. Ce dernier réagit à son environnement. Son comportement est en relation avec ses voisins.

III.2. Comportement des agents « Automate cellulaire de segmentation » :

Le modèle comportemental des automates cellulaires est prédéfini par certaines conditions relatives à l'environnement dans lequel ils évoluent. Ces comportements sont issus de leur stimulus local. Pour chaque agent, ce stimulus est estimé en évaluant les caractéristiques d'intensité en niveau de gris ou couleur de tous les pixels se trouvant dans le voisinage.

III.3. Algorithme principal :

III 3.1. Principe :

Pour appliquer le principe des automates cellulaires pour la détection des contours ou régions d'une image, l'idée est de calculer les intensités entre le pixel et ces voisins. Dans notre cas, on a opté après plusieurs tests, pour le voisinage de Moore (figure 4.1).

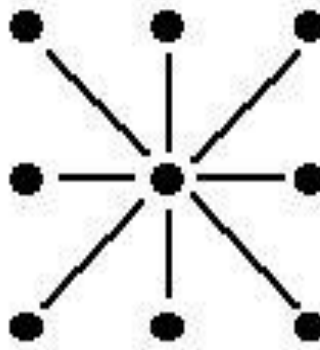


Figure 4.1 : Voisinage de Moore

L'affectation du pixel vers une région ou contour, s'appuie sur les calculs de la différence d'intensité entre le pixel et le pixel voisin.

Si, entre deux pixels consécutifs, les niveaux d'intensité sont très différents, l'automate va changer son état.

L'algorithme haut niveau est le suivant :

debut

Lancement de la segmentation

Pour chaque automate

Déterminer les voisins.

Calculer les paramètres de la segmentation (région ou contour).

Changer d'état

Fin Pour

Fin de segmentation

Afficher les cartes de segmentation sur l'environnement d'exécution

Fin.

III.3.2 Architecture de l'application :

L'organigramme relatif à notre système est comme suit:

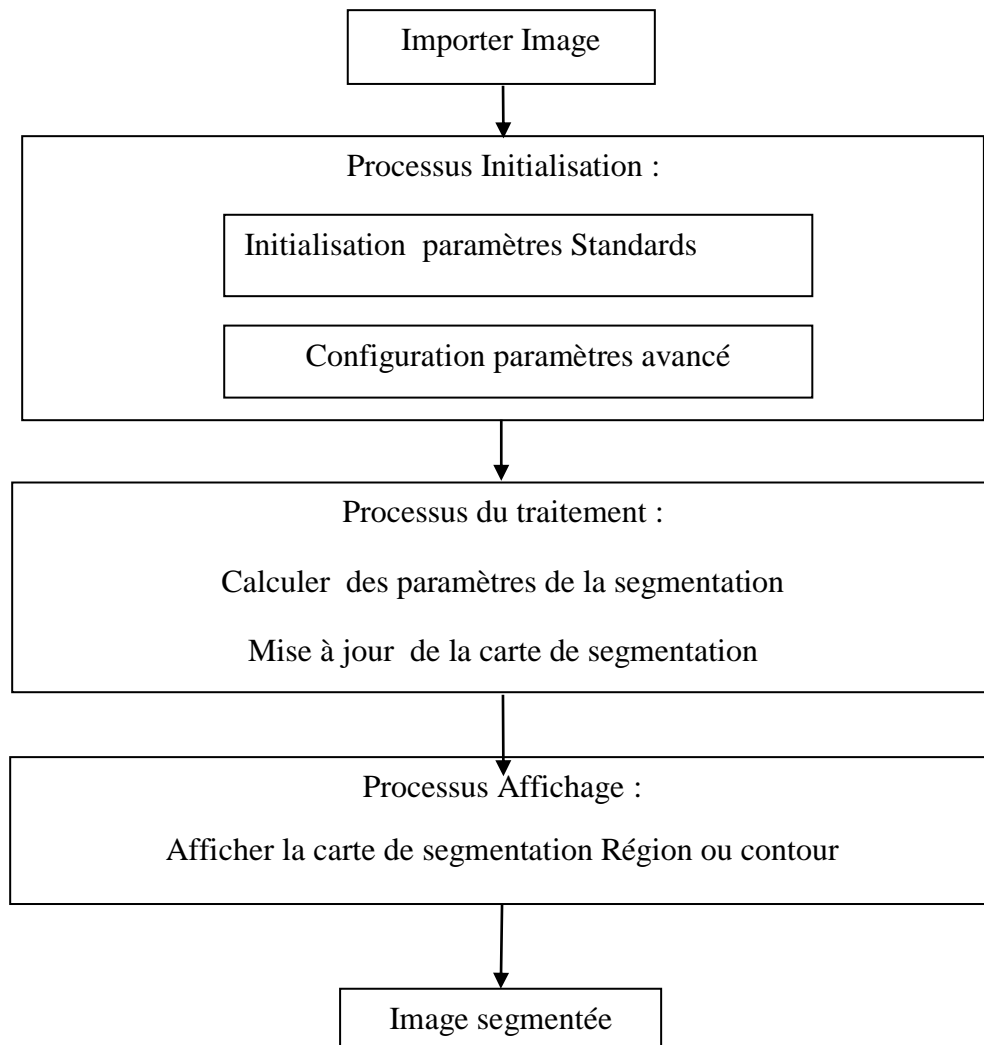


Figure 4.2 : Organigramme de l'algorithme proposé.

IV. Réalisation :

Dans cette partie on présente les différentes étapes de réalisation de notre projet qui a pour objectif d'implémenter l'algorithme proposé de segmentation contour et région à base des automates cellulaires. On présente d'abord l'environnement de programmation orientée agent puis on décrit la plateforme NetLogo utilisée. Ensuite on expose les différentes étapes de simulation et on présente la discussion des résultats obtenus.

En fait, une étape importante est effectuée au préalable et qui est la création de la base d'images :

Par manque et non accessibilité de disponibilité d'une bibliothèque d'image standard, on a créé notre propre bibliothèque à travers une collection d'images les plus utilisées dans ce domaine.

La tâche de collection des images nous a pris un temps considérable car d'une part la sélection des images ne se fait pas d'une manière aléatoire mais à travers un ensemble de critères (contours, contrastes, domaines...), d'autre part ce n'est pas facile de trouver ce genre d'images, pour cette raison une longue recherche a été faite pour résoudre ce problème.

Ces images ont été choisies avec soin pour toucher une large variété d'image et montrer la robustesse de notre travail. Trois types d'images, avec différentes extensions ont été introduites et regroupées en trois catégories : images médicaux (image radio), images niveau de gris, et images couleur.

