Spatial Object Aggregation Based on Data structure, Local Triangulation and Hierarchical analyzing method

Liu Yaolin, Martien Molenaar Menno-Jan Kraak
Department of Geo-Informatics
International Institute for Aerospace Survey and Earth Science
7500 AA, Enschede,The Netherlands
E-mail:{yaolin,Molenaar,Kraak}@itc.nl

Abstract

Aggregation for spatial objects plays an important role in database or model generalization in GIS since reducing spatial complexity is one of the major aims of model generalization. The aggregation operation can be considered as a geometrical operation under the control of semantic and spatial relation of objects. The process of spatial object aggregation is the process of spatial, semantic analyzing, decision-making and the implementation of operation. This process needs not only the geometric and thematic attributes of the objects and also the spatial and semantic relationship among the objects. It also needs the supporting data structure and analyzing tools in order to make analyzing, decision-making and computation easy in aggregation process. There is still lack of studies on how data structure supports aggregation process and also lack of the method for semantic analysis in aggregation process. This paper is to focus methods and process of spatial aggregation based on semantic and geometric characteristics of spatial objects and relations among the objects with the help of spatial data structure (Formal Data Structure), local constrained Delaunay triangulations and semantic hierarchy. The adjacent relation among connected objects and unconnected objects has been studied through constrained triangle as elementary processing unit in aggregation operation. The hierarchical semantic analytical matrix is given for analyzing the similarity between object types and between objects (including different levels). It will govern and guide aggregation process. The several different cases of aggregation have been presented in this paper.

Key words: Aggregation, spatial object, Hierarchy, Data Model.

1. Introduction

Aggregation for spatial objects can serve as basic tools for the generation at multiple representations of geo-data within the context of database generalization processes. Aggregation process for spatial objects needs only not analyzing and understanding semantic, geometric characteristic of spatial objects and the relations among them, but also supporting data structures. There is still lack of studies on how data structure supports aggregation process and also lack of the method for semantic analysis in aggregation process although there are several data structures used in the multiple representation of geo-database such as R-tree, B-tree, SDS and so on. The data that are organized by some data structure will be more effective and efficient to be analyzed and measured. The data structure plays an important role in identifying and evaluating conflicts and analyzing spatial relations among objects. In this research, we present a data model which based on Formal Data Structure (FDS) (Molenaar 1989a, 1991) and Local Constrained Delaunay Triangulation. The semantic relation matrix between different objects and between different object types will govern or guide aggregation process.

Aggregation process for spatial object can be divided into two steps: First, the semantic analysis of spatial objects: the main task of this step is the semantic similarity matrix established by expert knowledge based on the aggregation hierarchy and classification hierarchy as well as the requirements and purposes of database generalization. It is an application-dependant. An aggregation hierarchy plays a key role in linking the definition of spatial objects at different scale levels (Molenaar 1996, Peng 1997, Richardson 1993 and Smaalen 1996). It shows how composite objects can be build from elementary objects and how these composite objects can be put together to build more complex objects and so on (Hughes, 1991; Molenaar, 1993). Second, the geometric analysis and operation of spatial objects: the main task is analyzing the geometric characteristic of spatial objects, the spatial relations among the objects and implementing operation to objects.

In this paper, we introduce briefly some basic concepts which will be used in aggregation process in section 2. After introducing the properties of Formal Data Structure and Constraints Delaunay Triangulation we develop a data model which is an integration FDS and CDT in section 3. The hierarchical semantic matrix for aggregation is given in section 4. Some aggregation operations and examples of aggregation are described in section 5 and 6. The paper end with the conclusion in section 7.

2. Elementary Object, Elementary Object Type, Composite Object, Composite Object Type and Aggregation Hierarchy

Some of concepts should be made clear in aggregation process.

Elementary Object in spatial model is real world object that contains both thematic and geometric information and is normally represented in a database by means of an "object identifier" with associated thematic and geometric data. These elementary are divided into three types (point, line and area) based on their geometric attributes. Objects can also be categorized according to their thematic attributes. The definition of elementary object in a database (should be done before a database can be built) depend mainly on four factors as following:

- Application discipline
- User context
- Aggregation level or scale or resolution
- Classification level

On each level, different elementary objects are relevant. Elementary objects at one level may be aggregates of elementary objects at another level.

Elementary Object Type is an abstraction that represents a class of similar elementary objects. This means that the elementary objects in spatial model that have common pattern of both state and behavior within the framework of an application may be grouped into elementary object types. The elementary object type in turn may be organized into super object types and so on. An elementary object is instance of some elementary object type. Elementary object types together with the classification and aggregation hierarchies are important aspects in semantic data modeling and play a critical role in defining the concept of database generalization (Molenaar 1996).

