ISSUES IN SPATIAL DATABASES AND GEOGRAPHICAL INFORMATION SYSTEMS (GIS)

HANAN SAMET

COMPUTER SCIENCE DEPARTMENT AND
CENTER FOR AUTOMATION RESEARCH AND
INSTITUTE FOR ADVANCED COMPUTER STUDIES
UNIVERSITY OF MARYLAND

COLLEGE PARK, MARYLAND 20742-3411 USA

BACKGROUND (A PERSONAL VIEW!)

1. GIS originally focussed on paper map as output
   - anything is better than drawing by hand
   - no great emphasis on execution time
2. Paper output supports high resolution
   - display screen is of limited resolution
   - can admit less precise algorithms
   - Ex: buffer zone computation (spatial range query)
     a. usually use a Euclidean distance metric (L_2)
        - takes a long time
     b. can be sped up using a quadtree and a Chessboard distance metric (L_∞)
        - not as accurate as Euclidean — but may not be able to perceive the difference on a display screen!
        - as much as 3 orders of magnitude faster
3. Users accustomed to spreadsheets
   - GIS should work like a spreadsheet
   - fast response time
   - ability to ask “what if” questions and see the results
   - incorporate a database for seamless integration of spatial and nonspatial (i.e., attribute data)

GENERAL SPATIAL DATABASE ISSUES

1. Why do we want a database?
   - to store data so that it can be retrieved efficiently
   - should not lose sight of this purpose
2. How to integrate spatial data with nonspatial data
3. Long fields in relational database are not the answer
   - a stopgap solution as just a repository for data
   - does not aid in retrieving the data
   - if data is large in volume, then breaks down as tuples get very large
4. A database is really a collection of records with fields corresponding to attributes of different types
   - records are like points in higher dimensional space
     a. some adaptations take advantage of this analogy
     b. however, can act like a straight jacket in case of relational model
5. Retrieval is facilitated by building an index
   - need to find a way to sort the data
   - index should be compatible with data being stored
   - choose an appropriate zero or reference point
   - need an implicit rather than an explicit index
     a. impossible to foresee all possible queries in advance
     b. explicit would sort two-dimensional points on the basis of distance from a particular point P
        - impractical as sort is inapplicable to points different from P

6. Identify the possible queries and find their analogs in conventional databases
    - e.g., a map in a spatial database is like a relation in a conventional database (also known as *spatial relation*)
        a. difference is the presence of spatial attribute(s)
        b. also presence of spatial output
7. How do we interact with the database?
    - SQL may not be easy to adapt
    - graphical query language
    - output may be visual in which case a browsing capability (e.g., an iterator) is useful
8. What strategy do we use in answering a query that mixes traditional data with nontraditional data?
    - need query optimization rules
    - must define selectivity factors
        a. dependent on whether index exists on nontraditional data
        b. if no, then select on traditional data first
    - Ex: find all cities within 100 miles of the Mississippi River with population in excess of 1 million
        a. spatial selection first if region is small (implies high spatial selectivity)
        b. relational selection first if very few cities with a large population (implies high relational selectivity)

SPECIFIC SPATIAL DATABASE ISSUES

1. Representation
   - bounding boxes versus disjoint decomposition
2. How are spatial integrity constraints captured and assured?
   - edges of a polygon link to form a complete object
   - line segments do not intersect except at vertices
   - contour lines should not cross
3. Interaction with the relational model
   - spatial operations don’t fit into SQL
     a. buffer
     b. nearest to ...
     c. others ...
   - difficult to capture hierarchy of complex objects (e.g., nested definition)
4. Spatial input is visual
   - need a graphical query language

5. Spatial output is visual
   - unlike conventional databases, once operation is complete, want to browse entire output together rather than one tuple at-a-time
   - don’t want to wait for operation to complete before output
     a. partial visual output is preferable
        - e.g., incremental spatial join and nearest neighbor
     b. multiresolution output is attractive
6. Functionality
   - determining what people really want to do!
7. Performance
   - not enough to just measure the execution time of an operation
   - time to load a spatial index and build a spatially-indexed output is important
   - sequence of spatial operations as in a spatial spreadsheet
     a. output of one operation serves as input to another
        - e.g., cascaded spatial join
     b. spatial join yields locations of objects and not just the object pairs

CHALLENGES:

