The design of an urban GIS to manage frequent spatial updates.

Christopher M. Gold and Weiping Yang

Chaire industrielle en géomatique appliquée à la foresterie
Centre de recherche en géomatique
Pavillon Casault, Université Laval
Ste-Foy, Québec, Canada G1K 7P4
Tel:(418)656-3308 Fax:(418)656-7411
E-Mail: CGOLD@VM1.ULAVAl.CA

ABSTRACT

It is the nature of large urban regions that both cadastral and facilities information need to be updated frequently. In the context of traditional GISs this poses something of a dilemma: does one manage these modifications of the spatial structure merely in order to have one current “map”, erasing all knowledge of the previous state of the map, or does one attempt to preserve “snapshots” at regular time intervals, with all the difficulties of storage and comparison between the snapshots?

This dilemma is a direct result of the “batch” mentality inherent in the first modern GISs, where the topological structure was obtained by detecting all line intersections in the map area and then constructing the whole boundary network at one time. This required that all topological linkages be re-created after any modification - unlike a modern data base system, where all indexes are dynamically maintained however many records are inserted or deleted.

This updating dilemma is only a problem for batch-oriented GISs but, where true dynamic spatial data structure maintenance is available, updating may be handled in a sequential interactive fashion. The “topology“ is then kept up to date during, rather than after, an editing session. A consequence of this sequential approach is that editing operations may be assigned a time and date, and that a log file of the input commands may be preserved in order to re-generate the topology in an identical sequence to the original data entry. This “tape recording” may be stopped at any historical date desired, giving the complete map at that date. The tape may then be played further, bringing the map further up to date. Finally, new digitizing may be entered, updating the topology and adding new commands to the previous end of the tape.

It is suggested here that the maintenance of a historically valid map may be greatly simplified by using dynamic spatial data structures, and that this is of particular value in a rapidly-changing urban setting. Those commands (e.g. line segments), on the earlier portion of the tape, that were subsequently removed provide, at minimal cost, both a record of the past situation for legal or archival purposes, as well as an audit trail for validation of the map updating activities within the organization. Ongoing research at Laval University using Voronoi methods has provided an appropriate dynamic spatial data structure that is currently being developed for spatio-temporal event management.
INTRODUCTION

This paper is about the rather abstract subject of how to model or represent space within a computer, although the main interest is in the new applications and flexibility provided by a new way of thinking about the problem. I will therefore describe the approach used in some older systems, and the operational consequences. This will be compared with the new methods developed at Laval University, followed by a brief outline of the approach, and the implications. I will put this in context by describing a GIS application in interactive spatial decision support. The problems described apply to vector (line-drawn) rather than to raster (image) map information. To give a sense of scale, the basic Voronoi program operates as a few thousand lines of code, in a PC environment.

SPATIAL DECISION SUPPORT SYSTEMS

There are various ways of classifying spatial decision support systems (Gold, 1993a). For the purposes of this article I will distinguish between global optimization methods, which are usually slow and difficult to interact with, and what I have called the “flight simulator” approach, which largely depends on the operator or decision maker to propose tentative actions, which are then evaluated by the machine. (An example is evaluating the consequences of sketching a new highway, and identifying all land parcels within some fixed distance: which road requires the least amount of compulsory purchase?) This assumes that the primary need is the ability to make small-scale decisions rapidly, and thus to have a spatial decision support system that really is interactive. This means that “what-if” modelling is possible, as with a spreadsheet: the user suggesting a scenario and the machine responding sufficiently rapidly that the decision-maker has not forgotten the context of his query. The objective is not to provide a globally optimum solution, but to allow the operator to make reasonable (and non-catastrophic!) decisions.

PLANE VERSUS BOAT: TWO APPROACHES TO INTERACTION WITH THE MAP

There are two main types of interaction with the supplied terrain information, that can most easily be visualized as the “plane” and “boat” approaches, and procedures involving combinations of these will be used in many GIS-type queries. For example, “plane” type digitizing has lines being drawn with fine disregard for the existing map “underneath”, and only afterwards, with some relatively slow batch operation, are any spatial relationships established between the lines being drawn and the pre-existing map. (Many simple spatial queries are, by their nature, entirely of the “plane” type as they do not modify the spatial structure underneath.) By-contrast, “boat” type digitizing has the cursor interact with the existing portions of the map, as it, being on the same level, would be in danger of collision with the land if real-time feedback were not available.

