الجمهـورية الجزائرية الديمقـراطية الشـعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليـم الــعالي والبحث العلمــي

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

جامعة 20 أوت 1955 سكيكدة

UNIVERSITÉ 20 Août 1955 – Skikda

Faculté des Sciences

Département d'Informatique

Mémoire de fin d'études en vue de l'obtention du diplôme

Master : Systèmes Informatiques (SI)

Thème :

Une approche basée Agent pour le clustering

Réalisé par :

- Ouaddah Ilhem
- Sarroub Bouchra

Encadré par :

- Dr. Zeghida Djamel

Jury :

- Ramdan Chafika
- Benjeddou Zeyneb

2021/2022

Dédicaces

A ma chère famille, mes chères amis pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études.

Ilhem

Dédicaces

Avant tout je rends grâce à <ALLAH>de m'avoir donné la force et le courage durant ces langues années d'étude et pour l'accomplissement de ce travail.

J'ai le plaisir de dédier ce modeste travail :

- À mes parents.
- Tous tes qui m'ont souhaité le succès et le bonheur dans ma vie, Tous ceux qui me sont chers.
- Et tous les chercheurs qui peuvent avoir besoin à l'avenir.

Bouchra

Remerciement

Avant tout, nous remercions Allah le tout puissant qui nous a aidé et nous a donné la patience et le courage durant ces longues années d'étude et pour l'accomplissement de ce travail.

En second lieu, nous tenons à remercier notre encadreur Mr.ZEGHIDA Djamel pour ses précieux conseils et son aide durant toute la période du travail.

Nos remercîments et gratitudes vont également aux membres de jury qui nous ont fait l'honneur de jugser ce travail.

Enfin, on remercie tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce travail.

ملخص

يتم غزونا ، بشكل يومي ، بواسطة كتلة هائلة من البيانات ، تتم معالجتها بواسطة أنظمة معقدة تستهلك الزمان والمكان. يتطلب استخراج المعرفة والدراية من هذه البيانات وتحليل نتيجة هذا الاستخراج استخدام تقنيات وطرق قادرة على التكيف مع الطبيعة والمتطلبات القاسية والمتغيرة لهذه العلاجات. تجميع البيانات هو أحد تلك التقنيات التي تم تحديها وتصنيفها على أنها مشاكل في منتهى الصعوبة. دقة هذا الأخير بعيدًا عن متناول الأساليب الدقيقة ويتطلب استخدام تقنيات تقريبية. للبقاء في المنافسة ومن أجل النجاح في التجميع الفعال والجودة ، فإننا نستخدم باستمرار مناهج جديدة أو نجمع بين العمل الحالي.

الهدف من هذا العمل هو تحسين نتائج الخوارزميات العنقودية باستخدام نهج قائم على أنظمة متعددة العوامل. لتحقيق هذا الهدف. تم تنفيذ المتغيرات المقترحة في تطبيق جافا لسطح المكتب والذي يستخدم لتجميع البيانات في مجموعات بيانات ملف نصي قدمها المستخدم أو أنشأها.

الكلمات المفتاحية: تجميع البيانات ، التجميع الصلب ، التصنيف غير الخاضع للرقابة ، التحسين التوافقي ، AGR ، النظام متعدد العوامل ، SMA.

Résumé

On est envahi, au quotidien, d'une masse immense de donnée, traitée par des systèmes complexes et consommateurs en terme de temps et d'espace. L'extraction de connaissances et de savoir de ces données et l'analyse du résultat de cette extraction demandent l'utilisation de techniques et méthodes capables de s'adapter aux natures et aux exigences dures et changeantes de ces traitements. Le clustering de données est l'une de ces techniques défiées et classées comme problèmes NP-difficiles. La résolution de ces derniers, est hors de portée des méthodes exactes et exige le recours à des techniques approximatives. Pour rester compétitif et afin de réussir un clustering performant et de qualité, on ne cesse d'utiliser de nouveaux approches ou de combiner des existants travaux.

Le but de ce travail est d'améliorer les résultats des algorithmes de clustering utilisant une approche basé sur les systèmes multi-agents. Pour réaliser cet objectif. Les variantes proposées ont été implémentées dans une application bureau Java qui est utilisée pour faire le clustering des données en dataset du fichier texte introduits ou crée par l'utilisateur.

Mots clés: clustering de données, clustering dur, classificaion non-supervisée, optimisation combinatoire, AGR, Système multi-agent , SMA.

Abstract

We are invaded, on a daily basis, by an immense mass of data, processed by complex systems that consume time and space. The extraction of knowledge and know-how from these data and the analysis of the result of this extraction require the use of techniques and methods capable of adapting to the natures and the harsh and changing requirements of these treatments. Data clustering is one of those techniques challenged and classified as NP-hard problems. The resolution of the latter is beyond the reach of exact methods and requires the use of approximate techniques. To remain competitive and in order to succeed in efficient and quality clustering, we are constantly using new approaches or combining existing work.

The main goal of this work is to improve the results of clustering algorithms using an approach based on multi-agent systems. To achieve this goal. The proposed variants have been implemented in a Java desktop application which is used to cluster data into text file datasets introduced or created by the user.

Keywords: data clustering, hard clustering, unsupervised classification, combinatorial optimization, AGR, multi-agent system, SMA.

Table des matières

Table des Figures

Liste des tableaux

Introduction générale

1. Contexte de la recherche

L'automatisation de la résolution de problèmes en informatique a donné naissance à la classification non supervisée, c'est le fait de classifier les données dans des groupes, ce processus est appelé clustering. Les groupes créés sont appelés clusters, l'objectif de cette approche est de découvrir la structure de base des données pour en extraire de l'information.

Le Clustering (ou partitionnement de données), cette méthode de classification non supervisée rassemble un ensemble d'algorithme d'apprentissage dont le but est de regrouper entre elles des données non étiquetées présentant des propriétés similaires. Isoler ainsi des schémas ou des familles permet aussi de préparer le terrain pour l'application ultérieure d'algorithmes d'apprentissage supervisé (comme le K-means). Ce dernier est devenu une tâche extrêmement difficile et sa complexité ne cesse de croitre en conséquence des données massives qui ne cessent de grandir.

Le clustering est utilisé notamment lorsqu'il est coûteux d'étiqueter les données. C'est néanmoins un problème mal défini mathématiquement : différentes métriques et/ou différentes représentations des données aboutiront à différents regroupements sans qu'aucun ne soit nécessairement meilleur qu'un autre. Ainsi la méthode de clustering doit être choisie avec soin en fonction du résultat attendu et de l'utilisation prévue des données.

D'autre part, un système multi-agent (SMA) qui est un système composé d'un ensemble d'agents, situés dans un certain environnement et interagissant selon certaines relations. Un agent est une entité caractérisée par le fait qu'elle est, au moins partiellement, autonome. Ces systèmes furent utilisés, en Intelligence Artificielle distribuée, comme outil de résolution distribuée de problème et après, en industrie comme dans beaucoup d'autre doamine, comme outil de simulation et, dans l'ingénierie logicielle, comme support de conception.

2. Contribution

Notre travail vise à consolider l'utilisation des Agents et des Systèmes Multi-Agents comme outil de calcul et d'optimisation dans le domaine du Clustering de données. En utilisant le modèle organisationnel AGR (Agent, Groupe et Rôle) nous allons faire passer les agents comme objets à clusturer et les groupes comme clusters

3. Organisation du document

Notre mémoire de fin d'études se compose de quatre chapitres :

Chapitre 1 : Le clustering des données

Dans ce chapitre, nous présentons le clustering de données et son objectif, ses différentes méthodes et ses caractéristiques. Nous commençons par une petite définition du clustering, son historique, ses méthodes de fonctionnement et les avantages de ce concept.

Chapitre 2 : Les systèmes multi-agents (SMA)

Dans ce chapitre , nous présenterons l'approche orientée agents , dans lequel s'inscrit notre travail, nous élaborons ensuite les différentes caractéristiques et traits des systèmes multi-agents (SMA) .

Chapitre 3 : Proposition du système

Dans ce chapitre nous proposons l'architecture de notre système ainsi qu'une description détaillée à l'aide du modèle Agent, groupe, rôle (AGR) et les différentes classes de notre système.

Chapitre 4 : Implémentation

Le dernier chapitre est consacré à la présentation de différents environnements de développement et les langages de programmation utilisés (JAVA , MADKIT , ECLIPSE) pour réaliser notre projet.

Chapitre 1 : Le clustering

1. Introduction

Chaque jour, nous sommes obligés de traiter des quantités infinies d'informations car elles sont diverses. Ces processus facilitent la prise de décision et la résolution de problèmes, de la simple représentation des données à l'analyse avancée. À cette fin, nous utilisons des méthodes et des techniques pour améliorer et permettre ces traitements, y compris le triage. Ce dernier peut être supervisé et conserver très simplement son nom commun d'origine : classification ou dans ce cas non supervisé, pour plus de précision : clustering.

Dans ce chapitre, nous nous intéressons au clustering. Nous présenterons ses concepts et méthodes de base ainsi que des mesures d'évaluation et de validation.

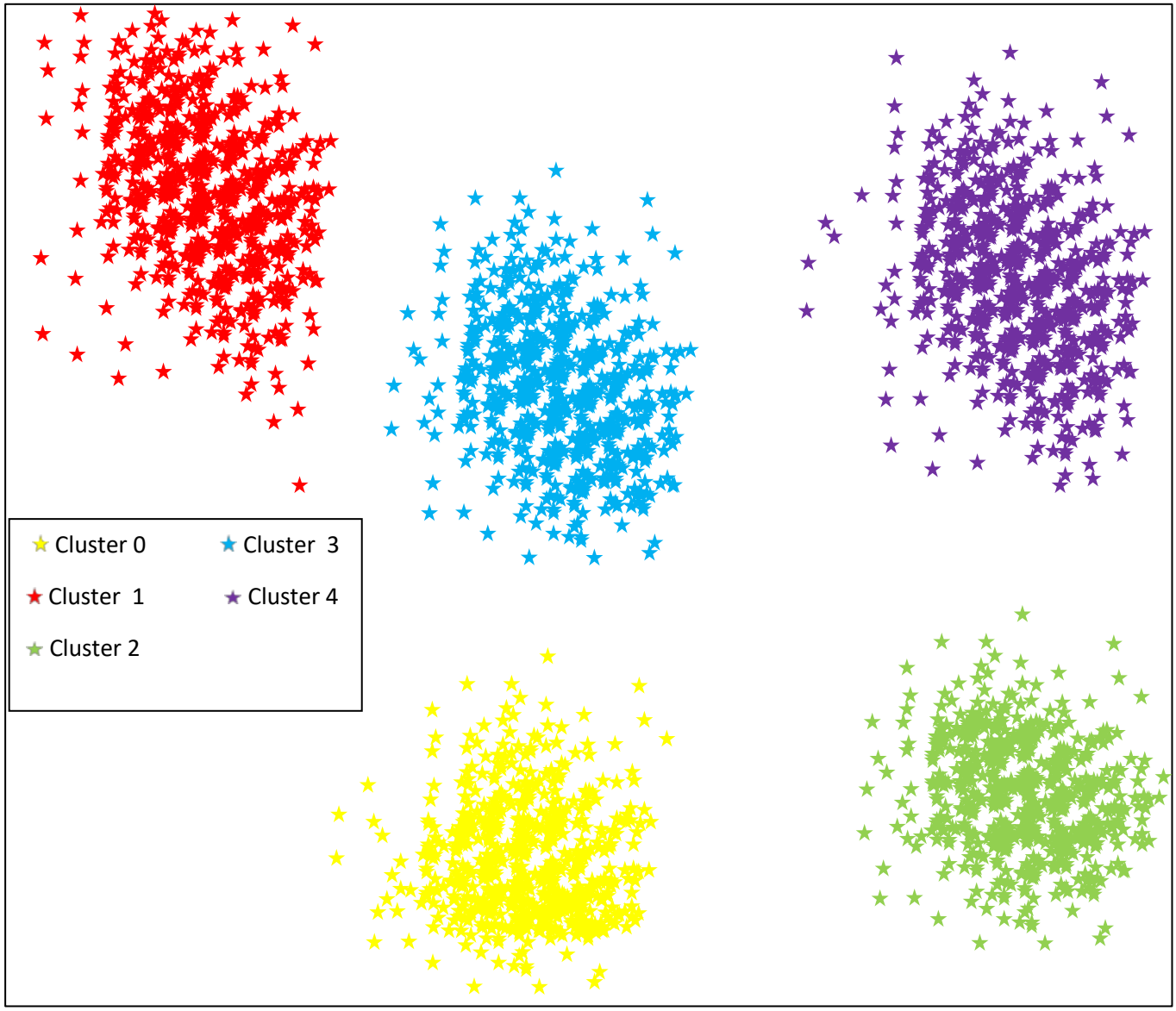

Figure 1.1 : Clustering.

2. Définition de Clustering :

Le clustering est la tâche de regrouper un ensemble d'objets (physiques ou abstraits) ou de données plus larges, de manière non supervisée (c'est-à-dire sans l'aide préalable d'experts), de telle sorte que le même ensemble d'objets (appelés clusters) soit plus proche (dans le sens de la sélection de critères de (dis)similarité) que les objets de groupes différents (clusters).

C'est une tâche majeure dans l'exploration de données exploratoire, une technique d'analyse de données statistiques largement utilisée dans de nombreux domaines, notamment l'apprentissage automatique, la reconnaissance de formes, le traitement du signal et le traitement des données, les images, la recherche d'informations, la bioinformatique, la compression de données et l'infographie…

L'idée est donc de découvrir des groupes au sein des données, de façon automatique. [1]

Pour établir l'équilibre, il minimise l'inertie à l'intérieur des clusters et maximise celle entre les clusters afin de bien les différencier. L'objectif peut être de hiérarchiser ou de répartir les données. La qualité d'un résultat de clustering dépend de la mesure de similarité utilisée par la méthode et la mise en œuvre.

3. L'historique

Le clustering apparaît premièrement dans l'anthropologie par *Driver* et *Kroeber* en 1932 [11], puis en psychologie par *Zubin* en 1938 [12], et *Robert Tryon* en 1939 [13], et il a été célébrisé par *Cattel* en 1943 [14]. Des difficultés de calcul ont retardé son développement jusqu'à la fin des années 1950 lorsque l'informatisation a entraîné une prolifération de techniques de clustering. Au début des années 1960 les techniques de clustering ont été émergées dans la biologie par *Sokal* et *Sneath* en 1963 [15]. Elles ont reçu l'attention des chercheurs de nombreuses autres disciplines, y compris la géographie (*Berry* et *Ray*, 1966) [16] le management (*Morrison*, 1967 [17]]; *Green*, *Frank*, et *Robinson*, 1967 [18]), les sciences politiques (*Kaiser*, 1967) [19], la psychiatrie (*Lorr*, 1966) [20]. L'urbanisation (*Wingo*, 1967) [21], l'économie (*Fisher*, 1969) [22] , et en mathématiques (*Jardine* et *Sibson*, 1968 [23]) [24], [25] En fait l'existence d'activités de clustering peut-être retracée il y a plus de cent ans, dans différentes disciplines, et dans différents pays, pour plus de détails voir [2] .

4. Objectifs du clustering:

Mirkin [2] a identifié une liste d'objectifs de clustering :

• Structure : Représentez les données sous la forme d'un ensemble d'objets similaires (généralement, la structure est l'objectif principal du clustering).

• Description : clusters basés sur les fonctionnalités.

• Corrélation : Découvrez les interrelations entre les différents aspects d'un phénomène.

• Généralisation : Étudier les propriétés des phénomènes liés aux données, un objectif qui nécessite une analyse à plusieurs niveaux des objectifs précédents.

• Visualisation : représente la structure du cluster sous la forme d'une image visuelle.

5. Les principales étapes du clustering

Généralement les méthodes de clustering doivent inclure les étapes suivantes [3][4] :

- Choix de données (optionnellement inclut l'extraction et/ou la sélection des caractéristiques)
- Définition d'un modèle (une fonction) de calcul de proximité appropriée aux données
- Classification ou regroupement
- Abstraction des données (s'il est nécessaire)
- Evaluation des résultats (s'il est nécessaire)

6. Les domaines d'application du clustering :

La classification non-supervisée fut, dans un premier temps, appliquée à :

Un grand nombre de problèmes provenant de divers domaines [5] :

- **En archéologie** pour regrouper des objets datant de l'âge de fer à partir de leur description [6].
- **En médecine** pour découvrir les classes de patients qui présentent des caractéristiques communes afin de détecter les patients atteints d'une même maladie [7]
- **En reconnaissance de forme** pour la construction des systèmes de reconnaissance de l'orateur [8] et dans la segmentation des images pour la détection et l'identification des zones homogènes [9].
- **En marketing** pour l'identification des clients ayant des comportements d'achat similaires afin d'établir des profils de clients et identifier les tendances [10].
- **En planification des villes** pour l'identification de groupes d'habitations suivant leurs type, valeur, localisation géographique
- **En segmentation d'images** pour détection des zones homogènes dans une image.
- **En Web log analyses** pour l'identification de profils d'utilisateurs à travers leur flux de clics (Clickstream)
- **En Text mining** pour la classification des textes selon leur similitude dans des dossiers automatiques.

7. Les types de clustering :

Généralement, le clustering se divise en deux visions de catégorisation, la première vision comporte le clustering dur, doux et floue. Alors que la deuxième comporte le clustering hiérarchique et non-hiérarchique.

7.1 Clustering dur :

Dans le clustering dur (hard clustering) chaque point de données appartient à un seul cluster.

	C1	C2	C3
A	1	0	0
B	0	0	1
C	0	0	1
D	1	0	0
E	0	1	0

Tableau 1.1 : Clustering Dur

7.2 Clustering doux :

Dans le clustering doux (soft clustering) chaque point de donnée peut appartenir à la fois à plusieurs clusters.

	C1	C2	C3
A	1	1	0
B	0	0	1
C	0	1	1
D	1	0	0
E	0	1	0

Tableau 1.2: Clustering doux

7.3 Clustering floue :

Permettant une représentation simple des incertitudes et imprécisions liée aux informations et aux connaissances.

