AMOEBA: HIERARCHICAL CLUSTERING BASED ON SPATIAL PROXIMITY USING DELAUNATY DIAGRAM

Vladimir Estivill-Castro, Ickjai Lee
Department of Computer Science,
The University of Newcastle,
NSW 2308, Australia.
Phone: (+612) 4921-6056, Fax: (+612) 4921-6929
Email: vlad@cs.newcastle.edu.au, ijlee@cs.newcastle.edu.au

ABSTRACT
Exploratory data analysis is increasingly more necessary as larger spatial data is managed in electro-magnetic media. We propose an exploratory method that reveals a robust clustering hierarchy. Our approach uses the Delaunay diagram to incorporate spatial proximity. It does not require any prior knowledge about the data set, nor does it require parameters from the user. Multi-level clusters are successfully discovered by this new method in only *O(nlogn)* time, where *n* is the size of the data set. The efficiency of our methods allows us to construct and display a new type of tree graph that facilitates understanding of the complex hierarchy of clusters. We show that clustering methods that adopt a raster-like or vector-like representation of proximity are not appropriate for spatial clustering. We conduct an experimental evaluation with synthetic data sets as well as real data sets to illustrate the robustness of our method.
Keywords: clustering, data mining, exploratory spatial analysis, Delaunay diagram

1 INTRODUCTION

Data gathering and data collection have long been one of the most expensive procedures in GIS (Geographic Information Systems). Developments in recent years have made these procedures more accurate, faster, easier and cheaper. As a result of this, we now face the challenge to find useful information and knowledge from the huge amount of geo-referenced data. Clustering is one of the central techniques in spatial analysis and spatial data mining (Ng and Han, 1994). GIS normally organizes data along many different themes or layers. Spotting and locating interesting groups in a particular theme is the starting point of exploratory data analysis and plays a critical role in exploratory investigations. Recently, a number of different clustering algorithms have been suggested in data mining (Ng and Han, 1994; Ester *et al.*, 1996; Zhang *et al.*, 1996; Agrawal *et al.*, 1998; Guha *et al.*, 1998; Wang *et al.*, 1999; Karypis *et al.*, 1999). They differ in their capabilities, applicability and computational requirements. Clearly no particular clustering method has been shown to be superior to all its competitors in all aspects. Typically, the problem is that clusters identified with one method can not be detected by other methods. Even a set of points declared noise by a certain method might be identified as a cluster by some other methods. We believe that this is due to two reasons. First, methods aim to be applicable in very general metric spaces. Second, methods adhere strongly to a clustering philosophy, based either on a hierarchical approach, a density approach or a nearest neighbor approach. Here, we concentrate on clustering geo-referenced 2-D point data, the most fundamental clustering task for exploratory data analysis in GIS. Then, we use the Delaunay Diagram (the dual of the Voronoi diagram) as our source of analysis. This is a

structure that is linear in the size of the data set but implicitly contains vast amounts of proximity information. For example, nearest-neighbors are efficiently computed. Our method combines hierarchical exploration with graph information and metric/density information to obtain remarkable robust clustering.

Today's digital spatial data sets are notoriously huge and complex. Further, their entities are not independent but spatially correlated and temporally correlated (Bailey and Gatrell, 1995). The unique nature of geo-referenced data impedes partitioning initial set of points into a number of mutually exclusive groups with similar, more homogeneous characteristics. GIS and data mining researchers have suggested criteria towards the ideal clustering approach. However, the main concerns are slightly different. The GIS community seems more interested in the quality of the clustering than the computational efficiency. For example, GAM (Openshaw, 1987) results in a robust exploratory tools, but it tests all circles, within a range of different radii, not to miss any possible clusters. As a result of this, it is neither realistic nor applicable to a very large data set. Data mining research seems more interested in enhancing time complexity by reducing either CPU or I/O costs (Ng and Han, 1994; Ester *et al.*, 1996), efficiency in a limited resources (Zhang *et al.*, 1996), subspace clustering (Agrawal *et al.*, 1998) or effectiveness in some sense (Guha *et al.*, 1998; Kang *et al.*, 1997; Karypis *et al.*, 1999; Openshaw, 1994).

Thus a clustering method for large sets of geo-reference data supporting exploratory analysis should meet requirements from both communities, GIS and data mining. These requirements are as follows:

- Multi-level, hierarchical, discovering clusters within clusters.
- Exploratory rather than confirmatory
- Incorporate spatial proximity
- Identify and manage local parameters rather than global
- No constraints (Minimize preconditions)
- Clustering with the whole data set rather than samples
- Efficiency (Fast enough) and effectiveness (Quality of clustering)
- Non-parametric (Do not impose models on the data, let the data speak first)

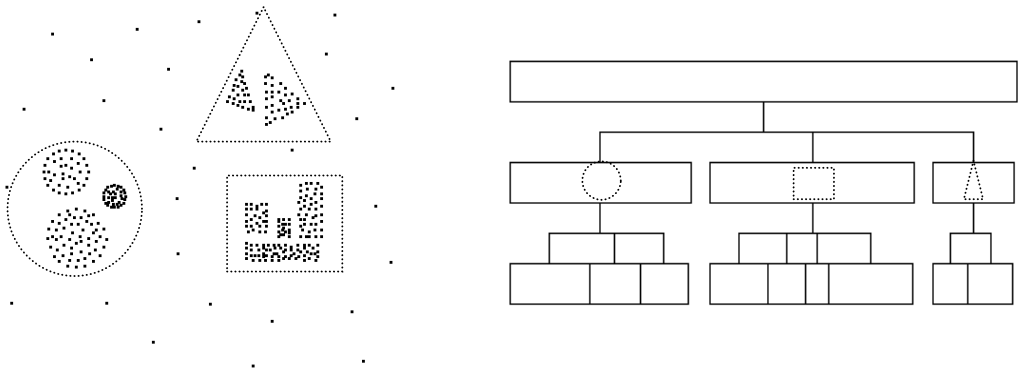
Current clustering methods fail to satisfy this combination of requirements. We present a clustering method named AMOEBA[1], which fulfills all these requirements. We explain in detail the suitability of AMOEBA for exploratory spatial analysis and spatial data mining. The remainder of this paper is organized as follows. Section 2 discusses these requirements in details. We examine the proximity graph compared to the conventional data models in section 3. In particular, we argue that it best describes spatial proximity. Section 4 discusses the inherent ambiguity of Delaunay triangulation and suggests solutions. Section 5 proposes the algorithm of the new clustering and analyzes its time complexity. In Section 6, we introduce a weighted dendrogram to help users understand the hierarchical structure while exploring data sets. The performance of AMOEBA is examined in Section 7. Finally, concluding remarks and future works are given in Section 8.