IV.1. Environnement de programmation :

IV.1.1. Programmation orientée agents :

La programmation orientée agents a été proposée par Yoav Shoham en 1993 comme un nouveau paradigme de programmation, que l'on peut voir comme une spécialisation de la programmation orientée objets.

Dans cette approche, les agents sont les éléments centraux du langage, de la même façon que les objets sont centraux pour les langages orientés objets. De plus, à chaque agent est associé un ensemble d'habiletés qui représentent ce que l'agent sait faire. En même temps, la programmation orientée agents suppose qu'on va développer des programmes dans lesquels plusieurs agents interagissent, ce qui met l'accent sur la dimension sociale des agents.

Parmi les nombreux logiciels orientés agents se distingue notamment la plateforme « NetLogo ». Cette dernière constitue un puissant outil logiciel et permet à des milliers d'agents de fonctionner en parallèle. Ce qui offre la possibilité d'explorer la connexion entre le comportement de l'entité autonome, avec le comportement qui peut émerger de l'interaction entre ces entités.

IV.1.2 Plate-forme d'implémentation (NetLogo) :

NetLogo est un environnement de modélisation programmable pour simuler des phénomènes naturels et sociaux. Il a été rédigé par Uri Wilensky en 1999.

NetLogo est particulièrement bien adapté pour la modélisation des systèmes complexes de développement au fil du temps. Les modélisateurs peuvent donner des instructions à des centaines ou des milliers d'agents tout fonctionnant de manière indépendante. Cela rend la possibilité d'explorer le lien entre le comportement du micro-niveau des individus et du niveau macro-tendances qui se dégagent de l'interaction de plusieurs individus.

NetLogo permet des simulations ouvertes et «jouer» avec eux, en explorant leur comportement sous diverses conditions. Il est également un environnement de création qui permet aux étudiants, aux enseignants et concepteurs de programmes à créer leurs propres modèles. NetLogo est assez simple que les étudiants et les enseignants peuvent facilement effectuer des simulations ou même construire leurs propres. Et, il est assez avancé pour servir d'un outil puissant pour les chercheurs dans de nombreux domaines.

NetLogo fonctionne sur la machine virtuelle Java, donc il travaille sur toutes les principales plateformes (Mac, Windows, Linux, et al.). L'approche proposée a été implémentée sur la plate-forme multi-agents NetLogo, afin de bénéficier des possibilités offertes par ce genre d'outil logiciel. La Figure 4.4 présente l'interface standard de cette plate-forme :

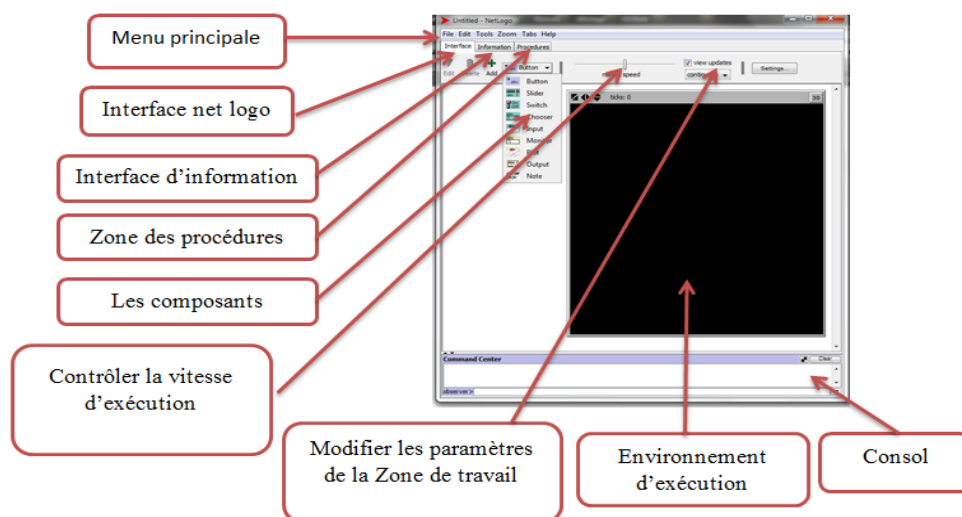


Figure 4.3 : interface Net logo

IV. Présentation de l'application :

L'interface graphique principale de cette application englobe un ensemble d'options à partir desquelles on peut accéder à notre système, la figure ci-dessous montre ses différentes options.