Composite Object is built from elementary objects that belong to different elementary object types. This means that the elementary objects are the constituents of composite object. Similar to elementary objects, composite objects at one level may be aggregates of composite objects at another level.

Composite Object Type is also an abstraction that represents a group of similar composite objects. An instance of the composite-object-type is referred to as a composite-object. A composite-type can be the elementary-type of another (super) composite-type. For example, object type Farm is a combination of the types Yard and Field. In other words, Yard is part of Farm, and so is the Field.

Aggregation Hierarchy shows how lower-order object types are combined to form a higher-order object type. A higher-order object type in the hierarchy is composite object type, whereas an object type that is part of the composite-type is elementary object type. This aggregation hierarchy has the following characteristics:

- Defining elementary objects;
- Composite object types in the hierarchy corresponding to higher abstraction levels;
- Elementary object types corresponding to lower abstraction levels;
- Specifying the elementary object types of elementary objects building a composite object of this type;
- The upward relationships PARTOF connecting a group of elementary objects with a certain composite object.
- Expressing the relationship between a specific composite object and its constituent parts at different levels.
- Replacing the elementary object types in a model with their composite-type will result in transforming the model from a lower abstraction level to a higher abstraction level;

An aggregation hierarchy has therefore a bottom-up character, in the sense that the elementary objects from the lowest level are combined to compose increasingly composite objects as one ascends in the hierarchy. The definition of aggregation hierarchy is application-dependant and it must be established before aggregation process. The different applications have different aggregation hierarchies, even though building these aggregation hierarchies are based on the same data set or thematic classification system. Suppose that we have a land use database in which the contents of the database can meet the requirements of land management at several levels and land evaluation for different land use at several levels. In order to meet different level management and different purpose of land evaluation at different levels, the database needs to be generalized corresponding with management level and evaluation level based on different aggregation hierarchy. The aggregation hierarchy used for land use management is completely different from the aggregation hierarchy for land evaluation. The same database for different land use evaluation at different levels will have different aggregation hierarchy for each one as well.

3.Supporting Data Model To Aggregation

The semantics relations and the spatial relations between objects must be not only analyzed but also geometric operations be implemented in the aggregation process. A spatial object may be influenced by neighboring objects with no restriction on the distance of separation. Now matter how far two object are separated, one can affect the other as long as they are neighbors. Such as:

- If two parcels have the same land use or high similarity and have adjacent relation between them, then aggregate them. This involves the geometric operation.
- Parcel A which violates the geometric constraint (too small in area) have adjacent relation with several land parcels. One of this set of land use parcels having the same land use or high similarity as parcel a should be selected out and aggregate with parcel A to form a new parcel. This involves the semantic analysis among objects and geometric operation.
- A set of spatial objects have regular pattern in space database, How to identify them from semantic and spatial aspects, then aggregate them. This involves semantic and spatial analysis and geometric operation.

Before aggregation process, the spatial objects with three components: the distance, the direction and the state of neighbors need to be examined and the semantic relations among the objects need to be analyzed. After aggregation, the geometric and textual properties of spatial objects need to be changed and the spatial relations need to be maintained dynamically. Dynamic maintenance of adjacent relations is critical issue in aggregation process. All these examining and analyzing show that aggregation need data model to support. If there is no data model to support these analysis and operations, then a process involving a probably heavy computation and complicated algorithm is necessary in order to complete aggregation process. A data model, to be adequate for automated aggregations, should provide the basis for describing spatial objects, and the topological relationship among them, through a well defined set of geometric primitives. Since aggregation decision-making relies on both spatial information and thematic information, the model should also indicate how the geometric aspect of a spatial object is linked with its thematic aspects.

Formal Data Structure model (FDS) for single valued vector maps developed by Molenaar (1989, 1991, 1995a), is an object-oriented topological (conceptual) data model. Although The FDS supports a number of elementary topological relationships, it does not support the spatial adjacency relationship among objects that are disconnected from each other. Topological relationships among "disconnected objects" are important to support spatial analysis and geometric operations that involve these kinds of objects.

Constrained Delaunay Triangulation (CDT) can be used for defining adjacent relation among disconnected objects, and conflict detection and displace of spatial objects, finding nearest neighboring object to a given object in generalization (Ware et al 1996, Chris B. Jones et al 1998, Wanning Pan 1997 etc). The CDT can also be used to measure the spatial relations such as measuring disjoint relation, distance relation and direction relation. CDT can be as the geometric primitives of the data model. For category database generalization, CDT is very useful to analyze and measure local spatial relationship but not to organize the whole data set since a simple area object will consist of a lot of triangles that will lead to date redundant too much and also lead to the difficulty of semantic analysis among objects.

Aggregation process needs not only an analysis process of database concerned from local to whole but also from whole to local. The data model should support this analysis process.