2 MORE ABOUT REQUIREMENTS

We now review in more detail the requirements outlined in the previous section. One of the unique characteristics of geo-referenced data is its generic complexity. Clusters may

[1] The clustering method suggested in this paper is named AMOEBA, since its process is similar to the reproduction of amoeba. The amoeba reproduces itself by dividing into two equal parts, a process known as binary fission. AMOEBA also reproduces itself by dividing into two parts in each level, one of which is of interest the other is not.

contain several numbers of sub-clusters. These sub-clusters may consist of smaller sub-clusters. This kind of hierarchy can be easily found in real world due to its hierarchical nature. For instance, a big city is usually surrounded by several numbers of small cities. Each small city may contain a number of towns in it. Therefore, a robust exploratory clustering method should be able to report clusters hiding in the data and reveal cluster hierarchy to help users explore and examine patterns at a different geographical scale.



(a) A point data set (n = 370, points represent Asian shops) (b) A cluster hierarchy

Figure 1. A point data set and its cluster hierarchy.

Intuitively, we can recognize that there are three large-scale clusters in Figure 1(a): a circle-shaped cluster, a triangle-shaped cluster and a rectangle-shaped cluster. Each large-scale cluster is composed of small-scale sub-clusters. For instance, the circle-shaped cluster consists of three sub-clusters. The diagram in Figure 1(b) reveals this hierarchy. With the aid of diagram, users not only recognize the total number of clusters, but also grasp easily the hierarchical structure of the data.

Definitions of what constitutes a cluster or noise vary from method to method. DBSCAN (Ester *et al*., 1996) takes a cloud of high density as a cluster while a maximal set of connected dense units in *k*-dimensions is a cluster for CLIQUE (Agrawal *et al*., 1998). Points merged based on a variety of criteria are clusters in bottom-up hierarchical clustering (Zhang *et al*., 1996; Guha *et al*., 1998; Karypis *et al*., 1999). A cluster is a group of objects, which have similar characteristics, more homogeneous among themselves that with respect to others, closer within themselves. Since clustering is for knowledge discovery, that is for revealing information and patterns in data, it should not be confirmatory, but exploratory (Openshaw, 1994). The methods should not impose a unique definition of neither cluster nor noise. In other words, exploratory approaches should not confirm whether a set of points is a cluster or not, but rather suggest all possible clusters. It is implicitly assumed in the clustering methods that clusters are of great importance for further analysis. Each cluster is of potential interest. Therefore, missing a cluster can cause incomplete and imprecise analysis. For example, the small-scale clusters in Figure 1(a) may represent concentrations of Asian shops while the large-scale cluster may represent suburbs. By missing out large-scale clusters, we are not able to analyze the relationship between suburbs and Asian shops. Similarly, it is no longer possible to examine the correct relationship between blocks and Asian shops without small-scale clusters. Naturally, detecting clusters is the fundamental task in clustering.

An essential characteristic of spatial entities is that they are correlated with each other in the sense of time, location, theme and attribute. As Tobler's famous proposition (1970) states: 'Everything is related to everything else, but near things are more related than

distant things'. Thus, proximity is critical to spatial analysis. Consequently, incorporating proximity is the main feature for successful spatial clustering.

In current methods the user supplies global parameters that influence the bias in the generalization of the clustering. These parameters are the number of clusters in the case of CLARANS (Ng and Han, 1994), but in other methods they are a density threshold to distinguish clusters from noise (Ester *et al*., 1996; Wang *et al*., 1999), or a threshold for controlling the granularity of clusters (Kang *et al*., 1997). Even in hierarchical clustering, user-supplied global parameters determine termination or merge conditions (Zhang *et al*., 1996; Guha *et al*., 1998; Karypis *et al*., 1999). However, these parameters are not easily determined without prior knowledge, hinders exploration. Unfortunately, slight changes to the parameter values translate into totally different results (Ankerst *et al*., 1999) (thus, raising suspicion on the clusters found because if so many results are possible, what is the true clustering structure in the data?). These global parameters should be avoided as much as possible. Moreover, it is unlikely that global parameters would ever be suitable. Geographical phenomena are the result of global order effects first and local order effect later (Bailey and Gatrell, 1995). Clusters, therefore, are the result of large scale or global effects and small scale or local effects. Clusters may share global first order effects, but have their own local second order effects. Global parameters should be used for a set of points only when the set of points is under the same circumstance. Thus, the use of global parameters should be minimized not only to allow for local variation, but for spotting true clusters.

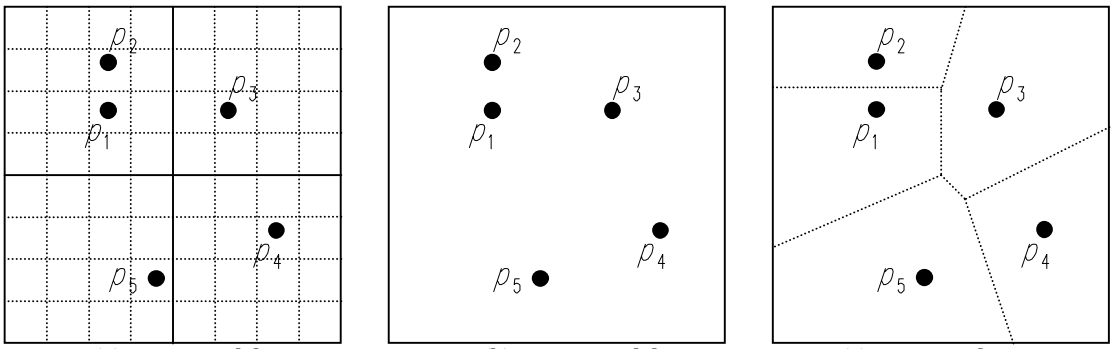
Openshaw (1994) emphasizes that subjectivity and precondition should be minimized in exploratory data analysis tasks. This means that constraints should not be imposed on the data set. His phrase 'Leave the data tell something' suggests that clusters should be stemmed from the data, not from the user. DBSCAN and GAM impose a certain radius of circle to find a high density of point set. Clustering in statistics assumes a certain mathematical distribution. These methods may produce incorrect or unexpected results when their assumptions are not met. Thus, assumptions should be minimized if they are inevitable.

Sampling is always at best not wrong, rather than right. It is used to represent the original data, typically to reduce CPU-time or computer memory requirements. With care, it may generate reasonable results. Given that every point is related to everything else, sampling can not represent everything in the original data. CURE (Guha *et al*., 1998) and ROCK (Guha *et al*., 1999) draw random samples from the database to reduce time complexity. However, one should notice that to gain confidence in the results, one would have to run the method several times, perhaps loosing such CPU-time gains. BIRCH (Zhang *et al*., 1996) performs a pre-clustering step to extract mathematical summaries that are stored in memory with a certain data structure called CF-tree. STING (Wang *et al*., 1997) and STING+ (Wang *et al*., 1999) also use a statistical information based on a grid to minimize the computational complexity. These compact values can be used for clustering to draw a good approximation. However, every member in the data set is not considered as equally important by use of either sampling or statistical summaries.

3 PROXIMITY GRAPH

Geographical data is represented by three basic topological concepts- the point, the line, and the area (Burrough, 1986). Despite of the fact that the point is the most primitive one, it is not easy to define point proximity as a discrete relation. Conversely, area data is

simple; two areas sharing a common boundary are neighbors. Similarly, two lines can be regarded as adjacent if they intersect each other. The spatial analysis of discrete non-connected objects (points) has been approached using distance concepts in the traditional vector and raster models[2]. As a result of this, the conventional data models are not able to hold a spatial adjacency properly for discrete objects (Gold, 1991).