	C1	C2	C3
A	0.65	0.25	0.10
B	0	0.3	0.7
C	0	0.4	0.6
D	1	0	0
E	0	0.2	0.8

Tableau 1.3 : Clustering Floue

7.4 Clustering hiérarchique :

Dans le clustering hiérarchique, les clusters sont combinés de manière itérative, pour finalement se retrouver dans une racine, on peut considérer comme un arbre binaire.

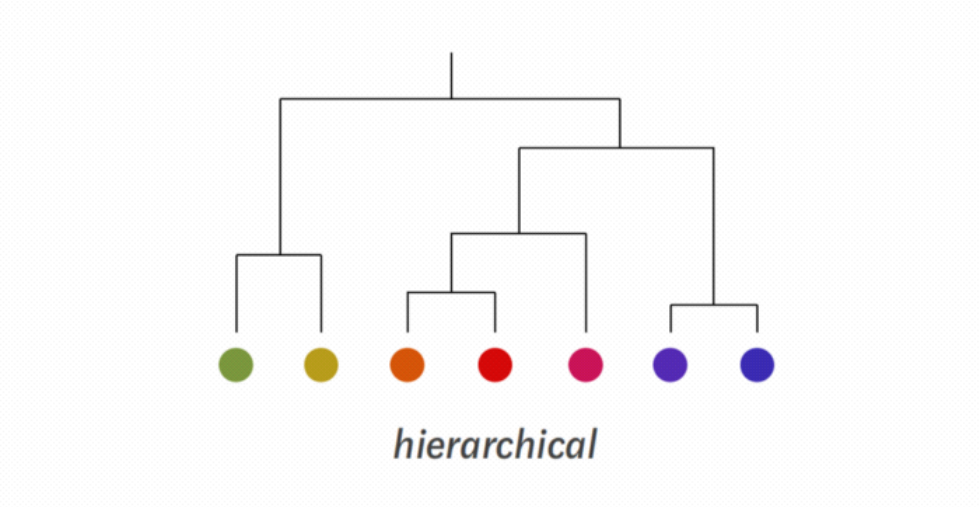

Figure 1.2: Clustering hiérarchique

7.5 Clustering non hiérarchique :

Tout simplement les résultats sont des divisions dures de l'ensemble de données

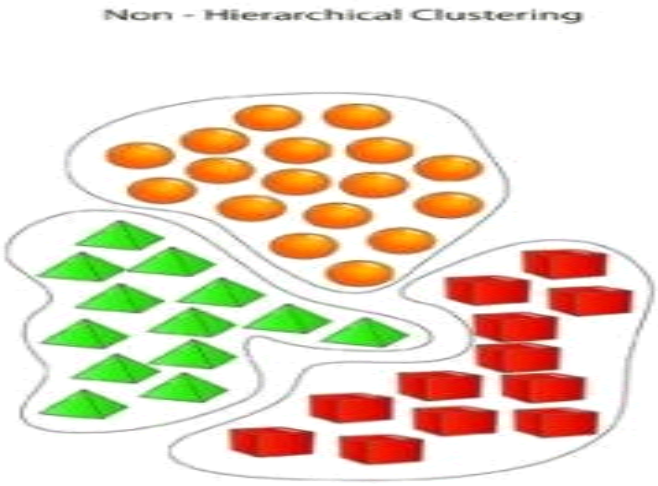

Figure 1.3 : Clustering non hiérarchique

8. Les méthodes de clustering de base

Dans cette section, nous dressons une taxonomie des différentes méthodes et algorithmes de clustering cités dans la littérature. Nous ne prétendons pas faire une liste exhaustive de l'ensemble des notions et méthodes existant dans le cadre du clustering, mais plutôt de donner un aperçu général du fonctionnement des différentes méthodes avec chacune ses avantages et ses inconvénients. Il est ainsi admis dans la communauté travaillant sur le clustering qu'aucun critère ni aucune méthode ne sont intrinsèquement meilleurs que d'autres sur l'ensemble des problématiques envisageables. Par contre, on considère que certaines méthodes sont plus adaptées que d'autres en fonction de l'application ciblée. On distingue classiquement les grandes familles de méthodes en clustering suivantes [26] [27]

- Les méthodes hiérarchiques.
- Les méthodes par partitionnement.
- Les méthodes à base de densité.
- Les méthodes basées sur un modèle.

8.1 Les méthodes hiérarchiques

Le fondement du clustering hiérarchique [28] est de construire une hiérarchie de clusters ou autrement dit, un arbre de clusters, connu aussi sous le nom de dendrogramme tel que présenté dans la figure 4.

Afin de parvenir à un tel arbre hiérarchique de clusters, il existe deux types d'approches, à savoir, l'approche ascendante, dite agglomérative clustering hiérarchique ascendant (CHA) et clustering hiérarchique descendant (CHD), dite divisive :

- **La méthode ascendante** (agglomérative ou Bottom up en anglais) construit l'arbre du bas vers le haut en démarrant avec autant de clusters que d'objets initiaux dans la base, puis fusionnant successivement les clusters considérés comme les plus similaires, jusqu'à obtenir un unique cluster racine, contenant l'ensemble des objets.

Algorithm agglomerative Hierarchical clustering

1) **Compute the dissimilarity matrix between all the date points.**
2) **Repeat**
3) **Merge cluster as $c_{a\cup b}= c_a \cup c_b$: Set new clusters cardinality as $N_{a\cup b} = N_a + N_b$**
4) **Insert a new row and column containing the distances between the new cluster $C_{a\cup b}$ and the remaining clusters.**
5) **Until only one maximal cluster remains or satisfaction of stop criterion .**

- **La méthode descendante** (divisive, Top down en anglais) construit l'arbre du « haut » vers le « bas » en démarrant avec un unique cluster contenant l'ensemble des objets de la base, puis divisant successivement les clusters de manière à ce que les clusters résultants soient les plus différents possibles, et ce jusqu'à obtenir des singletons (autant de clusters que d'objets dans la base).

Ensuite, une fois la hiérarchie formée, une étape optionnelle peut être ajoutée pour affiner le résultat. Il s'agit de déterminer le niveau L de coupure le plus approprié à appliquer sur l'arbre pour un regroupement des données aussi pertinent que possible. Par exemple, le choix du niveau de coupure ($L = 4$) dans le dendrogramme de la figure 4, renvoie le partitionnement suivant : $C = \{\{x1, x2, x3\}, \{x4, x5, x6\}\}$. Ce dernier paramètre L peut être choisi relativement au nombre de clusters désirés ou à l'aide d'une analyse statistique de la qualité des différentes partitions que l'on peut extraire de l'arbre.

Algorithm basic divisive hierarchical clustering

1) Start with the root node consisting all the data points.
2) Repeat
3) Split parent node into two parts c_1 and c_2 using bisecting k-means to maximize Ward's distance W(c_1, c_2)
4) Construct the dendogram.Among the current, choose the cluster with the highest squared error.
5) Untill Singelton leaves are obtained or satisfaction of stop criterion .

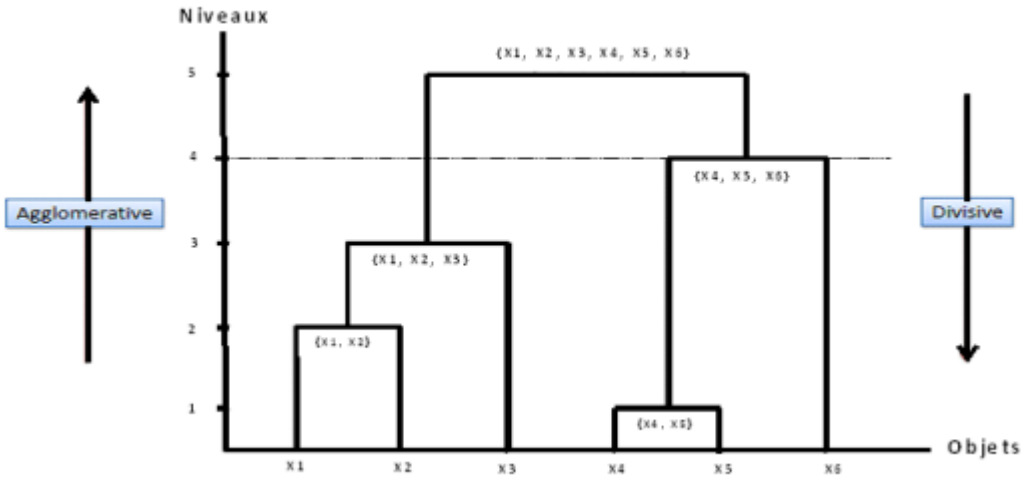

Figure 1.4 : exemple de dendrogramme

8.1.1 Le principe de fonctionnement

Le clustering hiérarchique initialise un système de classe comme un ensemble de singletons (dans le cas agglomératif) ou comme un seul cluster qui contient toutes les données (dans le cas divisif) et procède itérativement par fusion ou éclatement des clusters les plus adéquats jusqu'à ce qu'un critère d'arrêt soit satisfait. La concordance d'un cluster pour la fusion ou éclatement dépend de la (dis) similarité des éléments du cluster. Cela reflète l'hypothèse générale que les clusters contiennent des objets similaires. Pour fusionner (ou éclater) des clusters d'objets plutôt que des objets à part, on doit généraliser la distance entre des objets individuels à la distance entre clusters. De telles mesures de proximité désirées, sont appelées critères d'agrégation ou critères du lien (en anglais linkage metrics).

De nombreux critères d'agrégation ont été proposés dans la littérature. Parmi les critères couramment utilisés pour calculer la distance entre deux clusters, on cite [29] :

8.1.2 Distance entre clusters

- Le critère du saut minimal (Single-link en anglais) : Pour single link, la distance entre deux clusters est le minimum des distances entre toutes les paires d'objets appartenant à ces deux clusters différents. Autrement dit, la distance entre deux clusters C1 et C2 est définie par la plus courte distance séparant un objet de C1 et un objet de C2 :

$$D(C1, C2) = min(d(x, y)), x \in C1, y \in C2.$$

- Le critère du saut maximal (Complete-link en anglais), pour la méthode complete link, on utilise le maximum de ces distances, autrement dit, la distance entre deux clusters C1 et C2 est définie par la plus grande distance séparant un objet de C1 et un objet de C2 :

$$D(C1, C2) = max(d(x, y)), x \in C1, y \in C2.$$

- Le critère de la moyenne (Average-link en anglais), pour la méthode average link, on utilise la moyenne de ces distances ; ce critère consiste à calculer la distance moyenne entre tous les objets du cluster C1 et tous les éléments de C2 :

$$D(C1, C2) = \frac{1}{|C1| \times |C2|} \sum_{x \in C1} \sum_{y \in C2} d(x, y)$$

- Le critère de Ward : Le critère de Ward consiste à choisir à chaque étape le regroupement de clusters tel que l'augmentation de l'inertie intra clusters soit

minimale. Il ne s'applique que dans un espace Euclidien. La distance entre deux clusters C1 et C2 est définie par :

$$D(C1, C2) = \frac{|C1| \times |C2|}{|C1| + |C2|} \, d(gc_1, gc_2)$$

Avec : gc1 et gc2 sont respectivement les centres de gravité des clusters c1 et c2.

Avantages et inconvénients

Les avantages du clustering hiérarchique incluent :

- La facilité pour traiter différentes formes de similarité ou de distance entre les objets.

Les points faibles :

- Bien que ces méthodes soient largement utilisées, elles deviennent difficilement utilisables face à de larges bases de données, la complexité est quadratique en fonction du nombre d'objets de la base, puisque les distances entre toutes les paires d'objets possibles doivent être calculées.
- Le choix du critère d'arrêt qui reste mal défini.

8.2 Les méthodes par partitionnement

Le nombre de clusters k est connu a priori, celui-ci est le paramètre du démarrage de l'algorithme de partitionnement, le processus se base sur la distance, plus deux objets sont proches, plus ils ont la possibilité d'être regroupés dans le même cluster. Un cluster est un ensemble d'objets dans lequel chaque objet est plus proche du prototype de son cluster. Ce prototype est un centroïde s'il est la moyenne de tous les objets du cluster. Si ce centroïde n'est pas significatif, alors il est appelé un médoïde, ce médoïde est un objet particulier du cluster.

Algorithm 1 general prototype-based clustering

1) Select k initial prototypes.
2) Refine prototypes until convergence.
   2.1) Find the closest prototypes for n points.
   2.1) Recompute cluster prototypes.

8.2.1 Principe méthode de partitionnement

Avec une base de données de n objets ou n-uplets de données, une méthode de partitionnement construit k partitions des données, où chaque partition représente un cluster, où

chaque groupe doit contenir au moins un objet et chaque objet doit appartenir à un groupe exactement.

Pour obtenir une optimalité globale dans le clustering basée sur le partitionnement, il faudrait énumérer de manière exhaustive toutes les partitions possibles.

Il existe plusieurs méthodes de clustering par partitionnement, parmi elles on cite :

- **L'algorithme k-means**, où chaque cluster est représenté par la valeur moyenne des objets du cluster.
- **L'algorithme k-medoids**, où chaque cluster est représenté par l'un des objets

Ces méthodes de clustering heuristique fonctionnent bien pour rechercher des clusters de forme sphérique. Pour trouver des clusters avec des formes complexes et pour regrouper des ensembles de données très volumineux, les méthodes basées sur le partitionnement doivent être Étendu.

8.2.2 Algorithme K-Means

K-means est une méthode qui a été développée par MacQueen en 1967 (MacQueen,1967). Elle vise à partitionner un ensemble de données en K clusters homogènes, K est le nombre de clusters voulue ou fixé a priori, elle est dédiée aux tâches de clustering, elle permet de diviser une population donnée en K groupes homogènes appelés clusters. Le nombre de clusters K est déterminé par l'utilisateur selon ses attentes.

L'objectif du K-means est de segmenter les données en k-groupes, il faut spécifier avant de lancer l'algorithme combien de clusters ont désiré créer, c'est le paramètre k [30,31]. Ensuite, k points sont choisis semi aléatoirement comme centre des clusters. Toutes les instances sont assignées au centre le plus proche d'eux, ceci étant calculé avec la distance euclidienne.

Algorithm K-means clustering

1) Select K points as initial centroids.

2) Repeat

3) Form K clusters by assigning each point to its closest centroid.

4) Recompute the centroid of each cluster.

5) Until convergence criterion is met .

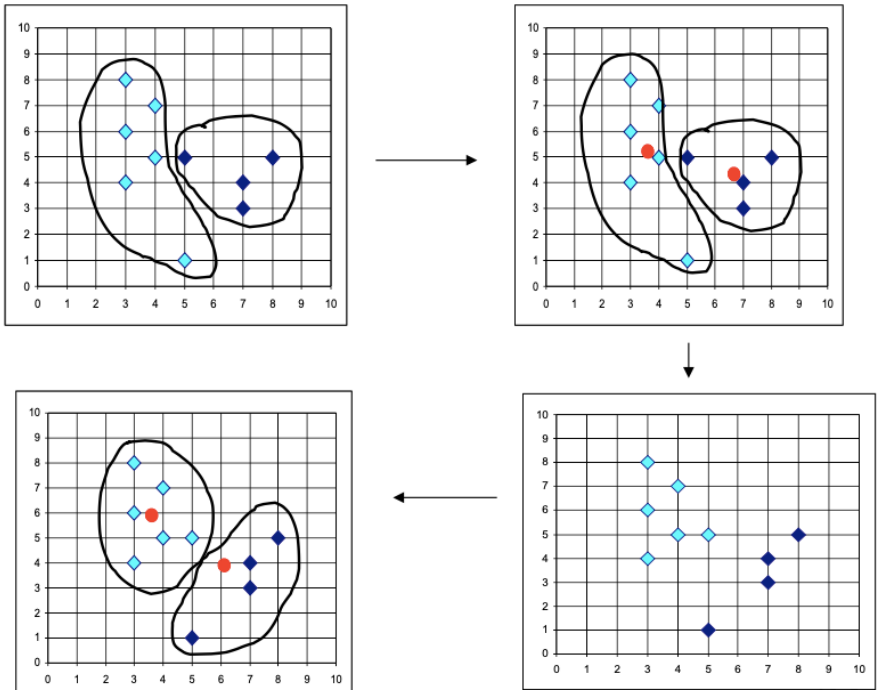

Figure 1.5 : Illustration de de l'algorithme K-means.

8.3 Méthodes basées sur la densité

Les clusters sont initialement formés comme la méthode single-link, mais avec des critères pour empêcher l'addition d'objets qui sont beaucoup plus éloignés du dernier objet ajouté au cluster, les objets, rejetés de cette façon, initient de nouveaux clusters [24], [52].

La méthode classique qui repose sur la recherche de régions denses (ou modes) est celle de Wishart 1969 [53]. La probabilité de la densité estimée au point x est définie comme suit [59], [60]:

$$\hat{\rho}(x) = \frac{k(x)}{n * v_n}$$

Tandis que la deuxième approche donne une valeur fixe au k et calcule pour chaque objet la taille de son voisinage $V(x)$ pour qu'elle contienne k objets. La première approche définit une valeur fixe de volume Vn autour des objets, et calcule pour chaque point x sa valeur (x) de nombre de points tombants dans son voisinage de taille Vn. Les objets sont alors étiquetés comme étant denses ou non selon que leur voisinage Vn contient plus ou moins de points que la

valeur du seuil k, ce seuil est mis à une valeur dépendante de la taille de l'ensemble de données n. La probabilité de la densité estimée au point x est définie comme suit [59], [62], [60]:

$$\hat{\rho} = \frac{k}{n * V_n(x)}$$

Un point est un mode ou dense dans l'algorithme de Wong et Lane [63] , [53].

Deux approches générales non-paramétriques proches de la méthode de Wishart ont été utilisées pour la définition de densité, l'approche de Parzen, proposée en 1962 [55] et appliquée au problème de classification par Specht en 1967 [56], et l'approche des plus proches voisins introduite par Fix et Hodges [57], [58] et explorée par Patrick [D'autres méthodes de clustering ont été développées, qui se basent sur la densité, ces méthodes peuvent trouver les régions de forte densité dans l'espace de données, chacune des régions étant prise pour signifier un cluster différent. La recherche de modes est faite en considérant une petite région locale de volume Vn autour de chaque point, et on calcule le nombre d'objets (i) tombant dans cette région pour chaque objet. Les points non denses sont alors enlevés et les points denses sont groupés par la méthode single-link.