We combine the advantage of FDS and CDT to develop a data model which is integration of FDS and CDT in aggregation process. The data model is shown in Figure 1. Figure 2 shows that logical structure of geo-database is organized in database based on data model.

In structure modeling, the objects in database are represented as an object class Geo_class, which is super class of point, line and area objects. It can contain some attributes and member function such as Distance () and Direction ().The data model consists of:

- three object types, namely point object, line object, area object, classified according to the geometric description of spatial object;
- five geometric data types (geometric primitives), including coordinates, node, edge, triangle and face, the definition of which is based on planar-graph theory at node-arc level;
- a set of links between geometric data types(g-g links), and a set of links between geometric data types and object types (g-f links). It supports a number of elementary topological relationships, including area-area, line-line, point-point, area-line, area-point, and line-point relationships.

Assumed that there are three objects O1,O2and O3 as shown in Figure 3, In a constrained Delaunay triangulation T, every boundary vertex of each object o_i corresponds to a triangle vertex and every boundary edge of each object o_i serves as a constraining edge. Each object o_i is defined by an unique object identifier and references to the triangles of T which lie within its boundary; each triangle t_i of T is described by an unique triangle identifier, references to each of its three constituent edges, plus a reference to the object within which it lies; each edge e_i is described by an unique edge identifier and references its start and end vertices; and each vertex v_i stores an unique vertex identifier plus x and y co-ordinate values. Plus supplementary topological information in the form of reference to the two triangles to which the edge belongs. If a triangle t_i lies within an object o_i then t_i is said to belong to..

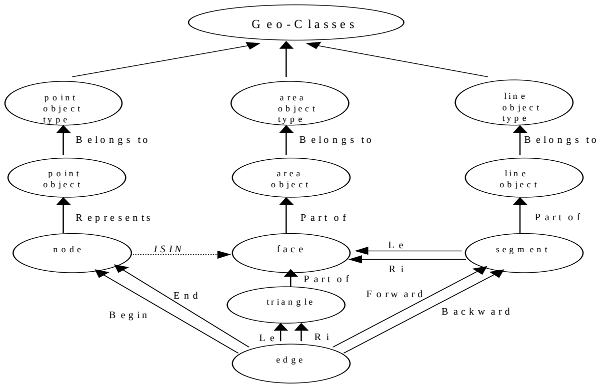


Figure 1.The Data model for aggregation(modified from Molenaar 1998)

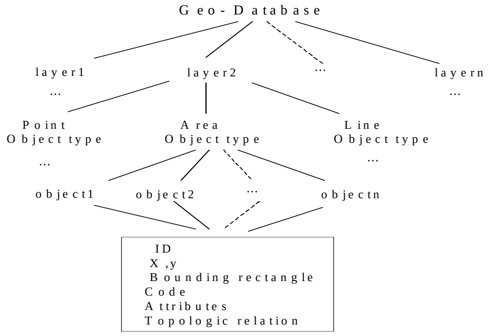


Figure 2. logical structure of geo-database

It has following properties:

- Its nodes of triangle must on the boundary of objects.
- The edges in a triangle of CDT are divided into two groups:
  - edges belonging to the part of the boundary of the object (named constrained edges).
  - edges not belonging to the part of the boundary of the object.
- The triangles in CDT are classified as three types:
  - Triangle having only one constrained edge in its three edges, seeing triangle b, g, i, h in Figure 3, and Tri1EdgeList denoting a set of this kind of triangles;
  - Triangle having two constrained edges in its three edges, seeing triangle a, f, k in Figure 3, and Tri2EdgeList denoting a set of this type of triangles ;
  - Triangle having no constrained edge in its three edges, seeing triangle d in Figure 3, and Tri0EdgeList denoting a set of this kind of triangles.
- The adjacent relationship can be grouped into three types in triangle net:
  - Two triangles sharing a common constrained edge, seeing triangle i and k, triangle f and d in Figure 3;
  - Two triangles sharing a common non-constrained edge, seeing triangle c and d, trianle h and k in Figure 3;
  - Two triangles sharing a common vertex; seeing triangle g and h in Fig.3.

- If two objects, o_i and o_j share a common node of triangle, they are adjacent.
- If two objects, o_i and o_j share a common constrained edge of triangle, they are adjacent.
- If two objects, o_i and o_j share a common triangle between them, they are adjacent, seeing object o3 and 01 in figure 3.

These good properties of the constrained Delaunay triangulation will be used for spatial analysis in aggregation process, such as rendering measurement of distance and direction among objects, maintaining topological links between points, lines and polygons, facilitating easier local proximity relations between the objects. The triangle is used as a basic unit to analyze geometric characteristics of object and the proximal relation among the objects in this research.