(a) raster model (b) vector model (c) Voronoi diagram

Figure 2. The raster, vector and Voronoi spatial tessellation (n = 5).

The raster approach has good implicit adjacency relationships, but they are between pieces of space, not perceived map objects (Gold, 1991). As it can be seen from Figure 2(a), each point has 4- or 8-connectivity neighbors, which share a common boundary depending on 4- or 8-connectivity. As a result of this, the points p_1 and p_2 are considered as neighbors, but the points p_4 and p_5 are not considered as neighbors in the raster model. However, the adjacency relationship heavily depends on the size of cell. If the cell size become sixteen times bigger (the solid line) than the dotted line, then the two points p_4 and p_5 become neighbors in the bigger cell size. This causes an inconsistent definition of proximity. On the other hand, the topology in the vector model is usually dependent on intersection. Namely, the topology is detected where intersection occurs. Thus, since the five points in Figure 2(b) do not intersect, they are not considered as neighbors each other. Therefore, the vector approach does not model point data a discrete proximity relationship properly. Distance concepts are often suggested to overcome this problem. Points lying within a certain distance are regarded as neighbors. This works in some sense. However, the adjacency relationship is still inconsistent, similarly to the problems mentioned about the raster model. Neither clustering using the raster-like proximity (Wang *et al.*, 1997; Guha *et al.*, 1998, Wang *et al.*, 1999) nor clustering using the vector-like proximity (Openshaw, 1987; Ester *et al.*, 1996) properly incorporate proximity.

The Voronoi diagram has been proposed as an alternative way to model proximity amongst points, overcoming the limitations of conventional data models (Gold, 1991). The Voronoi diagram approach has a number of merits over the conventional data models. First, it represents topology (spatial adjacency) explicitly. If two points share a Voronoi edge, then they are treated as neighbors. This adjacency information is so fundamental that constitutes the dual known as the Delaunay triangulation. Each Delaunay edge represents topology between two points. Second, the various generalized Voronoi diagrams can model points having different weight (weighted Voronoi diagram), *n*-th nearest distance concept (*n*-th nearest Voronoi diagram) and various distance metrics (Minkowski and Karlsruhe Voronoi diagrams) (Okabe *et al.*, 1992). Further, Delaunay triangulation possesses a number of

[2] Two fundamental data models in GIS. The former is also called object-based or feature-based model, the latter is called tessellation-based or field-based model.

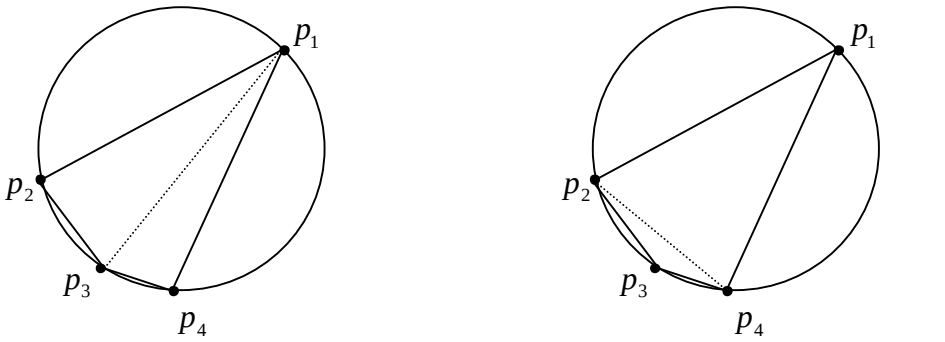
unique properties, which can be successfully used for clustering methods (Kang *et al*., 1997, Eldershaw and Hegland, 1997; Estivill-Castro and Houle, 1998). Moreover, it can be constructed in only $O(n\log n)$ time. The structure is succinct; the number of edges is linear in the number of points and the expected number of edges incident to point are limited to a constant value. Further, it is easy to extract the shape of clusters, which is useful for further analysis. In addition, the triangulation is equilateral as possible, so that the unexpected effect of long elongated edges can be minimized. Given these reasons, the Delaunay triangulation is a very solid candidate for extracting clustering information amongst discrete data objects.

4 DELAUNAY DIAGRAM AND DELAUNAY EDGES

In a proximity graph, points are connected by edges if and only if they seem to be close by some proximity measure (Liotta, 1996). If this simple rule is applied to clustering, two points belong to the same cluster if they are connected by a small enough Delaunay edge. Hence, analysis of Delaunay edges is very promising for clustering. However, this requires removing some ambiguities.

4.1 Problem with Delaunay Triangulation

In Delaunay triangulation, three points determine a Delaunay triangle if its circumcircle does not contain any other points in its interior. However, in the case of co-circular quadruples, this property becomes invalid. That is, when more than three points lie on a circumcircle (where all other points in the data set are outside the circle), a unique triangulation can not be defined. Whilst several possible alternatives of selecting Delaunay edges result in a valid dual of the Voronoi diagram, for the purposes of analyzing edges and proximity a problem arises.



(a) Triangulation with longer diagonal (b) Triangulation with shorter diagonal

Figure 3. Two possible triangulations when four points are cocircular (n = 4).

Consider Figure 3. There exist two possible triangulations when four points are cocircular. Proximity information and summary statistics are not the same in the two different cases. Two points p_2 and p_4 are not considered as neighbors in Figure 3(a), but they are in Figure 3(b). On the other hand, two adjacent points p_1 and p_3 in Figure 3(a) become unrelated in Figure 3(b). Furthermore, the sum (and thus, the average) of the lengths of Delaunay edges, the degrees (the number of edges incident to a point) and the standard deviation vary between the two cases. The Delaunay triangulation is non-informatory in cases were cocircularity occurs. Accepting any of the valid triangualtions inside the circle over the other alternatives would artificially introduce structure not really existing in the data. There are two remedies in order to overcome the problem. First, all possible edges are taken into account. In this case, the number of edges is no longer linear to the number of points. Moreover, it not only causes intersection between edges, but also more complex

structure. Secondly, diagonals are removed from the Delaunay triangulation. This Delaunay diagram makes the embedded planar map simple. If a face is not a triangle, it is because its points are cocircular. Thus, it eliminates the statistical side effects caused by triangles when cocircularity happens. This Delaunay diagram, constructed by removing all ambiguous diagonals in the Delaunay triangulation, is the base of edge proximity analysis in AMOEBA.

