The Relationship among GIS-Oriented Spatiotemporal Databases

Lixin Li and Peter Revesz
Dept. of Computer Science, Univ. of Nebraska-Lincoln, Lincoln, NE 68588
e–mail: {lli, revesz}@cse.unl.edu

Abstract: We overview three major types of GIS-oriented spatiotemporal databases: (1) point-based, (2) region-based, and (3) constraint-based. We analyze the relationship among these spatiotemporal databases and show how they can be translated into each other. We illustrate the translations with an example from National Agricultural Statistics Service (NASS) databases. Finally, we also discuss the advantages and disadvantages of using the various types of spatiotemporal databases.

1. Introduction

GIS (Langran 1992, Longley et al. 2001, Worboys 1995) applications increasingly require the use of spatiotemporal data. For example, land-use change through time is a typical spatiotemporal data. Future GIS systems need to efficiently manage spatiotemporal databases (STDBs) containing such data. The study of the representation and the algorithmic methods to query and visualize spatiotemporal data is still a growing research area. We argue in this paper that spatiotemporal data can be categorized by whether its representation is based on points (Section 2), regions (Section 3), or constraints (Revesz 2002) (Section 4). As Figure 1 shows, these are alternative representations translatable into each other.

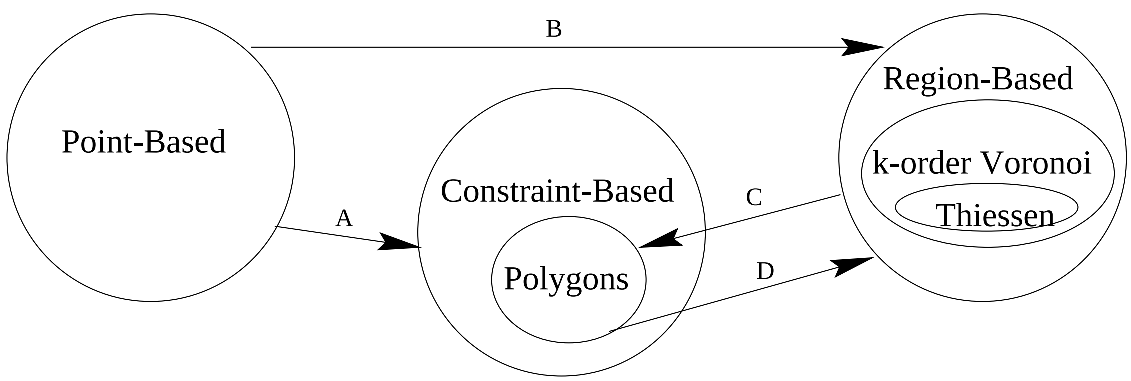

Figure 1: The relationship among the spatiotemporal databases.

(A) can be shape functions (Revesz and Li 2002b, Li and Revesz 2002, 2003, Zienkiewics and Taylor 2000), spline functions (Goodman and O'Rourke 1997), IDW (Revesz and Li 2002a, Shepard 1968), and Kriging (Deutsch and Journel 1998).

(B) can be functions that generate Voronoi regions or Worboys relations (Worboys 1995).

(C) can use linear arithmetic constraints to represent region boundaries and linear regression functions (Example 4.1) to represent the temporal part.

(D) can assign to each region a uniform value, such as the average of all the nearest neighbors.

2. Point-Based Spatiotemporal Databases

A point-based STDB consists of a set of point-based spatiotemporal relations. For 2-D space and 1-D time problems, a point-based spatiotemporal relation should have the schema of (x, y, t, w_1, w_2, ..., w_m). The attributes (x, y) specify point locations and t specifies a time instance. The last m attributes w_i $(1 \leq i \leq m)$ record the other features at location (x, y) and time t.

Since point-based STDBs are regular relational databases, we don't give an example in this paper. The disadvantage of point-based STDBs is that the measured information, such as SPI, only exists at certain sampled locations and times. Therefore, we are not able to query the values at unsampled locations and times.