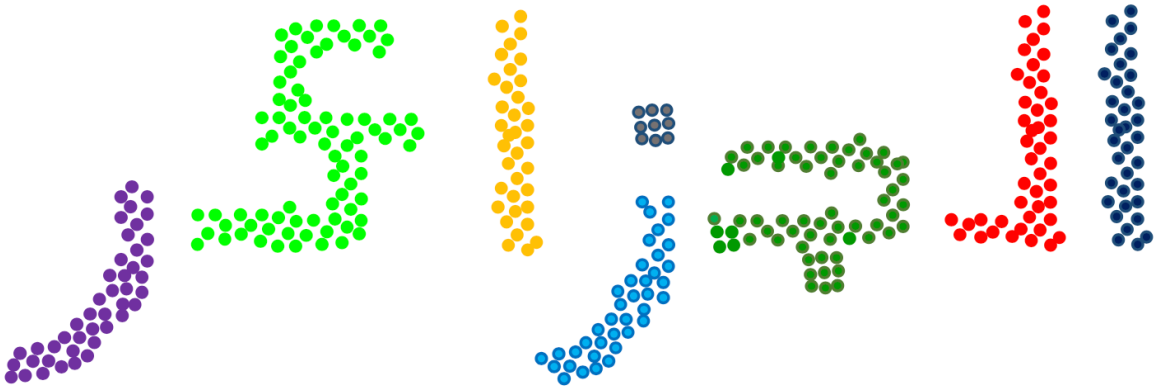

Figure 1.6: Clustering par densité.

La (Figure 6) représente le clustering des données par densité dans un espace bidimensionnel, chaque couleur représente un cluster différent.

Il existe deux types de méthodes basées sur la densité [68], [69]:

- Les méthodes connectives
- Les méthodes basées sur une fonction de densité

8.4 Méthodes basées sur la grille

Ces méthodes consistent à diviser l'espace de données en un nombre fini de cellules formants une grille, puis elles forment des clusters à partir des cellules denses. Ces méthodes ont été initialement proposées par Warnekar et krishma [32] pour organiser l'espace d'attributs, et leur popularité a augmenté après l'introduction de STING (STatistical Information Grid), CLIQUE (CLustering In Quest), et WaveCluster. L'avantage principal de ces méthodes est leur temps de traitement rapide, qui est indépendant du nombre d'objets, mais dépend uniquement du nombre de cellules de chaque dimension de l'espace quantifié [33], [34].

9. Structures de données

Dans la littérature, il est aisé de trouver plusieurs définitions de métrique de distance.

Cependant, toutes ces mesures ont une interprétation identique.

Dans ce qui suit, nous allons passer en revue les principales mesures de distance utilisées en clustering.

- Matrice de données

$$\begin{bmatrix} X_{11} & \cdots & X_{1f} & \cdots & X_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ X_{i1} & \cdots & X_{if} & \cdots & X_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ X_{n1} & \cdots & X_{nf} & \cdots & X_{np} \end{bmatrix}$$

- Matrice de similarité

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ : & : & : & & \\ d(n,1) & d(n,2) & \cdots & \ldots & 0 \end{bmatrix}$$

10. Fonction de proximité

- Métrique pour la similarité : La similarité est exprimée par le biais d'une mesure de distance
- Une autre fonction est utilisée pour la mesure de la qualité
- Les définitions de distance sont très différentes que les variables soient des intervalles (continues), catégories, booléennes ou ordinales
- En pratique, on utilise souvent une pondération des variables

- Similarité entre objets Les distances expriment une similarité

  Ex: la distance de Minkowski :

$$d(i,j) = \sqrt[q]{(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \cdots + |x_{ip} - x_{jp}|^q)}$$

où $i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ et $j = (x_{j1}, x_{j2}, , \ldots, x_{jp})$ sont deux objets p-dimensionnels et q un entier positif

Si q = 1 d est la distance de Manhattan

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \cdots + |x_{ip} - x_{jp}|$$

10.1 Similarité entre objets(I)

Si q = 2, d est la distance Euclidienne :

$$d(i,j) = \sqrt{(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \cdots + |x_{ip} - x_{jp}|^2)}$$

Propriétés :

$d(i,j) \geq 0$

$d(i,i) = 0$

$d(i,j) = d(j,i)$

$d(i,j) \leq d(i,k) + d(k,j)$

Exemple: distance de manhattan

	Age	Salaire
Personne1	50	11000
Personne2	70	11100
Personne3	60	11122
Personne4	60	11074

$d(p1, p2) = 120$

$d(p1, p3) = 132$

Conclusion: p1 ressemble plus à p2 qu'à p3

10.2 Variables binaires

Une table de contingence pour données binaires

		Objet j		
		1	0	sum
Objet i	1	a	b	a+b
	0	c	d	c+d
	sum	a+c	b+d	p

a= nombre de positions où i a 1 et j a 1

Exemple $oi = (1,1,0,1,0)$ et $oj = (1,0,0,0,1)$

$$a = 1, b = 2, c = 1, d = 2$$

11. Mesures de distances

Coefficient d'appariement (matching) simple (invariant pour variables symétriques):

$$d(i,j) = \frac{b + c}{a + b + c + d}$$

Exemple $oi = (1,1,0,1,0)$ et $oj = (1,0,0,0,1)$

$d(oi, oj) = 3/5$

. Coefficient de Jaccard $d(i,j) = \frac{b+c}{a+b+C}$

$d(oi, oj) = 3/4$

11.1 Variables binaires (I)

. Variable symétrique : Ex. le sexe d'une personne, i.e coder masculin par 1 et féminin par 0 c'est pareil que le codage inverse

. Variable asymétrique : Ex. Test HIV. Le test peut être positif ou négatif (0 ou 1) mais il y a une valeur qui sera plus présente que l'autre.

. 2 personnes ayant la valeur 1 pour le test sont plus similaires que 2 personnes ayant 0 pour le test

11.2 Variables binaires(II)

. Exemple

Nom	Sexe	Fièvre	Toux	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

- Sexe est un attribut symétrique
- Les autres attributs sont asymétriques
- Y et P ≡ 1, N ≡ 0, la distance n'est mesurée que sur les asymétriques

$$d(jack, mary) == \frac{0+1}{2+0+1} == 0.33$$

$$d(jack, jim) == \frac{1+1}{1+1+1} == 0.67$$

$$d(jim, mary) == \frac{1+2}{1+1+2} == 0.75$$

Les plus similaires sont Jack et Mary⇒atteints du même mal

11.3 Variables Nominales

- Une généralisation des variables binaires, ex: rouge, vert et bleu
- Méthode 1: Matching simple

  m: # d'appariements, p: # total de variables $d(i,j) == \frac{p-m}{p}$
- Méthode 2: utiliser un grand nombre de variables binaires
  - Créer une variable binaire pour chaque modalité (ex: variable rouge qui prend les valeurs vrai ou faux)

11.4 Variables Ordinales

- Une variable ordinale peut être discrète ou continue
- L'ordre peut être important, ex: classement
- Peuvent être traitées comme les variables intervalles remplacer x_{if} par son rang r_{if} $\in\{1,...,Mf\}$
- Remplacer le rang de chaque variable par une valeur dans [0, 1] en remplaçant la variable f dans l'objet I par

$$z_{if} = \frac{r_{if}\_1}{m_f\_1}$$

- Utiliser une distance pour calculer la similarité

12. Mesures d'évaluation et de performance

Dans la littérature, une grande variété d'algorithmes ont été proposés pour différentes applications et tailles d'ensemble de données. Tant que le clustering de données est un processus non supervisé, alors pas de classes prédéfinies ni d'exemples pouvant montrer la validité des résultats obtenus [35], Donc il est important d'utiliser des mesures spéciales d'évaluation et de performance pour mesurer la qualité d'un résultat de clustering. Les mesures d'évaluation sont majoritairement utilisées pour juger la qualité d'un résultat de clustering.

Il existe trois objectifs [36]: la séparabilité, la connectivité, et la compacité.

1. • La séparabilité des clusters signifie qu'ils sont séparés par paires, et pour chaque paire de clusters, il existe un hyperplan séparant les deux clusters dans un espace à d dimensions.
2. • La connectivité est le degré auquel les objets adjacents sont placés dans le même cluster, qui est défini par des algorithmes de voisinage, les plus couramment utilisés sont : les algorithmes KNN (k-plus proches voisins), ε-voisinage et NC.

3. • L'étanchéité du cluster est caractérisée par la densité d'objets autour du centre du cluster.[37]

12.1 Indices de validité des clusters

La qualité des résultats de regroupement peut être évaluée par des indicateurs de validité, qui sont divisés en trois types : La première catégorie représente les indicateurs de validité externes, dans lesquels l'évaluation des résultats est basée sur des impositions prédéfinies sur les données. Dans le second type, l'évaluation de clustering est basée sur les termes de quantité du vecteur d'ensemble de données entrelacé lui-même (comme une matrice de proximité), ce type est appelé validation interne. Dans le dernier type, l'évaluation se fait en appliquant le même algorithme au même ensemble de données mais avec des valeurs de paramètres différentes ou en comparant des structures obtenues par différents algorithmes de clustering, ce type est appelé validation relative [38].

12.1.1 Indice de Davies et Bouldin

Cet indice [39] peut être défini comme une combinaison de mesures d'homogénéité et de séparation :

- L'homogénéité

$$H_{i=\frac{1}{n_i}} \sum d(x, \mu_i)$$

Tels que : μ_i est le centre du cluster i, et d(x, μ_i) est la distance entre l'objet x et le centre μ_i,et n_i la taille du cluster i

- La séparation :

$$S_{(i,j)} = d(\mu_i, \mu_j)$$

Tel que : d(μi, μj) est la distance entre le centre du cluster i et le centre du cluster j. La fonction du Davies et Bouldin [39] alors est définie comme :

$$DB = \frac{1}{k} \sum_{i=1}^{k} DB_i$$

Tel que :

$$DB_i = \max_{1\leq j\leq k, i\neq j} \frac{H_i + H_j}{S_{i,j}}$$

12.1.2 Indice de Dunn

L'indice de Dunn [40] est désigné pour trouver les clusters compacts et les biens séparés [41]:

$$D_{n_c} = \min_{i=1\ldots,n_c} \left\{ \min_{j=i+1,\ldots,n_c} \left(\frac{d(c_i, c_j)}{\max_{k=1\ldots n_c} diam(c_k)} \right) \right\}$$

Tel que :

$d(c_i, c_j)$ est la distance entre deux clusters c_i et c_j définie comme :

$$d(c_i, c_j) = \min_{x\in C_i, y\in C_j} d(x, y)$$

$d(c_k)$ est le diamètre du cluster ck défini comme :

$$diam(C) = \max_{x,y\in C} d(x, y)$$

12.2 Indice de coefficient de silhouettes

Cet indice peut être aussi défini comme une combinaison de deux mesures, la cohésion et la séparation :

- La cohésion :

$$C(x_i) = \frac{1}{n_i - 1} \sum_{y\in c_i, y\neq x_i} d(x_i, y)$$

- La séparation :

$$SP(x_i) = \min_{i\neq j} \frac{1}{n_i} \sum_{y\in c_j} d(x_i, y)$$

Tel que : d(x, y) est la distance entre l'objet x et y, et ni la taille du cluster i.

L'indice de coefficient de silhouette [42], [43] est défini comme :

$$S(k) = \frac{1}{n}\sum_{i=1}^{n} S(x_i)$$

Tel que : n est la taille de l'ensemble de données, et k le nombre des clusters, et (xi) est définie comme suit :

$$S(x_i) = \frac{Sp(x_i) - C(x_i)}{\max(C(x_i), Sp(x_i)}$$

Cet indice est utilisé pour sélectionner la meilleure valeur du k (nombre des clusters), par le choix du k qui majoré l'indice (k) [43], [42].

Ces trois premiers indices sont internes.

12.3 Indice de Fowlkes-Mallows

Cet indice a été proposé par Fowlkes et Mallows [44] comme une mesure pour comparer deux résultats de clustering hiérarchique. Cependant, il est possible de l'utiliser pour des résultats de clustering non-hiérarchique. Pour bien définir cet indice, il faut d'abord définir les notions suivantes [45]:

- X l'ensemble d'objets de taille $|X| = n$
- $C = \{C_1, C_2, \dots C_k\}$ est un ensemble non-vide de sous-groupes disjoints de X tel que l'union de tous les sous-groupes donne à X:$\cup_i^k = {}_1C_i$.
- $C' = \{C'_1, C'_2, \dots, C'_p\}$ est un autre ensemble de partitions d'un autre clustering du même ensemble X.
- $M = (m_{ij})$ est la matrice de confusion des paires C, C', m_{ij} est le nombre des éléments de l'intersection des deux clusters C_i et C'_j:

$$m_{ij} = \left|C_i \cap C'_j\right|, 1 \le i \le k, 1 \le j \le$$

L'indice de Fowlkes et Mallows peut alors être défini comme suit [44],[45] :

$$FM(C, C') = \frac{\sum_{i=1}^{k}\sum_{j=1}^{p} m_{i,j}^2 - n}{\sqrt{(\sum_i |C_i|^2 - n)\left(\sum_j \left|C'_j\right|^2 - n\right)}}$$

Cet indice est un indice relatif.

12.4 Indice de variance intra cluster (SSE)

Cet indice est basé sur le concept de la minimisation des distances entre chaque centre de cluster et les autres objets du même cluster (distance intra cluster) [46], [47] :

$$Var(C) = \sum_{C_i \in C} \sum_{x \in C_i} d(x - \mu_i)^2$$

Cet indice est un indice interne.

12.5 Indices externes

Pour les indices externes, il nous faut une connaissance des vrais résultats (on a les objets de chaque classe).

- L'homogénéité : est l'indice qui vérifie si tous les objets d'un cluster sont de la même classe, il est donné par

$$Homogeneity = 1 - \frac{H(C|L)}{H(C)}$$

- La complétude : cet indice vérifie si tous les objets d'une classe sont assignés au même cluster, cet indice est donné par

$$Completeness = 1 - \frac{H(L|C)}{H(L)}$$

où $(C|L)$ est l'entropie conditionnelle des classes compte tenu des assignations des clusters, elle est donnée par : $H(C|L) = -\sum_{i=1}^{k}\sum_{j=1}^{q}\frac{n_{i,j}}{n}.\log\left(\frac{n_{i,j}}{n_j}\right)$

(C) représente l'entropie des classes, il est donné par :

$$H(C) = -\sum_{i=1}^{k}\frac{n_i}{n}.\log\left(\frac{n_i}{n}\right)$$

où C représente l'ensemble des classes et L représente l'ensemble des clusters, k est le nombre des clusters, q est le nombre des classes, n est le nombre total des objets, n_i et n_j sont respectivement la taille de la classe i et la taille du cluster j, et ni, représente le nombre des objets de la classe i assignés au cluster j.

L'entropie conditionnelle des clusters compte tenu des assignations des classes $(L|C)$ et l'entropie des clusters $H(L)$ sont définies symétriquement.

- V Mesure : cet indice est défini par les deux indices précédents comme suit :

$$VMeasure = 2.\frac{Homogeneity.Completeness}{Homogeneity + Completeness}$$

- La pureté : est le degré de points de données qui sont classifiés correctement, elle est donnée par :

$$Purity = \frac{1}{n}\sum_{n=1}^{k} n_{i,j}$$

- Précision : cet indice est donné par :

$$Precision(i,j) = \frac{n_{i,j}}{n_j}$$

- Rappel : cet indice est donné par :

$$Recall(i,j) = \frac{n_{i;j}}{n_i}$$

- G Mesure : cet indice est donné par :

$$GM(i,j) = \sqrt{Precision(i,j) * Recall(i,j)}$$

- F-Measure : cet indice est donné par :

$$F - measure = \frac{1}{n}\sum_{j=1}^{k} n_j \max_{t=1\ldots q} F(L_j, C_t)$$

Tel que :

$$F(L_j, C_t) = \frac{2 * Precision(i,j) * Recall(i,j)}{Precision(i,j) + Recall(i,j)}$$

Une grande valeur de ces mesures est nécessaire pour un bon clustering.

13. Conclusion

Dans ce chapitre, nous avons élaboré avec détails le clustering, qui est une des techniques statistiques largement utilisées dans la fouille de données. Il est dans un cadre d'apprentissage non supervisé, qui tente d'obtenir des informations sans aucune connaissance préalable, ce qui n'est pas le cas de l'apprentissage supervisé ainsi nous avons présenté les méthodes et les différentes fonctions de proximité, ainsi que les indices de validité dans la littérature pour le résoudre

Dans le chapitre suivant, nous allons présenter le développement orienté agents que nous allons ensuite utiliser pour traiter un problème de clustering.

Chapitre 2 : Les agents

1. Introduction

L'informatique est en train de changer de manière assez profonde et l'évolution des domaines d'application sont devenus complexe avec les systèmes d'intelligence artificielle (IA). Le thème agents et systèmes multi-agents (SMA) est actuellement un champ de recherche très actif, et l'objectif de ces recherches est de donner plus d'autonomie et d'initiative.

Ce chapitre présente les principales notions d'agent et les systèmes multi agents (SMA).

2. Concept d'agent

2.1. Définition d'agent

Il n'existe pas encore une définition commune d'un agent, pour cela et pour avoir une bonne vision de ce concept voici quelques définitions.

D'après *Ferber* : « un agent est une entité autonome, réelle ou abstraite qui est capable d'agir sur elle-même et sur son environnement qui dans un univers multi agent, peut communiquer avec d'autres agent, et dont le comportement est une séquence de ses observations, de ses connaissances, et de ses interactions avec d'autres agents » [70].

D'après *Yves Demazeau* : « un agent est une entité réelle ou virtuelle dont le comportement est **autonome**, évoluant dans un environnement qu'il est capable de percevoir et sur lequel il est capable d'agir et d'interagir avec les autres agents » [71].