4.2 Perimeter, Angle and Edge

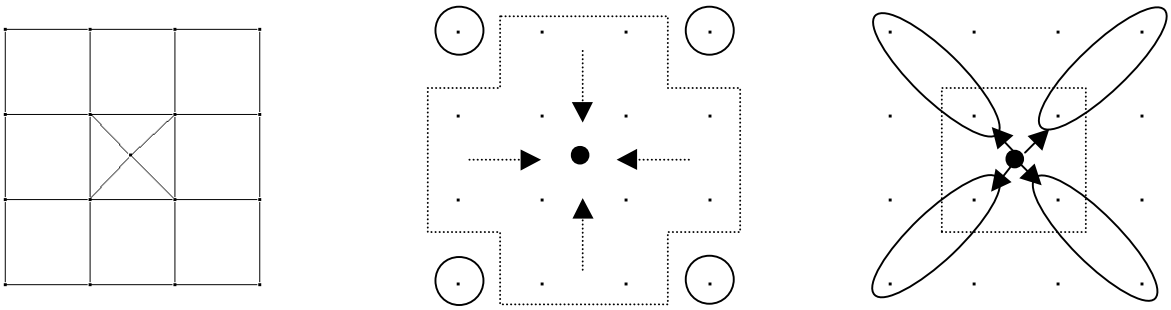
The Delaunay diagram has a number of unique characteristics and offers three direct alternatives for clustering based on edge analysis. These are perimeter (the sum of length of edges in a face), angle and length of Delaunay edge. Perimeter is of particular intuitive interest since the Delaunay preserves the empty circumcircle criterion (Okabe *et al*., 1992). Points determine a Delaunay face (interior) if and only if its circumcircle does not include any other points in the data set. Perimeter itself makes sense due to this reason, when perimeters are small, the points in the face most be close together and belong to the same cluster. If the perimeter is large, whilst the Delaunay equilaterality is maximized, it is because some point in the face is not in a common cluster. Unfortunately, angle and edge do not share this intuition because they directly do not determine the Delaunay diagram. However, as products of triangulation, they may encode significant proximity information (Eldershaw and Hegland, 1997).

Researchers (Estivill-Castro and Houle, 1998) have successfully derived the number of clusters using perimeter. The angle is widely used in point pattern analysis to decide whether a set of points is randomly distributed, regularly distributed or clustered (Boots, 1986). Edges are used to detect arbitrary shapes of clusters without requiring cluster numbers (Eldershaw and Hegland, 1997; Kang *et al*., 1997). Edges have a couple of advantages over perimeters for clustering. They are simpler than perimeters. Each edge represents an interaction between a pair of points while three points form each perimeter, complicating defining a discrete relation of proximity between a pair of points. Secondly, the length of edge inversely represents the strength of interaction between points (recall that distance decays effect (Bailey and Gatrell, 1995)). For instance, in Figure 3(a) the pair of points (p_2, p_3) is a stronger association (relation) than the pair of points (p_1, p_2).

5 AMOEBA

When there is no interaction or no correlation among points, the points do not form any significant clusters. In other words, there are no notably shorter Delaunay edges than the mean length of all edges. However, interactions result in data points that may either attract or repulse one another. Both attraction and repulsion will likely produce either clustered or regular distributions (note that physical models of attraction of repulsion are used for producing nice graph drawings (Eades *et al*., 1996)). If two points represent phenomena that attract each other, then they become closer so that they belong to the same cluster. If all endpoints of edges repulse one another, then the resulting distribution will be regular. Inversely, strong attraction among a certain number of points may bring about noise and powerful repulsion causes clusters. This is well explained in the aid of Figure 4. Figure 4(a) shows a Delaunay diagram with 17 regularly distributed points. If the point in the center in Figure 4(b) attracts all the points in a dotted cross-shaped geometric diagram except for four points within solid circles in the corner, then all the points within the cross-shaped diagram move towards the lure. Consequently, the four points within solid circles in Figure 4(b) appear as noise. This implies that noise is not only generated by repulsion with cluster members, but also by interactions among other points in the data set.

Similarly, if the center point in Figure 4(c) strongly pushes away four points in the dotted rectangle, then pairs of points within ellipse-shaped diagrams become clusters without any proper interactions amongst themselves. It is now obvious that both clusters and noise are generated not only by interactions between their neighbors, but also other members in the data set. Therefore, both all Delaunay edges (global effect) and neighbors (local effect) are taken into account in AMOEBA.

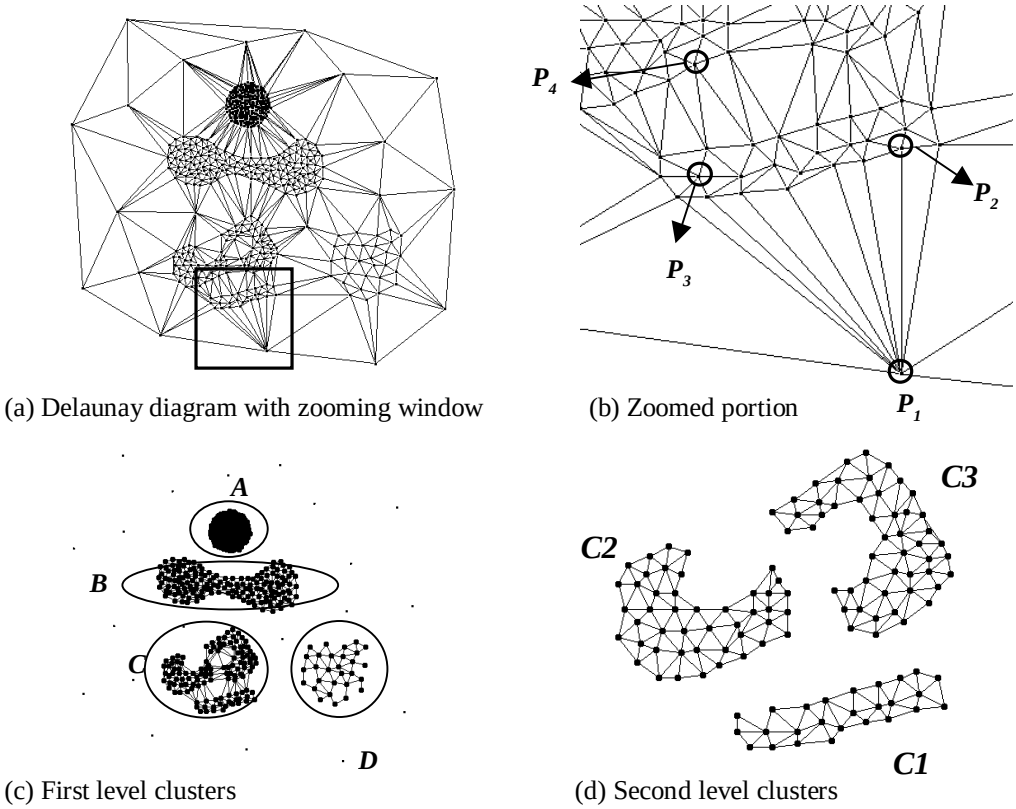


(a) A Delaunay diagram (n = 17) (b) Strong attraction (c) Strong repulsion

Figure 4. The effect of interaction.

5.1 Edge Analysis Criteria

Once the Delaunay diagram is constructed, clustering a set of points reduces to a one dimensional clustering problem. Namely, classifying Delaunay edges into two groups: edges are of interest when they are short, since they signal two points in the same cluster; alternatively edges are not of interest because they are too long. Note that this is the intuition behind single linkage clustering, which ensures a ratio between inter-cluster distance and intra-cluster distance. Edges which are not of interest should be removed, and then we can concentrate on finding sub-clusters within clusters developing further level of the clustering hierarchy.