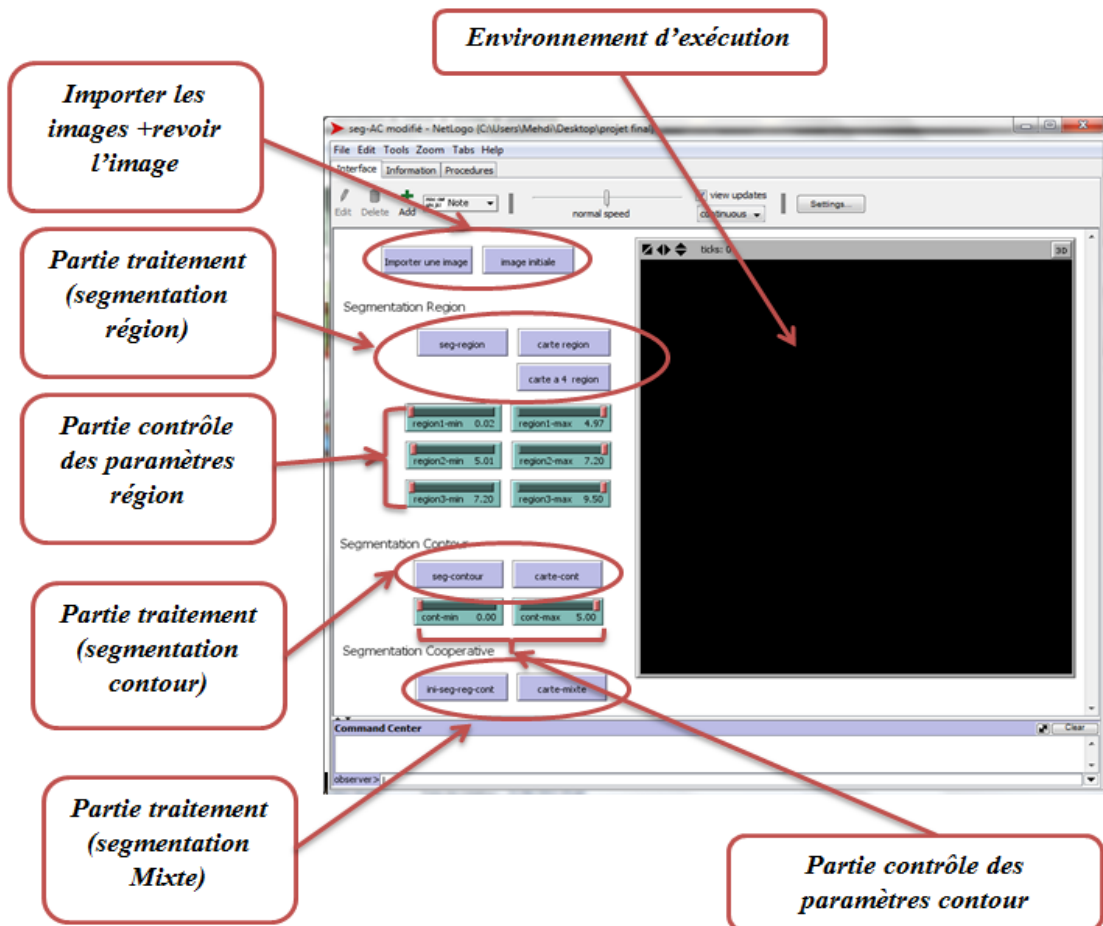


Figure 4.4 : interface graphique de l'application

Cette interface a été organisée en 6 parties de manière à offrir une bonne navigation à l'utilisateur.

- 1) **Importer les images et revoir l'image initiale** : Deux boutons, un pour l'importation d'une image, et l'autre pour visualisé l'image initial après la segmentation.

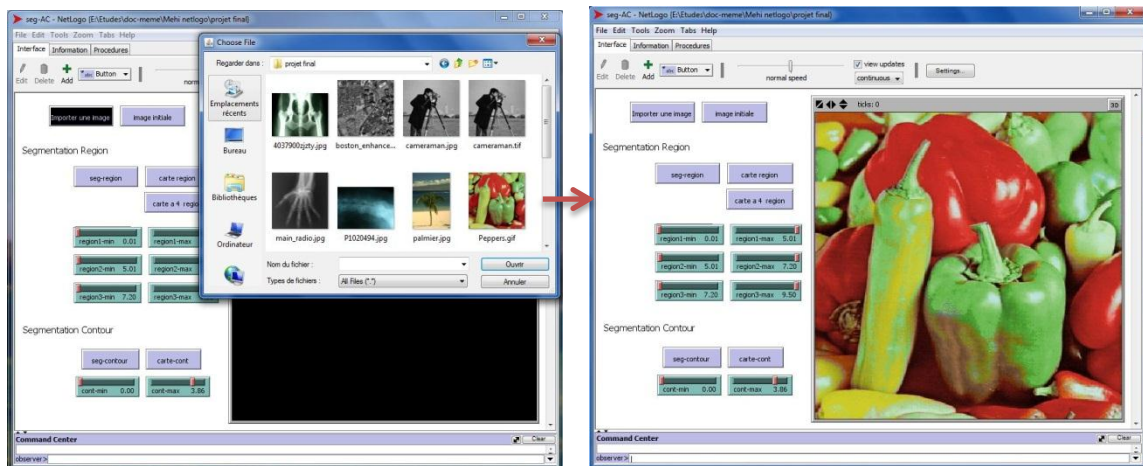


Figure 4.5 : Importation d'image

- 2) **Partie contrôle des paramètres** : ces curseurs permettent de modifier les paramètres par défaut, afin d'ajuster la segmentation de l'image ou la visualisation d'une région.
- 3) **Partie paramètre traitement (segmentation région)** : le bouton **seg-region** pour lancer la segmentation de région. **Carte-région**, pour la visualisation de l'image segmentée (Figure 4.7 : a). **Carte a 4 régions** (Figure 4.7 : b) pour la visualisation de l'image segmentée a 4 régions.

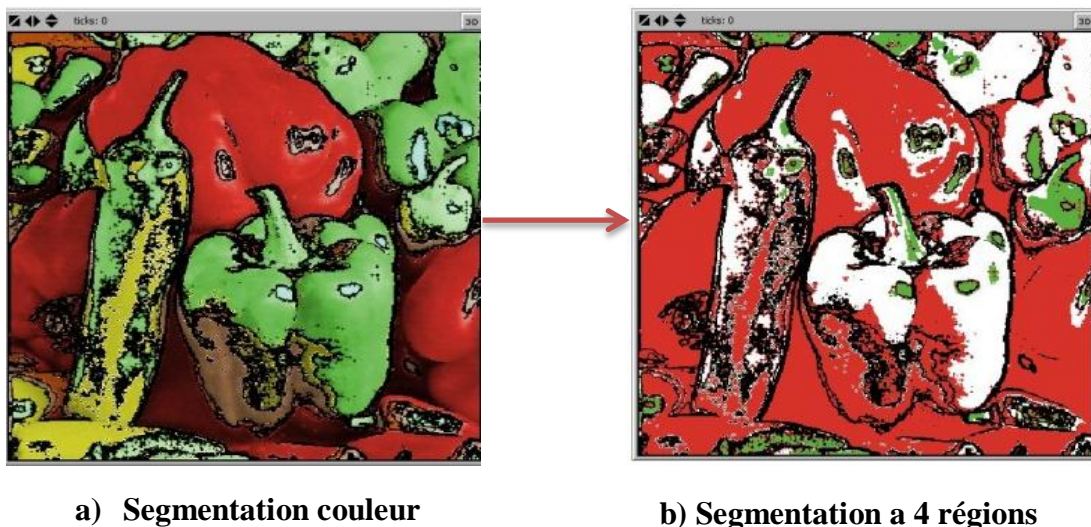


Figure 4.6 : Segmentation Région

- Pour une segmentation à quatre couleurs, les pixels de l'image sont répartis sur les quatre régions, selon l'espace de couleur (Figure 4.7) ci-dessous.