Jennings, *Sycara* et *Wooldridge* ont proposé 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 pour lesquels il a été conçu ». [72]

Les notions "situé", "autonomie" et "flexible" sont définies comme suit :

Situé : l'agent est capable d'agir sur son environnement à partir des entrées qu'il perçoit de celui-ci.

Autonomie : l'agent est capable de contrôler ses propres actions ainsi que son état interne, et d'agir sans l'intervention d'un tiers (humain ou agent).

Flexible : l'agent dans ce cas est capable de répondre à temps et ayant les propriétés suivantes :

- **Réactif** : Un agent peut percevoir l'environnement dans lequel il est situé, et réagir en conséquence aux changements portés à cet environnement et élaborer une réponse dans le temps requis.
- **Proactif** : l'agent doit avoir un comportement opportuniste et être capable de prendre l'initiative au bon moment.
- **Social** : l'agent doit être capable d'interagir et de coopérer avec les autres agents (logiciels et humains).

2.2. Comportement de l'agent

Un agent se caractérise essentiellement par la manière dont il est conçu et par ses actions. En d'autres termes par son architecture et son comportement [85]. Ce dernier terme est fondamental dans la définition et la modélisation d'un agent.

Le comportement caractérise l'ensemble des propriétés que l'agent manifeste dans son environnement. On peut le comprendre par les réponses de l'agent aux sollicitations de son environnement ou en regardant sa manière d'évoluer. Il est analysable sans connaître les détails d'implémentation. Il s'agit d'un phénomène qui peut être appréhendé par un observateur extérieur qui au regard des actions qu'entreprend l'agent, peut induire ou spécifier ce qu'une architecture est censée produire.

2.3. Types d'agent

Les agents sont classés en plusieurs types, selon les architectures, les capacités, et leur environnement, en trois types essentiels qui sont : les agents cognitifs, les agents réactifs, et les agents hybrides.

2.3.1. Les agents cognitifs

Ils disposent d'une base de connaissance comprenant les diverses informations liées à leurs domaines d'expertise et à la gestion des interactions avec les autres agents et leur environnement, qui permet aux agents de communiquer, de collaborer et d'agir, qui donne la particularité d'avoir un raisonnement assez développé c'est-à-dire la décision.

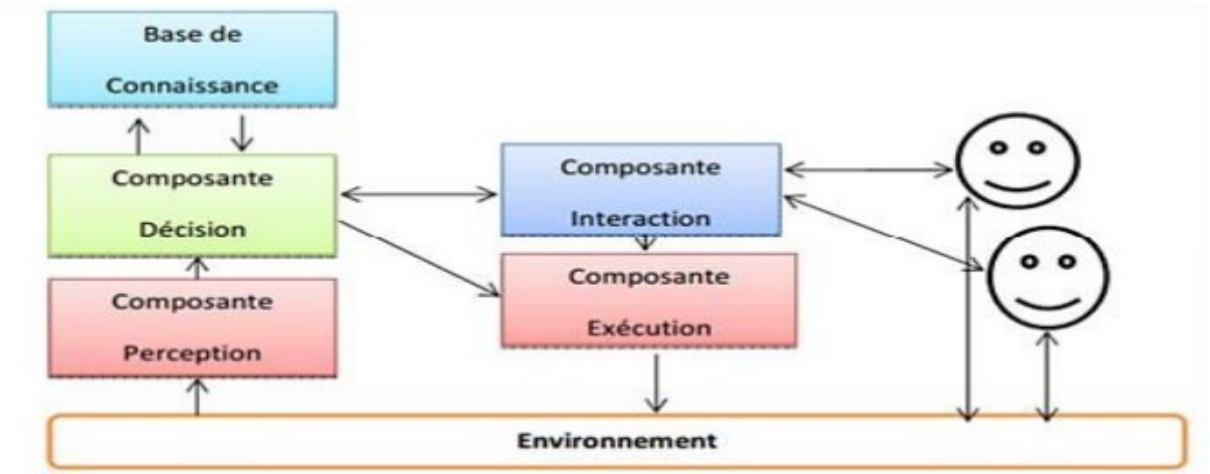

***Figure 2.1.Modèle d'un agent cognitif* [88]**

Ce type d'agent (Figure 2.1) se caractérise par :

- Une représentation explicite de l'environnement et du monde auquel il appartient ;
- Une réaction planifiée ;
- Une base de connaissances comprenant des informations et du savoir-faire ;
- Une mémoire pour mémoriser les anciens états.

2.3.2. Les agents réactifs

Ce sont des agents non intelligents, ils ne disposent que d'un protocole et d'un langage de communication réduit, ils ne peuvent répondre qu'à l'action programmée, ils sont constamment en état de veille sur leur changement d'environnement.

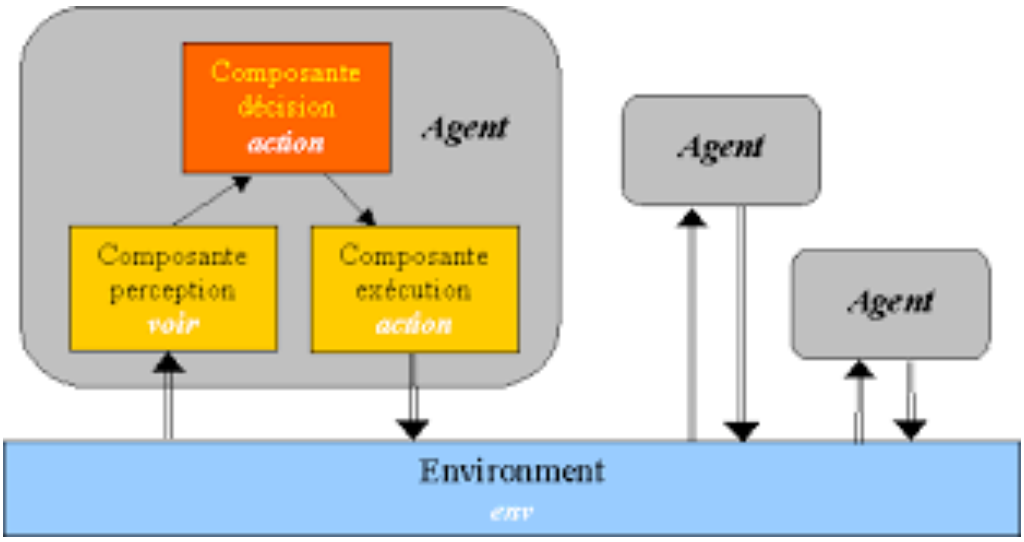

Figure2.2. Modèle d'un agent réactif

L'agent réactif (figure 2.2.) présente les caractéristiques suivantes :

- Pas de mémoire ;
- Pas de représentation explicite de son environnement;
- Prise de décision se basant sur le fait du Stimulus/Réponse ;
- Simple à mettre en œuvre.

2.3.3. Les agents hybrides ou mixtes

Les chercheurs ont essayé de combiner les deux types précédents, ils sont conçus pour allier des capacités réactives à des capacités cognitives pour obtenir une architecture hybride pour la résolution de tous types de problèmes.

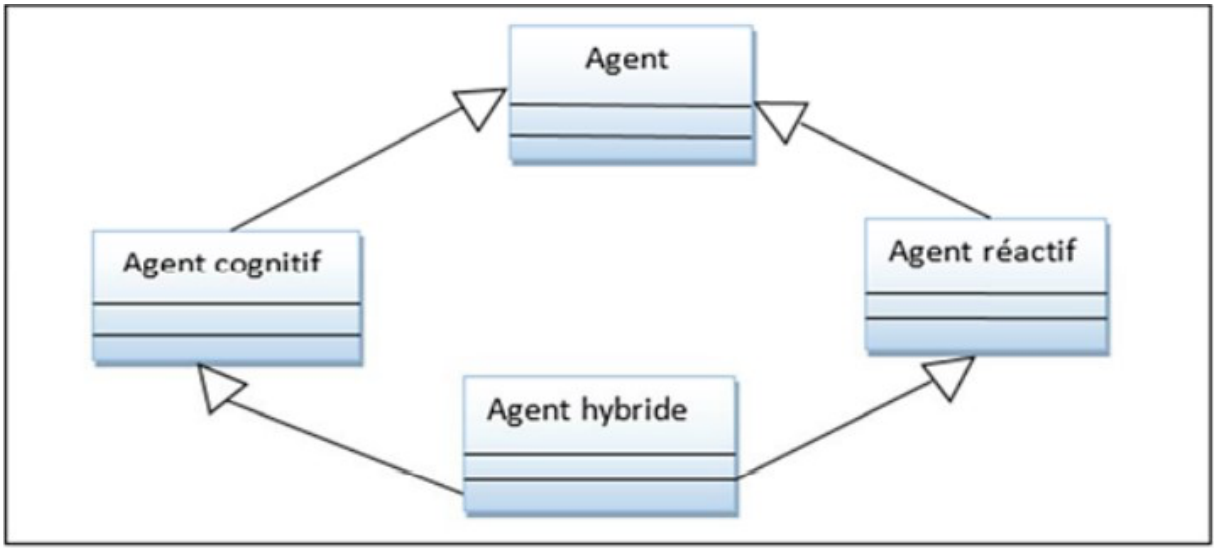

***Figure 2.3.Modèle d'un agent hybride* [88]**

2.4. Structure d'un agent

D'après *Ferber* [85] qui a proposé une structure qu'un agent possède : le savoir-faire, les croyances, le contrôle, l'expertise et la communication.

Savoir-faire : c'est une interface pour la déclaration des compétences et des connaissances de l'agent.

Les croyances : c'est la représentation de l'environnement de l'agent (les autres agents et lui-même) c'est l'agent connait le monde.

Le contrôle : c'est les buts, les intentions, les plans et les taches dans un agent.

L'expertise : c'est la connaissance sur la résolution du problème.

La communication : c'est pourquoi la création d'un langage commun à tous les agents pour garantir une bonne communication et une bonne coordination d'actions.

2.5. Propriétés d'agent

On peut définir les propriétés d'agents à partir des définitions comme suit :

• **Autonomie :** les agents contrôlent leurs actions et leurs états internes. Le système dans son ensemble est capable de réagir sans l'intervention d'un humain ou d'un autre agent.

• **Réactivité :** ils perçoivent leur environnement et réagissent aux changements qui s'y produisent dans le temps requis.

• **Initiative :** le comportement des agents est déterminé par les buts qu'ils poursuivent et par conséquent ils peuvent produire des actions qui ne sont pas seulement des réponses à leur environnement.

• **Habilité sociale :** pour satisfaire ses buts un agent peut demander l'aide d'autres agents avec lesquels il partage la réalisation de tâches.

• **Raisonnement :** un agent peut décider quel but poursuivre ou à quel événement réagir, comment agir pour accomplir un but, ou suspendre ou abandonner un but pour se dédier à un autre.

• **Apprentissage :** l'agent peut s'accommoder progressivement à des changements dans des environnements dynamiques grâce à des techniques d'apprentissage.

• **Mobilité :** dans des applications déterminées il peut être intéressant de permettre aux agents de migrer d'un nœud à un autre dans un réseau tout en préservant leur état lors de sauts entre nœuds.

3. Système multi-agents

3.1. Définition des SMA

Les systèmes multi Agents sont des systèmes distribués conçus et implantés idéalement comme un ensemble d'agents interagissant [102], le plus souvent, selon des modes de coopération, de concurrence et de coexistence.

Un système multi-agent peut être homogène ou hétérogène :

- Homogène : au niveau d'un système multi agent homogènes tous les agents ont la même structure interne, ceci incluant les objectifs, les connaissances des domaines, et les actions possibles. Ces agents choisissent leurs prochaines actions en utilisant la même procédure.
- Hétérogène : les agents peuvent être hétérogènes en ayant des objectifs différents, des modèles de domaines et des actions différentes.

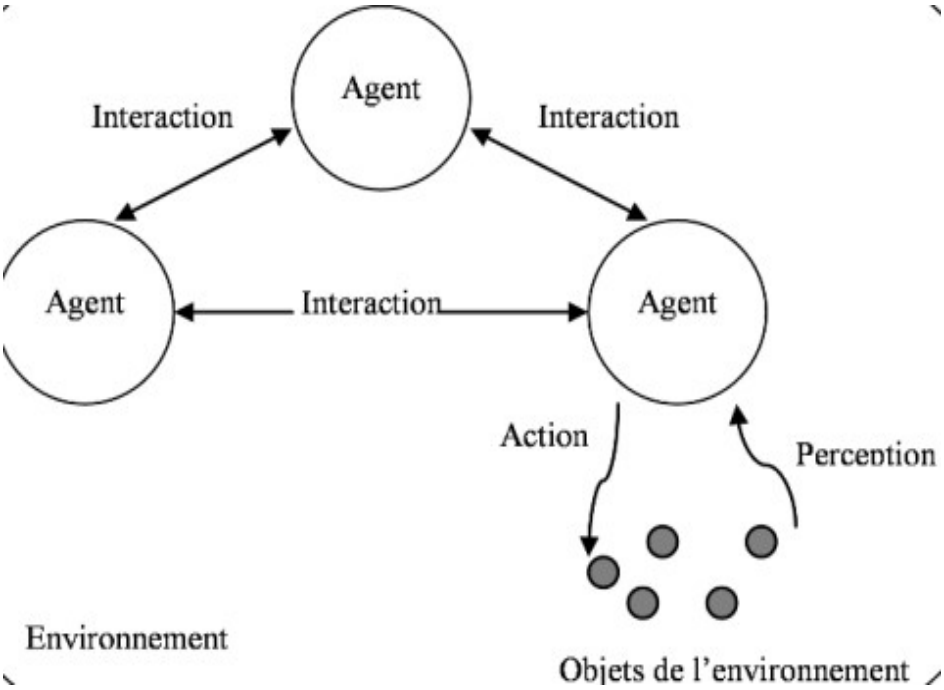

Figure 2.4.Représentation d'un système multi-agent selon Ferber[70]

3.2. Composant d'un système multi-agent

Selon la définition de l'approche Voyelle d'*Yves Demazeau*, un système multi-agent est constitué de quatre briques : **Agents, Environnement, Interaction** et **Organisation**.

Agents : architectures internes des agents on a parlé déjà dans la première partie.

Environnement : L'environnement est un élément important dans le système multi-agents. C'est grâce à lui que les agents peuvent coexister et interagir. L'environnement doit pouvoir

être perçu par les agents et ces derniers doivent pouvoir agir dessus et interagir au travers. Avec celle des comportements individuels, la spécification de l'environnement permet de définir la dynamique d'un SMA. Lorsque les agents sont réactifs, l'environnement détient une importance capitale. En effet, comme ces agents ne peuvent communiquer directement entre eux, il est le médiateur de leurs interactions.

Ils s'influencent mutuellement soit par leur position s'ils sont situés, soit par l'intermédiaire d'objets qu'ils perçoivent et modifient.

Types d'environnement

Selon le point de vue que l'on adopte, différents types d'environnements peuvent être identifiés :

- **Point de vue du système multi-agents :** l'environnement correspond à l'ensemble des entités extérieures au système.
- **Point de vue de l'agent :** l'environnement est tout ce qui est extérieur à lui-même.
- **Point de vue du concepteur :** il peut correspondre à l'état du système, ou représenter l'ensemble des outils permettant de simuler, de visualiser et d'évaluer le SMA.

Propriétés d'environnement

L'environnement possède certaines propriétés

- **Accessible ou inaccessible :** un agent a accès à l'état complet de l'environnement ou non.
- **Déterministe ou indéterministe :** le changement de l'état de l'environnement est uniquement déterminé par l'état courant et les actions des agents ou non.
- **Statique ou dynamique** : l'environnement peut changer quand l'agent est en action (réflexion) ou non.
- **Discret ou continu :** le nombre de perceptions et d'actions est limité ou pas.

En ce qui suit, nous présentons aussi quelques caractéristiques importantes des SMA :

Interaction : à partir des définitions du concept d'agent il existe deux types d'interactions dans les SMA : l'interaction d'un agent avec son environnement et l'interaction d'un agent avec les autres agents du système.

Organisation : L'organisation lie de façon interrelationnelle des éléments ou événements ou individus divers qui dès lors deviennent les composants d'un tout. Elle assure solidarité et

solidité relative, donc assure au système une certaine possibilité de durée en dépit des perturbations aléatoires [75].

3.3. Caractéristiques d'un SMA

Un SMA possède la plupart des caractéristiques suivantes [76] :

• **Distribution :** le système est décomposable, l'élément de base étant l'agent.

• **Décentralisation :** les agents sont indépendants, il n'y a pas de décisions centrales valables pour tout le système.

• **Autonomie :** un agent est en activité permanente et prend ses propres décisions en fonction de ses objectifs et de ses connaissances.

• **Echange de connaissances :** les agents sont capables de communiquer entre eux, selon des langages plus ou moins élaborés.

• **Interaction :** les agents ont une influence localement sur le comportement des autres agents.

• **Organisation :** les interactions créent des relations entre les agents, et le réseau de ces relations forme une organisation qui peut évoluer au cours du temps.

3.4. Coopération dans les SMA

La coopération est une caractéristique essentielle en univers multi-agents pour l'exécution d'une tâche. En effet, chaque agent ne possède qu'une vue partielle de l'environnement auquel il appartient. Grace à la coopération, il peut accomplir cette tâche avec beaucoup plus de performance. Très tôt, les chercheurs se sont intéressés à la formulation de ce concept utilisé pour la résolution du problème et aussi pour que le système ait un fonctionnement optimal [77]. L'aspect collectif lors de la résolution d'un problème et plus exactement la coopération des agents est un point important dans le domaine des SMA. Pour [85], la coopération revêt deux dimensions :

• La coopération peut être une attitude des agents qui décident de travailler en commun,

• Un comportement est qualifié de coopératif à partir des caractéristiques sociaux tels que le nombre de communications effectuées ou l'interdépendance des actions.

Ferber considère que plusieurs agents coopèrent pour atteindre un objectif commun si l'une des deux conditions suivantes est vérifiée :

a. L'ajout d'un nouvel agent accroit différentiellement les performances du groupe.

b. Il existe des conflits potentiels d'accession à des ressources et l'action des agents sert à éviter ou à sortir de tels conflits.

