Robust Evaluation of Spatial Queries*

(Extended Abstract)

Renato Barrera†, Max J. Egenhofer, and Andrew U. Frank††
National Center for Geographic Information and Analysis
and
Department of Surveying Engineering
Boardman Hall
University of Maine
Orono, ME 04469
{renato, max, frank}@mecan1.maine.edu

Abstract

Evaluation of queries with requests for the aggregation of many detailed values in a database are of particular importance in Geographic Information Systems. They occur whenever a sum or a count for an area is requested and the individual data elements are stored. Geographic databases may keep versions of the same map with different levels of precision and these could be used to produce the answer more rapidly, perhaps with less precision. The more aggregated and less precise a representation is, the fewer instances are recorded and the less storage is occupied. For certain users and tasks, a result with less precision may be usable. For example, the display of an overview map may be sufficiently detailed based on the selection of a few significant objects.

A trade-off between precision of the answer and response time asks for optimization. If one includes the factor of time available to perform a certain operation, it is possible to treat this as a trade-off between precision of the result and processing time: the more time available, the more precise one can determine the result. The requirements for such a system are (1) to perform incremental evaluations and (2) to assess how much a partial result deviates from the final, "most precise" result so that users can be informed about the limitations of the answer.

---

* This work was partially funded by grants from Digital Equipment Corporation under Sponsored Research Agreement No. 414, TP-765536 and BW-213860, and Intergraph Corporation. Additional support from NSF for the NCGIA under grant number SES 88-10917 is gratefully acknowledged.

† Intergraph Corporation, Mail Stop IW17A2, One Madison Industrial Park, Huntsville, AL 35807-2174.

†† Department of Geo-Information (E127), Technical University Vienna, Gusshausstrasse 27-29, A-1040 Vienna, Austria.

Barrera, Renato, Max J. Egenhofer, and Andrew U. Frank. "Robust Evaluation of Spatial Queries." Paper presented at the 5th International Symposium on Spatial Data Handling, Charleston, South Carolina, USA 1992.

1 Introduction

Databases are grounded in mathematical precision (Ullman, 1982). Solutions to equations are exactly true and logical deductions are not approximate. Only recently probabilistic and fuzzy concepts were added to mathematics (Zadeh, 1968) and query processing, but only in very limited areas. Traditionally, databases follow the "absolute precision" semantics of query evaluation (with some exceptions in the area of statistical databases (Olken *et al.*, 1990)). All data in the database is used to process a query and a single, precise result is produced, which follows from standard logic and arithmetic. This strategy guarantees that the result is precise—based on the data available; however, it does not consider the processing time or the time the user has to wait for the result.

In interactive Geographic Information Systems (GISs), users expect almost instantaneous answers (Smith and Frank, 1990). GISs are often used to answer queries that require aggregation of data, such as counting and summing, for many small entities in a large area. For example, one may ask for the area of the state of Maine, which can be deduced from the area of the individual parcels, or the average age of the population for the whole U.S., which could be deduced from Census data that is stored per census block. Obviously this method is inappropriate for processing such queries: in all these cases, results were computed beforehand and stored and an answer is found with a single look-up.

In a GIS, users are often not interested in high precision, but rather have a quick, but possibly appropriate answer. First, consider how precise one needs to know the area of Maine and how well this quantity is defined. Second, interactive users of a GIS are often exploring the data in novel ways and attempt to solve new problems. They experiment with the power of the query language and need sometimes several attempts until they formulate the right query. They do not want to wait until the system has computed a precise answer, only to find out that they have asked the wrong question.

Processing spatial queries over very large amounts of detailed data will frequently exceed the time users are willing to wait for an interactive result and consume extensive processing resources. For example, processing the query for the area of the state of Maine by adding up the areas of all parcels in Maine will give a fairly precise result in acres—for the price of a long waiting time. Users have to wait until the entire data set has been checked, and prior to the end of query processing, no information is available. Sometimes they could find out quickly that they have asked the wrong query. A method to produce answers quickly, but perhaps imprecisely, is important in this situation; if more precision is required, the user may ask for a *refinement* of the result.

