A Multi-Granularity Approach for Adaptive Mass Vector Dataset Access and Transmission

Changxiu Cheng[1], Feng Lu[1] and Mingbo Zhang[1,2]

[1]State Key Laboratory of Resources and Environmental Information System
Institute of Geographical Sciences and Natural Resources Research
Chinese Academy of Sciences, Beijing, 100101, P. R. China
[2]Shandong University of Technology, Shandong, 255049, P. R. China

Abstract

Online large volume vector data access and transmission is a crux for many GIS related applications. In this paper, a concept of granularity is introduced and a multi-granularity based pyramid model for large volume vector data generalization is set forward. A virtual vector-raster-vector conversion process is utilized to generate pyramid vector snapshots to avoid overlaps and gaps which may emerge with geometric generalization under current object-relational representation frameworks for vector data. Granularity value is associated with display resolution and different representation layers are selected online according to the display scale. It is argued the presented approach can rationally reduce the data volume transmitted online and keep the visualization and querying effects.

I. INTRODUCTION

An important requirement for GIS applications is the ability to transmit and display numerous geometric objects swiftly (Chan and Chow, 2002). Under most circumstances, complete fetching and transmitting all of the geometric objects located in the display window is time-consuming and unnecessary, especially for online GIS applications concerning large volume spatial data, where some retrieved geometric objects may be too small in viewing zone to be displayed meaningfully on the screen. For example, a polygon with hundreds of segments may occupy only a pixel at a small display scale, but the system will retrieve all these segments from the dataset and then repeatedly draw them at the pixel location. It not only wastes the CPU time, but also increases the memory loading. This problem is more serious for network related GIS applications where the network loading is a crux and resource wasting is intolerant.

One possible solution is to store the same map in different scale. Thus a geometric object will be stored as multiple copies in a database, each with a different level of detail. Some example can be found in Timpf and Frank (1995), Jones and Kidner (1996), Bertolotto & Egenhofer (2001), Tu *et al.,* (2001) and Cecconi (2003). Another alternative is to store geometric objects in their entirety as usual, and apply a line simplification algorithm to obtain the desired scale. See for instance in Oosterom and Schenkelaars (1995), Timpf (1998), Buttenfield (1999), Wei *et al.* (2000), Zhou and Jones (2001). The former approach is easy to get implemented, but introduces a high degree of data redundancy, and has the potential problem of update anomaly (Chan and Chow, 2002). The latter one is an ideal approach for progressive transmission (Buttenfield, 1999), but is difficult for implementation, and can't be seamlessly integrated with the commercial geographic database systems (Spaccapietra *et al.,*2000). Moreover, the approach may still retrieve more data than necessary from database systems (Chan and Chow, 2002).

Cartographic generalization is a core technology for building multi-scale databases. It can reduce the account of data and adjust the information according to the given scale and theme (Robert & Geoffrey, 1999; Wei *et al.,* 2000; Bertolotto and Egenhofer, 2001; Cecconi, 2003). There are two fundamental types of generalization: (i) quantitative generalization, which means a gradual increasing/decreasing of map content fetched with scale changing; and (ii) qualitative generalization, which means transformation of elementary spatial information to more abstract forms. Such a procedure is obviously complex and thus time-consuming (Cecconi, 2003). Although automatic cartographic generalization has been on the research agenda for three decades, it is not fully automatic, and without widely used up to now (Harrie and Sarjakoski, 2002).

In this paper, we introduce the concept of granularity and present an efficient and fully automatic generalization algorithm based on granularity to derive multiple representations. These representations can be organized to build a multi-granularity database, which highly reduces the data volume transmitted, and simultaneously keeps acceptable visualization and querying effects.

The remainder of this paper is organized as follows. Section 2 introduces the concept of granularity and presents a method for building multi-granularity geographic databases. Section 3 evaluates the performance of multi-granularity databases

1082-4006/03/0901~2-29 $5.00

with a case study. Section 4 draws a conclusion.

