Segmentation hiérarchique des cartes topologiques

Mustapha Lebbah[*,**], Hanane Azzag[**]

[*] LIM&BIO - UFR (SMBH)- Université Paris 13,
74, rue Marcel Cachin 93017 Bobigny Cedex France
[**]LIPN - UMR 7030
Université Paris 13 - CNRS
99, av. J-B Clément - F-93430 Villetaneuse
{hanane.azzag, mustapha.lebbah}@lipn.univ-paris13.fr

Résumé. Dans ce papier, nous présentons une nouvelle mesure de similarité pour la classification des référents de la carte auto-organisatrice qui sera réalisée à l'aide d'une nouvelle approche de classification hiérarchique. (1) La mesure de similarité est composée de deux termes : la distance de Ward pondérée et la distance euclidienne pondérée par la fonction de voisinage sur la carte topologique. (2) Un algorithme à base de fourmis artificielles nommé AntTree sera utilisé pour segmenter la carte auto-organisatrice. Cet algorithme a l'avantage de prendre en compte le voisinage entre les référents et de fournir une hiérarchie des référents avec une complexité proche du *nlog(n)*. La segmentation incluant la nouvelle mesure est validée sur plusieurs bases de données publiques.

1 Introduction

Le problème de la classification de données est identifié comme une des problématiques majeures en extraction des connaissances à partir de données. Depuis des décennies, de nombreux sous-problèmes ont été identifiés, comme par exemple la sélection des données ou des variables, la variété des espaces de représentation (numérique, symbolique, etc), l'incrémentalité, la nécessité de découvrir des concepts, ou d'obtenir une hiérarchie, etc. La popularité, la complexité et toutes ces variantes du problème de la classification de données, (Jain et al. (1999)), ont donné naissance à une multitude de méthodes de résolution. Ces méthodes peuvent faire appel à des principes heuristiques ou encore mathématiques.

Les méthodes qui nous intéressent dans ce travail, sont celles qui permettent de faire de la classification non supervisée de données en utilisant les cartes topologiques (appelées aussi SOM :Self-organizing Map). Celles-ci sont souvent utilisées parce qu'elles sont considérées à la fois comme outils de visualisation et de partitionnement non supervisé de différents types de données (quantitatives et qualitatives). Elles permettent de projeter les données sur des espaces discrets qui sont généralement en deux dimensions. Le modèle de base, proposé par Kohonen (Kohonen (2001)), est uniquement dédié aux données numériques. Des extensions et des reformulations du modèle de Kohonen ont été proposées dans la littérature, (Bishop et al. (1998);

Lebbah et al. (2007)).

Pour l'apprentissage des cartes topologiques, les critères de qualité sont plus difficiles à définir ; ils s'articulent autour de l'interprétation des regroupements ou des partitions obtenues. Par conséquent un premier problème se pose : celui de la segmentation (partitionnement) de la carte. On retrouve dans la littérature plusieurs méthodes ou propositions de segmentation de la carte qui utilisent des critères de similarité standard qui ne tiennent pas compte du voisinage introduit par la carte, (Vesanto et Alhoniemi (2000)). Elles se résument, souvent, à l'utilisation d'un algorithme de regroupement (classification hiérarchique ou les K-moyennes) combiné à un indice de qualité pour déterminer la partition idéale. Le second problème qui nous intéresse dans cet article est celui du choix de l'algorithme de segmentation de la carte. Ainsi, nous avons introduit une nouvelle classification hiérarchique que l'on va appliquer sur les référents (représentants) de la carte. Cette nouvelle méthode nommée AntTree introduite par (Azzag et al.) s'inspire du comportement d'auto-assemblage observé chez une population de fourmis réelles et leurs capacités à s'accrocher entre elles pour construire des structures vivantes.

La suite de notre article est organisée comme suit : dans la section 2, nous présentons les principes généraux des cartes SOM avec la nouvelle mesure proposée, ainsi que la nouvelle méthode de classification hiérarchique utilisée pour la segmentation de la carte topologique. La section 3, quant à elle, est consacrée aux résultats et à l'étude comparative réalisée sur des bases de données numériques. La dernière section rassemble les conclusions faites au cours des expérimentations et présente des perspectives.