For functional modeling, the model possesses two function members. Distance() is a function that return the metric distance from one object to another. Direction() returns the bearing from one object to another. As previously mentioned, distance and direction are two important parameters on geometric aspects in aggregation process. Apart from Distance () and Direction (), the data model combines aspects of objects-oriented and topologic data model. Point, line, area objects are represented with their geometric and thematic aspects. Their geometric representation contains information about topologic object relationships, whereas their thematic description is structured in object classes that may form generalization hierarchies. Such class hierarchies in combination with the topological object relationships of the data model supports the definition of aggregation hierarchies of objects. These classification and aggregation hierarchies play an important role in linking the definition of spatial objects at several scale levels. They govern and guide the aggregation process.

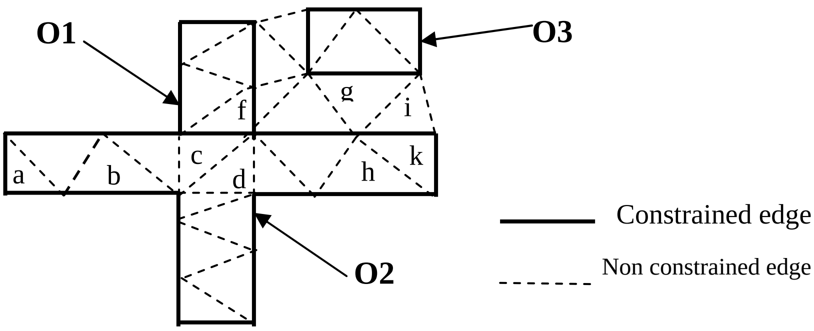


Figure 3 Structure of Constrained Delaunay Triangulation structure

4. Semantic analysis of aggregation

Whether the two adjacent objects or an adjacent group of objects can be merged or aggregated depends on if the attributes of the two objects are same or similar. If the attributes of them are same or similar, they can be merged or aggregated. Otherwise not.

The closeness or similar among objects and object types can be described by the similarity. The similarity is application-dependent. The larger the value of similarity between objects is, the closer or more similar they are. The similarity among elementary objects, elementary object types and composite object types etc can be established by expert knowledge based on the aggregation hierarchy and classification hierarchy and requirements and purposes of database generalization. A matrix can be used to represent similarities among objects and object types as shown in Figure 4 . The matrix is symmetric and reflexive one, and has the property which s_{ij} is equal to s_{ji} ($s_{ij} = s_{ji}$) and s_{ii} is equal to s_{jj} ($s_{ii} = s_{jj} = 1$) in the matrix. s_{ji} is a value between o and 1. This matrix has the characters of hierarchy and shows the similarity between same level of objects and between different levels of objects. This will provide potential possibility for between same level of objects and between different levels of objects to be merged or aggregated. This also means that different levels of objects can be kept in a spatial database. The similarity matrix will be used as a look-up table for guiding or governing the aggregation process of spatial objects in semantic to a certain application. An example for computing similarity matrix from classification hierarchy is as following:

$$s_{ij} = \begin{cases} 1 & \text{(same object or same object type)} \\ 1 - e_i / e_r & \text{(different object or different object type)} \end{cases}$$

Where: e_{max} denotes thec maximum link number of edge between different nodes in classification hierarchy. e_i denotes the link number of edge between different nodes in classification.

SIMILARITY	**Obj1**	**Obj2**	**…**	**Sub-type 1**	**Sub-type 2**	**…**	**Sup-type 1**	**Sup-type 2**	**…**
Obj1	S_{11}	s_{12}	…	S_{14}	S_{15}	…	s_{17}	s_{18}	…
Obj2		s_{22}	…	S_{24}	S_{25}	…	s_{27}	s_{28}	…
…			…	…	…	…	…	…	…
Sub-type1				S_{44}	S_{45}	…	s_{47}	s_{48}	…
Sub-type2					S_{55}	…	s_{57}	s_{58}	…
…						…	…	…	…
Sup-type1							s_{77}	s_{78}	…
Sup-type2								s_{88}	…
…									…

Figure 4 Matrix of similarity

Where: obj1, obj2 etc denote different elementary objects; Sub-type1, sub-type2 etc denote different elementary object types; sup-type1, sup-type etc denote composite object type. s_{ij} denotes similarity value between matrix elements.

5. Geometric Analysis of Aggregation

That the two objects or a group of objects having same thematic attributes can or can not be merged or aggregated depends on if they have adjacent (connectivity or unconnected) relationship between them or the space between them is small than the threshold. This means that the adjacent objects (or neighbors) must be identified and the space between them must be computed before aggregation operation. Those tasks can be conducted by analyzing the spatial relationship among the geometric primitives using CDT.