3.5. Coordination dans les SMA

La coordination d'actions est l'ensemble des activités supplémentaires qu'il est nécessaire d'accomplir dans un environnement multi-agents dans la création d'un ordre commun au sens physique. Ainsi, dans une société composée d'agents autonomes qui poursuivent des objectifs individuels, la coordination est une méthode indispensable pour agencer de manière cohérente les interventions de chacun. Elle définit une relation d'ordonnancement et de dépendance entre les actions. La coordination est nécessaire :

- Lorsque les agents ont besoin d'informations et de résultats que d'autres agents peuvent fournir ;
- Lorsque les ressources communes sont limitées ;
- Pour éviter des actions inutiles ;
- Pour permettre aux agents de satisfaire des objectifs dépendants ;

De ces faits, le but de la coordination est de trouver parmi un ensemble de comportement d'agents qui interagissent, une collection de comportement qui réalise d'une façon satisfaisante les objectifs les plus importants des agents.

La coordination de multiples agents est essentielle pour la viabilité des systèmes dans lesquels ces agents partagent des ressources. La pluparts des recherches en intelligence artificielle distribuée se concentrent sur le développement de stratégie de coordination en différé.

Ces stratégies préfabriquées peuvent devenir rapidement inadéquates si le modèle du monde du concepteur du système est incomplet ou incorrect ou si l'environnement change dynamiquement.

L'apprentissage est un mécanisme inestimable qui permet aux agents de développer des stratégies de coordination qui satisfont les demandes des environnements et les exigences des agents individuels.

3.6. Apprentissage dans les SMA

L'apprentissage est un axe de recherche de plus en plus important dans le domaine des SMA car il permet d'avoir des systèmes adéquats, performants et évolutifs. Il devient une nécessité lorsque l'environnement est changeant.

L'apprentissage des agents peut porter sur quatre axes différents [79]:

- **L'apprentissage centré agent** concerne ce qu'un agent peut apprendre sur lui-même ou sur les autres agents. Cet apprentissage porte sur le comportement de l'agent, ses stratégies et ses décisions.

- **L'apprentissage centré environnement** se focalise sur ce que l'environnement peut apprendre à l'agent. Cela peut porter sur de nombreux objets tant la diversité des environnements peut être grande. En effet, dans un environnement à forte dynamique, un agent peut apprendre sur des parties lui étant nouvellement apparues. Ce type d'apprentissage se retrouve naturellement dans des systèmes de vie artificielle.

- **L'apprentissage centré interaction** porte sur les moyens mis en œuvre par les agents pour communiquer ou interagir. Cela peut être l'apprentissage de nouveaux langages d'interactions ou bien de nouveaux thèmes de communication.

- Enfin, **l'apprentissage centré organisation** s'occupe de faire évoluer les rôles des agents au sein de leur société. Lorsque le système est supervisé, les agents sont guidés dans leur démarche.

Pour effectuer ces différents types d'apprentissage, plusieurs moyens ont été développés. L'apprentissage peut être la conséquence d'attribution directe ou indirecte de récompenses aux agents.

Il peut aussi être induit par l'interaction et la coopération au sein d'un groupe. Ce dernier consiste en la capacité d'un agent à apprendre grâce aux interactions coopératives qu'il a avec ses congénères. Les agents doivent alors avoir à leur disposition des moyens de communication et des protocoles associés à l'apprentissage.

3.7. La communication dans les SMA

La communication dans les SMA c'est un élément important parce que c'est la base de toute interaction et de l'organisation, Une communication peut être définie comme une forme

d'action locale d'un agent vers d'autres agents [90]. Par conséquent la communication entre agents est de deux types direct ou indirect.

3.7.1. Communication directe

Elle est propre aux agents cognitifs [103], Communication par envoi de message on considère deux cas [80]:

- Envoi synchrone de messages (le message est traité dès sa réception) ;
- Envoi asynchrone de message (le message peut être conservé à un traitement ultérieur).

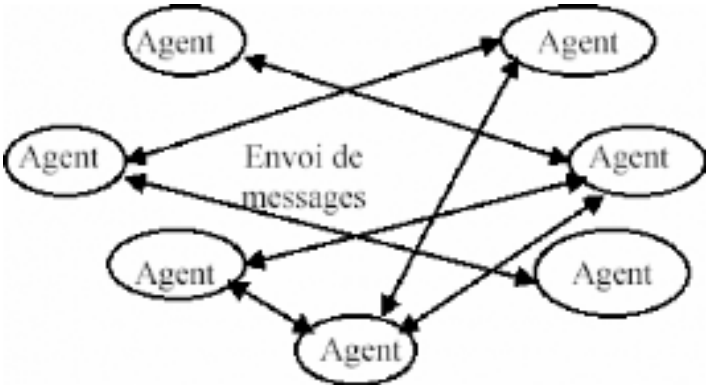

Figure 1.5.Communication par envoi de message

a suppri

3.7.2. Communication indirecte

Elle est utilisée pour les agents réactifs, la communication se fait via un tableau noir ou bien à travers l'environnement.

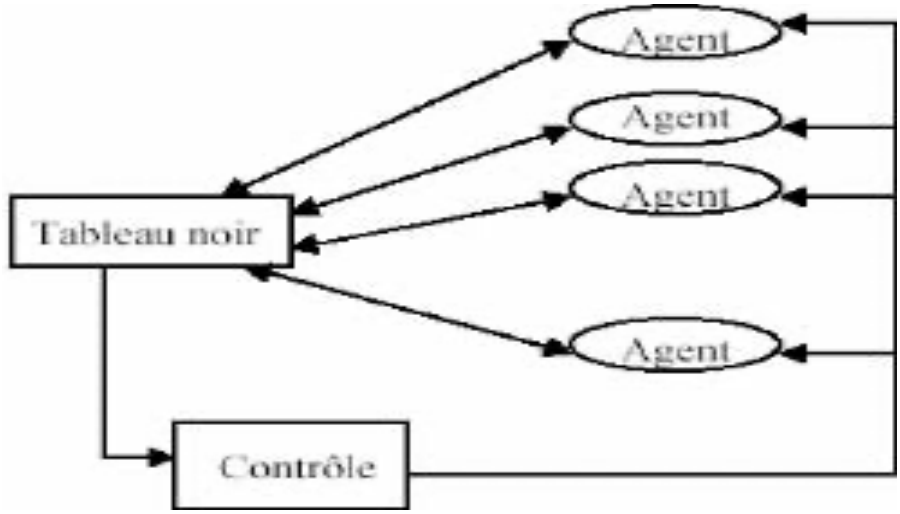

Figure 2.6.Communication par tableau noire

3.8. Les langages de communication

La nécessité d'un langage commun à l'ensemble des agents pour la communication entre eux, pour pouvoir agir dans un environnement commun et éventuellement résoudre des problèmes, pour cela il existe deux langages de communication multi-agent les plus répandus sont : KQML (knowledge Query and Manipulating Language) et FIPA-ACL (Agent Communication Language de FIPA) [83] .

3.8.1. Le langage KQML est un langage qui :

- Support la communication inter agents ;
- Possède une quarantaine de performatifs et de règles ;
- Les performatifs sont des commandes qui ont une certaine ressemblance avec des verbes utilisée de façon performatives dans le langage naturel ;
- Les types de message sont des assertions, instructions de routage, commandes persistantes….

3.8.2. Le langage FIPA-ACL

Le langage FIPA-ACL est un langage basé sur les actes de langage comme le langage KQML, FIPA-ACL est plus riche au niveau de la sémantique, était développé pour répondre aux critiques sur KQML.

3.9. Domaine d'application des SMA

Dans le monde de la recherche : On distingue généralement trois types d'utilisation : La simulation de phénomène complexes, la résolution de problèmes, et la conception de programmes.

- **Dans l'industrie :** L'automatisation des processus de production, la logistique, les robots coopératifs, maison intelligente.
- **Dans l'information :** L'assistance personnelle, la recherche d'information, la gestion du Works flow, maison intelligente.
- **Dans la simulation :** C'est un système constitué d'agents virtuels qui simulent des actions physiques / sociales, citons comme exemple la simulation des processus industriels [85].

3.10. Prise de décision dans les SMA

La prise de décision dans les SMA fait intervenir les deux notions importantes d'observabilité et d'incertitude qui seront développé ci-dessous.

3.10.1. Observabilité

L'observabilité d'un environnement caractérise l'ensemble des informations qui sont accessibles à un agent. Suivant ces informations, nous pouvons distinguer différents types

d'observabilité. Un agent qui exhibe un comportement de localité et d'imprécision des perceptions, introduit la notion d'observabilité partielle. Dans ce cas, les agents seront contraints de faire face à ce manque de connaissances et devront décider au mieux comment agir en fonction des informations dont ils disposent.

Nous différencierons tout d'abord l'observabilité de l'état global du système et l'observabilité de l'état local de l'agent. Le terme état global désignera l'état du système multi-agents (agents et environnement). L'état local d'un agent sera défini selon le degré de délibération de l'agent. Il pourra correspondre aux perceptions courantes de l'agent ou bien à une représentation interne de ses connaissances.

Le degré d'observabilité influence la complexité de la prise de décision. Intuitivement, si les états de l'environnement sont connus alors il est assez facile de décider comment agir. Au contraire, Si nous ne disposons que d'informations partielles sur les autres agents, nous sommes contraints d'envisager leurs comportements possibles, ce qui rend la décision très complexe.

3.10.2. Incertitude

Il existe plusieurs causes d'incertitude dans un système multi-agents [79]. Parmi ces causes, on peut citer :

- L'imprécision et les limitations des capteurs,
- La modélisation incomplète de l'environnement,
- La dynamique des interactions entre agents.

L'imprécision des capteurs entraîne des incertitudes sur l'information perçue par l'agent et par conséquent sur l'environnement. En fait, tout ce qui est hors de portée des capteurs se trouve alors non-observable et par conséquent incertain.

L'incomplétude de la modélisation de l'environnement, entraînent par ailleurs des incertitudes sur le résultat des actions des agents. L'incomplétude de la modélisation est essentiellement due au concepteur. Ce dernier peut, en effet, avoir une connaissance incomplète de tous les phénomènes et lois de l'environnement.

Par ailleurs, les agents évoluent dans un même environnement. Leurs actions peuvent donc entrer en interaction, conduisant alors à des effets non souhaités ou non envisagés. Le fait qu'un agent ne connaisse pas les actions entreprises par les autres agents ajoute donc un degré d'incertitude sur le résultat de ses actions.

Pour qu'un agent évolue dans un environnement déterministe, il serait nécessaire que le système soit parfaitement modélisé, que les capteurs et effecteurs des agents soient d'une précision irréprochable et que l'agent connaisse exactement les états et comportements des autres agents. Il paraît évident qu'un tel système est difficilement possible.

3.10.3. Délibération et décision

L'observabilité et l'incertitude sont à l'origine de nombreuses difficultés posées lors de la prise de décision dans les systèmes multi-agents. Lorsque l'observabilité est partielle et le résultat des actions incertain, il est d'autant plus difficile pour les agents de coopérer ou de se coordonner.

Il est possible de contourner le problème en faisant appel à une entité centrale prenant toutes les décisions et coordonnant les actions de chacun. Dans certaines applications, une telle solution est cependant impraticable. Chaque agent devra donc décider de manière autonome comment agir dans un environnement incertain tout en faisant face au manque d'observabilité.

3.11. Les méthodologies SMA

Les chercheurs dans ce domaine se sont basés sur des outils, méthodes et des modèles déjà existants, pour proposer une démarche qui permettra au concepteur d'être guidé afin de passer d'un cahier de charge à une implémentation et de pouvoir gérer le cycle de vie globale d'une application, ces méthodes peuvent être classées selon trois axes :

1) Les extensions d'approche orienté objet : AAII, Gaia, Aalaadin, ADELFE.
2) Les approches drivées de l'ingénierie de la connaissance : DESIRE, MAS-commonKADS.
3) Les méthodes moins classiques : voyelles.

3.11.1. La méthodologie ADELFE

ADELFE (Atelier pour le Développement de Logiciels à Fonctionnalité Emergente) est une méthodologie dédiée à la conception des systèmes adaptatifs, sont utilisées lorsque l'environnement est imprévisible ou bien le système est ouvert. Les agents modélisés par ADELFE sont coopératifs. [84].

3.11.2. La méthodologie DESIRE

DESIRE c'est une méthode qui manipule les structures des connaissances, les décompositions, l'ordonnancement et la délégation des taches, les échanges de l'information, et c'est une méthode directement issue de l'ingénierie des connaissances.

3.11.3. La méthodologie Voyelle

Est une méthode de haut niveau, repose sur des principes purement multi-agent, dans cette méthode la priorité est la liberté totale de choix ce qui problématique lorsque l'on veut concevoir des systèmes, mais il y'a le problème de quel modèle utilise, et comment faire la décomposition et quels outils choisir.

- **Le modèle AGR** (Agent/Groupe/Rôle) La première organisation d'agents présentée est nommée AGR pour Agent-Groupe-Rôle définie dans [107] . Deux niveaux d'abstraction organisationnelle sont pris en compte : l'organisation et le groupe. Ce modèle repose sur les composants suivants [80]:

Agent : est une entité autonome communicante qui joue des rôles au sein de différents groupes.

Groupe : représente un regroupement d'agents qui associé à un ensemble de rôles. Il définit la structuration organisationnelle d'un système multi agents usuel. Chaque agent peut être membre d'un ou de plusieurs groupes.

Rôle : est considéré comme la représentation abstraite d'une fonction, d'un service ou d'une identification d'un agent au sein d'un groupe particulier. Chaque agent peut avoir plusieurs rôles. Un même rôle peut être tenu par plusieurs agents.

- **Le modèle AGRE** (Agent/Groupe/Rôle/Environnement), en AGR, l'environnement était la chaîne de transmission des messages agent-agent et aussi la chaîne par laquelle un groupe rend le travail effectué à l'autre comme il est vu que l'environnement est une toute chaîne d'interaction.

En AGRE on peut dire que l'on souligne une troisième chaîne de communication : l'aire où l'agent se trouve « physiquement ».

Principe d'AGRE

- Les agents sont les unités dans « le petit monde » multi-agent où ils se manifestent leurs existences au travers d'un de ces deux « modes » : l'organisation est un type de monde c'est-à-dire une position dans une organisation signifie une appartenance à un groupe, et le monde physique qui est un espace où les agents perçoivent et agissent grâce à leurs corps.
- Un agent est situé simultanément dans un monde organisationnel et physique.

- Dans le monde organisationnel, l'agent peut jouer plusieurs rôles. Dans le monde physique, l'agent ne peut avoir qu'un seul corps
- Un mode est la voie d'action d'un agent sur un espace d'interaction c'est-à-dire un agent communique seulement s'il joue un rôle dans un groupe. L'environnement est très important car il est le medium entre les agents. Les actions individuelles des agents sont façonnées par un environnement constamment modifié par le collectif d'agents [86]. Dans l'AGRE, est le mécanisme d'auto-organisation qui se réalise tant au niveau organisationnel, à travers des rôles, qu'au niveau physique, à travers le corps. Un agent peut donner support à plus d'un système auto-organisateur en AGRE grâce à l'espace d'interaction ou des types de modes [87].

3.12. Les plateformes de développement

La difficulté de réalisation des SMA est essentiellement due à la complexité de ces systèmes. Afin de rendre leur réalisation plus accessible, des environnements de développement (ou plates-formes multi-agents) de ces systèmes ont donc été construits à partir d'architectures et de langages existants. Les plates- formes multi-agents facilitent le développement de SMA, en mettant à la disposition du concepteur des fonctions de base pour la création d'agents et pour l'interaction entre agents. En outre, les plates-formes offrent la possibilité à un concepteur non spécialiste des SMA, de pouvoir réaliser un tel système.

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

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

3.12.1. Plateforme Jack

Jack est décrit comme étant un environnement pour construire, exécuter et intégrer Des systèmes multi-agents commerciaux, écrite en Java et utilisant une approche orientée composants. Elle est développée par la société australienne Agent Oriented Software Pty. Les agents sont basés sur le modèle BDI (Belief, Desire, Intention) développés à l'Australian Artificiel Intelligence Institute (AAII) [104].

3.12.2. Plateforme MadKit

MadKit est un environnement basé sur la méthodologie Aalaadin ou AGR (agent / groupe/ rôle). L'outil fournit un éditeur permettant le déploiement et la gestion des SMA. La gestion faite via cet éditeur offre plusieurs possibilités intéressantes. L'outil offre aussi un utilitaire pour effectuer des simulations [105].

3.12.3. Plateforme JADE

Jade est une plate-forme Java pour les systèmes multi-agents respectant le standard FIPA. JADE a été développée par l'université de Parme et C-SELT – centre de recherche télécom Italien. Le but de JADE est de simplifier le développement des systèmes multi-agents en assurant la conformité des standards par un ensemble complet de services et agents. En se conformant aux standards FIPA : service de nom, service de pages jaunes, messages transportés et service d'analyse, et une bibliothèque de protocole d'interactions de FIPA, Jade possède trois modules principaux nécessaires aux normes FIPA. Ils sont lancés à chaque démarrage de la plate-forme [106] :

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

4. Conclusion

L'objectif de ce chapitre était d'introduire les différents concepts d'agent et systèmes multi-agents avec un survol de méthodologies et modèles orientées agent. Le chapitre suivant sera consacré à la modélisation de notre système avec le modèle organisationnel AGR et une présentation détaillée de ce dernier avec le langage de modélisation AUML.

Chapitre 3 : Conception

1. Introduction et problématique

Nous nous trouvons souvent confrontés face au problème de clustering des datasets. L'automatisation de cette tâche est de très grande importance. Afin de parvenir à une solution à ce problème, nous présenterons dans ce chapitre la conception de notre vision en l'adaptant avec une approche basé agents pour un clustering de données.

L'agent groupe rôle (AGR)

1.1.Définition

Le modèle AGR est un méta-modèle (langage pour décrire des modèles) organisationnel pour la modélisation des SMA, il est basé sur trois concepts primitifs, Agent, Groupe et Rôle qui sont structurellement connectés et ne peuvent pas être définis par d'autres primitives. Ils satisfont à un ensemble d'axiomes qui unissent ces concepts [91].