2 Segmentation topologique et hiérarchique

Les cartes auto-organisatrices présentées par Kohonen ont été utilisées pour la classification et la visualisation des bases de données multidimensionnelles. Une grande variété d'algorithmes des cartes topologiques est dérivée du premier modèle original proposé par Kohonen. Ces modèles sont différents les uns des autres, mais partagent la même idée de présenter les données de grande dimension en une simple relation géométrique sur une topologie réduite.

Dans cette section, nous décrivons la version originale des cartes auto-organisatrices. Ce modèle consiste en la recherche d'une classification non supervisée d'une base d'apprentissage $A = \{z_i \in \mathcal{R}^d, i = 1..N\}$ où l'individu $\mathbf{z}_i = (z_i^1, z_i^2, ..., z_i^j, ..., z_i^d)$ est de dimension d. Ce modèle classique se présente sous forme d'une carte possédant un ordre topologique de N_c cellules. Les cellules sont réparties aux nœuds d'un maillage. La prise en compte dans la carte $\mathcal{C}$ de la notion de proximité impose de définir une relation de voisinage topologique. Ainsi, la topologie de la carte est définie à l'aide d'un graphe non orienté et la distance $\delta(c, r)$ entre deux cellules c et r étant la longueur du chemin le plus court qui sépare les deux cellules c et r. Afin de modéliser la notion d'influence d'une cellule r sur une cellule c, qui dépend de leur proximité, on utilise une fonction noyau $\mathcal{K}$ ($\mathcal{K} \geq 0$ et $\lim_{|x|\to\infty} \mathcal{K}(x) = 0$). L'influence mutuelle entre deux cellules c et r est donc définie par la fonction $\mathcal{K}(\delta(c, r))$. A chaque cellule c de la grille est associée un vecteur référent $\mathbf{w}_c = (w_c^1, w_c^2, ..., w_c^j, ..., w_c^d)$ de dimension d. Les phases principales de l'algorithme d'apprentissage sont définies dans la littérature et consistent

à minimiser la fonction coût suivante :

$$\mathcal{J}(\phi, \mathcal{W}) = \sum_{\mathbf{z}_i \in A} \sum_{r \in C} \mathcal{K}(\delta(\phi(\mathbf{z}_i), r)) ||\mathbf{z}_i - \mathbf{w}_r||^2 \quad (1)$$

Où ϕ affecte chaque observation $\mathbf{z}$ à une cellule unique de la carte $\mathcal{C}$. Dans cette expression $||\mathbf{z} - \mathbf{w}_r||^2$ représente le carré de la distance euclidienne.

A la fin de l'apprentissage, la carte auto-organisatrice détermine une partition des données en p sous ensembles. Cette partition et les sous-ensembles associés seront notés par $\mathcal{P} = \{P_1, \ldots, P_c, \ldots, P_p\}$. A chaque sous ensemble P_c on associe un vecteur référent $\mathbf{w}_c \in \mathcal{R}^d$ qui sera le représentant ou la "moyenne" de l'ensemble des observations de P_c.

Souvent l'utilisation des cartes topologiques est suivie par une segmentation des cartes ou un regroupement des référents. Cette tâche, est réalisé à l'aide d'algorithmes de partitionnement tel que K-moyennes, ou la classification ascendante hiérarchique CAH (Jain et Dubes (1988)). Le choix des deux sous-ensembles qui vont être regroupés est réalisé à l'aide d'une mesure de similitude définie entre deux sous-ensembles. Différents critères de similitude sont proposés dans la littérature, (Jain et Dubes (1988); Ambroise et al. (1998)). Souvent ces critères ne tiennent pas compte de la topologie ou de l'auto-organisation des référents obtenue avec la carte topologique. La mesure de similitude la plus connue est celle du critère de Ward définie comme suit :

$$Indice_w = \left(\frac{n_c n_r}{n_c + n_r} \right) ||\mathbf{w}_c - \mathbf{w}_r||^2 \quad (2)$$

tel que n_c et n_r indiquent le nombre d'observations affectées pour le sous-ensemble P_c et P_r.