TRADITIONAL GIS SPATIAL OPERATIONS

This distinction is of importance for the development of interactive “flight-simulator” decision support systems. Traditional GIS queries may be of two types: those that do not, and those that do, modify and rebuild any topology or spatial relationships. Examples of the first include totals of polygon areas, or all polygons adjacent to some specified polygon type. Whenever the query
involves the three classic spatial GIS operations, however, then topology modification or construction is needed. These operations are usually quoted as: polygon reclassify and merge; buffer zone or corridor construction; and polygon overlay. These have been considered the “difficult” GIS operations because, as traditionally performed, they all involve rebuilding the polygon structure - and this is a global operation, because it depends on detecting all the intersections of potential boundary segments. For many years this was the central operation, with lots of special cases, and very much a trade secret.

This operation was an expensive global process, which became the bottle-neck and focus point of GIS processing, because of the absence of a meaningful operational definition of the term “local” - thus enforcing a complete rebuild after every topological change. Thus the “plane” model had no interaction with the pre-existing map structure, and after all the lines were added the global polygon rebuild was required. In addition, note the emphasis on “polygon”. Most activities were for regional analysis, and the polygon topological structure could be validated in part for data entry errors. For data in the form of stream networks, or independent data points, there was less, or no, topological validation. Islands were not readily associated with their containing polygons, as no line intersections were observed. The model of space was inadequate for flexible spatial operations.

A SPACE-FILLING, DYNAMIC SPATIAL MODEL

So much for the “plane” approach. Enter the “boat” model of space. Here the operator, pen or cursor is part of the map world, and interacts directly with it. This allows local modifications of the map without global reconstruction. This is achieved by rejecting the line-intersection model of space and replacing it with the proximal model: all space is always fully occupied, and consists of tiles or bubbles around each map object (point or line segment). This is also called the Voronoi model. Objects that were disconnected in the line-segment model may well be neighbours in the Voronoi model, if their bubbles touch. A bubble is defined as enclosing all locations closer to that map object than to any other. The “boat” representing the operator or cursor is also a map object with its own bubble, and thus has a set of neighbours with which it interacts. This permits many operations to be performed in a local fashion, rather than globally, as well as new operations not even feasible with global procedures.

Research in computer generation of static Voronoi diagrams has become increasingly active in the last few years, as a potentially valuable tool for handling the difficult conversion from finite-precision coordinate geometry to the more desirable processing of spatial information in graph-theoretic terms. Many approaches have been developed for constructing Voronoi diagrams for point sets, and this has proved useful for various interpolation problems. A few workers have developed static (i.e. global) algorithms for sets of points plus line segments, but to my knowledge we at Laval are the only developers of dynamic methods for points and line segments, as well as developing a wide range of applications - some being emulations of traditional GIS, and some being completely new. The main benefit comes from the dynamic maintenance of the spatial data structure, for example permitting the user to work interactively with questions about reorganizing administrative districts, evaluating the zones, of influence of traffic pollution with suggested new roads, or for traffic flow analysis.
Using the Voronoi or bubble spatial model lines may be drawn, deleted or intersected, and points may be created, deleted or moved directly by the user - as in a fully interactive digitizing system. Figure 1 shows the generation of a line segment “bubble” within the well known point Voronoi diagram. Also shown is the equivalent Delaunay triangulation. Figure 2 shows the triangle-switching operation used to maintain the spatial structure interactively as one of the points (labelled P) moves to the left (from Gold, 1993b).