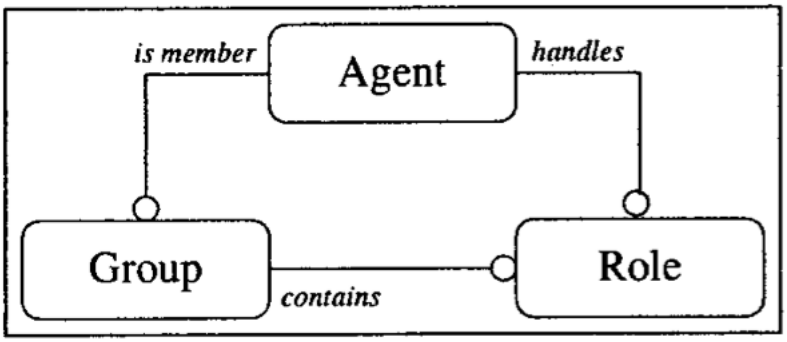

Figure 3.1 : le modèle de base

- **Agent :** un agent est une entité active et communicante jouant des rôles au sein de groupes. Un agent peut avoir plusieurs rôles et être membre de plusieurs groupes. Une caractéristique importante du modèle AGR est qu'aucune contrainte n'est imposée sur l'architecture d'un agent ou sur ses capacités mentales. Un agent peut être aussi réactif qu'une fourmi, ou aussi intelligent qu'un humain, sans aucune restriction.
- **Groupe :** un groupe est un ensemble d'agents partageant une caractéristique commune, un groupe est utilisé comme contexte pour un modèle d'activités, et est utilisé pour partitionner les organisations. Deux agents peuvent communiquer si et seulement s'ils appartiennent au même groupe, mais un agent peut appartenir à plusieurs groupes. Cette fonctionnalité permettra de définir des structures organisationnelles.
- **Rôle :** le rôle est la représentation abstraite d'une position fonctionnelle d'un agent dans un groupe. Un agent doit jouer un rôle dans un groupe, mais un agent peut jouer

plusieurs rôles. Les rôles sont locaux pour les groupes et un rôle doit être demandé par un agent. Un rôle peut être joué par plusieurs agents.

Les rôles peuvent être décrits comme dans Gaïa [109] par des attributs tels que sa cardinalité (combien d'agents peuvent jouer ce rôle). Il est également possible de décrire des contraintes structurelles entre les rôles. Une contrainte structurelle décrit une relation entre des rôles définie au niveau organisationnel et imposée à tous les agents. Dans AGR nous proposons deux contraintes structurelles : correspondance et dépendance. Une contrainte de correspondance stipule que les agents jouant un rôle joueront automatiquement un autre rôle. Le méta-modèle AGR est représenté figure 1 en UML.

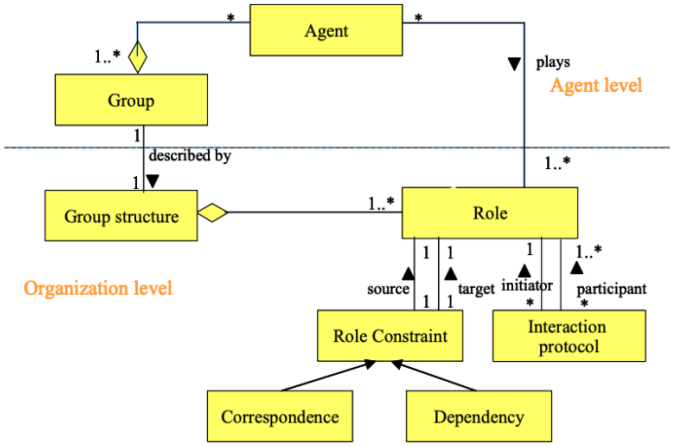

Figure 3.2 : Meta Model UML de l'AGR [91]

1.2.Le diagramme "Cheeseboard"

Dans le diagramme Cheeseboard ou bien plateau de fromages, un groupe est représenté par un ovale qui ressemble à un plateau. Les agents sont représentés comme des quilles qui se dressent sur le plateau et traversent parfois le plateau lorsqu'ils appartiennent à plusieurs groupes [92].

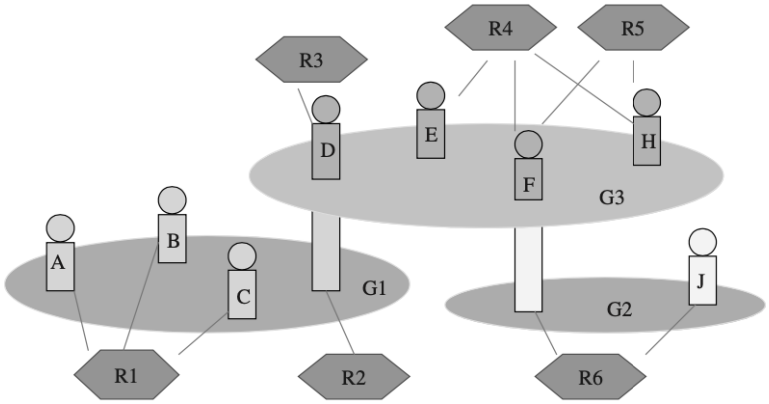

Figure 3.3 : La notation "Cheeseboard" pour décrire des organisations concrètes

Un rôle est représenté par un hexagone et une ligne relie cet hexagone aux agents. La figure donne un exemple d'organisation concrète utilisant le diagramme Cheeseboard. Dans cette image, l'agent F est membre à la fois de G2 et de G3, jouant les rôles R4 et R5 dans G2, et R6 dans G3 [92].

1.3. Structures organisationnelles

La notation cheeseboard, même si elle est très adaptée à l'organisation concrète, elle n'est pas adaptée à la description des relations au sein de l'organisation à un niveau abstrait, c'est-à-dire à la définition des structures organisationnelles. Ainsi, J.Ferber, O.Gutknecht, et F.Michel dans [91] ont introduit une notation pour décrire les structures organisationnelles.

Afin d'exprimer les diagrammes organisationnels d'une manière plus simple et pratique, Ils proposent un ensemble d'objets graphiques.

Dans cette notation, les structures de groupe, c'est-à-dire la représentation abstraite des groupes, sont représentées sous forme de rectangles dans lesquels les rôles, représentés sous forme d'hexagones, sont situés. Les contraintes sont représentées par des flèches entre les rôles. Ils utilisent deux types de flèches. Les grandes flèches sont utilisées pour la correspondance et les flèches fines sont utilisées pour modéliser les dépendances.

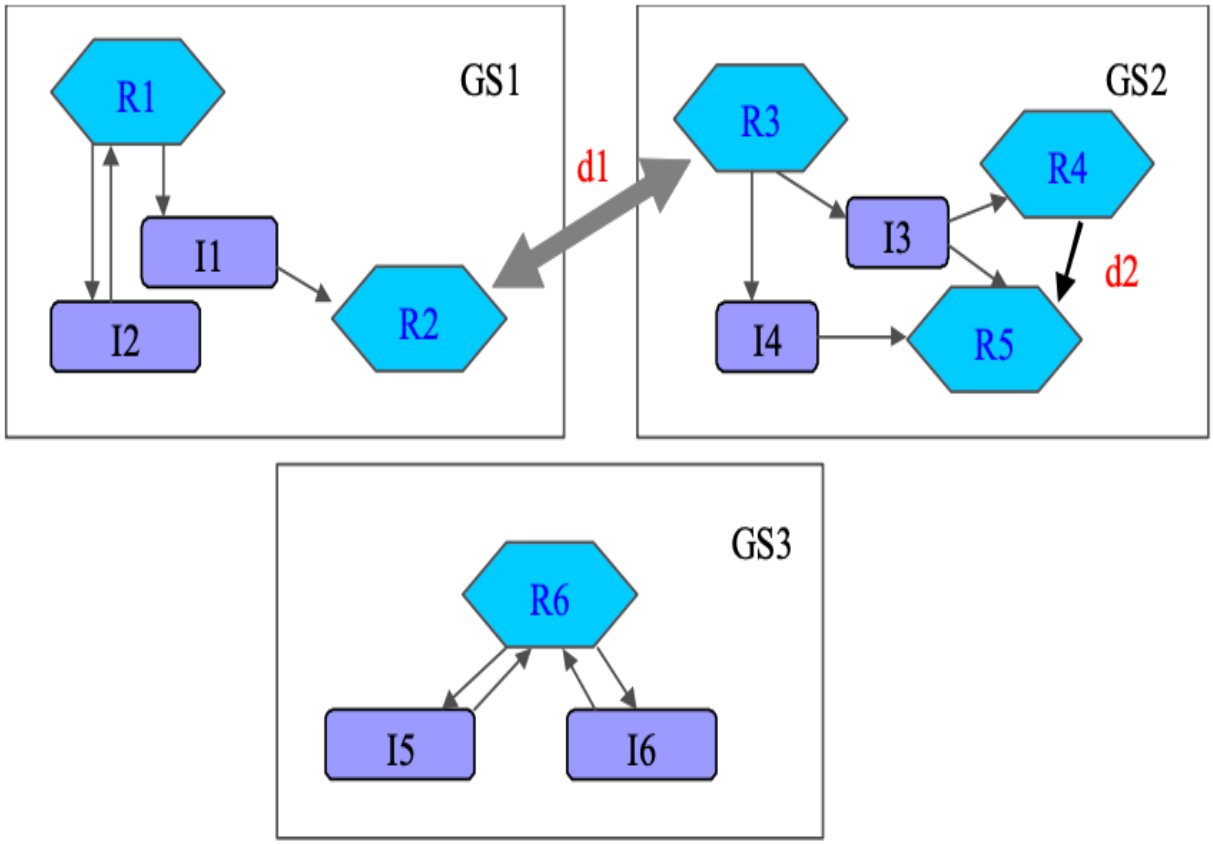

Figure 3.4 : Représentation de la structure organisationnelle

1.4. Les axiomes d'AGRE

- Tout agent est membre (au moins) d'un groupe.
- L'agent communique seulement s'il joue un rôle dans un groupe.
- Dans le monde organisationnel, l'agent peut jouer plusieurs rôles, mais dans le monde physique il ne peut avoir qu'un seul corps
- Les rôles et les corps sont les modes d'existences d'un agent sur un espace d'interaction.
- Un agent est situé simultanément dans un monde organisationnel et physique.
- Les agents sont les unités dans le « petit monde » multi agents ou ils manifestent leur existence à travers un des modes physiques ou organisationnelles.
- Deux agents communiquent seulement s'ils sont membres d'un même groupe.
- Un rôle est défini dans la structure d'un groupe.

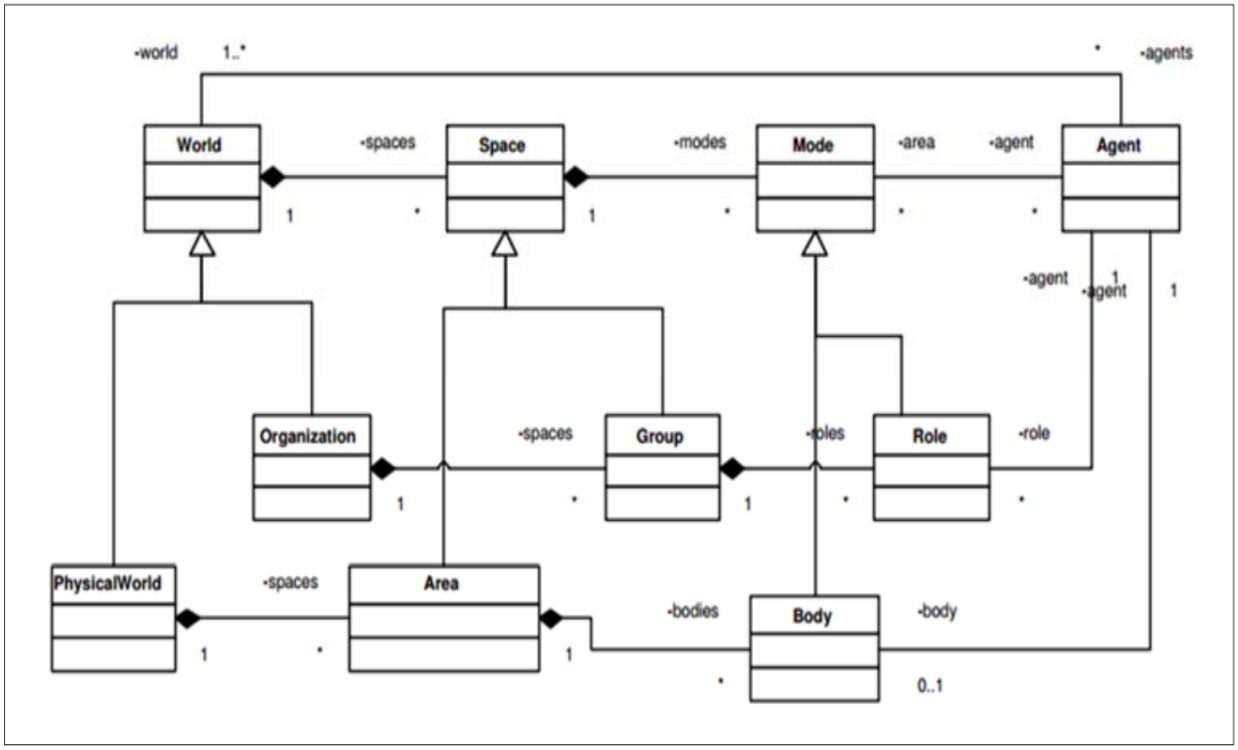

Figure 3.5 : Model UML avec AGRE.

Dans les deux modèles, l'environnement est vu comme toute chaîne d'interaction. Dans ce sens, en AGR, l'environnement était la chaîne de transmission des messages agent-agent et aussi la chaîne par laquelle un groupe rend le travail effectué à l'autre. En AGRE on peut dire que l'on souligne une troisième chaîne de communication : l'aire où l'agent se trouve situé « physiquement ».[88]

2. Modélisation AGR

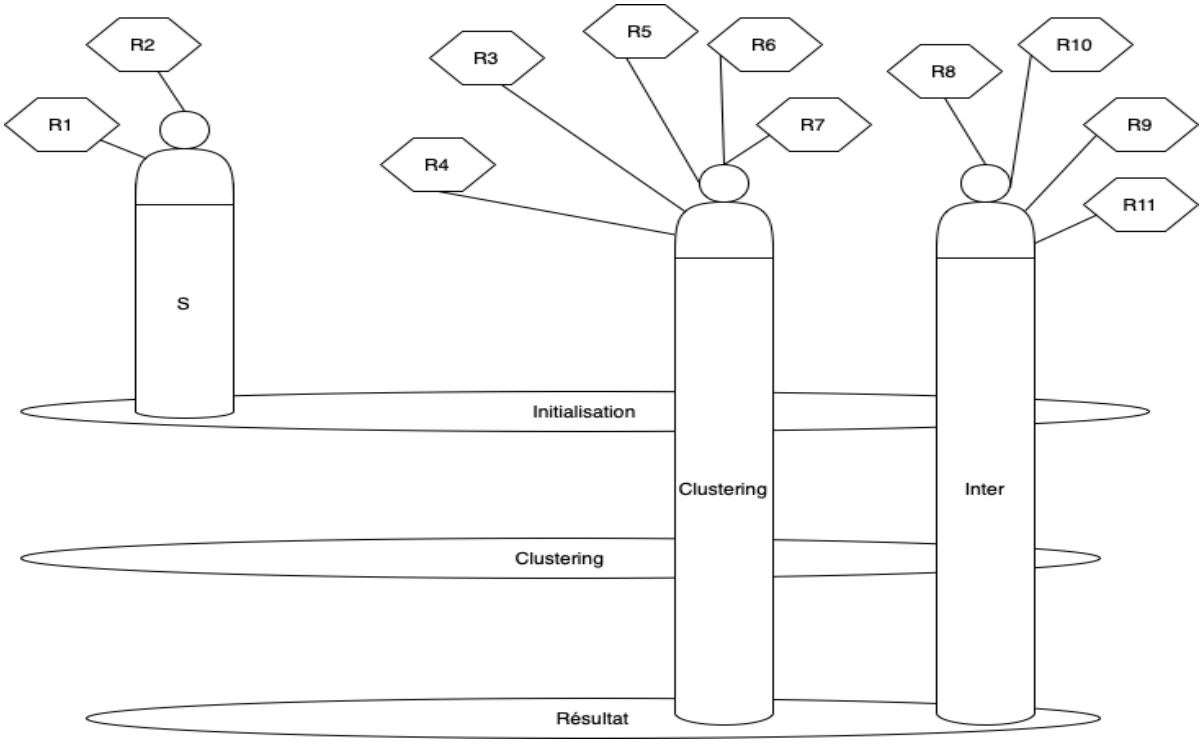

Figure 3.6 : modélisation AGR du système

GROUPE	AGENT	ROLE
G1 : Initialisation	S : Start	R1 : Initialisation R2 : Lancer les agents
	Clustering	R3 : Calculer combien de points
		R5 : initialisation des vecteurs
	Inter : Interaction	R8 : Récupération du nombre des données
G2 : Clustering	Clustering	R4 : Placement des vecteurs et points et calcule de la distance
		R6 : Comparer entre les distances calculés
	Inter : interaction	R9 : Récupération des vecteurs et points R10 : classement des résultats de la comparaison des distances
G3 : Résultats	Clustering	R7 : Envoie des résultats de la comparaison
	Inter : interaction	R11 : Récupération des résultats du clustering

3. AUML (Agent Unified Modeling Language)

Les logiciels développés en agents présentent des spécificités par rapport aux méthodes logicielles plus traditionnelles, quelques tentatives d'adaptation d'UML à ces caractéristiques ont été faites, ce qui a donné naissance à AUML - Agent UML - dont la documentation principale se trouve dans [93].