En considère deux partitions : $\mathcal{P}^{t-1}$ comme la partition avant regroupement des deux sous-ensembles P_c et P_r associés aux deux référents c et r, et $\mathcal{P}^t$ la partition obtenue en regroupant les sous ensembles P_c et P_r. On peut montrer que la différence entre l'inertie des deux partitions est égale au critère de regroupement de Ward (2). Ainsi à chaque étape de la classification on calcule une matrice de similarité associée à la nouvelle partition. Par conséquent, à chaque étape de l'algorithme, on choisit une nouvelle partition qui limite l'augmentation de l'inertie intra-classe. Notons que cette propriété ne garantit pas l'optimisation globale du critère. L'algorithme peut être décrit en 5 étapes :

1. Initialiser la matrice de similarité avec la partition obtenue avec la carte.
2. Trouver les deux sous ensembles les plus proches selon le critère de Ward.
3. Regrouper les deux sous ensembles P_c et P_r en un seul sous ensemble.
4. Mettre à jour la matrice de similarité de la nouvelle partition.
5. Répéter 2

Nous rappelons que la classification hiérarchique ou tout autre méthode de segmentation des cartes topologiques sont couplées à un indice de qualité qui permet de choisir la taille de la

partition optimale. Afin d'optimiser l'algorithme de segmentation de la carte, nous proposons dans ce papier deux modifications. La première consiste à utiliser un algorithme de classification hiérarchique qui supprime l'étape 4 et fournit automatiquement la taille de la partition "idéale". Ceci se résume à utiliser un algorithme de classification hiérarchique qui utilise une seule et unique matrice de similarité, qui est celle de la partition à l'instant $t = 0$ ($\mathcal{P}^0$). Cet algorithme sera détaillé par la suite dans la section 2.1. Notre deuxième proposition consiste à modifier la mesure de similarité de regroupement, afin de prendre en compte le voisinage de la topologie fournie par la carte.

Le critère de Ward mesure la perte d'inertie après chaque fusion de deux sous ensembles, il est donc nécessaire de considérer la modification de la topologie de la carte après fusion en pondérant l'indice de Ward par un paramètre quantifiant ce changement. Nous proposons de quantifier le changement topologique par une pondération du critère de Ward avec une mesure qui prend en compte la nouvelle structure de la carte, cette dernière est définie comme suit :

$$\sum_{u \in C} K(\delta((c,r),u))) \text{ telle que } \delta((c,r),u) = \min\{\delta(c,u), \delta(r,u)\}$$

Cette pondération permet donc de quantifier ce changement topologique de la carte. Afin de de prendre en compte la proximité des référents sur la carte, nous proposons de soustraire une quantité à cette mesure de façon à diminuer la perte d'inertie mesurée par le critère de Ward, selon la proximité des sous-ensembles sur la carte topologique. Finalement, la nouvelle mesure devient :

$$\begin{aligned} Indice_{Neigh-W} &= \left(\sum_{u \in C} K(\delta((c,r),u))) \right) \frac{n_c n_r}{n_c + n_r} ||\mathbf{w}_c - \mathbf{w}_r||^2 \\ &- K(\delta(c,r))(n_c + n_r)||\mathbf{w}_c - \mathbf{w}_r||^2 \end{aligned} \quad (3)$$

Cette mesure heuristique est constituée de deux termes. Le premier terme correspond à la perte d'inertie des observations après fusion des deux sous ensembles P_c et P_r. le deuxième terme permet de rapprocher les sous-ensembles correspondants à deux référents voisins sur la carte, afin de conserver l'ordre topologique entre les différentes partitions. En effet, si c et r sont voisins sur la carte, la valeur de $\delta(c,r)$ sera alors basse et dans ce cas celle de $K(\delta(c,r))$ sera élevé ; le second terme a donc comme effet de réduire davantage le premier terme. Il est évident que pour un voisinage nul, notre mesure se réduit à calculer le critère de Ward. Il est donc possible de dire que notre mesure permet d'obtenir une solution régularisée du critère de Ward : la régularisation étant obtenue grâce à la contrainte d'ordre topologique introduit dans notre proposition de mesure.

Finalement l'utilisation de notre mesure permet de définir une matrice de similarité qui tient compte à la fois de la perte d'inertie et de la topologie de la carte. Pour traiter cette matrice nous allons présenter dans la section suivante l'algorithme de classification hiérarchique basé sur les fourmis artificielles.