(a) Delaunay diagram with zooming window (b) Zoomed portion

(c) First level clusters (d) Second level clusters

Figure 5. A data set with Delaunay diagram showing the principle of AMOEBA (n = 620).

Generally, the mean length of all edges is a reasonable cut-off value when clusters are well separated and density does not vary too much between clusters. However, this is not always the case. For instance, the high density of cluster ***A*** in Figure 5(c) reduces the mean length of edges to a value much smaller than some distances between points inside less dense clusters. This is because edges in the very high-density cluster are significantly shorter. There are four types of edges in Delaunay diagrams. There are edges within a cluster, between clusters, between cluster and noise and between noise points. The edges that should be removed at a certain level include all edges between cluster and noise, all edges between noise points and some edges between clusters. For instance, consider the four clusters (***A***, ***B***, ***C***, and ***D***) on the first level of a hierarchical cluster of points in Figure 5. Edges between clusters ***A*** and ***B*** are of no interest for the first level. However, edges between clusters ***C1***, ***C2*** and ***C3*** are of interest at this level to identify ***C***, as their parent cluster. These decisions are heavily dependent on the shape and distribution of points.

If points in clusters ***A***, ***B*** and ***D*** removed and the same number of randomly distributed points is inserted (dissolving these clusters), then clusters ***C1***, ***C2*** and ***C3*** become interesting clusters on the first level. Therefore, the standard deviation (representing the type of distribution), the sum of local edges (local effect) and the sum of total edges (global effect) are altogether used in AMOEBA to determine the filter values to identify clusters. Obviously, all edges incident to noise must be removed in all cases. One characteristic of noise is that the mean of edges incident to noise tends to be significantly greater than the global mean.

Table 1. The comparison between four points shown in figure 5(b).

Statistics \ Point	P_1	P_2	P_3	P_4	Global
# of edges	11	5	6	6	1807
Local mean	71.81	17.58	8.17	12.71	10.83

As shown in Table 1, a noise point P_1 has a local mean of incident edges approximately seven times greater than the global mean. On the other hand, The mean of point P_3, lying completely within cluster ***C1*** in Figure 5, does not deviate so much from the global mean. Similarly, a point P_4 lying on the border of cluster ***C2*** is relatively close to the global mean. Generally speaking, points lying on the border of clusters tend to have greater local mean than points completely within cluster, since border points have edges between clusters or edges between cluster and noise. Noise tends to have greater local mean than border points. Therefore, the ratio between local mean and global mean is a robust index for detecting noise. That is, we define the *noise index NI(p)* of a point *p* as follows:

$$NI(p) = \text{Local Mean} / \text{Global Mean} \quad (1)$$

where, Global Mean represents the mean of all points in the graph and Local Mean denotes the mean of points with edges incident to point *p*. Therefore, the noise index of a particular point implies its deviation from other points.

As mentioned earlier, the distribution (and density) of points also plays a role in deciding whether a set of points is a cluster or not. In order to make the cut-off value for short and long edges less sensitive to the variations in density within clusters, AMOEBA calculates a tolerance value that combines the standard deviation and the noise index. Relatively high cut-off value should be applied to edges of points lying within clusters in order not to

remove interesting interactions. Inversely, relatively small cut-off value is set for noise to eliminate all uninteresting edges. For this reason, the tolerance value is defined as follows:

$$T(p) = \text{Standard Deviation} / NI(p) \quad (2)$$

Finally, this edge analysis is captured in a criterion *function F(p)*. The cut-off value for edges incident in *p* is defined as the sum of both global mean and tolerance value.

$$F(p) = \text{Global Mean} + T(p) \quad (3)$$

In each level of hierarchy, edges greater than or equal to *F(p)* are eliminated and edges less than the criterion function survive. Surviving edges are used for detecting clusters in the next level.

5.2 Definitions

The following definitions are relative to a level in the clustering hierarchy.

Definition 1 *Passive Edge*: A Delaunay edge which is greater than or equal to the criterion function *F(p)* at a certain level. Passive edges are removed from the proximity graph, since they are not of interest any more. Thus, they are no longer used for further clustering.

Definition 2 *Active Edge*: A Delaunay edge that is less than the criterion function *F(p)* at a certain level. Active edges and points incident to them form a new proximity graph in each level of hierarchy. The newly created proximity graph is a subgraph of the previous graph and is used for detecting sub-clusters.

Definition 3 *Active Path*: A path in the current proximity graph where every edge in the path is an active edge.

Definition 4 *Cluster*: A set of points connected by active paths in the same level. Alternatively, If there exists an active path between a pair of points, the pair belongs to the same cluster.

Definition 5 *Noise*: A point that has no active edge incident to itself. In other words, all edges incident to noise are passive edges.

Definition 6 *Passive Cluster*: A cluster where there still exists an active path between any pair of points in the cluster after another further AMOEBA's splitting process. An AMOEBA splitting process is the classification into interesting (formally active edges) and non-interesting (passive edges). Passive clusters are the leaves of the clustering hierarchy because their points are in one connected component before and after the AMOEBA's binary fission. Actually, because of the statistical considerations in AMOEBA, there is no large difference among the strength of interactions amongst points in passive clusters; i.e. the splitting process does not generate meaningful sub-clusters. Thus, no further binary fission is required for passive clusters.

Definition 7 *Active Cluster:* A cluster where there does not exist an active path between all pairs of points in the cluster after another further AMOEBA's split. Here, a cluster has significantly varying internal interactions in some sense. The edge split is considered to generate meaningful sub-clusters. Thus, further "binary fission" is required to reveal the structure of active clusters. For instance, clusters ***A***, ***B*** and ***D*** in Figure 5(c) are passive clusters by the first level, while cluster ***C*** is an active cluster. The active cluster requires further binary fission towards the next level. As a result, the active cluster ***C*** produces meaningful sub-clusters ***C1***, ***C2*** and ***C3*** which are passive clusters in the next level; refer to Figure 5(d).

5.3 The Algorithm

procedure AMOEBA(*Graph*)
begin
  WriteCluster(*Graph*); //Write the *Graph* as a cluster

```
   NumberOfEdges = CalculateMeanAndStdev(Graph, GlobalMean, GlobalStDev);
   if (NumberOfEdges less than or equals to ONE)     return;  // Zero or one edge is incident
   for each node v in Graph do{
      EdgeList = Graph.adjacent_edges(v);                  // Extract edges incident to node v
      LocalMean = CalculateLocalMean(EdgeList);
      for each edge e in EdgeList do{
         ToleranceValue = GlobalStDev * GlobalMean / LocalMean;
          if (e.distance()  > (GlobalMean + ToleranceValue))
             Graph.delete_edge(e);
      }
   }
   if (Graph.degree(v) equals to zero)
      Graph.delete_node(v);                              // Eliminate noises
   ConnectedComponents(Graph, ComponentNumber); // Calculate connected components
   for each connected component c do{
      SubGraph = ConstructSubGraph(Graph, ComponentNumber, c);
      if (NumberOfEdges not equals SubGraph.number_of_nodes())
        AOEMEBA(SubGraph);
   }
end
```

Construction of the Delaunay diagram is the first step. The diagram is a connected planar plane-embedded graph passed to the AMOEBA algorithm. Recursively, all points in a connected components are reported as a cluster. Thus, every edge is tested for the criterion function in Equation (1) where the global values are adjusted to the current level. After eliminating passive edges and noise only active edges remaining in the diagram, but may define new connected components. AMOEBA' reproduction algorithm is called for each connected component unless no new connected components are created from the passive cluster and the end of the recursion.