Néanmoins, le langage AUML ne supporte actuellement que des notions faibles d'agentivité, représentant les agents comme des objets, tout en employant des diagrammes de machine d'état pour modéliser leur comportement et des diagrammes d'interaction étendus pour modéliser leurs actes de communication[94], ne supporte pas les abstractions cognitives ou sociales. Caire dans [95] commente que, bien que cette notation soit utile, elle ne porte aucun concept d'agent dans sa base, déclarant également que spécifier le comportement d'un objet en termes de protocoles d'interaction ne transformera pas un tel objet en agent.

L'une des principales contributions de l'AUML est le document sur les diagrammes d'interaction que l'on trouve dans [93] où l'on tente d'en faire une extension des diagrammes UML vers le support des actes de communication et la modélisation de la communication inter-agents. Dans son format original, ce document proposait également une notation pour la représentation à choix multiples et le parallélisme.

3.1. L'objectif de AUML

Comme UML sur lequel il s'appuie en partie, le but d'Agent UML est d'offrir aux développeurs une notation qu'il sert à analyser, concevoir, et implémenter SMA. L'idée clé d'Agent UML est de réutiliser autant que possible les diagrammes issus d'UML lorsqu'ils correspondent aux besoins des concepteurs des SMA et d'étendre UML – grâce à ses capacités d'extension (stéréotypes, valeurs étiquetées, contraintes) – lorsque les agents et les objets sont différents et que les agents ne peuvent pas être représentés par des diagrammes UML.

Un tel exemple d'extension se trouve dans les diagrammes de séquence Agent UML avec les trois connecteurs AND, OR et XOR. L'utilisation de l'Agent UML dans la documentation est clairement visible dans la spécification FIPA Interaction (voir [96]) puisque tous les protocoles d'interaction sont documentés avec un schéma exprimé en Agent UML.

Il convient de noter, cependant, qu'au lieu de nous fier à l'UML de l'OMG, nous avons l'intention de réutiliser UML partout où cela a du sens. En d'autres termes, AUML ne devrait pas se limiter à UML uniquement - ne vouloir capitaliser sur UML que le cas échéant. La philosophie générale, alors, est la suivante : quand il est logique de réutiliser des portions d'UML, alors faites-le ; quand cela n'a pas de sens d'utiliser UML, utilisez autre chose ou créez quelque chose de nouveau. [97]

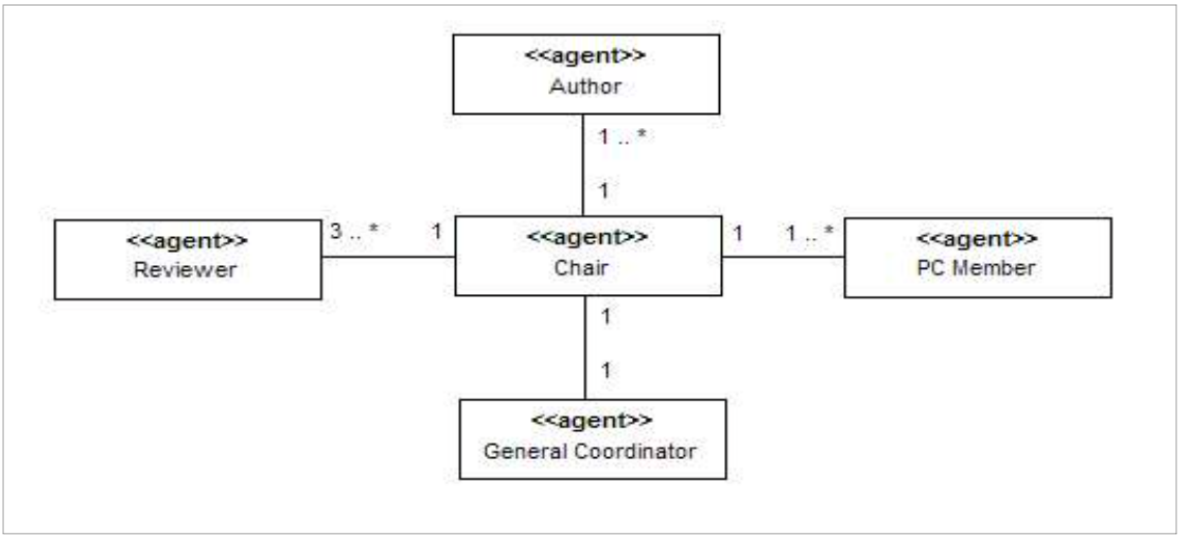

Figure 3.7 : Exemple de diagramme de classes pour le niveau conceptuel, montrant les classes d'agents.[98]

3.2. Comparaison entre Objet et Agent

Contrairement à un objet qui invoque ses méthodes, un agent est capable d'évaluer les messages entrants (Acte de communication (AC)) par rapport à ses objectifs, plans, tâches, les préférences et le modèle du monde interne (B. BAUER). La figure si dessous compare un agent et un objet. [80]

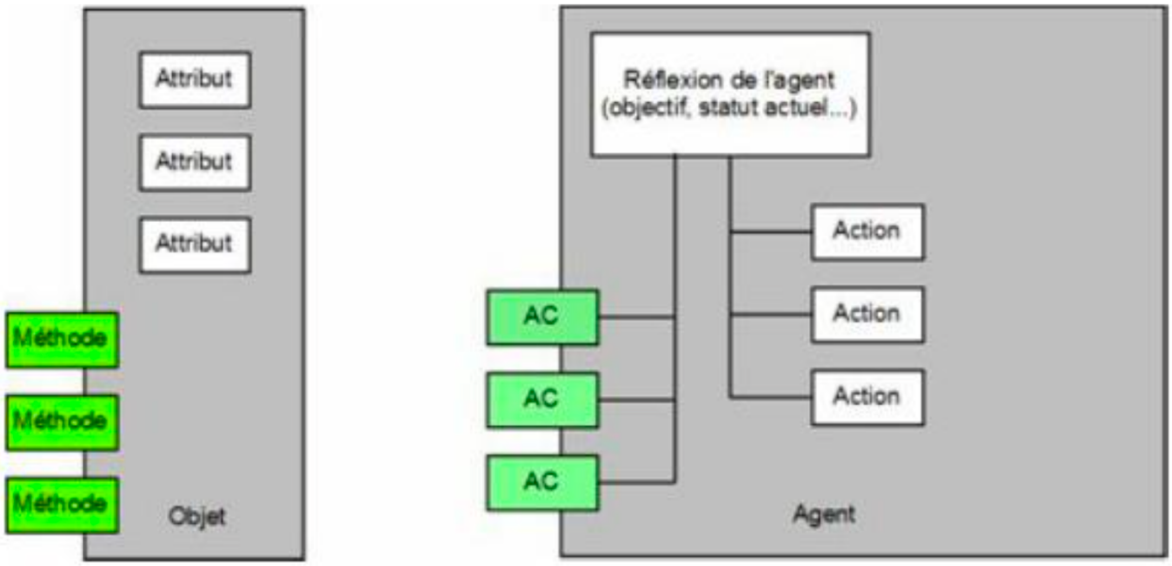

Figure 3.8 : Comparaison entre un agent et un objet

3.3.Diagramme de classes AUML de notre système

Le diagramme de classe montre la structure statique des classes dans le système et les différents types de relations qui existent entre ces classes. Il montre également les caractéristiques (les propriétés et les opérations) des classes et les contraintes qui s'appliquent à la façon dont les objets sont connectés [98]

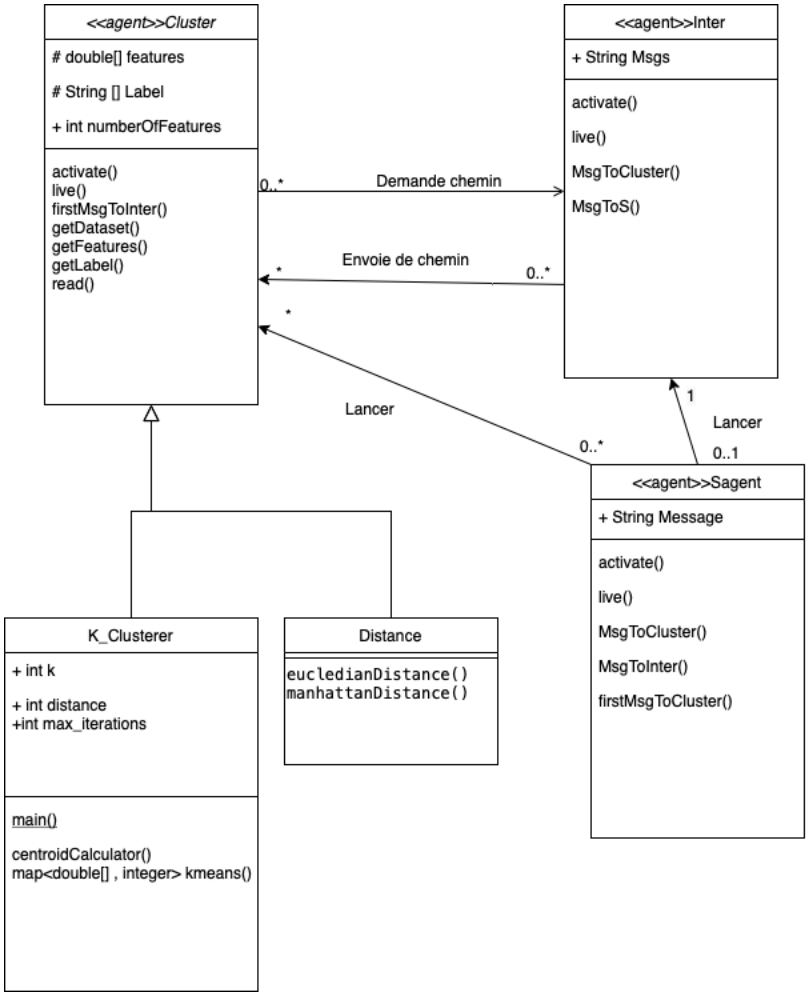

Figure 3.9 : Diagramme de classes d'agents de notre système

4. Conclusion

Dans ce chapitre nous avons présenté les langages de modélisation utilisé pour réaliser notre idée de base et nos approches du clustering par agents. Ainsi que les classes de notre application.

Dans le chapitre suivant, nous allons présenter l'implémentation de notre application en précisant les outils utilisés dans notre travail.

Chapitre 4 : Implémentation

1. Introduction

Après avoir élaborer la conception de notre application, nous abordons dans ce chapitre le dernier volet de ce mémoire, qui a pour objectif de présenter et exposer la réalisation et l'implémentation de notre application, nous présenterons les outils de développement matériels et logiciels utilisés. Après la présentation des outils nous montrons quelques captures d'application illustrant son fonctionnement.

2. Étude technique

L'étude technique est une phase d'adaptation de conception à l'architecture technique. Elle a pour but de décrire au plan fonctionnel la solution à réaliser d'une manière détaillée ainsi que la description des traitements.

Cette étude, qui suit l'étude détaillée, constitue le complément de spécification informatique nécessaire pour assurer la réalisation du futur système.

2.1. Matériels utilisés

Pour la réalisation de notre application, nous avons eu recours à plusieurs moyens matériels et logiciels. Le développement de l'application est réalisé via deux ordinateurs portables ayant les caractéristiques suivantes :

Caractéristiques	PC 1	PC 2
Marque	Apple MacBook air	Lenovo DESKTOP-2MU4VCC
Processeur	Intel Core i5 bicœur à 1,4 GHz (Turbo Boost jusqu'à 2,7 GHz)	Intel Core i3-4005U 1.70GHz
RAM	4GB	4 GO
Disque dur	256 gb PCIE	500Gb
Système d'exploitation	MacOS Catalina	Windows 10 professionel

Tableau 4.1 : Matériel de base

3. Environnement de développement

Afin de réaliser un projet, un ensemble d'outils logiciels sont nécessaires pour pouvoir accomplir le travail demandé :

3.1. Les Outils et langage de développement

3.1.1. JAVA

Figure 4.1: Logo java

Le langage java 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 et assistants personnels.

3.1.2. Eclipse IDE

Figure 4.2 : logo Eclipse IDE

L'IDE Eclipse est célèbre pour l'environnement de développement intégré (IDE) Java, mais il compte un certain nombre d'IDE y compris notre IDE C/C++, JavaScript/TypeScript IDE, PHP IDE, et plus encore. [99]

Il permet de combiner la prise en charge de plusieurs langues et d'autres fonctionnalités dans l'un de ses packages par défaut, et Eclipse Marketplace permet une personnalisation et une extension pratiquement illimitées.

Une collection impressionnante d'outils peut être facilement installée dans l'IDE du bureau Eclipse, y compris des constructeurs d'interface graphique et des outils de modélisation, de création de graphiques et de rapports, de tests, etc.

3.1.2.1. Eclipse Windowbuilder

WindowBuilder est composé de SWT Designer et Swing Designer, Il facilite la création d'applications Java GUI sans passer beaucoup de temps à écrire du code. Il utilise le concepteur visuel WYSIWYG et les outils de mise en page pour créer des formulaires simples dans des fenêtres complexes ; le code Java sera ensuite généré. Il permet d'ajouter facilement des contrôles par glisser-déposer et ajouter des gestionnaires d'événements aux contrôles, modifier diverses propriétés des contrôles à l'aide d'un éditeur de propriétés, internationaliser l'application et bien plus encore.[100]

3.1.3. MADKIT

Figure 4.3 : LOGO madkit

La plate-forme Madkit a été développée en 1998 à l'Université de Montpellier par Olivier Gutchneck lors de sa thèse et Jacques Ferber son encadrant.

Un aspect clé de MADKit est que, contrairement aux approches conventionnelles qui sont principalement centrées sur l'agent, MaDKit suit une approche centrée sur l'organisation

(OCMAS). Par conséquent, MaDKit est construite sur le modèle organisationnel AGR (Agent/Groupe/Rôle) et ne s'appuie pas sur un modèle d'agent prédéfini : (tout type d'Agents jouent des rôles dans des groupes et créent ainsi des sociétés artificielles [101]

Cette plateforme base son implémentation sur les concepts d'agents, de groupe et de rôle. Les agents sont lancés par le noyau de Madkit, qui propose notamment les services de gestion des groupes et de communication. Cette plate-forme est surtout intéressante pour l'approche organisationnelle qu'elle met en avant lors de l'analyse et de la conception d'un système multi-agents.

Ainsi que trois grands principes d'architecture : Micro noyau agent, Agentification des services, Modèle graphique componentiel. MADKIT est développé en Java 1.1 et testé sous Unix, Windows et MacOS. (MaD).[80]

MadKit comporte plusieurs caractéristiques et fonctionnalités, Parmi ces caractéristiques on compte :

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

4. Réalisation du travail et résultats

4.1. Présentation des interfaces

Dans ce qui suit, nous allons présenter les différentes interfaces de notre application :

A. L'interface Principale

L'utilisateur doit rentrer le nombre maximal de clusters, le nombre maximal d'itérations et la métrique souhaité (Euclidienne ou Manhattan) :

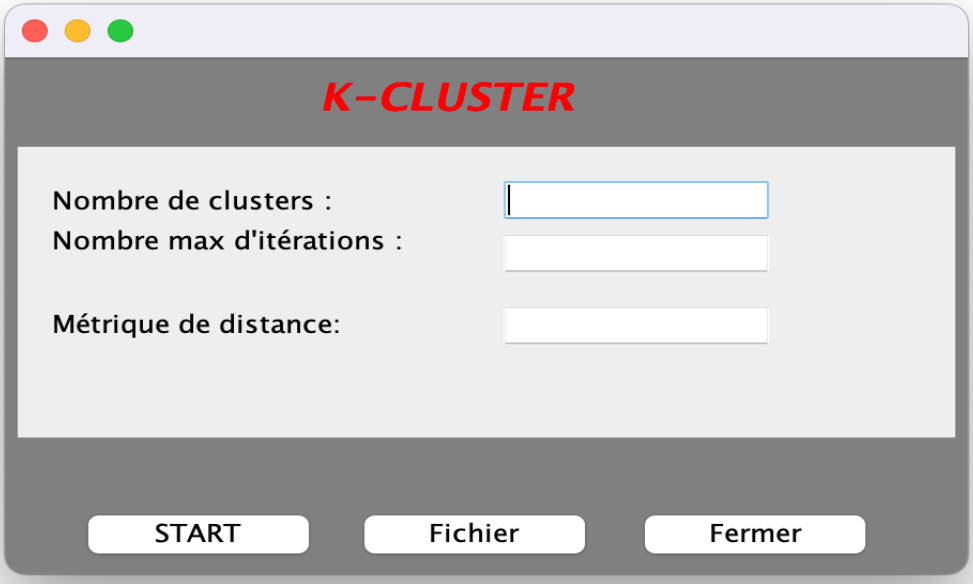

Figure 4.4 : Fenêtre principale de l'application

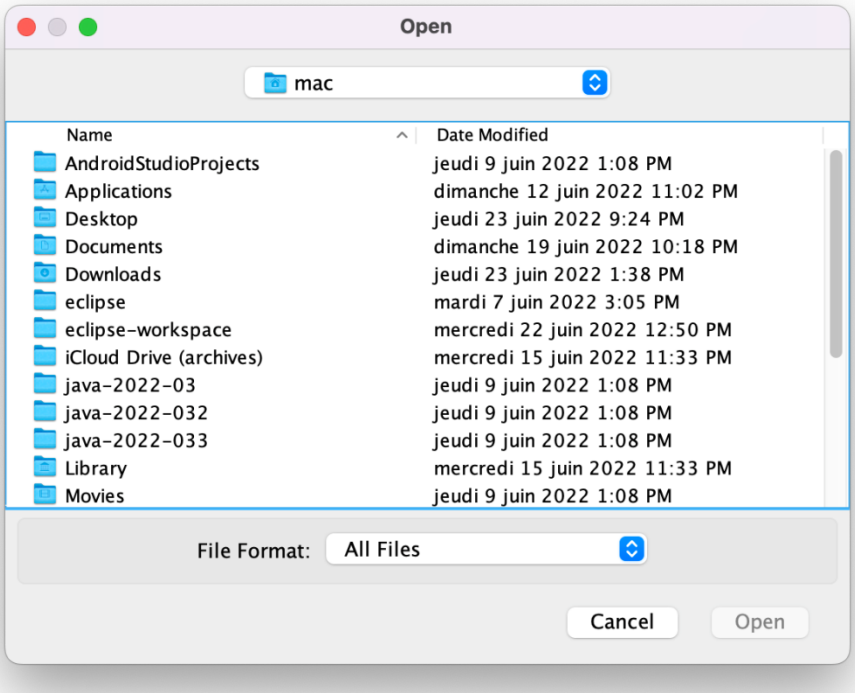

Figure 4.5 navigation vers le fichier texte

Après que l'utilisateur insert les données nécessaires, Le bouton « START » sert à lancer les agents, ils trouveront le chemin du fichier texte (iris.txt) dans le dossier de l'application et le clustering sera ensuite effectué et stocké dans un autre fichier texte nommé « results.txt ».