In the future we will increasingly see computer networks and access to databases over communication networks. Query execution will make such access "transparent, " i.e., automatic and not of concern to the user. However, access to data not stored locally may be much slower or in certain cases may even be impossible, because the remote data storage system is for some technical or administrative reasons currently unavailable. In such cases, execution strategies that trade absolute precision for a quick answer are important.

This paper proposes a novel alternative to the standard query processing strategy by trading precision for time. This strategy is different from other approaches to overcome the limitations of traditional query processing, e.g., to allow for incomplete information (Imielinski and Lipski, 1984), fuzzy queries (Zemankova, 1989), or uncertain values in the result (Barbara *et al.*, 1990; Morrissey, 1990). These efforts dealt with cases in which precise data is not available; here we attempt to produce quickly imprecise results, even when precise data is available. The robust evaluation of queries guarantees within a given time frame an answer (with an estimation of its precision), and offers users to get incremental improvements of the precision by allotting further processing time.

There is some confusion in common usage of terms describing data quality. We will use the terms *precise, correct* and *usable* in the following sense: *Precise* describes the attribute for any answer that was made by logical conclusion over the entire data set available. This is different from *correct* (or true), which means that the result corresponds to reality, a premise that data processing cannot guarantee. Finally, the term *usable* describes an answer that is sufficiently close to the precise result so that the user can make the intended decision.

Imprecision of a query result does not imply its uselessness. Users have particular purposes when querying, and if a measure of quality is associated with each result, they may assess whether or not the answer is appropriate for their goal. For example, if the query for the area of the state of Maine has to be answered within 2 seconds, the result may be 30, 000 square miles, ±10%. This answer may be already sufficient if the user wants to know whether Maine is larger then Florida (55, 000 square miles).

There are a number of similar situations, all characterized by an evaluation method by which an initial answer with limited precision can be given quickly and, with additional processing time and more data inspected, the results can be refined. The remainder of this paper focuses on the application of the robust query evaluation for GISs (Section 2), analyzes the problem (Section 3), and discusses methods to assess the precision of approximate results (Section 4). The conclusions in Section 5 summarize the design criteria for a robust query evaluation system.

2 Robust Queries for Geographic Data

This section describes some particular situations of spatial queries and geographic analysis, in which robust query evaluations are appropriate.

2.1 Aggregates over Areas

Many GIS users are interested in queries for aggregated information such as summing up values of instances, counting the number of objects within a certain area, and determining minima, maxima, and averages (Egenhofer and Frank, 1986). For example, "What is the size of an area?" "What is the total population?"

The dependency between a value of a composite object, such as a state, and values of its components, such as its counties, has been modeled on a conceptual level as propagation (Egenhofer and Frank, 1989). The actual calculation of propagated values may be particularly expensive if the aggregates have a very large number of components.

2.2 Multiple Representations

GISs may contain multiple records that represent the same objects (Buttenfield 1989; Bruegger 1989), either at different levels of resolution or using different perspectives for modeling. For example, data may be used to produce maps of different scales and, in lieu of deriving the different representations, they may be stored explicitly. If data is concurrently represented at different levels of resolution, an opportunity for using these different resolutions for robust query evaluation presents itself. A query processor can exploit the different levels of detail so that certain queries can be solved without the need to access the large amount of data of the most detailed representation.

The following scenario shows how a robust evaluation of queries over multiple representations may improve performance (Becker and Widmayer, 1990). It also demonstrates that the method is applicable for a broader range of operations than just "sum" and similar operations. To identify whether a region R contains a given point p, the point-in-polygon operation is first applied to the coarsest representation of R. If p happens to be far away from R's boundary, this search provides a usable result and no further access to more detailed representations is necessary. Otherwise, it will be necessary to repeat the operation at a more detailed level.

2.3 Gradual Evaluation

The user may need a query result within a short time to make an urgent decision. The time available may be less than the time necessary for a precise answer. It is then appropriate to process starting at the highest level and the result will be successively refined until the time available is used up. Depending on how well the result approximates the required value and if there is additional time available, the user may opt for further refinement of the result, while partial decisions are already made, based on the result with limited precision.