5.1. Detection of Neighboring Conflict Objects

For effective aggregation process, the number and location of objects that violate the constraints need to be identified rapidly. The neighborhood of the violated objects must be identified in order to aggregate them. We now describe a method for identifying the neighbor set of a particular object based on the data model which we developed. The procedure is primarily concerned with finding triangles which are constrained edge connect with o_i (i.e. triangles that share a constrained edge with o_i) through triangles t which belong to o_i . Any such triangle which does not belong to o_i is added to List, provided it has not already been added. Given an object o_i , returns a list of the identifiers associated with all triangles connected with o_i (not belonging to o_i) through the adjacent relations with triangles which belong to (within o_i). The process of finding adjacent triangles of o_i continues until the all triangles t within o_i

have been processed. The adjacent triangles of o_i are of interest in that they provide a convenient way of identifying its connectivity and proximal neighboring objects. These neighbors are found by examining each of o_i's adjacent triangles in turn. If a particular adjacent triangle t_p belongs to an object o_j then o_i and o_j are connectivity. If, however, t_p belongs to F, then each of the objects contiguous to its vertices needs to be found. It follows that each of these objects (with the exception of o_i itself) is proximal to o_i.

The process is described as follows, using Figure4 as an example for illustration:

- Let o_a be in Figure 4 is the object in consideration.
- For o_a, store t_i ($t_i \in o_a$) to TriObjectList.
- For each t_i ($t_i \in$ TriObjectList $\wedge$ ($t_i \in$ Tri1EdgeList $\vee$ $t_i \in$ Tri2EdgeList)), do the following:

{

- Find out neighbor triangle t_j which is connected with t_i through the constrained edge (the boundary of the object o_a, t_j not belonging to o_a).
- Store t_j to the TriNeighborList.

}

- For $t_j \in$ TriNeighborList

{

- Get area neighbor object of o_a by t_j with the conception of o_a.
- Store it in ObjectNeighborList if it does not exist in the ObjectNeighborList.

}

- For o_a and $o_i \in$ ObjectNeighborList, do the following:

{

- Get a set of triangles between o_a and o_i from TriNeighborList, and store them to TempList.
- For each triangle $t_k \in$ TempList, do the following:

  {

  - Compare l_t (the average edge length of t_k) with the threshold $d_{threshold}$.
  - If $l_t > d_{threshold}$, do nothing, else store o_i in ObjectConflictList preparing for aggregation.

  }

}

Here, note that the average edge length of a triangle represents approximately the distance between two objects.

5.2. Detection of Conflict Set of Objects with regular Distribution Pattern

How to identify a set of objects with regular pattern in spatial and semantic is very important in aggregation process Although there are many possible patterns that are of interest to aggregation, the following discussion concentrates on the detection of groups of objects which have regular pattern in direction (see below for the definition) within a larger group. The particular example to be consider concerns the detection of (linear) groups of landuse elementary objects (Figure 8). The human eye will detect such a group when:

- The distance between o_i and neighboring elementary objects in the group are similar and are normally less than the distance to the nearest objects outside the group, from any member of the group.
- The objects in the group have normally the same attributes.
- $o_i \in$ ObjectConflictList as seed to repeat the procedure of above.

The procedure starts with the object in ObjectConflictList which has been gotten by DetectNeighborObject. The object in ObjectConflictList will be as seed to look for new conflict objects or its neighboring objects within $d_{threshold}$ to it. The each object in ObjectConflictList will be processed using DetectNeighborObject method. The new conflict objects will been added in ObjectConflictList if they are not added before. If the object in ObjectConflictList has been processed, the sign is assigned for it. This means that the object in ObjectConflictList with sign does not need to be processed any more. The objects with sign in ObjectConflictList will be a pattern group of objects if they have the pattern of spatial distribution.

The procedure of detecting a regular group of objects within a larger group is formulated as follows:

- For $o_i \in$ ObjectConflictList, do the following:

{

1.Get all area neighboring conflict objects of o_i using above processing method if o_i has not been processed, store them in ObjectConflictList if they are not in the ObjectConflictList, and store o_i in ObjectGroupList .

2.For $o_j \in$ ObjectConflictList and o_j not being processed and $o_j \neq o_i$, do the following:

{

- $o_j \Rightarrow o_i$ and repeat above step 1.
- Assign a sign the object o_j that has been processed.
- Until all objects have been processed in ObjectConflictList.