3. Region-Based Spatiotemporal Databases

A region-based STDB database has both spatial and temporal parts. The spatial part has schema *(region-id, boundary)*. The *region-id* is a unique identifier of each polygonal shaped region, and the *boundary* is the sequence of its corner vertices. The spatial part can be stored in an Arc/GIS database. The temporal part has schema (*region-id*, t, w_1, w_2, $\ldots$, w_m), where t is the time attribute and each w_i represents some other characteristics of the region.

Nebraska_Corn_Space_Region

county	boundary
1	{ (-656160.3, 600676.8), (-652484.0, 643920.3), ... , (-661759.8,532153.1)
⋮	⋮

Nebraska_Corn_Time_Region

county	year	practice	acres	yield	production
1	1947	irrigated	2700	49	132300
1	1947	nonirrigated	81670	18	1470060
1	1947	total	84370	19	1602360
⋮	⋮	⋮	⋮	⋮	⋮

Table 1: A region-based spatiotemporal database with separate spatial and temporal relations.

Example 3.1 A NASS (National Agricultural Statistics Service) region-based spatiotemporal database shows the yearly corn yield and production in each county of the state of Nebraska. The spatial part of the database is shown in the upper half of Table 1 which uses the vector representation of counties in Nebraska, while the temporal part is shown in the lower half of Table 1. Note that in the relation *Nebraska_Corn_Time_Region*, {*county*, *year*, *practice*} is the primary key, because it is the minimal set of attributes that functionally determine the other three attributes. The attributes used in these two tables are the following:

- *county*, the common attribute between the spatial and temporal relations, is the Federal Information Processing Standards (FIPS) id that is unique for each county in a state.
- *boundary* is a sequence of corner vertices on the boundary of a county. For example, the county with id 1 is a polygon that is represented by its 12 corner vertices.

- *year* is the time.
- *practice* is one of the following types: irrigated, nonirrigated, and total.
- *acres* is the number of acres that are harvested with total = irrigated + nonirrigated.
- $yield = bushels/acres$.
- $production = acres \times yield = total\ bushels$ (total = irrigated + nonirrigated).

Although region-based STDBs cover the continuous area and do not have the problem of missing information as in point-based STDBs, the information accuracy in region-based STDBs is reduced by assigning one uniform value to each region.

4. Constraint-Based Spatiotemporal Databases

Constraint databases (Kanellakis et al. 1995, Revesz 2002) provide a natural representation for spatiotemporal objects when their trajectory can be described as simple mathematical functions. Constraint-based STDBs can represent both point-based and region-based spatiotemporal objects, where each attribute is associated with an attribute variable. A constraint-based relation is a finite set of constraint tuples. The value of the attributes in a relation is specified implicitly using (arithmetic) constraints such that each constraint tuple is a conjunction of constraints using the same set of attribute variables.

Nebraska_Corn_Space_Constraint

county	easting	northing	
1	x	y	$81.15x - y \leq -15827909.86$, $-0.02x - y \geq -356911.63$, $-1.11x - y \leq -108714.89$
1	x	y	$7.69x - y \geq -2116031.10$, $33.21x - y \geq -7938392.36$, $29.71x - y <= -7135823.80$
⋮	⋮	⋮	⋮

Nebraska_Corn_Time_Constraint

county	year	practice	acres	yield	production	
1	t	irrigated	a	y’	p	$1947 \leq t$, $t \leq 2000$, $a = 3453t - 6729678$, $y' = 2t - 3890$, $p = 540016t - 1054525400$
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 2: A constraint-based spatiotemporal database with separate spatial and temporal relations.