2.4 Evaluation with Given Error Bounds

A user may require an answer and be able to tolerate a determined level of error. In this case the evaluation can proceed from the highest level available and refine the result until the required precision is reached—achieving the usable result in minimal time with minimal processing cost.

2.5 Graphical Presentation

Methods of robust query evaluation can be applied to graphical presentations. For raster images, e.g., remote sensing data, it may often be sufficient to select a generalized representation to satisfy the user requirements. This is particularly important if one considers (1) the size of the data sets, (2) the time used for their transfer from storage to the user's machine, and (3) for rendering.

Various strategies are feasible:

- Select a sample, e.g., every tenth row and column. (Such a "compressed" image may be stored redundantly.)
- If the image is stored as a quad tree (Samet, 1989), select only nodes up to a certain depth of the tree.

Likewise for vector-based systems, boundaries of very large objects may be drawn sufficiently precise by dropping insignificant points. Depending on the time available, a more or less detailed representation may be selected (Douglas and Peucker, 1973; Ballard, 1981). There is evidently a strong conceptual link to map generalization (McMaster and Buttenfield, 1991).

3. Formalization of the Problem

In order to make the discussion specific, we formalize the assumptions for robust evaluation of queries. This step will allow us to analyze the problem and to describe solutions and their limitations.

3.1 Data Structure: Hierarchy of Spatial Areas

Space is partitioned in areal units such that no two areal units overlap and that all the areal units together make up the entire space (complete partition). We assume a hierarchy of partitions of space such that each level forms a partition and the subdivisions at the hierarchically lower levels make up a subdivision at the next higher level. This models, for example, political subdivisions where the world is divided into nations; nations into states; states into counties; and counties into towns.

As a notation we use indices such that all area units a_{11}, a_{12}, and a_{13} make up unit a_1. The number of indices indicates the level of subdivision and the last index identifies the area within the area of the leading sequences of indices. For example, a is the total area, $a_1 \ldots a_n$ are the first subdivision, $a_{j1} \ldots a_{jm}$ are the areal units of the second subdivision that make up areal unit a_j, etc. For each spatial unit of the lowest hierarchical level, there is a vector of attributes that describes the properties of this area:

$$f(a_{n\ldots z}) = v_1 \ldots v_l$$

3.2 Aggregation

Each areal unit in the hierarchically higher levels has a vector of attributes, which form aggregate values for the attributes of the subdivisions in this areal unit. The method to aggregate depends on the type of the data value (essentially the "scales of measurement" (Stevens, 1946)). There is evidence that only a limited set of functions can be applied in this situation. For example:

- Attributes of type 'count'—aggregation by summation, but also minimum, maximum, and variance may be used.
- Nominal value (e.g., land use classes)—the aggregate may be the set of all the values that occur or the value (and its frequency) that occurs most often.

3.3 Distributed Databases

If the data for a region a_{ijk} is stored in a separate database—for example assume that the data is stored in a county government database—then aggregate data can be stored at higher levels (and may even be kept at various locations—for example the state aggregates may be kept in many agencies that use it). If the aggregate data is to be kept precise, then each update to any of the detailed databases at the third level must be

communicated to all the levels above (and from there distributed to additional storage sites).

The amount of communication required to propagate each change is impractical. It would be more rational to allow the higher level (e.g., state) databases to be slightly out of synchronization, only updating them when the accumulated changes add up to more than a certain threshold. The higher-level database is then always accurate up to a certain margin of error. Provided this error margin is acceptable, requests can be satisfied without a need to access the county database.

3.4 Queries

Queries for attribute values (or functions of these values) of areal unit at any hierarchical level can be retrieved directly from the database. If a query asks for an aggregate value of an area that is not one of the precomputed ones, the area requested can be approximated with areal units of the highest level (fully contained) and then the approximation can be improved using areal units of lower levels.