}

}

- Get a regular group of objects
- Empty ObjectConflictList

5.3. Extraction of Skeleton Line

Supposed that an area object violating the geometric constraints have similar value in similarity matrix with its neighboring objects, in order to the area balance of each class after aggregation, the area object must be skeletonized. Each part of area object after skeletonising the area object is assigned to neighboring objects (seeing figure 6). Extraction of skeleton line of double line river is also very useful in river generalization. Here we concentrate on the extraction of the skeleton line of polygonal objects. The extraction of the skeleton line of polygonal objects starts with triangle t having two constrained edges (seeing triangle a as shown in figure 3) or triangle tt having no constrained edge (seeing triangle d as shown in figure 3), and find out all adjacent triangles ttt having only one constrained edge through sharing non-constrained edge between two triangles and end with t or tt, and computes the mid-point of non-constrained edge of ttt , and computes the mid-point of non-constrained edge and center point of triangles tt. The skeleton line will detected by connecting all special points. The procedure is as following:

- For t_{2i} ($t_{2i} \in$ Tri2EdgeList) or t_{0i} ($t_{0i} \in$Tri0EdgeList), do the following:

{

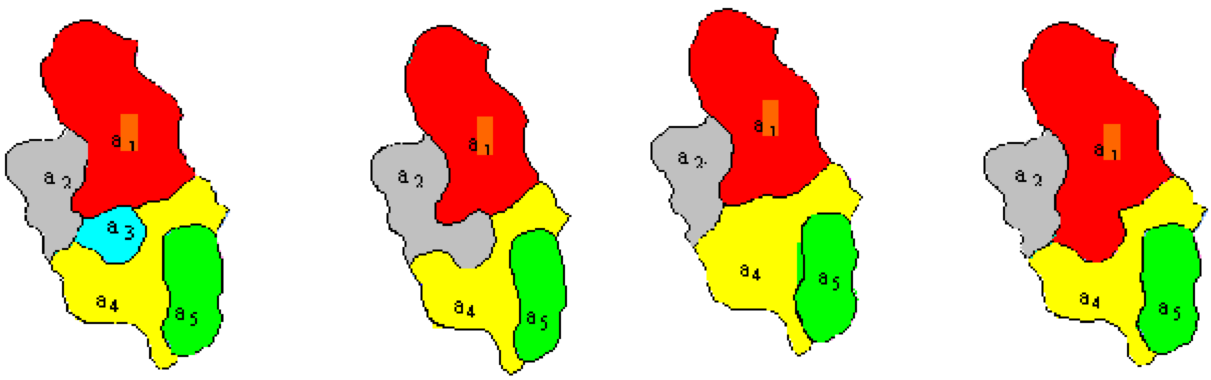
- Start with t_{2i} or t_{0i}, find out all adjacent triangles through sharing non-constrained edge between two triangles and end with t_{2i} or t_{0i}.
- Compute the mid-point of non-constrained edge of t_{2i} and t_{1i} ($t_{1i} \in$ Tri1EdgeList).
- Compute the center point and mid-point of non-constrained edge of t_{0i}.
- Get a sequence of points which describe the skeleton line of the object.
- Connect the opposite point of the non-constrained edge of t_{2i} with the mid-point of the constrained edge of t_{2i}, and connect the two mid-point of two non-constrained edge of t_{1i}, and connect the mid-point of three mid-point of three non-constrained edges with center point of t_{0i} respectively.

}

6. Examples of Aggregation Operation

The following part gives fiv different cases in aggregation based on the methods above.

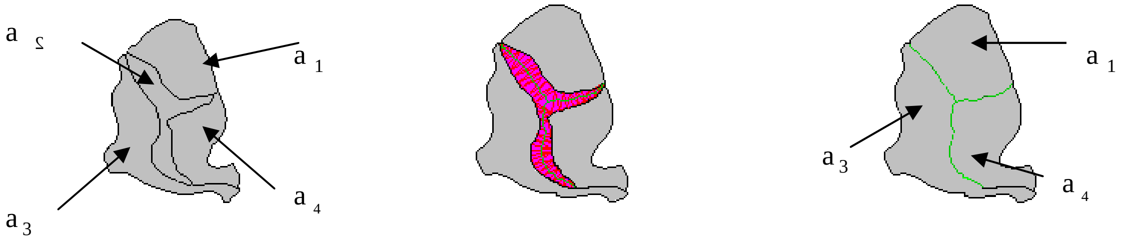
Case1: we assume there are five area objects a_1, a_2, a_3, a_4 and a_5 as shown in figure 5 (a) which belong to different object types as shown in Figure. The area of object a_3 is less than area threshold which each object must have in database. So object a_3 must be aggregated with one of a_1, a_2, a_4 and a_5. which one should be selected in a_1, a_2, a_4 and a_5 to merge with a_3 depends on the similarity and relations among a_3, a_1, a_2, a_4 and a_5. a_3 has connectivity relation with a_2, a_4 and a_5. a_3 has proximal relation with a_5. We take similar value between objects from similarity matrix, compare their value size and select largest one for a group of objects having connectivity relation such as a_1, a_2, a_3, a_4. The two objects having largest value will be aggregated as shown in Figure 5 (b), (c) and (d). For a_3 and a_5, the distance between them must be measured first. If the distance is less than distance threshold which database specifies, the similarity value can be taken from similarity matrix. If the distance is greater than the distance threshold, there is no need to process further for a_3 and a_5.