II. MULTI-GRANULARITY GEOGRAPHIC DATABASE

Granularity and multi-representation for spatial objects

Granularity is a concept used to measure the degree of detail for describing some objects and phenomena. In a vector based spatial data representation, it is defined as the minimum distance unit between two coordinates, as the concept of resolution in a raster based data representation. Any two points in a vector based dataset must obey: *(formula 1)*

$$Abs(Point1.x-Point2.x)=k*\text{the value of granularity and}$$
$$Abs(Poin1.y-Poinx2.y)=k*\text{ the value of granularity}$$
$$\text{where } k=0,\ 1,\ 2,\ 3,\ldots\ldots,n. \qquad (1)$$

Given a map, the value of granularity is inversely proportional to the display scale. A pyramid or multi-scale geographic database can be derived through changing the granularity values (ref. Figure 1(a)) (Neild, 2003).

In the multi-granularity databases, every representation is derived from the original data according to a given granularity value. These representations with bigger granularity value have fewer points, as shown in Figure 1(b). It's similar to the pyramid models for raster data. The original layer is on the bottom of the pyramid, and the more abstract representations are located on higher layers of the pyramid. The pyramid simulates the procedure of accessing geographical information, i.e. information get exposed gradually for the area of interests.

Method of object generalization

Classical geometrical generalization technologies apply to topology models where common features are only stored once in a dataset, which means operations on a common arc will take same effects for the polygon objects containing the arc. Currently mass vector data are commonly stored in RDBMSs with object-relational models where every feature is independent on other features and has its own geometric coordinates, which means many copies for a common arc will be stored in a dataset to form complete polygon objects respectively. It makes no difference between two models for line feature generalization. But polygon generalization within the object-relational model will bring out some overlaps and/or gaps (Oosterom, 1993), as shown in Figure 2.

We use a virtual vector-raster-vector conversion procedure to solve this problem. A vector dataset is firstly converted to a raster dataset using an appointed granularity as the cell size. If the granularity is big enough, a vector dataset derived from the raster dataset will have fewer vertices than the original vector dataset. Different granularity values will generate different representations with different detail level for spatial data.

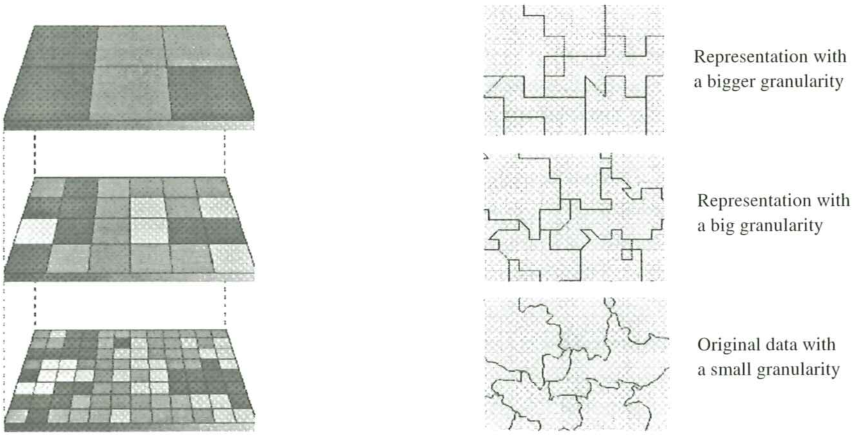
The algorithm of generating the multi-representations is described as follows.

1). Transform original coordinates to a new reference system with its granularity value as the minimum graduation. Then any points within a same cell will be shifted to the center of the cell.

$$X=int((X-MapExtent.XMin)/\ \text{granularity value})*\ \text{granularity value}+\text{granularity value}/2$$
$$Y=int((Y-MapExtent.YMin)/\ \text{granularity value})*\ \text{granularity value}+\text{granularity value}/2 \qquad (2)$$