2.1 Classification hiérarchique

Pour segmenter la carte nous avons utilisé une approche biomimétique inspirée du comportement d'auto-assemblage observé chez les fourmis réelles. Ces dernières construisent des structures vivantes en se connectant progressivement entre elles, (Anderson et al. (2002)).

Le modèle développé utilise des règles comportementales afin de construire des heuristiques pour la classification non supervisée hiérarchique. Dans notre modèle chaque fourmi artificielle représente une donnée $\mathbf{z}$ à classer. Ces fourmis artificielles vont construire de manière similaire un arbre en appliquant certaines règles où les déplacements et les assemblages des données sur cet arbre dépendent de leurs similarités.

Le principe d'AntTree est le suivant : chaque donnée (fourmi) à classer $\mathbf{z}_i$, $i \in [1, N]$ (N est le nombre de données initiales) représente un nœud de l'arbre à assembler.

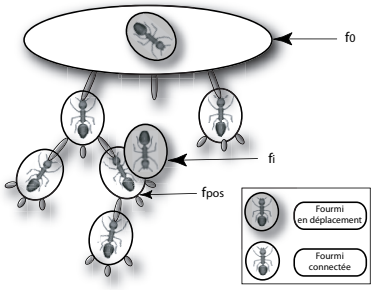

FIG. 1 – *Construction de l'arbre par des fourmis : les fourmis qui sont en déplacement sont représentées en gris et les fourmis connectées en blanc.*

Initialement toutes les fourmis artificielles sont positionnées sur un support noté f_0 (voir figure 1). A chaque itération, une donnée $\mathbf{z}_i$ est choisie dans la liste des données triée au départ. On notera par la suite par f_i chaque fourmi représentant une donnée $\mathbf{z}_i$ à classer dans l'arbre. Cette fourmi va chercher alors à se connecter sur sa position courante, sur le support (f_0) ou sur une autre fourmi (donnée) déjà connectée (noté f_{pos}). Cette opération ne peut aboutir que dans le cas où elle est suffisamment dissimilaire à f_+ (la fourmi connectée au support f_0 ou à f_{pos} et dont la donnée est la plus similaire à $\mathbf{z}_i$). Dans le cas contraire, la fourmi f_i associé à la donnée $\mathbf{z}_i$ se déplacera de manière déterministe dans l'arbre suivant le chemin le plus similaire indiqué par f_+. Le seuil permettant de prendre ces décisions ne va dépendre que du voisinage local. Pour étiqueter les données nous allons ensuite considérer que chaque sous arbre va représenter une classe trouvée. Dans (Azzag et al.) l'auteur détaille de manière plus complète les règles comportementales définissant les différents algorithmes de cette approche.

Notons qu'AntTree a l'avantage d'avoir une complexité proche du $n \log(n)$. Une étude détaillée a été réalisé dans (Azzag et al.), elle confirme que par rapport à d'autres algorithmes en n^2 les temps nécessaires par AntTree peuvent être jusqu'à 1000 fois inférieur à ceux de la

CAH sur de grandes bases de données et ceci pour une qualité égale.

Ces temps vont encore être réduits puisque AntTree s'applique sur l'ensemble des référents fournit par la carte topologique, $\mathcal{W} = \{\mathbf{w}_1, ..., \mathbf{w}_p\}$. Ceci réduit considérablement la complexité de la segmentation de la carte. Ainsi la structure d'arbre recherchée est celle qui représente le mieux l'ensemble des référent $\mathcal{W}$ (avec la nouvelle mesure de similarité (3)). Chaque nœud parent de l'arbre est plus représentatif du nœud fils. Ainsi l'algorithme, AntTree-SOM-Neigh-W, résumant les étapes élémentaires pour la segmentation de la carte topologique (SOM) peut être présenté comme suit :

- **Entrée** : $\mathcal{W} = \{\mathbf{w}_1, ..., \mathbf{w}_{N_c}\}$, l'ensemble des référents constituant la carte topologique à la fin de l'apprentissage.
  - Calcul de la matrice de similarité en utilisant la formule (3)
  - Construction de l'arbre avec l'algorithme AntTree.
- **Sortie** : structure des référents sous forme d'arbre.