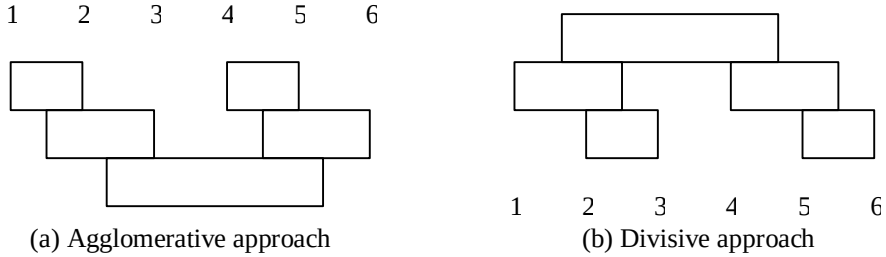
5.4 Time Complexity

Time complexity is of great importance, especially for very large data set. AMOEBA is specially designed for clustering of spatial data. AMOEBA requires $O(n\log n)$ time for constructing the Delaunay diagram in the initialization phase. Once the diagram is constructed, AMOEBA repeatedly calls itself until termination conditions are met. At the root level of the clustering hierarchy, the input number of edges is $O(n)$, since it is linear to the number of input points (n). In each proximity graph, testing all points for the criterion function (1) takes $O(n)$ time complexity since extracting local edges for each point only takes constant time. Since computing connected components is linear to the sum of both number of edges and number of points, this step only requires $O(n)$ time. Therefore, each AMOEBA procedure requires O(n) time complexity. The structure of AMOEBA's binary fission is similar to that of binary tree, removing a constant fraction of the edges at each level (the cut-off value is very likely to be close to the median of the edge lengths). Thus, the expected depth is at most $O(\log n)$. Consequently, the time complexity of proposed clustering is $O(n\log n)$ time. This compares extremely favorably with other clustering methods for spatial clustering.

6 WEIGHTED DENDROGRAM

Hierarchical clustering is traditionally represented by a dendrogram. Each node in the tree graph represents a cluster. In agglomerative clustering, each cluster consists of only a

single point in the first level. These primitive clusters are successively merged into super clusters until the whole data set becomes a cluster. On the other hand, the whole data set represents a cluster in the first level in partitioning clustering. This cluster is repeatedly divided into sub-clusters until each point forms a cluster.



(a) Agglomerative approach (b) Divisive approach

Figure 6. Two Dendrograms (n = 6).

Dendrograms display the hierarchy of clusters in some sense. However, as the number of points increases, the depth of tree becomes longer. This not only decays the readability of the display, but requires more space to fit it. In addition, it is difficult to figure out the ratio of noise at each level. Moreover, the relative size of a cluster is not easily recognized. It is also hard to compare the significance of either intra-level clusters or inter-level clusters. For these reasons, dendograms do not represent well the hierarchical structure revealed by AMOEBA, especially when early stops (homogeneous clusters) occur. These drawbacks are resolved by incorporating weights to clusters into the dendrogram. Weights are explained by the length of rectangles in the diagram. This diagram is called Pendrogram[3] . Recall that points having no active edges are noise (Definition 5 in Section 5.2). Therefore, two points connected by a single active edge form a leaf in the Pendrogram. With the aid of Pendrograms, users can easily count both the total number of clusters and the number of intra-level clusters. For instance, Figure 7 suggests 8 clusters in total, 4 clusters in the first level and 3 clusters in the second level. Analogies and comparisons between clusters sizes and level can easily reveal the relative significance of clusters. The size of noise creates a gap (or mismatch) in each level, which equally separates intra-level clusters.

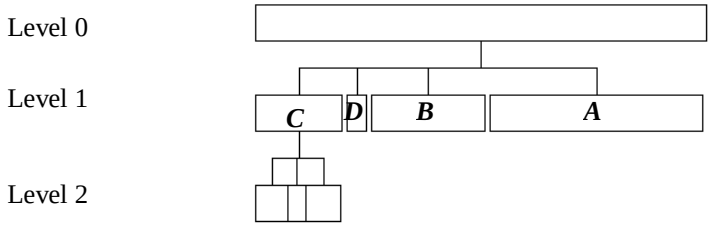


Figure 7. Pendrogram (n = 620, the points are the same as those in Figure 5).

As illustrated in Figure 5(c), noise is significant after the first AMOEBA's binary fission. Thus, clusters in the first level are separated with a large gap. On the other hand, the split of cluster C in Figure 5(d) generates no noise. Therefore, sub-clusters ***C1***, ***C2*** and ***C3*** are separated without gaps in Figure 7. Pendrograms have a number of merits over traditional dendrograms. They enhance the readability of the display. In addition, they suggest the distribution of interactions within clusters. Interactions in clusters ***A***, ***B*** or ***D*** are considered as homogeneous, since they do not derive any sub-clusters. Meanwhile, interactions within cluster ***C*** are to be considered heterogeneous. Further, it makes possible

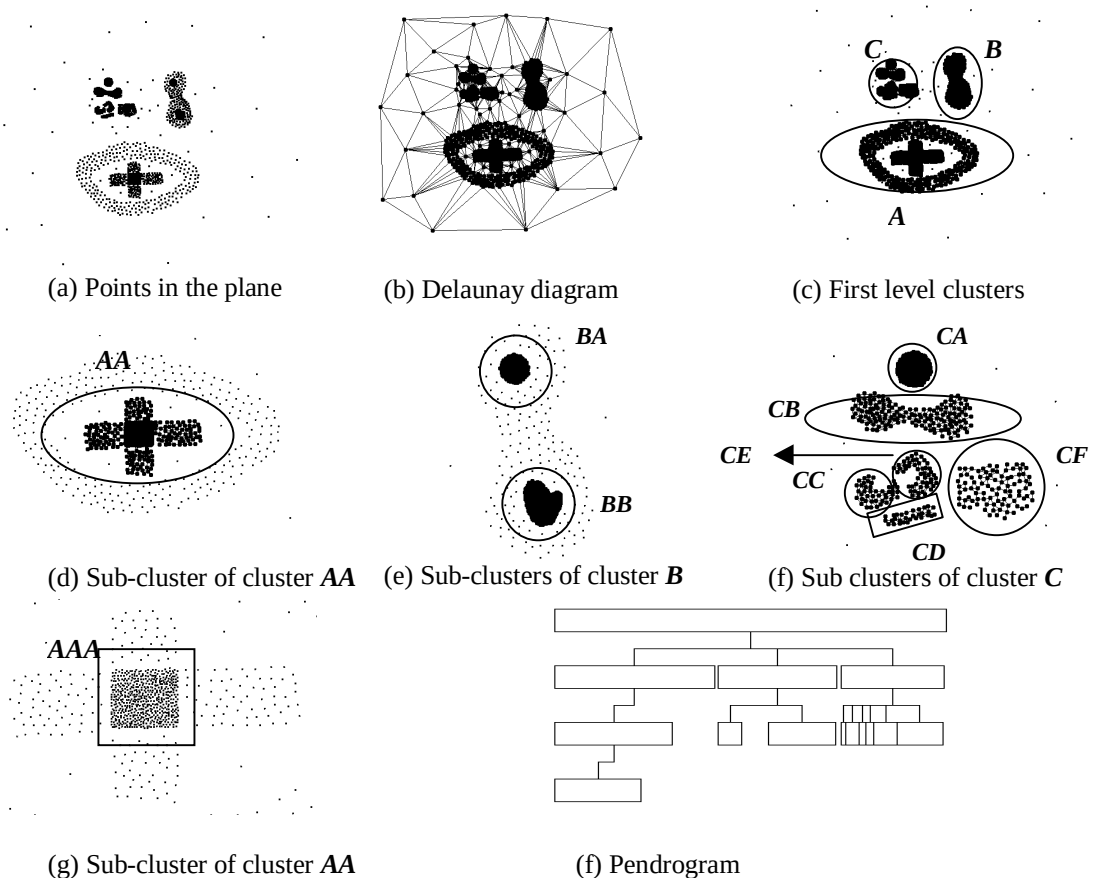
[3] Pendrogram denotes weighted dendrogram; from the Greek pento (meaning weight) and dendrogram.