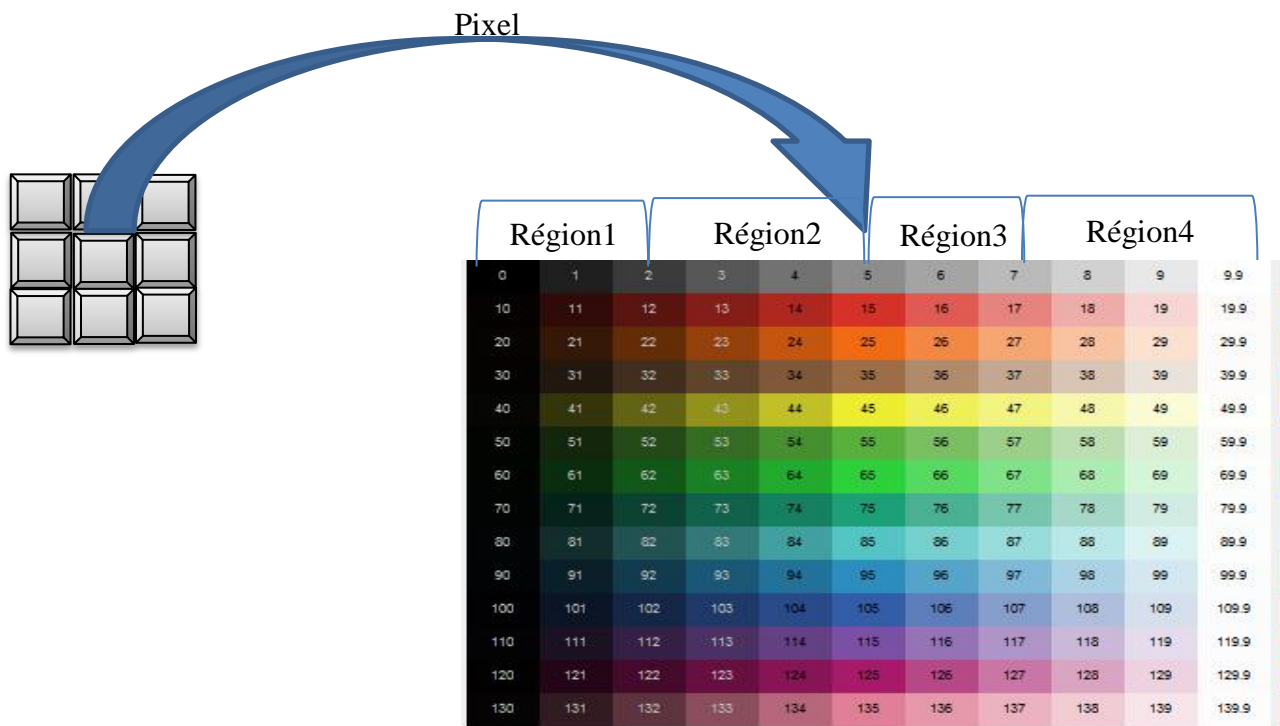


Figure 4.7 : Réaction du pixel dans l'espace de couleur Net Logo

- Pour une segmentation région plus avancée, la répartition des « pixels » se fait de la même manière mais avec les couleurs d'origine de l'image et d'une manière automatique.
- 4) Partie contrôle des paramètres de la Segmentation contour :** ces curseurs permettent de modifier les paramètres de la simulation, afin d'ajuster la segmentation de l'image et la visualisation des contours.
 - 5) Partie paramètre traitement (segmentation contour) :** le bouton seg-contour pour lancer la segmentation de contour. Carte-Cont, pour la visualisation des contours détectés sur l'image (Figure 4.8).

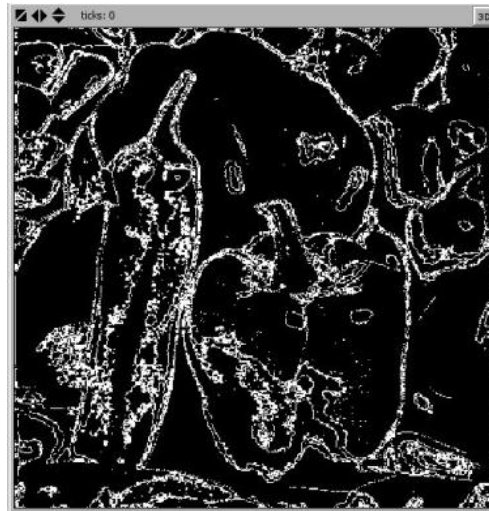


Figure 4.8 : Segmentation Contour

Nous donnons une illustration de la segmentation contour en 3D présenter dans la figure 4.9 ci-dessous

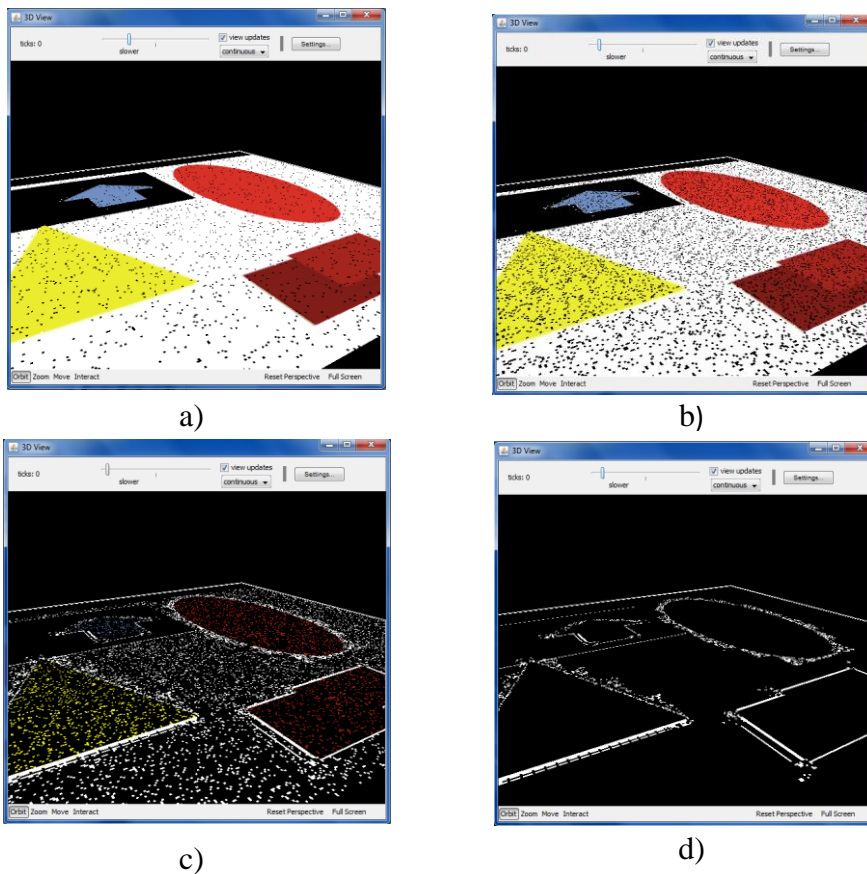


Figure 4.9 : Procédé de la Segmentation Contour (Visualisation 3D)

Pendant le procédé de la segmentation, on peut voir le passage du niveau micro, représenté par l'évolution des agents vers le niveau macro représenté par la carte segmentée (Figure 4.9).