More difficult are complex queries, where one asks for a selection of the area based on some criteria. For example, "Determine the area in which population density is over a certain threshold." A possible strategy is to select all highest-level areal units and determine whether any of them has a population density higher than the threshold. These areas may be summed up. In a first refinement step, this is repeated for all areal units of the next lower level.

4. Assessment of Error of the Result

Approximate results are usable if the user has a method to assess the precision of the value provided. It is, of course, not advocated to provide users with approximate answers without informing them about the approximation and allowing them to decide whether the information provided is of sufficient quality to be usable.

If the information is provided graphically, the scale of the map shown and other aspects of the graphics often imply the quality of the data. There is research underway to find good ways to communicate data quality to the user and help them assess the results computed (Beard *et al.*, 1992). Here we sketch some methods to compute those aspects of the data quality that are associated with the approximation produced by a robust query execution strategy.

4.1 Standard Deviation

The traditional method to assess error associated with an observation is to give the *standard deviation*, often written like 3, 450 square miles ± 24 square miles. This indicates that the precise value is with 66% probability between 3, 426 and 3, 474 (assuming a normal distribution of the error).

There are a number of reasons, why this may not be appropriate:

- The error may not be normal distributed (because data values systematically increase in time, thus a time lag in the update results in a systematic underestimate).
- The not so likely but possible larger deviation of the result from the expected one may be very significant information.

4.2 Maximum Deviation

The error associated with a value may be described by giving the interval bounds for the maximum or minimum value—possibly together with the most likely one. In the scenario of distributed databases with updates delayed until a certain threshold is met, bounds for the maximum deviation are set by the thresholds for updating the aggregate value.

5 Conclusions

A number of design criteria for a robust query processor can be derived from the above examples:

- Produce approximate results either within a time limit or a precision requested.
- Estimate the deviation from the precise results.
- Continue the evaluation if additional time has been allotted or if the precision requested has not been reached.
- Reuse results from previous queries.

Of course, a query once evaluated may be stored in the aggregate database and used for the evaluation of later queries—assuming that data that was once interesting for a user may again be useful. This is certainly a better strategy than to attempt to precompute any value possible.

The query evaluation mechanism and the user interface may interact such that any result, from the first approximation on, is made available to the users and they can stop processing whenever a usable result is achieved. Also the interface must allow the user to further refine the result—even if it was a complete answer to the query within the precision requested—without redoing the initial computations.

Substantive research is necessary to classify the aggregation methods that are appropriate for a GIS and to see how they interact with a robust evaluation strategy. What values can be precomputed? Which method to help the user assess the usability of the result is most appropriate and how is it computed?

References

D. Ballard. 1981, Strip Strees: A Hierarchical Representation for Curves. *Communications of the ACM*, 24(5):310-321.

D. Barbara, H. Garcia-Molina, and D. Porter. 1990, A Probabilistic Relational Data Model. In: F. Bancilhon, C. Thanos, and D. Tsichritzis, editors, *Advances in Database Technology. Lecture Notes in Computer Science Vol. 416*, pages 60-74, Springer-Verlag, New York, NY.

K. Beard, B. Buttenfield, and S. Clapham, 1991, NCGIA Research Inititative 7: Visualization if Spatial Data Quality, Technical Report 91-26, National Center for Geographic Information and Analysis.

B. Becker, and P. Widmayer. 1990, Spatial Priority Search and Access Techniques for Scaleless Maps, Technical Report, Institute for Informatics, University of Freiburg, Germany.

B. Bruegger. 1989, Hierarchies over Topological Data Structures. In: *ASPRS-ACSM Annual Convention*, pages 137-145, Baltimore, MD.

B. Buttenfield. 1989, Multiple Representations: Initiative 3 Specialist Meeting Report. Technical Report 89-3, National Center for Geographic Information and Analysis.

D. Douglas and T. Peucker. 1973, Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature. *Canadian Cartographer*, 10(2):112-122.

M. Egenhofer and A. Frank. 1986, Connection between Local and Regional: Additional `Intelligence' Needed. In: *FIG XVIII. International Congress of Surveyors, Commission 3, Land Information Systems*, Toronto, Ontario, Canada.