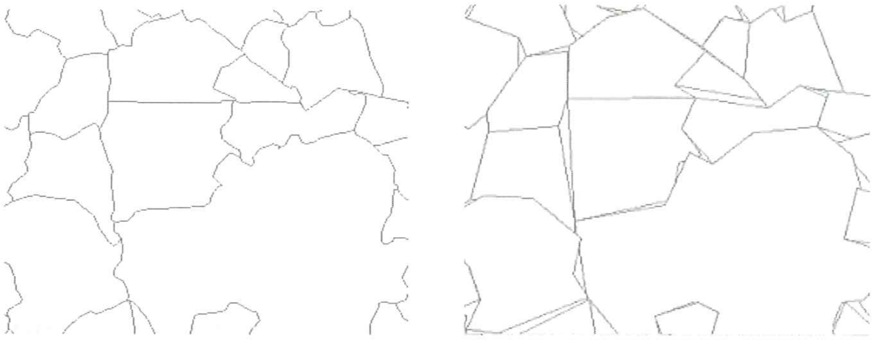
2). Remove redundant points. There are two kinds of redundant point existing in the generated vector dataset, i.e., points in same location and points in same line. After removing these points, the generated dataset will have fewer vertices.

The excessive or unrealistic (self-intersecting) simplification (Prasher and Zhou, 2003) won't emerge in this method. And yet the topological constraints (Egenhofer *et a*.. 1994; Zhou *et*



(a) Pyramid of raster data (b) Multi-granularity representation

Figure 1. Pyramid model and multi-granularity representation



(a) Before generalization (b) After generalization

Figure 2. Overlaps and gaps brought by polygon generalization on object-relational models

al., 2002) and semantic constraints (Zhou *et al.,* 2002) won't be violated, for there are only zoom transformation between these representations and the original data, and no any semantic information is omitted or changed. For example, neighboring spatial objects will also be adjacent, and a river flowing along a road will still be itself after generalization.

Display associating with granularity

One of the most important capabilities for the multi-granularity databases is providing tailored information to users. Current display granularity is defined as true length on the earth's surface that 1 pixel corresponds to, and it will change as users zoom the maps. For example, current display granularity will decrease during the process of zooming in, once it falls below the granularity of next representation layer which expresses more details, the layer will be fetched from the datasets to substitute current layer displayed. The granularity value corresponding to 1 pixel is calculated with formula 3.

Granularity value = Max (ViewHeightM/
ViewHeightPix,ViewWidethM/ViewWidethPix) (3)

where,

ViewHeightM is the height of a map view using meter as the unit

ViewHeightPix is the height of a map view using pixel as the unit

ViewWidthM is the width of a map view using meter as the unit

ViewWidthPix is the width of a map view using pixel as the unit

III. PERFORMANCE ANALYSIS

An experience is carried out with a dataset containing 873,097 polygons which are composed by 14,242,044 vertices. The extent of the dataset is about 5000*4000 km. The display window is 600 *400 pixels.

Step 1. Calculate the granularity value when the dataset is fully viewed with formula 3 and get 10,115m.

Step 2. Generate the representation layer according to the above granularity value.

Step 3. Interpolate four granularity values between 1 and 10,115 by arithmetical progression. The serials of granularity values is 1, 3035, 5564, 7587, 9104, and 10115.

Step 4. Derive representation layers according to above granularity values with the algorithm described in section 2.2. These layers form a multi-granularity database.

Performance evaluation

Compression ratio

Table 1 shows the vertex numbers of the six representation layers and corresponding compression ratio. It reflects that the compression ratio increases rapidly when the granularity arguments. The number of vertices of the overview representation layer is only 8.7% (1-91.3%) percent of original data, with an imperceptible displaying distortion.

ADVN calculation

Although the vertex numbers of displayed layers increase rapidly along with map zooming in, the spatial extent will simultaneously dwindle rapidly with enlarged display scale. It makes fewer vertices be transmitted and displayed. It is interesting to evaluate the relationship between the number of displayed vertices and map zooming.

Firstly, we interpolate some values between any two discrete granularities as shown in Table 2. Then the average displayed vertex number (ADVN) of a representation layer can be easily calculated with formula 4. For example, the ADVN without compression is 14242044*(1/10115)*(1/10115)˜0.14, and the ADVN of overview layer is 3331245*(3035/10115)*(3035/10115)˜299910.86. The ADVN can also be interpolated for different granularities. When current granularity value is above 10115, the ADVN will be the same to the ADVN on overview. Table 2 shows the results.