1. Incorporation of geometry into database queries without user being aware of it!
   - find geometric analogs of conventional database operations (e.g., ranking semi-join yields discrete Voronoi diagram)
   - extension of browser concept to permit more general browsing units based on connectivity (e.g., shortest path), frequency, etc.
2. Spatial query optimization
   - different query execution plans
   - use spatial selectivity factors to choose among them
3. Graphical query specification instead of SQL
4. Incorporation of time-varying data
   - how to represent rates?
5. Incorporation of imagery
6. Develop spatial indices that support both location-based ("what is at X"?) and feature-based queries ("where is Y"?)
7. Incorporate rendering attributes into database objects or relations
   - queries based on the rendering attributes
   - Ex: find all red regions
   - query by content (e.g., image databases)
8. GIS on the Web and distributed data and algorithms
9. Knowledge discovery
10. Interoperability

SELECTED REFERENCES
(Also see http://www.cs.umd.edu/~hjs/pubs.html)

1. F. Brabec and H. Samet. Visualizing and Animating R-trees and Spatial Operations in Spatial Databases on the Worldwide Web. in *Visual Database Systems 4 (VDB4)*, Chapman & Hall, London, 1998, pp. 123-140. http://www.cs.umd.edu/~hjs/quadtree http://www.cs.umd.edu/~hjs/pubs/v4s.pdf
2. C. Esperança and H. Samet. Experience with SAND/Tcl: a scripting tool for spatial databases. *Journal of Visual Languages and Computing*, 13(2):229-255, April 2002. http://www.cs.umd.edu/~hjs/pubs/sandtcl.ref.pdf
3. G. R. Hjaltason and H. Samet. Ranking in Spatial Databases. *Advances in Spatial Databases — 4th Symposium, SSD'95*, Lecture Notes in Computer Science 951, Springer-Verlag, Berlin, 1995, 83-95. http://www.cs.umd.edu/~hjs/pubs/incnear.pdf
4. G. R. Hjaltason and H. Samet. Incremental distance join algorithms for spatial databases. *Proceedings of the ACM SIGMOD Conference*, pages 237-248, Seattle, WA, June 1998. http://www.cs.umd.edu/~hjs/pubs/incjoin.pdf
5. G. R. Hjaltason and H. Samet. Distance browsing in spatial databases. *ACM Transactions on Database Systems*, 24(2):265-318, June1999. http://www.cs.umd.edu/~hjs/pubs/incnear2.pdf
6. G. R. Hjaltason and H. Samet. Speeding up construction of PMR quadtree-based spatial indexes. *VLDB Journal*, 11(2):109-137, November 2002. http://www.cs.umd.edu/~hjs/pubs/bulkload.pdf
7. H. Samet, H. Alborzi, F. Brabec, C. Esperança, G. R. Hjaltason, F. Morgan, and E. Tanin. Use of the SAND

spatial browser for digital government applications. *Communications of the ACM*, 46(1):63-66, January 2003.http://www.cs.umd.edu/~hjs/pubs/cacm03.pdf

8. H. Samet and F. Brabec. Remote thin-client access to spatial database systems. *Proceedings of the 2nd National Conference on Digital Government Research*, pages 75-82, 409, Los Angeles, CA, May 2002. http://www.cs.umd.edu/~hjs/pubs/dgo02.ps
9. H. Samet, F. Brabec, and G. R. Hjaltason. Interfacing the SAND spatial browser with FedStats data. *Proceedings of the 1st National Conference on Digital Government Research*, pages 41-47, Los Angeles, CA, May 2001. http://www.cs.umd.edu/~hjs/pubs/dgo01.pdf
10. E. Tanin, F. Brabec, and H. Samet. Remote access to large spatial databases. *Proceedings of the 10th International Symposium on Advances in Geographic Information Systems*, pages 5-10, McLean, VA, November 2002. http://www.cs.umd.edu/~hjs/pubs/acmgis02.ps
11. H. Samet. *Applications of Spatial Data Structures: Computer Graphics, Image Processing, and GIS*, Addison-Wesley, Reading, MA, 1990. ISBN 0-201-50300-0.
12. H. Samet. *The Design and Analysis of Spatial Data Structures*, Addison-Wesley, Reading, MA, 1990. ISBN 0-201-50255-0.
13. H. Samet. Spatial Data Structures. in *Modern Database Systems: The Object Model, Interoperability, and Beyond*, W. Kim, Ed., Addison-Wesley/ACM Press, 1995, 361-385. http://www.cs.umd.edu/~hjs/pubs/kim.ps

14. H. Samet and W. G. Aref. Spatial Data Models and Query Processing. in *Modern Database Systems: The Object Model, Interoperability, and Beyond*, W. Kim, Ed., Addison-Wesley/ACM Press, 1995, 338-360. http://www.cs.umd.edu/~hjs/pubs/kim2.ps