B. La fenêtre de résultats

Dans cette fenêtre l'utilisateur pourra visualiser les résultats du clustering, il aura la main de les modifier et seront ensuite modifier dans le fichier texte également.

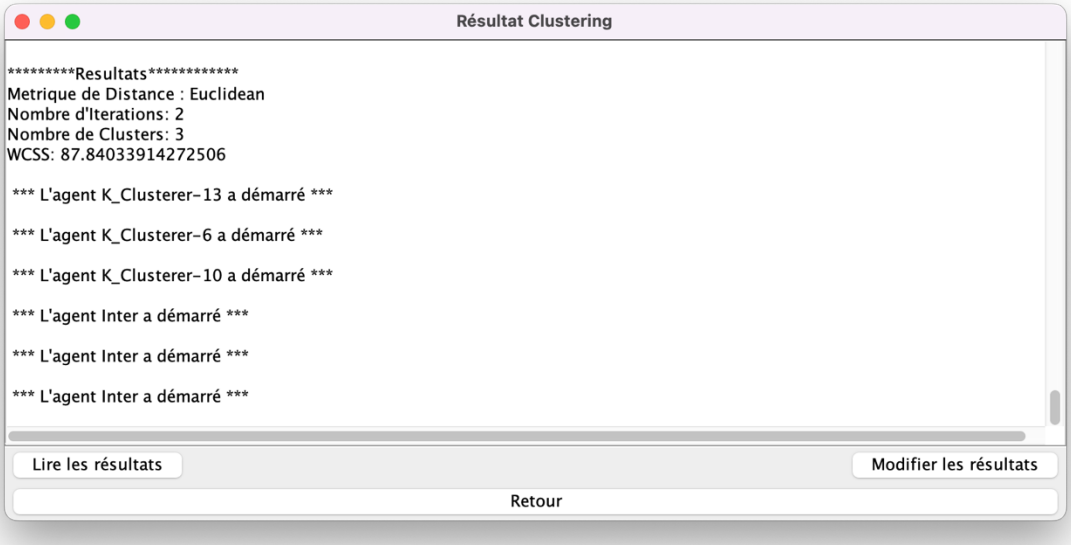

Figure 4.6 : Fenêtre du résultat du clustering (L'affichage des agents lancés)

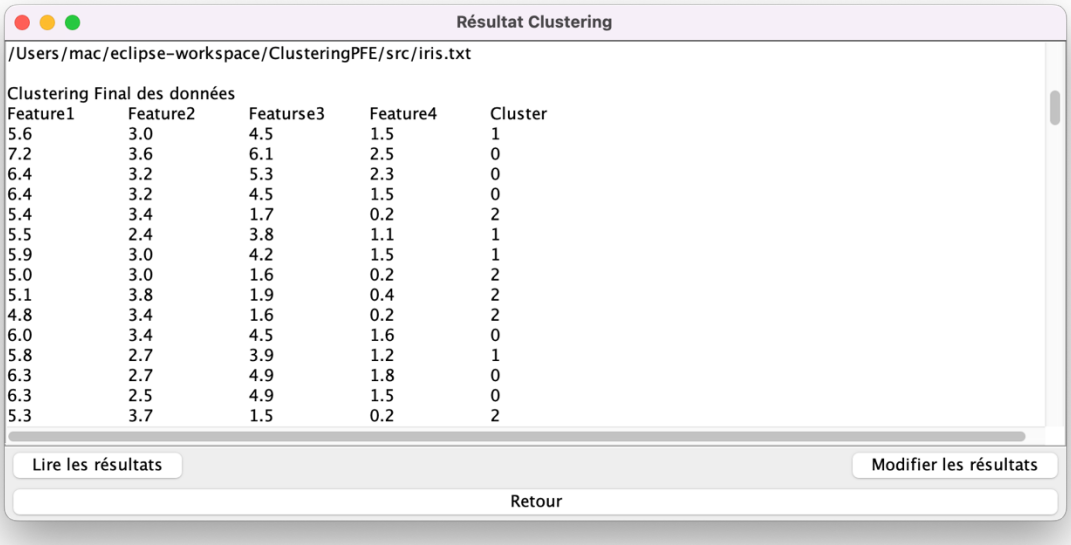

Figure 4.7 : Fenêtre du résultat du clustering (L'affichage du résultat du clustering)

5. Conclusion

Dans ce chapitre, nous avons présenté l'environnement et le processus de développement. Nous avons exposé ainsi le résultat de développement à l'aide des aperçus écran de notre application qui consiste à effectuer un clustering basée agents.

Après avoir fini le travail, nous avons constaté que le principe AGR a donné de bons résultats et a montré son efficacité.

Conclusion générale

Conclusion générale

L'utilisation du concept agent et des systèmes multi-agent était liée au domaine de la résolution distribuée de problème. Par la suite, les SMAs étaient connus comme outil de simulation ou moyen de conception.

Dans le cadre de ce projet de fin d'étude, il était question d'utiliser les agents comme outil de résolution. Pour ce, nous avons présenté une approche de clustering de données à base d'agents pour explorer les capacités de calcul des Agents et des SMAs. À cette fin, nous avons utilisé le modèle AGR pour la représentation des Agents, des Groupes et des Rôles de chaque agent du clustering, à l'aide de l'algorithme de clustering K-Means.

Nous avons créé des ***agents*** jouant le rôle d'***objet*** à clusturer dans des ***clusters*** substitués par des ***groupes*** au niveau agent. Cela était réalisé sous la plateforme MADKIT, et en utilisant le langage de programmation JAVA.

Même si notre travail est une première ébauche, nous pouvons conclure que l'utilisation des agents comme outil de résolution ou d'optimisation est très pertinent et encourageant.

En perspectives, nous pensons que plusieurs améliorations peuvent être apportées à notre système pour consolider les pouvoirs de calcul des agents et des SMAs en domaine d'optimisation en général, et en clustering spécifiquement.

En plus, l'application du concept agent, pour d'autres domaines d'applications sera, sans doute, un soutien très apprécié et une consolidation très souhaitée des capacités de calcul et d'optimisation des agents.

Référence :

[1] https://onconano2.webnode.fr/diagnostique/puce-%C3%A0-adn/experience-1
[2] B. Mirkin, Clustering for Data Mining: A data Recovery Approach, National Research University Higher School of Economics, 2005
[3] A. JAIN, M. MURTY et P. FLYNN, Data Clustering: A Review, The Ohio State University, 1999
[4] A. K. Jain et R. C. Dubes, Algorithm for Clustering Data, 1988
[5] Cleuziou, Guillaume. Une méthode de classification non-supervisée pour l'apprentissage de régles et la recherche d'information. PhD thêsis, Université d'Orléans, 2004
[6] Hodson, Frank Roy, Sneath, Peter HA, and Doran, JE. Some experiments in the numerical analysis of archaeological data. Biometrika, 53(3-4) :311–324, 1966.
[7] Belacel, Nabil. M´ethodes de classification multicritère : méthodologie et applications à l'aide au diagnostic médical. 2000
[8] Furui, Sadaoki. Speaker-independent isolated word recognition using dynamic features of speech spectrum. IEEE Transactions on Acoustics, Speech, and Signal Processing, 34(1) :52–59, 1986.
[9] Ester, Martin, Kriegel, Hans-Peter, Sander, J¨org, Xu, Xiaowei, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In Kdd, volume 96, pages 226–231, 1996
[10] Tay, Francis EH and Shen, Lixiang. Economic and financial prediction using rough sets model. European Journal of Operational Research, 141(3) :641–659, 2002
[11] H. E. Driver et A. L. Kroeber, «Quantitative expression of cultural relationships,» Berkeley : University of California Press, 1932.
[12] J. Zubin, «A technique for measuring like-mindedness,» The Journal of Abnormal and Social Psychology, pp. 508-516, 1938.
[13] R. C. Tryon, «Cluster analysis : correlation profile and orthometric (factor) analysis for the isolation of unities in mind and personality,» Ann Arbor(Michigan) : Edwards Brothers, 1939
[14] R. B. Cattell, «The description of personality: basic traits resolved into clusters,» The Journal of Abnormal and Social Psychology, p. 476–506, 1943
[15] R. R. Sokal et P. H. A. Sneath, «Principles of numerical taxonomy,» Freeman, 1963
[16] B. Berry et M. Ray, «Multivariate socioeconomic regionalization: a pilot study in central Canada,» Ostry, S. & Rymes, T. (eds), Papers on regional statistical studies, p. 75–130, 1966,
[17] D. G. Morrison, «Measurement Problems in Cluster Analysis,» Management Science, pp. 775- 780.
[18] P. E. Green, R. E. FRANK et P. J. ROBINSON, «CLUSTER ANALYSIS IN TEST MARKET SELECTION,» Management Science, pp. 387-400, 1967.
[19] H. F. Kaiser, «An objective method for establishing legislative districts,» midwest journal, pp. 200-213, 1966
[20], M. Lorr, Explorations in typing psychotics, Pergamon Press, 1966, p. 241.
[21] L. Wingo, «Recent Patterns of Urbanization among Latin American Countries,» Urban Affairs Quarterly, vol. 2, pp. 81-109, 1967.
[22] D. Fisher, Clustering and Aggregation in Economics, Johns Hopkins University Press, 1969.
[23] N. Jardine et R. Sibson, «The construction of hierarchic and non-hierarchic classifications,» The Computer Journal, vol. 11, p. 177–184, 1968.
[24] K. D. Bailey, Cluster Analysis, Wiley, 1975.

[25] K. Krippendorff, «Clustering,» chez Multivariate Techniques in Human Communication Research, New York, NY: Academic Press, 1980, pp. 259-308
[26] Berkhin, Pavel. A survey of clustering data mining techniques. In Grouping multidimensional data, pages 25–71. Springer, 2006
[27] Xu, Dongkuan and Tian, Yingjie. A comprehensive survey of clustering algorithms. Annals of Data Science, 2(2) :165–193, 2015
[28] Johnson, Stephen C. Hierarchical clustering schemes. Psychometrika, 32(3) :241–254, 1967.
[29] Hartigan, John A. Statistical theory in clustering. Journal of classification, 2(1) :63–76, 1985
[30] J. A. Hartigan, "Clustering Algorithms," John Wiley and Sons Inc. New York, 1975
[31] S. Z. Selim and M. A. Ismail, "K-means type algorithms: a generalized convergence theorem and characterization of local optimality," IEEE Transactions on Pattern Analysis and Machine Intelligence 6, pp. 81-87, 1984.
[32] C. S. Waenekar et G. Krishna, «A Heuristic Clustering Algorithm Using Union of Overlapping Pattern-Cells,» Pattern Recognition, vol. 11, n° %12, pp. 85-93, 1979.
[33] P. N. Tan, M. Steinbach et V. Kumar, «Cluster Analysis: Basic Concepts and Algorithms,» 2005.
[34] C. C. Aggarwal, Data Clustering Algorithms and applications, 2013.
[35] J. Han, J. Pei et M. Kamber, Data Mining: Concepts and techniques 3rd Edition, Morgan Kaufmann Publishers, 2011.], [M. Halkidi, Y. Batistakis et M. Vazirgiannis, Cluster Validity Methods: Part 1, 2002
[36] A. M. Bagirov, S. Taheri et N. Karmitsa, Partitional Clustering via Nonsmooth Optimization, 2020
[37] Zebiri Ibrahim , Optimisation par loups gris adaptée pour le Clustering de données,2020
[38] S. Theodoridis et K. Koutroumbas, Pattern Recognition, 2nd Edition, 1999
[39] D. L. Davies et D. W. Bouldin, «A Cluster Separation Measure,» 1979.
[40] J. C. Dunn, «Well Separated Clusters and Optimal Fuzzy Partitions,» Journal of Cybernetics, vol.4, n° %11, pp. 95-104, 1974
[41] M. Halkidi, Y. Batistakis et M. Vazirgiannis, Clustering Validity Checking Methods: Part 2, 2002
[42] P. J. ROUSSEEUW, «Silhouettes: a graphical aid to the interpretation and validation of cluster analysis,» 1986
[43] K. a. Rousseeuw, «Finding Groups in Data: An Introduction to Cluster Analysis,» 1990
[44] E. B. Fowlkes et C. L. Mallows, «A Method for Comparing Two Hierarchical Clusterings,» Journal of the American Statistical Association, vol. 78, n° %1383, pp. 553-569, 1983
[45] S. Wagner et D. Wagner, Comparing Clusterings – An Overview, 2007
[46] C. Ramdane, «Le Clustering des données : une nouvelle approche évolutionnaire quantique,» 2006
[47] P. LLOYD, «Least squares Quantization in PCM,» 1982
[48] E. D. Dolan et J. J. Moré, «Benchmarking optimization software with performance profiles,» 2001
[49] A. M. Bagirov, S. Taheri et N. Karmitsa, Partitional Clustering via Nonsmooth Optimization,2020
[50] J. W. Carmichael, J. A. George et R. S. Julius, «Finding Natural Clusters,» Systematic Zoology, vol. 17, pp. 144-150, 1968.

[51] J. W. Carmichael et P. H. A. Sneath, «Taxometric Maps,» Systematic Biology, vol. 18, n° %14, pp. 402-415, 1969.
[52] B. S. Duran, Cluster analysis, 1970.
[53]D. Wishart, «Mode Analysis: a generalization of nearest neighbour reduces chaining effects,» chez Numerical taxonomy, Academic press, 1969, pp. 282-311.
[54] D. Horn, «A Study of Personality Syndromes,» Personality, vol. 12, n14, 1943.
[55] E. Parzen, «On Estimation of a Probability Density Function and Mode,» Annals of Mathematical Statistics, vol. 33, pp. 1065-1076, 1962.
[56] D. F. Specht, «Generation of Polynomial Discriminant Functions for Pattern Recognition,» 1967.
[57] E. Fix et J. J. Hodges, «Discriminatory Analysis: Nonparametric Discrimination: Consistency Properties,» International Statistical Review, vol. 57, n° %13, pp. 238-247, 1989.
[58] E. Fix et J. L. Hodges, Discriminatory Analysis: Nonparametric Discrimination: Small Sample Performance, 1952.
[59] A. K. Jain et R. C. Dubes, Algorithm for Clustering Data, 1988.
[60] D. G. Stork, P. E. Hart et R. O. Duda, Pattern Classification, 2nd Edition, John Wiley & Sons, 2012.
[61] E. A. Patrick et F. P. Fischer, «A generalized k-Nearest Neighbor Rule,» 1970.
[62] B. S. Everitt, S. Landau, M. Leese et D. Stahl, Cluster Analysis 5th Edition, London: John Wiley & Sons, 2011.
[63] M. A. Wong et T. Lane, «A kth Nearest Neighbor Clustering Procedure,» Journal of the Royal Statistical Society. Series B (Methodological), vol. 45, n° %13, pp. 362-368, 1983.
[64] D. C. WUNSCH et R. XU, Clustering, 2009.
[65] J. A. Hartigan, Clustering Algorithms, Toronto: John Wiley & Sons, 1975
[66] K. Fukunaga, «Introduction to Statistical Pattern Recognition, Second Edition,» 1972.
[67] B. Silverman, Density Estimation for Statistics and Data Analysis, CRC Press, 2018.
[68] R. Pradeep et S. Shubha, «A survey of clustering techniques, » International Journal of Computer Applications, 2010.
[69] P. Berkhin, «Survey of clustering data mining techniques, » chez Grouping Multidimensional Data, pp. 25-71.
[70] J.FERBER, 1995
[71] Dema, 1996
[72] R.JENNING, et al. 1998
[73] WIES 99
[74] JARR 02
[75] Morin, 1977
[76] Boissier, 2004
[77] Wooldridge et al., 1995
[78] Demazeau et al., 1990
[79] Russell et al., 2006
[80] B.BENDJAMA I.SOUAMES Optimisation par colonie de fourmis basée agent pour le problème d'emploi du temps des cours universitaires.
[81] 15201417
[82] FINI 95
[83] FIPA99
[84] BERNON, et al., 2005
[85] J.FERBER, et al., 1998

[86] C.LENAY, et al., 1994
[87] J.FERBER, et al., 2005
[88] Boulhout Ilyas, Étude de l'impact du principe Influence/Réaction dans la modélisation Agent de variantes Améliorées d'algorithmes à fourmis.
[89] Vercouter, 2004
[90] PHILIPE, et al. 2005
[91] Jacques Ferber, Olivier Guknecht, Fabien Michel, AGENT/GROUP/ROLES: SIMULATING WITH ORGANIZATIONS,2003.
[92] Jacques Ferber, Olivier Gutknecht, and Fabien Michel, From Agents to Organizations: An Organizational View of Multi-agent Systems.
[93] Huget, M.P., et al.: Interaction Diagrams. Working Documents. AUML Official Website (2003), http://www.auml.org/]
[94] Vicari, R.M., Gluz, J.C.: An Intelligent Tutoring System (ITS) View on AOSE. International Journal of Agent-Oriented Software Engineering (2007)
[95] Caire, G.: Agent Oriented Analysis Using Message/UML. In: Wooldridge, M.J., Weiß, G., Ciancarini, P. (eds.) AOSE 2001. LNCS, vol. 2222, p. 119. Springer, Heidelberg (2002)
[96] http://www.fipa.org/repository/ips.php3
[97] Marc-Philippe Huget, James Odell and Bernhard Bauer
[98] M. Fowler, «UML Distilled,» 1997
[99] https://www.eclipse.org/ide/
[100] https://www.eclipse.org/windowbuilder/
[101] https://www.madkit.net/madkit/madkit.php