to compare or contrast between intra-level clusters or between inter-level clusters. This is much simpler and illustrative than dendrograms.

7 PERFORMANCE EVALUATION

We used a synthetic data set and a real world data set for illustrating the robustness of AMOEBA. The remarkable results are provided in the following sub-sections.

7.1 Synthetic Data set



(a) Points in the plane (b) Delaunay diagram (c) First level clusters

(d) Sub-cluster of cluster ***AA*** (e) Sub-clusters of cluster ***B*** (f) Sub clusters of cluster ***C***

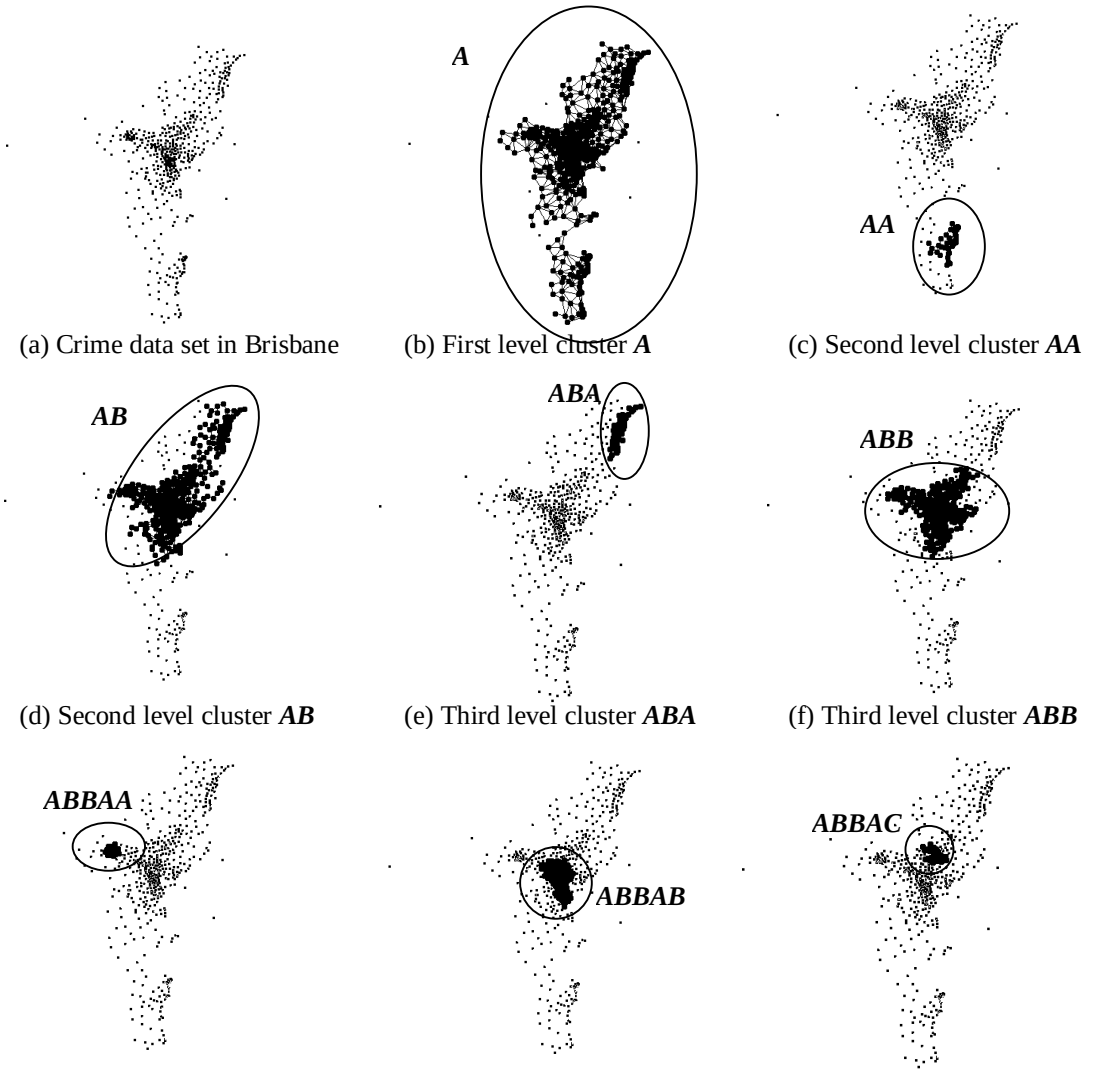
(g) Sub-cluster of cluster ***AA*** (f) Pendrogram

Figure 8. A synthetic data set (n = 2500).

As shown in Figure 8(c), three clusters ***A***, ***B*** and ***C*** are identified in the first level. These clusters are active clusters; thus they are further split. All clusters produced in the second level (***BA***, ***BB*** and ***CA*** ~ ***CF***) are passive clusters except for the cluster ***AA***. The active cluster in this level produces its sub-cluster ***AAA***. Consequently, AMOEBA correctly suggests 14 clusters. Obviously, AMOEBA spots not only arbitrary shape of clusters, (that is, not only convex clusters as opposed to K-means), but different density clusters.