- 6) Segmentation coopérative : permet d'afficher une carte de segmentation mixte en utilisant les résultats des deux segmentations (contour – région)

V. Résultats expérimentaux :

V.1. Discussion :

Afin d'estimer l'efficacité et les performances de notre algorithme, on l'a appliqué sur un ensemble d'images. L'environnement matériel d'exécution est un Micro-ordinateur, doté d'un processeur 2Duo 2.4Ghz et 4 Go de RAM.

A travers un ensemble de jeux d'essais, on a constaté la grande vitesse d'exécution de notre algorithme et son efficacité. Il travaille en temps réel et donne des bons résultats illustrés dans la figure 4.10.

1.1 Segmentation contour :

Afin de évaluée les résultats obtenus, on présente dans cette section une comparaison entre les résultats obtenus par notre système et ceux obtenus en appliquant un filtre linéaire de Sobel. Ce filtre linéaire utilise une opération mathématique appelée convolution pour transformer les points de contour de l'image d'entrée en blanche et le reste en noire. Le filtre de Sobel est le plus exploité dans les applications industrielles pour ces résultats dits satisfaisantes.

La figure 4.10 présente quelques résultats obtenus et comparés avec ce filtre.

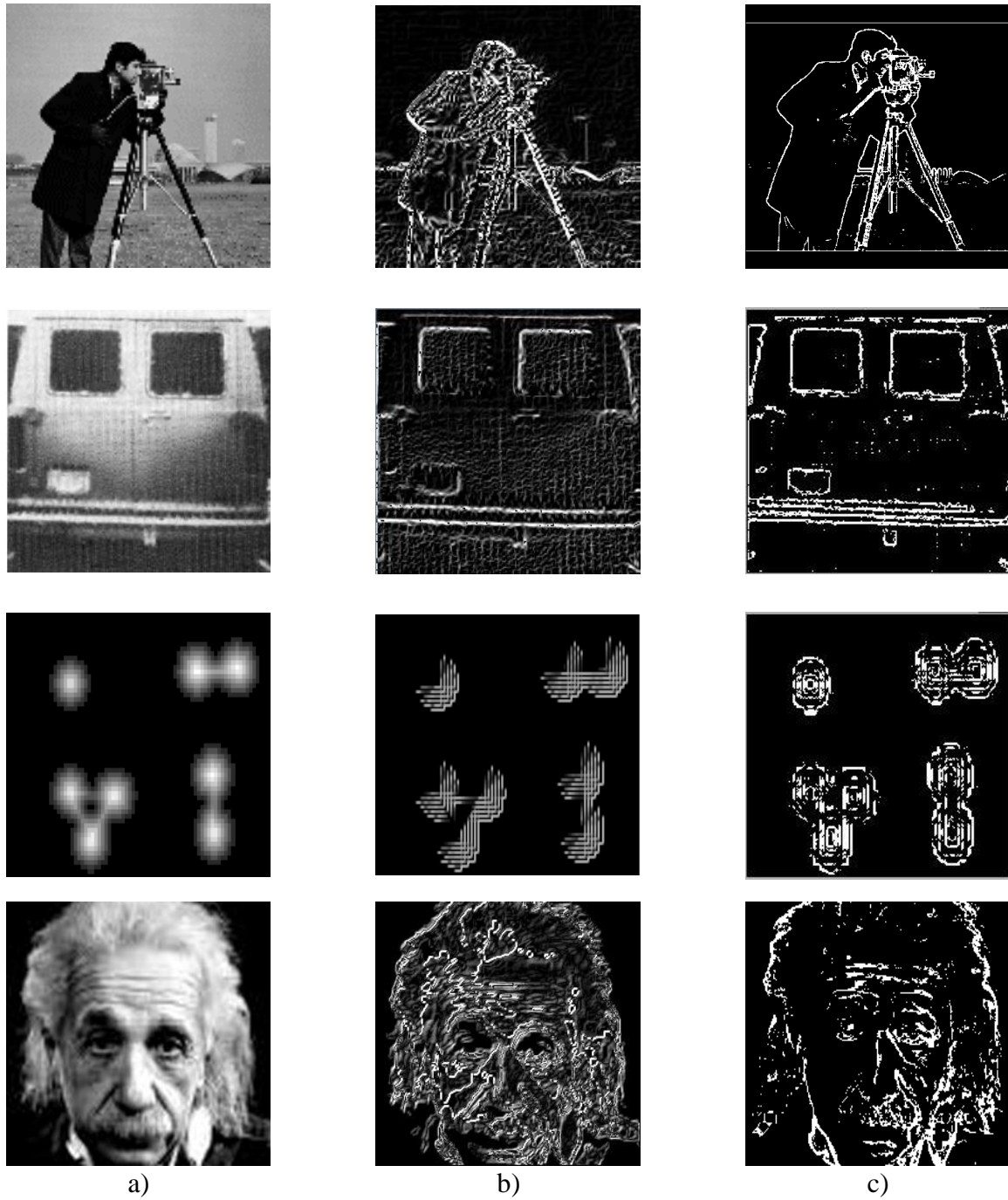


Figure 4.10 : (a) Images originaux, (b) Application du filtre Sobel,
(c) Application de l’algorithme proposé

D'après ces résultats, on remarque très bien que les images résultantes, tout en appliquant le filtre de Sobel contiennent des contours non précise par rapport à ceux obtenus par notre système. Le filtre de Sobel pose le problème de sur-segmentation lorsqu'il s'agit de traiter une image complexe comparativement à notre système qui permet l'extraction des contours les plus adéquats.

1.2 Segmentation région :

Cette deuxième partie, est consacrée essentiellement à comparer nos résultats avec les résultats obtenu en appliquant les trois méthodes de segmentation région : croissance de régions, division-fusion et lignes de partages des eaux (LPE) sur des images de type niveau de gris et couleur.

- Segmentation par croissance de régions : Ce type de segmentation consiste à faire croître des régions en y ajoutant successivement les pixels adjacents qui satisfont un critère d'homogénéité. La croissance s'arrête lorsque tous les pixels ont été traités.
- Segmentation par division fusion de régions : L'approche segmentation par division de régions consiste à diviser l'image originale en régions homogènes au sens d'un critère donné. Après la phase de fusion de régions fondée sur l'analyse d'un graphe d'adjacence.
- Segmentation par LPE : L'idée de cette technique consiste à utiliser des marqueurs, considérés comme des minima locaux, qui ont pour rôle d'imposer la présence de certaines régions seulement, de manière à contraindre la topologie de la LPE.