L'arbre fournit une partition de la carte topologique $\mathcal{P} = \{P_1, ..., P_s\}$ où la valeur de l'indice s représente le nombre de sous arbres connectés au support fournit par AntTree. Ainsi, dans le même processus nous proposons de segmenter la carte et de fournir le nombre de sous ensembles constituant la partition sans aucun indice de qualité.

3 Validation

Afin de pouvoir évaluer la qualité de la classification obtenue, nous avons utilisé des bases de données comportant un nombre variable d'observations (Blake et Merz (1998)). Nous avons également testé notre approche sur des données artificielles (Azzag et al.), engendrées par des lois gaussiennes avec des difficultés diverses (recouvrement des classes, variables non pertinentes, etc.) ainsi que des données réelles. Le tableau 1 présente pour chaque base, le nombre de classes réelles (Cl_R), la dimension de l'espace des données (d) et le nombre de données total (N). La figure 2 montre quelques exemples avec des difficultés variables utilisés pour tester notre modèle.

Nous avons comparé notre modèle avec la segmentation de la carte utilisant l'algorithme AntTree combiné à l'indice de Ward (formule2), sans aucune information du voisinage (modèle appelé SOM-AntTree-W). Dans ces expérimentations, la comparaison des différents résultats est mesurée à l'aide de trois critères externes. On peut utiliser ces indices lorsque la segmentation souhaitée est connue, en particulier sur nos jeux de données. Il s'agit de la comparaison entre la segmentation proposée et une segmentation souhaitée. Ainsi nous avons utilisé, le taux de bonne classification (appelé aussi pureté) en utilisant l'étiquette connue de chaque donnée ; l'indice de Rand, qui calcule le pourcentage du nombre de couples d'observations ayant la même classe et se retrouvant dans le même sous ensemble après segmentation de la carte, et le troisième indice est celui de Jaccard qui est similaire à l'indice Rand sans prendre en considération les couples d'observations correctement classées dans des sous ensembles différents, (Saporta (2006)).

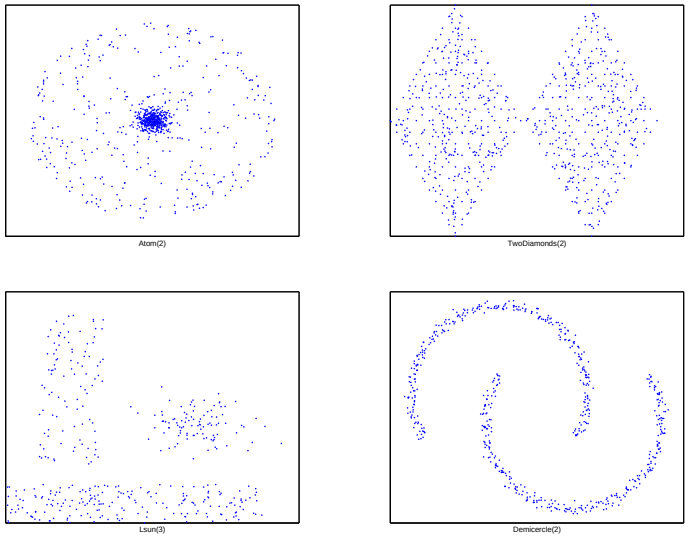

FIG. 2 – *Quelques exemples de jeux de données (Atom (2), TwoDiamon (2), Lsun (3), demi-cercle (2)). Le numéro qui suit le nom de la base indique le nombre de classe*

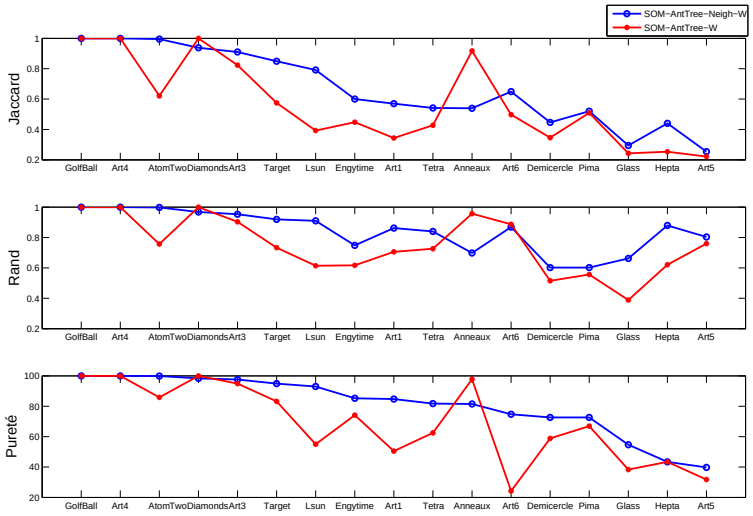

FIG. 3 – *Visualisation des différents indice de qualité : Jaccard, Rand, pureté*