7.2 Real Data set

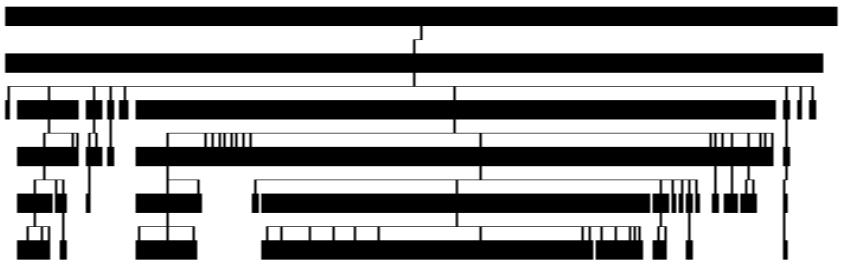
Despite of the complex structure of real data set, AMOEBA discovers all possible clusters hidden in the data set. For instance, a cluster ***A*** is identified in the first level as shown in Figure 9(b), directly overlaying the main urban area. This active cluster is further partitioned into two clusters ***AA*** and ***AB*** in the next level. All possible clusters appear in the Pendrogram in Figure 9(j). Users are more interested in relatively stronger clusters. Clusters with more edges (interactions) are easily selected by simply choosing relatively larger rectangles in the Pendrogram.



(a) Crime data set in Brisbane (b) First level cluster ***A*** (c) Second level cluster ***AA***

(d) Second level cluster ***AB*** (e) Third level cluster ***ABA*** (f) Third level cluster ***ABB***

(g) Fifth level cluster ***ABBAA*** (h) Fifth level cluster ***ABBAB*** (i) Fifth level cluster ***ABBAC***



(j) Pendrogram from level 0 to 5

Figure 9. Crime data set from Brisbane, Australia (n = 494).

8 FINAL REMARKS

AMOEBA is a clustering method especially designed for spatial data sets. It incorporates global first-order effects and local second-order effects. Hence, it is less sensitive to noise, outliers and the type of distribution. High densities in small areas affecting global indicators do not misguide it. It does not require any parameters from user and generates multi-level clusters. High quality clustering is reported, since AMOEBA incorporates proper spatial proximity and gradually analyzes the entire data set. AMOEBA strategy is accompanied with a visualization of the resulting clustering hierarchy named Pendrogram. All this within $O(n\log n)$ expected time.

Phenomena geographically coincide in the real world. There is no reason why a clustering hierarchy must be a partition. A variety of types of crime may occur in the same railway station or more than two shopping centers may coincide in a large geographical scale. Further work will consider clustering with coincident points. Future works also include finding relationship between themes and the use of information on correlated themes for determining better termination conditions. Incorporating *n*-nearest neighbors or other estimates (perimeter, angle and area) attracts our interests as well. Clustering weighted points using weighted Voronoi diagram is also of interest.

9 ACKNOWLEDGEMENTS

This research is partly supported by a grant from the University of Newcastle Postgraduate Research Scholarship.

10 REFERENCES

Agrawal, R., Gehrke, J., Gunopulos, D. and Raghavan, P. (1998). "Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications", *Proceedings of the ACM SIGMOD International Conference on Management of Data*, June, 1998, Seattle, Washington, 94-105.

Ankerst, M., Breunig, M. M., Kriegel, H. -P. and Sander, J. (1999). "OPTICS: Ordering Points To Identify the Clustering Structure", *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 31 May - 3 June, Philadelphia, Pennsylvania, 49-60.

Bailey, T. C. and Gatrell, A. C. (1995). *Interactive Spatial Analysis*, Wiley: New York.

Boots, B. N. (1986). "Using angular properties of Delaunay triangles to evaluate point patterns", *Geographical Analysis*, 18(3), 250-260.

Burrough, P. A. (1986). *Principles of Geographical Information Systems for Land Resources Assessment*, Oxford: New York.

Eades, P., Lin. X. and Tamassia, R. (1996), "An algorithm for drawing a hierarchical graph", *International Journal of Computational Geometry and Applications*, 6(2), 145-155.

Eldershaw, C. and Hegland, M. (1997). "Cluster Analysis using Triangulation", in Noye, B. J., Teubner, M. D. and Gill, A. W. (eds.), *Computational Techniques and Applications: CTAC97*, World Scientific: Singapore, 201-208.

Ester, M., Kriegel, M. -P., Sander, J. and Xu, X. (1996), "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, 2-4 Auguest, 1996, Portland, Oregon, 226-231.

Estivill-Castro, V. and Houle, M. E. (1998), "Robust Clustering of Large Geo-referenced Data Sets", *Proceedings of the 3rd Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 26-28 April, 1999, Beijing, China, 327-337.

Gold, C. M. (1991). "Problems with handling spatial data - The Voronoi approach", *CISM Journal ACSGC*, 45(1), 65-80.

Guha, S., Rastogi, R. and Shim, K. (1998). "CURE: An Efficient Clustering Algorithm for Large Databases", *Proceedings of the ACM SIGMOD International Conference on Management of Data*, June, 1998, Seattle, Washington, 73-84.

Guha, S., Rastogi, R. and Shim, K. (1999). "ROCK: A Robust Clustering Algorithm for Categorical Attributes", *Proceedings of the Internatioal Conference on Data Engineering*, 23-26 March, 1999, Sydney, Australia, 512-521.

Kang, I., Kim, T. and Li, K. (1997). "A Spatial Data Mining Method by Delaunay Triangulation", *Proceedings of the 5th International Workshop on Advances in Geographic Information Systems (GIS-97)*, November, 1997, LasVegas, Nevada, 35-39.

Karypis, G., Han, E. and Kumar, V. (1999). "CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling", *IEEE Computer: Special Issue on Data Analysis and Mining*, 32(8), 68-75.

Liotta., G. (1996). “Low Degree Algorithm for Computing and Checking Gabriel Graphs”, *Report No. CS-96-28*, Department of Computer Science in Brown University, Providence.

Ng, R. and Han, J. (1994). "Efficient and Effective Clustering Method for Spatial Data Mining", *Proceedings of 1994 International Conference on Very Large Data Bases (VLDB'94)*, September, 1994, Santiago, Chile, 144-155.

Okabe, A., Boots, B. N. and Sugihara, K. (1992). *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, John Wiley & Sons: West Sussex.

Openshaw, S. (1987). "A mark 1 Geographical Analysis Machine for the automated analysis of point data sets", *International Journal of GIS*, 1(4), 335-358.

Openshaw, S. (1994). "Two exploratory space-time-attribute pattern analysers relevant to GIS", in Fotheringham, S. and Rogerson, P. (eds.), *Spatial Analysis and GIS*, Taylor and Francis: London, 83-104.

Tobler, W. R. (1970). "A computer movie simulating urban growth in the Detroit region", *Economic Geography*, 46(2), 234-240.

Wang, W., Yang, J. and Muntz, R. (1997). " STING: A Statistical Information Grid Approach to Spatial Data Mining", *Proceedings of the 23rd VLDB Conference*, August, 1997, Athens, Greece, 186-195.

Wang, W., Yang, J. and Muntz, R. (1999). "STING+: An Approach to Active Spatial Data Mining", *Proceedings of the Internatioal Conference on Data Engineering*, 23-26 March, 1999, Sydney, Australia, 116-125.

Zhang, T., Ramakrishnan, R. and Linvy, M. (1996). "BIRCH: An Efficient Data Clustering Method for Very Large Databases", *Proceedings of the ACM SIGMOD International Conference on Management of Data*, June, 1996, Montreal, Canada, 103-114.