(a) (b) (c) (d)

Figure 5 Case 1 of aggregation operation

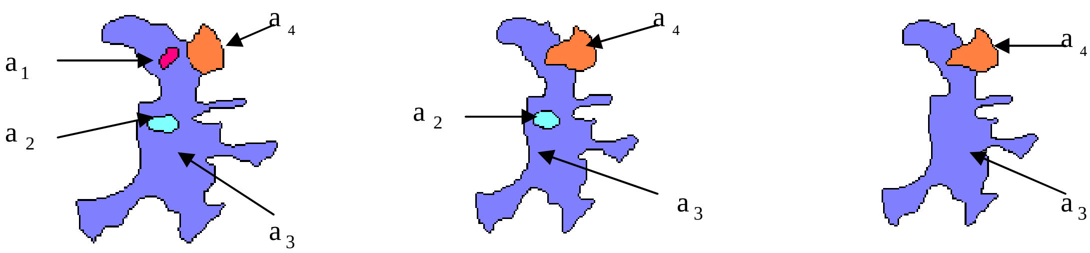
Case2: Suppose that there are four objects a_1, a_2 , a_3 and a_4 which belong to different object types as shown in Figure 6 (a). These objects have connectivity relation among them. a_5 is too narrow and should be aggregated with its neighboring objects. a_5 has the same similarity value from similarity matrix with other three objects. So a_5 should merge partly with other three objects. First the skeleton line of a_5 is created (c.f. Figure 6 (b)). This line will become a part of edges of a_1, a_3 and a_4 . Then the edges of a_1, a_3 and a_4 and topological relation among them are reorganized (c.f. Figure 6 (c)).



(a) (b) (c)

Figure 6 Case 2 of Aggregation Operation

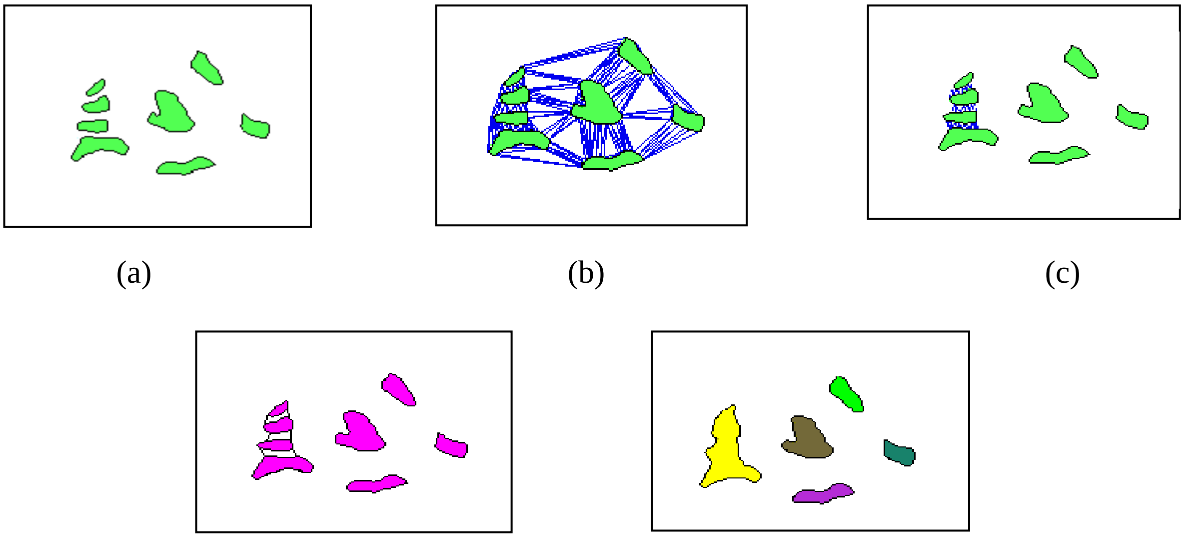
Case3: suppose that there are three objects a_1, a_2 , a_3 and a_4 which belong to different object types as shown in Figure 7 (a) . a_3 contains a_1 and a_2 , and they have inclusion relation. That a_1 collapses into a_3 or merges with a_4 depends on the similarity and distance between a_1 and a_4 . If the distance between the two objects is less than the distance threshold and the similarity value between a_1 and a_4 is greater than one between a_1 and a_3, a_1 will be merged with a_4 (c.f. Figure 7 (b)). Otherwise a_1 will collapse into a_3.



(a) (b) (c)

Figure 7 Case 3 of Aggregation operation

Case 4: Suppose that there are a group of objects which belong to the same object type as shown in Figure 8 a. These objects have proximal relation among them (c.f. figure 8 b). First a subgroup of objects which violate the distance threshold must be identified using above method (c.f. figure 8 c). Then the conflict objects will be merged (c.f. figure 8 d, e) and at same time the relationship among these objects will be changed.