Bases	Cl_R	d	N
Atom	2	3	800
Anneaux	2	3	1000
Dem-icercle	2	2	600
Engytime	2	2	4096
Glass	7	9	214
GolfBall	1	3	4002
Hepta	7	3	212
Lsun	3	2	400
Pima	2	8	768
Target	6	2	770
Tetra	4	3	400
Two diamonds	2	2	800
WingNut	2	2	1016
ART1	4	2	400
ART2	2	2	1000
ART4	2	2	200
ART5	9	2	900
ART6	4	8	400

TAB. 1 – *Jeux de données utilisées dans l'évaluation.* Cl_R *: le nombre de classes réelles ;* d *: la dimension de l'espace des données ;* N *: le nombre de données total*

Le tableau 2 indique les performances atteintes avec notre modèle AntTree-SOM-Neigh-W en comparaison avec le modèle utilisant simplement le critère de Ward. La figure 2 montre la variation des trois indices utilisés pour comparer ces deux segmentations de la carte.

Sur les graphiques de la figure 2, nous pouvons observer que les courbes obtenues par notre modèle sont de plus grandes amplitudes sur l'indice de Rand ainsi que l'indice de Jaccard. Ceci confirme l'avantage de considérer le voisinage dans AntTree-SOM-Neigh-W. Cependant notre modèle obtient de moins bon résultats sur la base Anneaux (le pic représenté sur les deux premières courbes). Ceci s'explique par le fait que notre modèle retrouve beaucoup plus de classes sur cette base. En effet AntTree-SOM-Neigh-W utilise l'information du voisinage et cette dernière lui permet de détecter des sous-ensembles de manière plus précise que le modèle classique.

Nous observons également sur le tableau 2 un résultat global qui indique que les puretés sont améliorées à chaque fois que l'on introduit le voisinage dans la segmentation de la carte. Par exemple, avec la base $pima$ de 67% à 72.4%, avec le même nombre de classes trouvées. Avec la base $Hepta$ on obtient des résultats identiques de l'ordre de 43.4%. Pour la base $Anneaux$ on constate une baisse de pureté, on passe de 97.8% à 81.5%.

Finalement, on peut constater, globalement, une claire amélioration de la pureté lorsqu'on utilise la nouvelle mesure de regroupement proposée (SOM-AntTree-Neigh-W). La prise en compte de la topologie de la carte améliore nettement les résultats de la segmentation. Nous rappelons ici, qu'il existe d'autres algorithmes de segmentation de la carte n'utilisant que la

Bases / %	SOM-AntTree-W	SOM-AntTree-Neigh-W
Atom (2)	85.87 (5)	99.9 (7)
Anneaux (2)	97.8 (6)	81.5 (5)
Demi-cercle (2)	58.833 (2)	72.67 (4)
Engytime (2)	74.14 (5)	88.04 (7)
Glass (7)	38.32 (5)	59.81 (6)
GolfBall (1)	100 (3)	100 (4)
Hepta (7)	43.4 (4)	43.4 (4)
Lsun (3)	55 (3)	93 (5)
Pima (2)	67 (5)	72.4 (5)
Target (6)	83.25 (5)	94.42 (6)
Tetra (4)	62.5 (3)	81.75 (5)
Twodiamonds (2)	100 (4)	100 (5)
WingNut (2)	95.67 (3)	87.11 (5)
ART1 (4)	50.5 (4)	84.75 (4)
ART2 (2)	94.9 (4)	97.7 (4)
ART4 (2)	100 (3)	100 (5)
ART5 (9)	31.78 (4)	50.33 (6)
ART6 (4)	24.25 (2)	78.75 (4)