Example 4.1 The region-based NASS corn data in Example 3.1 can be represented by spatial and temporal parts in the constraint-based STDB shown in Table 2. In the spatial part of Table 2, each county polygon can be either convex or concave, we break each county polygon into a set of adjacent triangles for the convenience of implementation, where each triangle can be represented by a conjunction of three linear arithmetic constraints. For example, the county with FIPS code 1 consists of three triangles. The temporal part of Table 2 compresses the region-based

Nebraska_Corn_Time_Region relation in Example 3.1 by storing the linear regression functions of *acres*, *yield* and *production* according to *year*. The linear regression functions show the trend of these attributes during the past years. So they are useful to predict the future events. For example, what will the the yield of irrigated corn in county 1 approximately be in 2010? The entire *Nebraska_Corn_Time_Constraint* relation consists of only 279 constraint tuples, while *Nebraska_Corn_Time_Region* contains 14, 788 original tuples. Although the accuracy in this example is not very high because of the approximation by only one linear regression function, we can improve it by inserting more segments of lines. Revesz et al. (2001) discusses the detail of how to use piecewise linear functions to compress data in constraint databases. We also could use non-linear approximation functions to increase the accuracy. Note that because of the tradeoff between the accuracy and storage, the above improvements will increase the number of tuples in the temporal relation (i.e. *Nebraska_Corn_Time_Constraint*) of Table 2.

In summary, users can query the measured information at any location and time in constraint-based STDBs. This is similar to region-based STDBs: there is no missing information. Moreover, constraint-based STDBs are more sophisticated than region-based STDBs. In region-based STDBs, all the points in a region can only be given one uniform value. However, in constraint-based STDBs, the points in a region can be assigned with different values. If we could find appropriate functions (constraints) of x, y and t to interpolate the values inside each region and store them in the constraint-based STDBs, we can get a good result.

References

C. V. Deutsch and A. G. Journel. *GSLIB: Geostatistical Software Library and User's Guide*. Oxford University Press, New York, 2nd edition, 1998.

J. E. Goodman and J. O'Rourke, editors. *Handbook of Discrete and Computational Geometry*. CRC Press, Boca Raton, New York, 1997.

P. C. Kanellakis, G. M. Kuper, and P. Revesz. Constraint query languages. *Journal of Computer and System Sciences*, 51(1):26–52, 1995.

G. Langran. *Time in Geographic Information Systems*. Taylor and Francis, London, 1992.

L. Li and P. Revesz. A comparison of spatio-temporal interpolation methods. In M. Egenhofer and D. Mark, editors, *Proc. of the Second International Conference on GIScience 2002*, volume 2478 of *Lecture Notes in Computer Science*, pages 145–160. Springer, 2002.

L. Li and P. Revesz. Interpolation methods for spatio-temporal geographic data. *Computers, Environment and Urban Systems*, 2003.

P. A. Longley, M. F. Goodchild, D. J. Maguire, and D. W. Rhind. *Geographic Information Systems and Science*. John Wiley, Chichester, 2001.

P. Revesz. *Introduction to Constraint Databases*. Springer, New York, 2002.

P. Revesz, R. Chen, and M. Ouyang. Approximate query evaluation using linear constraint databases. In *Proc. Symp. on Temporal Representation and Reasoning*, pages 170–175, Cividale del Friuli, Italy, 2001.

P. Revesz and L. Li. Constraint-based visualization of spatial interpolation data. In *Proc. of the Sixth International Conference on Information Visualization*, pages 563–569, London, England, 2002a. IEEE.

P. Revesz and L. Li. Representation and querying of interpolation data in constraint databases. In *Proc. Second National Conference on Digital Government Research*, pages 225–228, Los Angeles, CA, 2002b.

D. Shepard. A two-dimensional interpolation function for irregularly spaced data. In *Proc. 23nd National Conference ACM*, pages 517–524. ACM, 1968.

M. F. Worboys. *GIS: A Computing Perspective*. Taylor & Francis, 1995.

O. C. Zienkiewics and R. L. Taylor. *Finite Element Method*. Butterworth Heinemann, London, 2000.