La figure 4.11 présente les résultats de la segmentation par ces méthodes sur différentes images comparés avec le résultat de l'algorithme proposé.

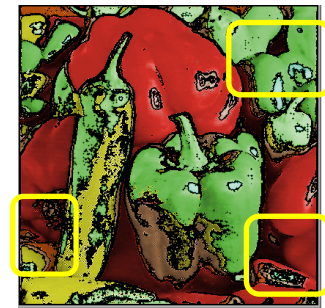
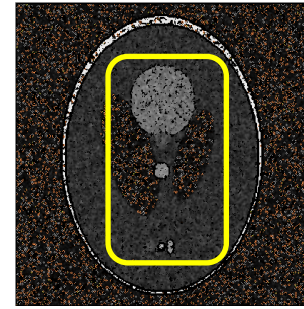


Image original

Croissance de région

Algorithme proposé



Image original

division fusion de régions

Algorithme proposé

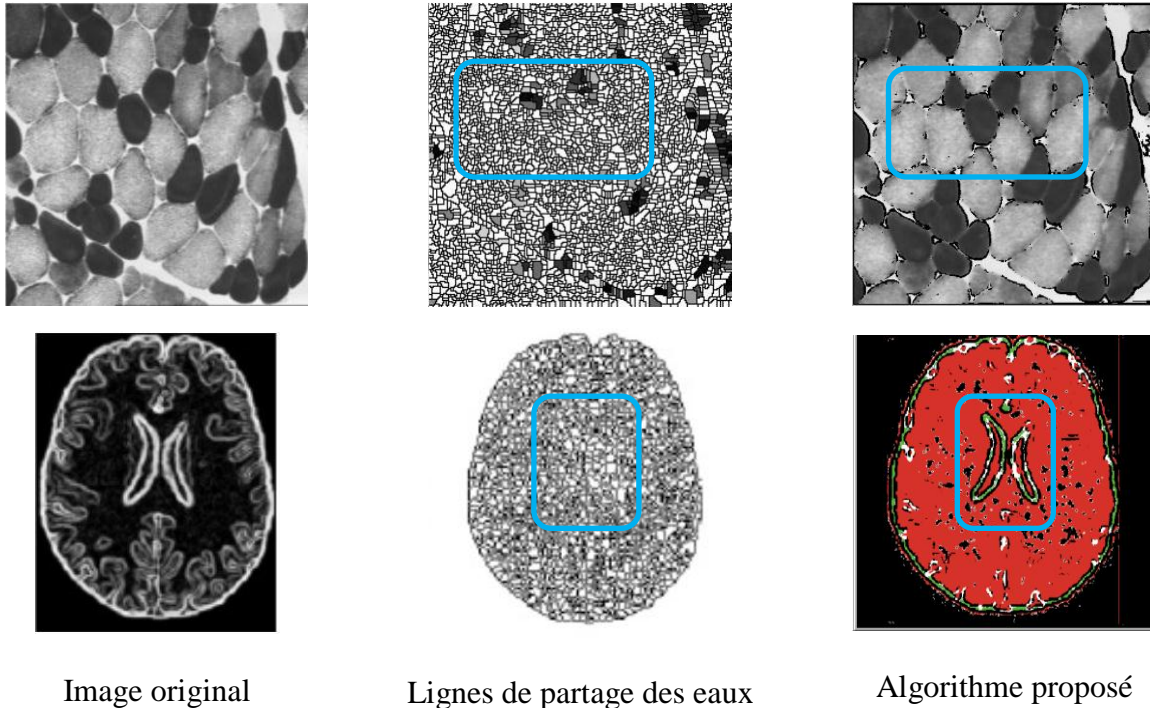


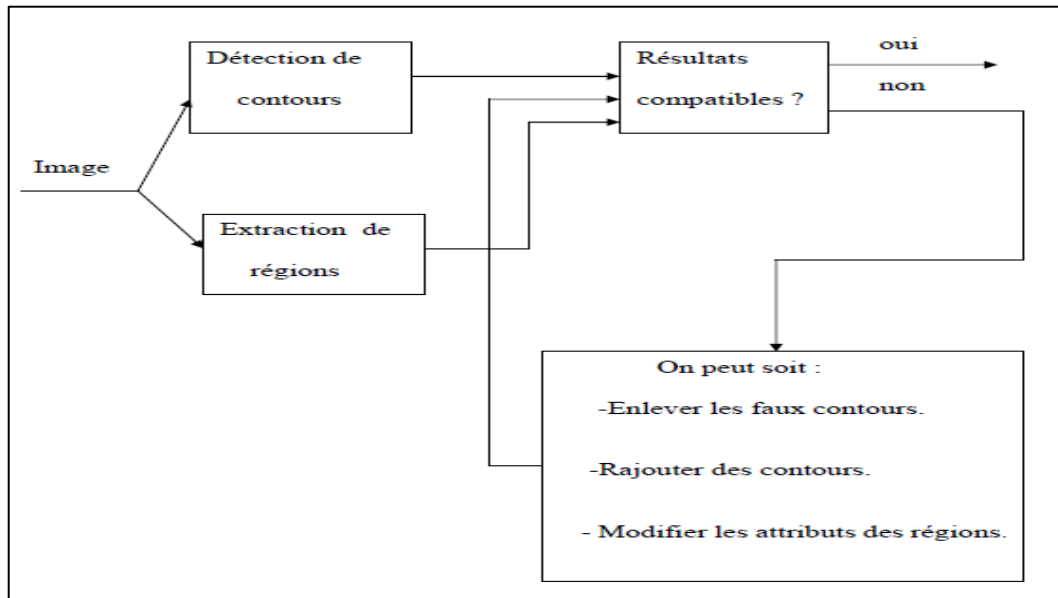
Figure 4.11 : résultats comparative de segmentation région

Globalement les résultats obtenus par notre système, sont très satisfaisants. Comparativement aux méthodes utilisées dans la littérature qui présentent quelques inconvénients pour certains types d'images : «Sur-segmentation» (Lignes de partage des eaux), «Erreur de localisation » (Division-fusion) et côtés faibles «weak edges» (Croissance de région), en plus elles sont couteuses en temps de calcul, notre algorithme s'adapte bien à différents problème posé par les méthodes précédents et remédie au problème de temps d'exécution.