ADVN = Number of vertices in map *
(Current view area/Overview area) (4)

Tabale 1. Compression result of multi-granularity database

Granularity	Number of vertices	Compression ratio
1	14242044	0%
3035	3331245	76.61%
5564	1867289	86.89%
7587	1469370	89.68%
9104	1310899	90.80%
10115	1239250	91.30%

where

Current view area/Overview area =

(The current granularity value /
The granularity value on overview)^2

Performance evaluation

The comparison of ADVN with/without compression is shown in Figure 3. It indicates that the maximum ADVN with compression is only 8.997% (1281361.59/14242044) of the total vertex number of original data, and the ratios in other displayed layers are less than it. Therefore, the data volume manipulated keeps relatively steady with the multi-granularity database at any time.

Figure 4 shows the change of ADVN with map zooming. With zooming in, more and more vertices will be fetched from original data, and the ADVN will increase according to a certain trend. Once achieving a critical value, another more abstract layer will be available and the data will be refreshed from the layer. And the ADVN falls quickly. Then the ADVN will increase again according to its trend, and quickly fall when achieving next available snapshot.

IV. CONCLUSION AND DISCUSSION

A multi-granularity adaptive geographic database is effective for online mass vector transmission and visualization. It can efficiently reduce the number of manipulated vertices and cut down the response time, simultaneously keeping acceptable visualization and querying effects. A vector-raster-vector conversion can be used to avoid the overlaps and gaps generated with current object-relational data models.

It deserves further research that how to build a multi-granularity database with right size. The redundant data and update anomaly is inevitable in multiple-layer databases. So the number of representations should be as few as possible. Then the

Table 2. ADVN with/without compression in the multi-granularity database

Granularity	Number of vertices	ADVN without compression	ADVN with compression
1	14242044	0.14	0.14
506		35640.32	35640.32
1012		142561.28	142561.28
1517		320340.4	320340.4
2023		569681.76	569681.76
2529		890303.76	890303.76
3034		1281361.59	1281361.59
3035	3331245	1282206.4	299910.86
3540		1744404	408019.88
4046		2278727.04	532999.2
4552		2884330.72	674651.22
5058		3561215.04	832975.93
5563		4307831.12	1007611.05
5564	1867289	4309380	565007.23
6069		5127135.84	672224.04
6575		6017721.2	788989.6
7081		6979587.2	915100.84
7586		8010621.75	1050280.84
7587	1469370	8012733.84	826684.06
8092		9114908.16	940396.8
8598		10290475.2	1061681.56
9103		11534788.46	1190058.96
9104	1310899	11537322.89	1061944.83
9610		12855451.21	1183271.03
10114		14239228.11	1310639.81
10115	1239250	14242044	1239250
11000		14242044	1239250
11200		14242044	1239250