TAB. 2 – *Comparaison entre AntTree-SOM-W, AntTree-SOM-Neigh-Wen utilisant l'indice pureté. Le numéro qui suit la pureté indique le nombre de sous ensembles de la partition trouvée. SOM :Self-Organizing Map (carte topologique) ; AntTree : classification hiérarchique inspirée des fourmis artificielles*

distance euclidienne tel que la CAH ou les K-moyennes, mais ces derniers nécessitent l'utilisation d'un indice de qualité afin de définir la taille de la partition optimale des référents, (Vesanto et Alhoniemi (2000); Vesanto et Sulkava (2002); Ambroise et al. (1998)). Avec notre modèle couplé à la nouvelle mesure 3 aucun indice n'est nécessaire pour obtenir la partition optimale. La figure 4 montre sur deux jeux de données le résultats de la segmentation de la carte. La figure à gauche indique comment la carte apprend la topologie du nuage suivi à droite de la structure de l'arbre fourni par l'algorithme de classification hiérarchique AntTree. Chaque classe est représentée par un sous arbre connecté au support. Aussi, chaque nœud de la carte et de l'arbre représente un référent qui est associé à un sous-ensemble de données.

4 Conclusions et perspectives

Dans ce travail nous avons développé une nouvelle mesure de similarité pour la segmentation des cartes auto-organisatrices afin de prendre en compte le voisinage entre les référents d'une carte. Nous avons également introduit une nouvelle approche de segmentation utilisant un algorithme de classification hiérarchique basé sur le principe des fourmis artificielles. En effet, ce dernier a l'avantage d'être très rapide tout en fournissant la taille de la partition en une seule étape. Par conséquent, le temps total de la segmentation de la carte est amélioré en comparaison aux autres méthodes qui parcourent l'ensemble des partitions possibles en choisissant une avec un l'indice de qualité optimal, (Vesanto et Alhoniemi (2000)). Lors de la comparaison avec la

distance euclidienne simple nous avons pu remarquer que notre nouvelle approche apporte des résultats compétitifs sur plusieurs bases de données.

Nous avons pu constater, qu'il existe des bases pour lesquelles notre méthode n'est pas efficace. Pour améliorer ces résultats, des perspectives peuvent être déduites. La première consiste à améliorer la mesure de similarité qui constitue un critère important dans la segmentation de la carte. En effet dans cette mesure nous devons tenir compte plus du voisinage et de l'inertie intra-classe, (Yacoub et al. (2001)). Par conséquent, il serait intéressant de vérifier si on retrouve les deux termes de notre mesure en calculant la perte d'inertie de la carte topologique. La seconde perspective concerne la méthode de segmentation qui a été utilisée, Il serait intéressant de penser à améliorer l'algorithme AntTree en l'hybridant avec une heuristique qui supprime les sous ensembles (classes) de petit effectif.