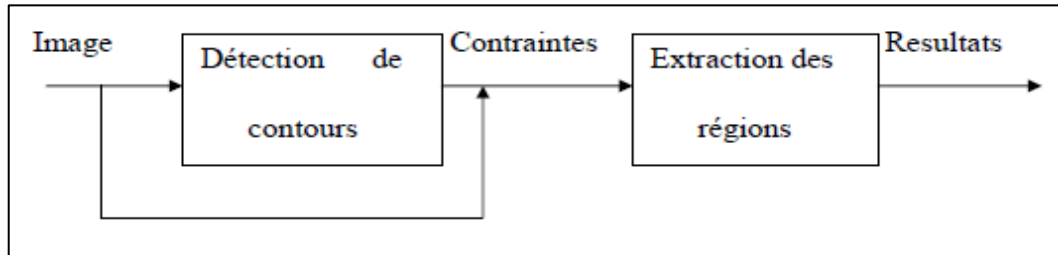
1.3 Segmentation mixte :

Cette approche combine l'approche par région et l'approche par contour. Elle exploite conjointement l'information contour et l'information région afin de pallier les insuffisances de chacune des deux approches. Peu de travaux existe dans cet axe. La coopération peut-être associative ou séquentielle.

Dans le cas de coopération associative, la compatibilité des résultats fournis par chaque approche est recherchée, et dans l'autre cas, les résultats de première méthode mise en œuvre fournissent des contraintes à la seconde Figure 4.12.



a) Coopération séquentielle Région-Contour



b) Coopération associative Région-Contour

Figure 4.12 : Segmentation Mixte

Dans notre cas on utilise la segmentation coopérative : l'automate peut changer d'état (pixel contour ou pixel région) selon son voisinage pour faire une segmentation contour ou une segmentation région).



a) image original

b) image segmenté

Figure 4.13 : résultats de Segmentation Mixte
--

Toutefois, les régions obtenues ne correspondent pas, dans tous les cas, aux objets représentés dans l'image. Les limites des régions obtenues sont habituellement imprécises et ne coïncident pas exactement aux limites des objets de l'image.

Les détecteurs de contours utilisés peuvent être simples, comme les opérateurs de Sobel. Les résultats de cette segmentation sont les prétendants des limites des objets de l'image.

Toutefois, ils peuvent présenter de fausses détections et ils ne sont généralement pas fermés la coopération entre les segmentations par régions et par contours contribue à une meilleure prise en compte des caractéristiques des entités de l'image et, par conséquent, à une meilleure segmentation.

VI. Conclusion :

Dans ce chapitre, on a présenté en détail notre application, qui est une nouvelle méthode de segmentation d'image contour et/ou région par les automates cellulaires. L'algorithme a été testé sur des images en niveaux de gris, et comparé avec des résultats de méthode standard (Sobel) pour une segmentation contour, et il a été aussi comparé avec les méthodes (LPE, Croissance de région, division fusion), pour une segmentation région pour différents types d'images niveau de gris et couleur. Les résultats sont très satisfaisants, tant en qualité de segmentation obtenue qu'en temps d'exécution.

Conclusion générale et perspectives

Ce mémoire de fin d'étude a pour objectif l'étude et l'utilisation de la simulation orientée agents et appliquée à l'une des problématiques les plus complexes en traitement d'image : segmentation région et contour.

Dans ce travail nous avons essayé de remplacer les méthodes traditionnelles de traitement d'image qui utilisent une approche séquentielle, par une approche qui consiste à utiliser les systèmes multi-agents. Ce système est représenté par les automates cellulaires qui sont capable de s'auto-organiser pour créer un comportement global et complexe et de faire émerger collectivement une segmentation optimale de l'image. Les premiers travaux utilisant les automates cellulaires ont donnés naissance à des nouvelles méthodes pour la résolution des problèmes dans différents domaines d'application.

Nous avons bénéficié de la maniabilité et la vitesse d'exécution des automates cellulaires pour mettre au point un algorithme efficace de segmentation d'images. La segmentation optimale émerge progressivement des comportements des agents.

Nous avons mené une étude de synthèse et d'art d'une manière tout à fait significative, avant de proposer un système de segmentation d'images à l'aide des automates cellulaires. Cette étude convoite à la fois le domaine de la simulation multi-agents et le domaine de la résolution distribuée de problèmes. Bien que l'interaction entre les deux reste encore très forte dans notre manière de concevoir les systèmes réactifs, chacun de ces domaines a d'une certaine manière gagné son autonomie.

Notre application offre aux traiteurs d'images, un outil pour tester ses idées sur la manière d'optimiser la segmentation d'image. Cette approche est non dédiée à un type d'image particulier.

Du point de vu résultats obtenus, ces derniers sont très satisfaisantes, et démontrent l'adaptabilité des automates cellulaires pour la segmentation face à une grande variété d'images.

En plus du travail de synthèse présenté sur le domaine, la contribution de ce travail est double : Il s'agit d'une part, de proposer un nouvel algorithme appliqué à la détection de contours et/ou région sur une image niveau de gris ou couleur, en ajoutant de nouveaux comportements, permettant l'optimisation de la recherche. D'autre part, il s'agit de réaliser et d'implémenter cet algorithme de détection contour et/ou région sur une plate-forme multi-agents, afin de bénéficier des possibilités de simulation temps réel offerte par cet outil.

Notre système reste ouvert et plusieurs points peuvent être étudiés dans le futur, donc on estime de prolonger ce travail en réalisant ces perspectives :

- On pourrait instancier notre système pour étudier dans un premier temps le comportement sur des vidéos captures via caméra.

- Il serait intéressant d'appliquer d'autres stratégies d'initialisation des paramètres de notre système et aussi de règles de comportement appliqués sur une large gamme d'images et voir s'ils peuvent influencer la qualité du traitement.

- On pourrait adapter notre système pour le domaine de la vision par ordinateur, afin d'effectuer une détection de visages pour une applications directes en vidéo-surveillance, biométrie, robotique, commande d'interface homme-machine.

Nous devons avouer que nous avons passé un temps considérable en réalisant ce travail motivé par l'importance du sujet proposé à travers les objectifs soulignés au début de ce document.

Référence Bibliographique

[Aristote, 2002] Aristote. ‘La Métaphysique’, ‘la Politique’, ‘Ethique à Nicomaque’, ‘les physiques’ etc... Thomas d’Aquin : tous les commentaires des œuvres d’Aristote. (2002)

[Chate et al., 1987] H. Chate and P. Manneville, Intermittence spatio-temporelle et automates cellulaires probabilistes, C. R. Acad. Sci. Paris Sér. II Méc. Phys. Chim. Sci. Univers Sci. Terre 304, no. 12, 609–613. (1987).

[Clavin et al., 1986] P. Clavin, D. d’Humières, P. Lallemand, and Y. Pomeau, Automates cellulaires pour les problèmes à frontières libres en hydrodynamique à deux et trois dimensions, C. R. Acad. Sci. Paris Sér. II Méc. Phys. Chim. Sci. Univers Sci. Terre 303 (1986), no. 13, 1169–1174. MR 89k :76004. (1986).