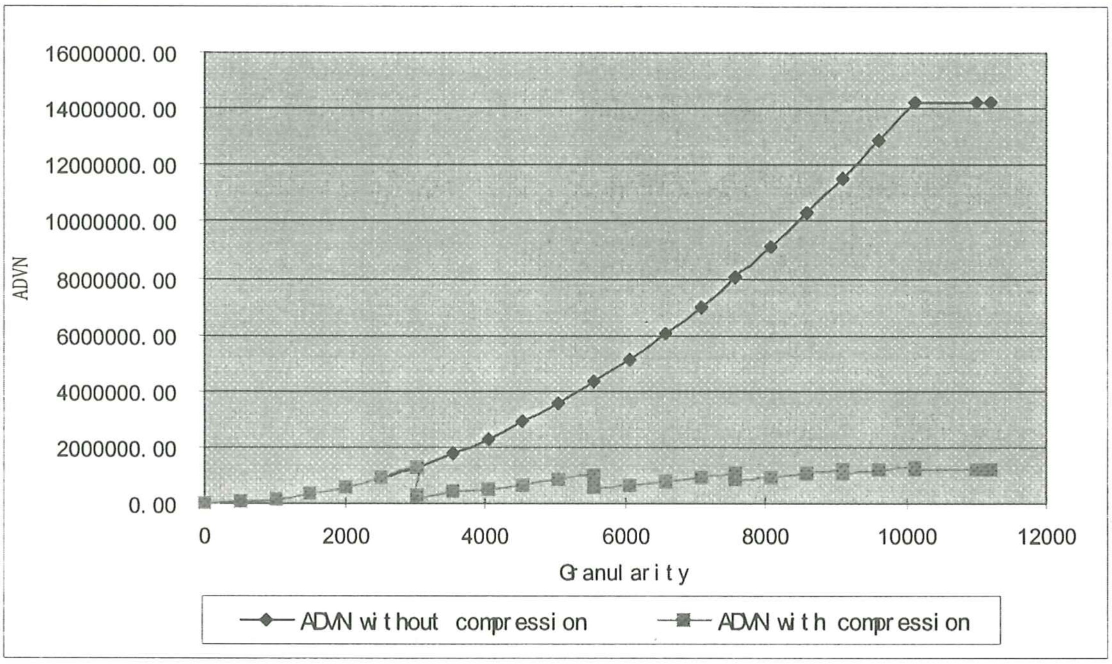


Figure.3. Comparison of ADVN with/without compression

trade-off between storage and performance needs more investigation. Moreover, how to integrate the generalization with other technologies, such as data cache and pre-fetching to fasten massive vector data transmission also deserves detailed study.

ACKNOWLEDGMENTS

This research was supported by the High Technology Research and Development Plan of China under the grant No. 2003AA135070 and the National Natural Fundation of China under the grant No. 40201043. The authors would also like to thank the anonymous reviewers for their constructive comments that have substantially improved this paper.

REFERENCE

[1] Bertolotto, Michela, Max J. Egenhofer, 2001, Progressive transmission of vector map data over the world wide web. *GeoInformatica 2001*, 5(4):345-373.

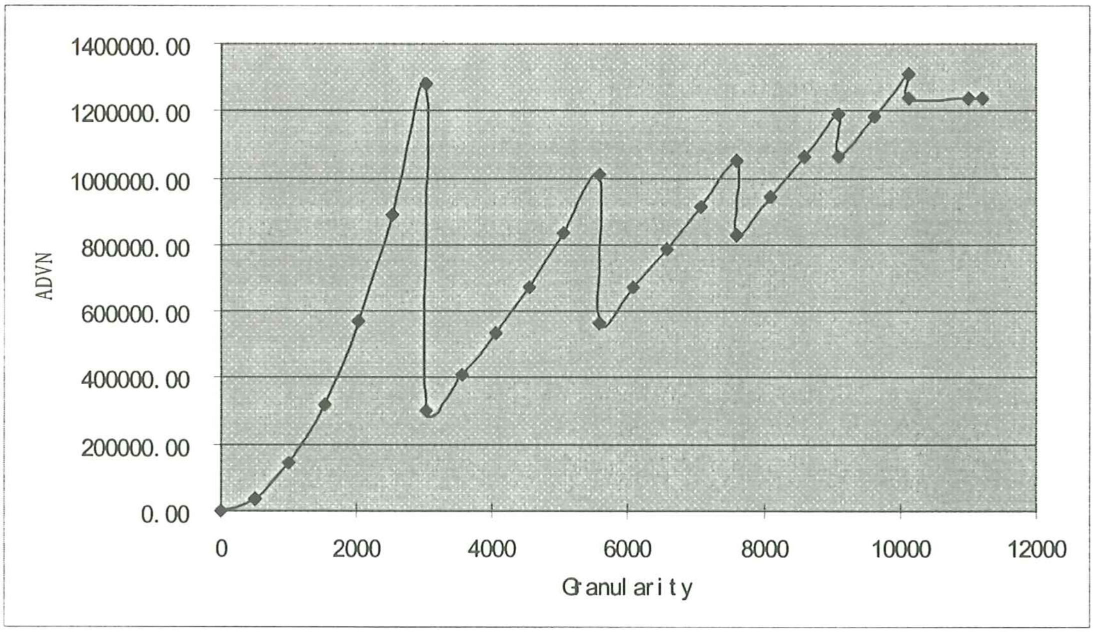


Figure 4. ADVN with compression

[2] Buttenfield, Barbara P., 1999, Progressive Transmission of Vector Data on the Internet: A Cartographic Solution, In *Proceedings, 18th International Cartographic Association Conference*, 1999.