Because of the proximal model of space that is used, it is not necessary to draw on top of, or point precisely at, an existing object in order to select it for use (e.g. to snap a line to it). Merely pointing nearer to that object than to any other will suffice. (This is the same as pointing anywhere within an object’s bubble.) This follows pretty closely the human gesture when selecting something on a map, and some work has been done on relating the Voronoi spatial model to human perception of spatial relationships (Gold, 1992d). Polygon building in an incremental manner is thus greatly simplified. (Figure 3 shows a simple example.) The usual difficulties with the “polygon merge” operation are handled in a well-behaved local fashion, as individual line segments may be added or deleted as required. The other classic single-coverage operation, the construction of corridors or buffer zones around selected features such as roads, also becomes a trivial operation, since all corridor boundaries must fall within the proximal zone (Voronoi region) of the point or line segment component of the feature. No intersection tests or other global operations are required. The last standard GIS operation, “polygon overlay”, is usually thought of as, for example, combining coverages “A” and “B” to produce coverage “C”. Within an incremental system this may be performed by tracing each segment of coverage A through coverage B, generating the required additional colours and intersections, of each line segment, to form the final combined coverage C, which is itself built as a Voronoi diagram. This step is repeated tracing B through A to form a complete polygon overlay, but for many applications this second step is not really required.

OPENING UP THE FUTURE

So much for standard polygon GIS operations. Operations on incomplete polygon structures, and those having apparent dangling segments, are readily handled - either by flagging them as errors, or else permitting them, as required. Network structures are managed with exactly the same tools. Interpolation problems, both with point data and with linear data elements or faults, follow the same path also - indeed most of these techniques were originally developed for terrain modelling purposes. Many restrictions due to the old topological structures are alleviated, new combinations of queries become feasible, and dynamic interactive operations may be developed.

Another class of operations is not normally considered feasible within a GIS: the real-time interaction of the user with the digital map - the “virtual reality”. As we have seen with the interactive digitizing example, the “boat” approach allows the user to query or respond to the existing map, as well as to modify it. Two immediate applications come to mind. The fast suggests a robotics application, where the “boat” or cursor may be given various rules about topography, slope, obstacles or roads and will attempt to determine a feasible path, perhaps with human assistance from time to time. (Static Voronoi diagram “skeletons” are already used for this, but merely to provide a feasible graph through a maze.) The integration of polygons, networks and interpolation within the same structure permits many similar dynamic applications.
The second use of the interactive “boat” approach concerns our original “flight-simulator” view of spatial decision support systems. Here the operator is making local tactical suggestions (perhaps a new road segment), and the machine is expected to evaluate the proposed action. In the pollution case, perhaps a buffer, zone is to be constructed about the branch road, overlaid with the property map, and a summary of the affected properties given. Given the local nature of the Voronoi method this is a rapid operation, and does not require the rebuilding of large portions of the map. The response to the user must be fast - a few seconds - because the nature of the “flight simulator” is to “fly” the problem, adjusting things and trying again, until two things happen: the operator gets a “feel” of the way the system is responding, and a reasonably satisfactory action (e.g. the road selection) is accepted. In many operational settings this is preferable to a somewhat artificial “optimal” solution, that nevertheless appears unreasonable in the field.

CONCLUSION: THE REST IS HISTORY

The final aspect of our research that I wish to cover follows on directly from the incremental nature of the method. This means that actions were performed in a particular sequence, and these may be recorded, played back, and modified. This introduces the idea of “the map as a movie”. A good example is urban property mapping. Given the current property map, it is necessary to update it with the latest information on a regular basis. Thus the original map is modified. What happens, however, when it is required to evaluate the history of a particular area? The whole original map has to be compared with the updated one, creating problems, and this is compounded if a long-term history of various uses is required, e.g. to evaluate planning strategies.

In the incremental approach, any changes to the map are merely appended to the previous information, together with relevant date information. The “tape recording” of the map is then played back to any desired date, for evaluation. Comparisons may then be made at any location, as to what changes occurred, and when. As with the whole Voronoi approach, this is event-driven over time, not a series of snapshots, removing much redundancy. For further details see Gold (1993b).

This brief review has been intended to sketch out the general implications of a new, more flexible approach to modelling space and time. It is object-oriented, rather than coordinate-oriented, and we are only just beginning to enumerate its applications.
Because this work has followed a consistent path over some years, and because much of it is found in conference proceedings, we have taken the liberty of listing some of the results by Laval researchers, in the hope that they may be of interest. This does not suggest that no work has been done by others!


Figure 1a  Point Voronoi diagram

Figure 1b  Addition of line segment

Figure 2a  P and Q are not neighbours

Figure 2b  P and Q have become neighbours

Figure 3  Voronoi regions of a simple map