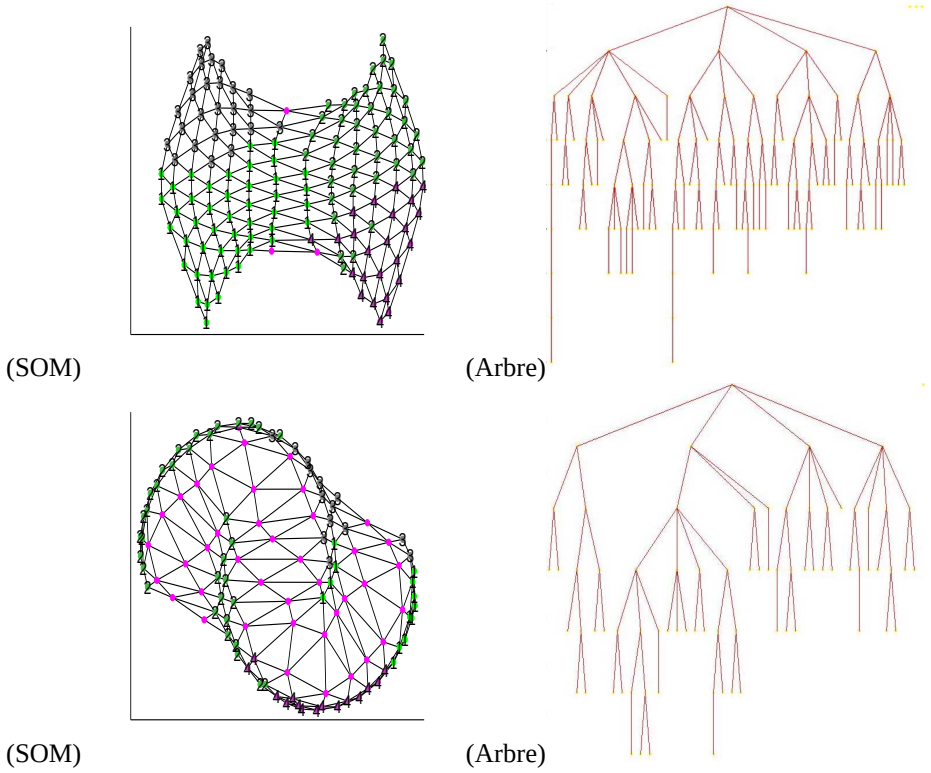

FIG. 4 – *Segmentation de la carte topologique SOM avec l'algorithme AntTree. Chaque numéro du noeud de la carte SOM représente une étiquette de la classe fournie par l'algorithme AntTree. Les bases utilisées sont : "TwoDiamont" et Demi-cercle*

Références

Ambroise, C., G. Séze, F. Badran, et S. Thiria (1998). Hierarchical clustering of self organizing map for cloud classification. *Neurocomputing 30*, 47–52.

Anderson, C., G. Theraulaz, et J. Deneubourg (2002). Self-assemblages in insect societies. *Insectes Sociaux 49*, 99–110.

Azzag, H., C. Guinot, et Y. . Venturini, Gilles". *Swarm Intelligence and Data Mining"*, Volume 34 of *Swarm intelligence and data mining, Studies in Computational Intelligence*, Chapter Data and text mining with hierarchical clustering ants, pp. 153–190. Springer-Verlag.

Bishop, C. M., M. Svensén, et C. K. I.Williams (1998). Gtm : The generative topographic mapping. *Neural Comput 10*(1), 215–234.

Blake, C. et C. Merz (1998). Uci repository of machine learning databases. technical report. Technical report, University of California, Department of information and Computer science, Irvine, CA, available at : *ftp ://ftp.ics.uci.edu/pub/machine-learning-databases*.

Jain, A. K. et R. C. Dubes (1988). *Algorithms for clustering data*. Prentice Hall advanced reference series :Computer Science.

Jain, A. K., M. N. Murty, et P. J. Flynn (1999). Data clustering : a review. *ACM Computing Surveys 31*(3), 264–323.

Kohonen, T. (2001). *Self-organizing Maps*. Springer Berlin.

Lebbah, M., N. Rogovschi, et Y. Bennani (2007). Besom : Bernoulli on self organizing map. In *International Joint Conferences on Neural Networks. IJCNN 2007, Orlando, Florida, August 12-17*.

Saporta, G. (2006). *Probabilités, analyse des données et statistiques*. Editions Technip.

Vesanto, J. et E. Alhoniemi (2000). Clustering of the self-organizing map. *Neural Networks, IEEE Transactions on 11*(3), 586–600.

Vesanto, J. et M. Sulkava (2002). Distance matrix based clustering of the self-organizing map. In *ICANN '02 : Proceedings of the International Conference on Artificial Neural Networks*, London, UK, pp. 951–956. Springer-Verlag.

Yacoub, M., F. Badran, et S. Thiria (2001). A topological hierarchical clustering : Application to ocean color classification. In *ICANN '01 : Proceedings of the International Conference on Artificial Neural Networks*, London, UK, pp. 492–499. Springer-Verlag.

Summary

In this paper, we present a new similarity measure used in clustering self-organizing map which will be reached using a new approach of hierarchical clustering. (1) The similarity measure is composed about two terms: the distance from weighted Ward and the Euclidean distance weighted by neighbourhood function. (2) An algorithm inspired from artificial ants named AntTree will be used to cluster self-organizing map. This algorithm has the advantage to take into account the neighbourhood between the referents and to provide a hierarchy of the referents with a complexity close to the $n \log(n)$. The SOM clustering including new measure is validated on several public data bases.