[3] Cecconi, Alessandro, 2003, *Integration of Cartographic Generalization and Multi-scale Database for Enhanced Web Map*. Ph.D. dissertation. http://www.geo.unizh.ch/publications/acecconi/phd/pdf/phd.pdf. 2003.

[4] Edward, P. F., k. Chan, K.W. Chow, 2002, On Multi-scale Display of Geometric Objects. *Data & Knowledge Engineering* 40: 91-119.

[5] Egenhofer, M. J., E. Clementini, P. Di Felice, 1994, Evaluating inconsistencies among multiple representations. In *Proceedings of the Sixth International Symposium on Spatial Data Handling, SDH'94*, p. 901-920, Edinburgh, Scotland, 1994.

[6] Harrie, Lars, Tapani Sarjakoski, 2002, Simultaneous Graphic Generalization of Vector Data Sets. *GeoInformatica 2002*, 6(3):233-261.

[7] Jones, C.B., Kidner, D. B., Luo, L., Bundy, G. L. and Ware, J. M, 1996, Database design for a multiscale spatial information system", *The International Journal of Geographical Information Systems*, 10(8): 901-920.

[8] Neild, James, 2003, Raster data management with ArcSDE. Http://www.esri.com/news/arcuser/0702/ over_arcsde1of2.html.

[9] Prasher, Sham, Xiaofang Zhou, 2003, Efficient Update and Retrieval of Objects in a Multiresolution geogeographic database. In *Proceedings of the 15th International Conference on Scientific and Statistical Database Management* .

[10] Robert, WeiBel and Dutton Geoffrey, 1999, Generalizing spatial data and dealing with multiple representations. *Geographical information systems: Principles and Technical Issue*, 1:125-155.

[11] Spaccapietra, Stefano, Christine Parent, Christelle Vangenot, 2000, *GIS Databases: From Multiscale to MultiRepresentation. Lecture Notes in Computer Science*, 2000 (1864) 57-70

[12] Timpf, Sabine, 1998, Map Cube Model – a Model for Multi-scale Data. In *8th International Symposium on Spatial Data Handling*, 190~201.

[13] Timpf, Sabine, Andrew U. Frank, 1995, A Multi-scale Data Structure for Cartographic Objects. In *Proceedings of ICC'95, (International Symposium on Computer- Assisted Cartography.)*, in Barcelona, Vol. 1, pp: 1389-1396

[14] Tu, Shengru, Xiangfeng He, Xuefeng Li, Jay J. Ratcliff, 2001, A systematic approach to reduction of user-perceived response time for GIS web services. Http://portal.acm.org/citation.cfm?id=512172 &dl=ACM&coll=portal&CFID=12045330&CFTOKEN=78417218.

[15] van Oosterom, Peter, 1993, The Gap-Tree, and approach to "on-the-flay" Map Generalization of an Area Partitioning. *GISDATA Specialist Meeting on Generalization.*

[16] van Oosterom, Peter, Vincent Schenkelaars, 1995, The Development of an Interactive Multi-Scale GIS. *International Journal of Geographical Information Systems,* 9(5): 489-507.

[17] Wei, Zu-Kuan, Sook-Kyoung Cho, Jae-Hong Kim, Hae-Young Bae, 2000, A new spatial data representation in GIS. Http://www.geocomputation.org/2000/GC019/Gc019.htm, 2000.

[18] Zhou, Xiaofang, Sham Prasher, Masaru Kitsuregawa, 2002, Database Support for Spatial Generalisation for WWW and Mobile Applications. In *Proceedings of the 3rd International Conference on Web Information Systems Engineering (WISE'2002)*, pages 239-246. December 2002, Singapore.

[19] Zhou, Sheng, Christopher B. Jones, 2001, Design and Implementation of Multi-Scale Databases. In *Proceedings of the 7th International Symposium, SSTD,* 2001,