M. Egenhofer and A. Frank. 1989, Object-Oriented Modeling in GIS: Inheritance and Propagation. In: *AUTO-CARTO 9, Ninth International Symposium on Computer-Assisted Cartography*, pages 588-598, Baltimore, MD.

T. Imielinski and W. Lipski. 1984, Incomplete Information in Relational Databases. *Journal of the ACM*, 31(4):761-791.

R. McMaster and B. Buttenfield. 1991, *Map Generalization: Making Rules for Knowledge Representation*, Longman, London.

J.M. Morrissey. 1990, Imprecise Information and Uncertainty in Information Systems. *ACM Transactions of Information Systems*, 8(2):159-180.

F. Olken, D. Rotem, and P. Xu. 1990, Random Sampling from Hash Files, SIGMOD 1990, *SIGMOD RECORD*, 19(2):375-386.

H. Samet. 1989, *The Design and Analysis of Spatial Data Structures*. Addison-Wesley Publishing Company, Reading, MA.

T. Smith and A. Frank. 1990, Very Large Spatial Databases: Report from the Specialist Meeting. *Journal of Visual Languages and Computing*, 1(3):291-309.

S. Stevens. 1946, On the Theory of Scales of Measurement. *Science Magazine*, 103(2684):677-680.

J. Ullman. 1972, *Principles of Database Systems*, Computer Science Press, Rockville, MD.

L. Zadeh. 1968, Probability Measures of Fuzzy Events, *Journal of Methematical Analysis and Applications*, 10:421-427.

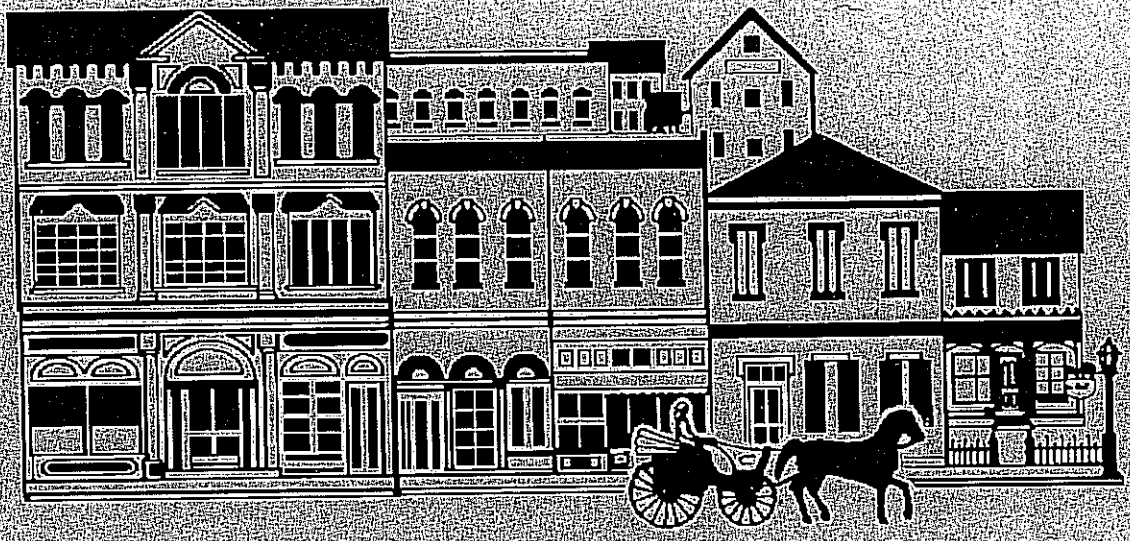
M. Zemankova. 1989, FIIS: A Fuzzy Intelligent Information System. *Database Engineering*, 12(2):75-84.

PROCEEDINGS

5TH

NAL SYMPOSIUM ON

SPATIAL DATA HANDLING



IGU Commision on GIS

August 3 - 7, 1992
Charleston, South Carolina
USA

Volume 1