[Dempster, 1998] M. Dempster. A Self-Organising Systems Perspective on Planning for Sustainability. Thèse de doctorat, University of Waterloo, School of Urban and Regional Planning. (1998).

[Djemame et al., 2011] S. Djemame, O. Djidel, M-C. Batouche. “ Image Segmentation Using Continuous Cellular Automata”, IEEE 10th International Symposium on Programming and Systems, ISPS 2011, Alger, Algeria, (25-27 avril 2011).

[Drogoul, 1993] A. Drogoul. De La Simulation Multi-Agent A La Resolution Collective de Problèmes. These, Université Paris VI. (1993)

[Drogoul et al., 2003] A. Drogoul, Diane Vanbergue, and Thomas Meurisse. Simulation Orientée Agent : où sont les agents ? Actes des Journées de Rochebrune, Rencontres interdisciplinaires sur les systèmes complexes naturels et artificiels, (2003).

[Emaszko, 2004] D. Emaszko. Projet de semestre Application POEtique (projet générique) (2004).

[Ferber, 1995]. J. Ferber. Les Systèmes Multi Agents: vers une intelligence collective (1995)

[Ferber,1999]. J. Ferber. Multiagent Systems : A Modern Approach to Distributed Artificial Intelligence, edited by Gerhard Weiss, MIT Press,(1999)

[Frisch et al., 1986] U. Frisch, B. Hasslacher, and Y. Pomeau, Lattice gas automata for navier-stokes equation,PhysRevLet 56, no. 1505. (1986).

[Gardner et al., 1970], B. Drossel, Schwabl. F., « Self organized criticality in a forest-fire model », Physica A, vol.191, 1992, p. 47-50. (1970).

[Gardner, 1970] M .Gardner, « The fantastic combinaisons of John Conway's new solitaire game “ Life “ », Scientific American,(1970).

[Gardner, 1971] M .Gardner, « On Cellular Automata, Self-reproduction, the Garden of Eden, and the Game of Life », Scientific American, vol. 224, 1971, p. 112-118,(1971).

[Hadj et al., 2006] 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).

[Heylighen, 2002] F.Heylighen . The science of self-organisation and adaptivity. In The Encyclopedia of Life Support Systems. UNESCO Publishing-Eolss Publishers,(2002).

[Holcombe et al., 2006] M.Holcombe, Simon Cakley, and Rod Smallwood. A general framework for agent-based modelling of complex systems. Technical report, Department of Computer Sciences, University of Sheffeld, (2006).

[Jennings et al.,1998] N.JENNINGS, SYCARA, K. et WOOLDRIDGE, M. (1998). A roadmap of agent research and development. Autonomous Agents and Multi-Agent Systems, 1:7–38 ,(1998).

[Kramm, 2005]A.Kramm Automates Cellulaires Mémoire de maîtrise d’informatique Université Paris VIII Septembre 2005 Sous la direction de Pierre Audibert P 15 (2005).

[Kulakowski, 2000] K. Kulakowski. Automaty komorkowe. AGH, (2000).

[**Louis, 2002**] P.Louis .automates cellulaires probabilistes mesures stationnaires, mesures de Gibbs associées et ergodicité .P5 (2002).

[**Maxwell, 1890**] J. Maxwell, Scientific paper ii, Cambridge University Press,(1890).

[**Melkemi, 2010**] K.Melkemi, « Szgmentation d'image à l'aide d'automate cellulaire » proposition d'un projet fin d'étude, université de Biskra (2010)

[**Minsky, 1965**]. M. Minsky “Matter, Mind and Models”, Proceedings of IFIP Congress, 1965, p. 45-49.(1977).

[**Moigne, 1990**] J. Moigne , La modélisation des systèmes complexes, Dunod, (1990).

[**Montheillet, 1997**] F.Montheillet,., Briottet, L. 1997. « Prévision des hétérogénéités de déformation dans un agrégat de deux phases viscoplastiques ». La revue des Métallurgie-CIT/Science et génie des matériaux. ,(1997).

[**Morin, 1977**] E. Morin. La Méthode 1. La Nature de la Nature(1977).

[**Pritsker, 1979**] A.Alan B.Pritsker. Compilation of definitions of simulation. SIMULATION,33(2):6163.<http://sim.sagepub.com>.<http://dx.doi.org/10.1177/00375497790330025> (1979)

[**Popovici et al., 2002**] A.Popovici and D.Popovici “Cellular Automata in Image Processing” .Proceedings of the 15th International Symposium on the Mathematical Theory of Networks and Systems, (2002).

[**Ralambondrainy, 2009**] T. Ralambondrainy. Observation de simulations multi-agents à grande échelle. PhD thesis, Université de la Réunion,(2009).

[**Ramat, 2006**] E.Ramat. Introduction à la modélisation et à la simulation à événements discrets. pages 49–74, (2006).

[**Russell et al., 2006**] S.Russell et P.Norvig. Artificial Intelligence : A Modern Approach. Prentice Hall, (2006).

[Sansonnnet, 2004] J-P.Sansonnnet Processus d'émergence de formes sur des populations d'agents à grain fin (2004). **[Simon, 1962]** H. Simon. The architecture of complexity. Proceedings of the American Philosophical Society, pages 467–482, 1962(**2004**).

[Simon, 2001] H.Simonin,. Le modèle satisfaction-altruisme : coopération et résolution de conflits entre agents situés réactifs, application à la robotique. Thèse de doctorat, Universiy Montpellier II(2001).

[Toffoli, 1984] T. Toffoli, Cellular automata as an alternative to rather than an approximation of differential equations in modeling physics,PhysicaD 10 (1984), 117–127.(1984)

[Varenne, 2001] F.Varenne (2001). What does a computer simulation prove? In Society for Computer Simulation (SCS), editor, Simulation in Industry - Proc. of the 13th European SimulationSymposium,pagess49-554.,<http://hal.archives-ouvertes.fr/hal-00004125/en/>.(2001)

[Victor,1984] G.Victor, Simulating physics with cellular automata,PhysicaD 10, 96–116.(1984).

[Wolfram, 1986]. S.Wolfram., « Random Sequence Generation by Cellular Automata. Advances in Applied Mathematics », vol. 7, 1986, p. 123-169,(1986).

[Wongthanavas, 2007] S.Wongthanavas, V. Tangvoraphonkchai “Cellular Automata-based Algorithm and its Application in Medical image processing”, vol III, pp.41-44, ICIP (2007)