(a) (b) (c)

(d) (e)

Figure 8 Case 4 of Aggregation Operation

Case 5: Suppose that there are a group of objects with proximal relation among them which belong to different object types First, a subgroup of objects which violate the distance threshold must be identified using above method. Second the similarity among these conflict objects in geometric need to be analyzed. Third, these conflict objects will be divided into several groups based on the similarity among these objects. Finally, each group of objects will form a new object and the spatial relations among them will also be changed.

7. Conclusion

The data model which is presented in this paper integrates the advantages of Formal Data Structure and Local Constraint Delaunay Triangulation. This data model is used in aggregation process avoids involving a probably heavy computation and complicated algorithm in aggregation operation.

The data model for aggregation has a lot of good properties, which allow to measure spatial relations such as disjoint relation, distance and direction relation among objects.

The similarity matrix with hierarchical character reflects the semantic relation among spatial object and object type and will provide potential possibility for between same level of objects and between different levels of objects to be merged or aggregated. The similarity matrix can be used as a look-up table for guiding or governing the aggregation process of spatial objects in semantic to a certain application.

Acknowledge

Research work was supported by International Institute for Aerospace Survey and Earth Science, Ministry of Education PR.China and National Bureau of Surveying and Mapping.

Reference

Bundy,G.LI., Jones, Chris B. and Furse E. 1995, Holistic Generalization of large-scale cartography data, in GIS and Generalization (eds Muller J.C., Lagrange, J.P. and Weibel, R.), pp. 106-116.

Jones,Chris B. and Ware, M.J., 1998, Nearest neighbor search for linear and polygonal objects with constrained trianglations, 8 *th* International Symposium on Spatial Data Handling, Delft, The Netherlands, pp. 13-19.

McMaster, R. B. 1991 Conceptual frameworks for geographical knowledge, in Map Generalization: Making Rules forKnowledge Representation (eds Buttenfield, B. P. and McMaster, R. B.), Longman House Essex, United Kingdom, pp. 21-39.

Molenaar, M., 1989, Single valued vector maps- a concept in GIS. GeoInformationssysteme, 2, 18-26.

Molenaar, M. 1991, Terrain objects, data structures and query spaces, in Geo-Informatik, (ed Schilcher, M.), Siemens-Nixdorf Informationssysteme A.G., Munchen, 1991, pp. 53-70.

Molenaar, M. 1993, Object hierarchies and uncertainty in GIS or why is standardisation so difficult, GeoInformations-Systeme, Vol. 6, No. 3, pp. 22-28.

Molenaar, M. 1996, Multi-scale approaches for geo-data, International Archives of Photogrammetry and Remote Sensing, Vol. XXXI, Part B3, Vienna, Austria, pp.

Muller, J. C. 1991, Generalization of spatial data bases, in Geographic Information Systems (eds Maguire, D. J., Goodchild, M. F. and Rhind, D. W.), Vol. 1, Longman Scientific & Technical Ltd, New York, pp. 457-475.

Muller, J.C., Weibel, R., Lagrange, J.P. and Salge, F. 1995, Generalization: state of the art and issues, in GIS and Generalization (eds Muller, J.C., Lagrange, J.P. and Weibel, R.), Taylor & Francis, pp. 3-17.

Peng, W. 1997, Automated Generalization in GIS, PhD Thesis, Wageningen Agricultural University (and ITC), The Netherlands, 188p.

Peuquet, D. 1984, A conceptual framework and comparison of spatial data models, Cartographica, Vol. 21, No. 4, pp.66-113.

Pilouk, M. 1996, Integrated modelling for 3D GIS, PhD Thesis, Wageningen Agricultural University, The Netherlands, 200p.

Richardson, D. E. 1993, Automated spatial and thematic generalization using a context transformation model, PhD. Thesis, R&B Publications, Canada, 149p.

Smith, J.M. and Smith, D.C.P, 1977, Database abstractions: aggregation and generalization, ACM Transactions on Database System, 2, pp.105-133

Thompson, P.J. 1989, Data with semantics: Data models and data management, Van Nostrand Reinhold, New York.

van Smaalen, J. W. N. 1996, A hierarchic rule model for geographic information abstraction, SDH 96, pp. 4B.31-4B.41.

Ware, M.J., and Jones, Chris.B., 1996, A Spatial Model For Detecting (and Resolving Conflict Caused By Scale Reduction. 7 *th* International Symposium on Spatial Data Handling, Delft, The Netherlands, pp. 9A15-23.

Weibel, R. 1995, Three essential building blocks for automated generalization, in GIS and Generalization (eds Muller J.C., Lagrange, J.P. and Weibel, R.), pp. 56-69.