

The Digital Simulation of Geographic Space and Time

Christopher M. Gold

Centre de recherche en géomatique

Pavilion Casault

Université Laval

Québec, Qc

Canada G1K 7P4

E-Mail: Christopher.Gold@scg.ulaval.ca

1. Introduction

This paper attempts to address the issue of simulation within geographic space in a general manner. It is clear that there are various types of simulation for various applications, but it is also clear that each of these applications has been treated as a specific case. This is based partly on history, as each discipline developed its own tools, but also due to the absence of any consistent conceptual model of space, and thus of any general purpose data structure. Encouraged by our work to date in the use of the Voronoi spatial model for various different applications [5], this paper is an attempt to "put it all together" and talk about simulation in general terms within space and time.

The paper starts with a discussion of the properties of the space which we inhabit. By inhabiting it, we impose a variety of conditions on the space we are working in, such as reasonable ranges of scale, limited definitions of coordinates, and relations to other objects embedded in this space that we will call "geographic". Attempting to transfer a model of this space to the computer imposes additional, and usually undesirable, problems, primarily concerned with discretization. As critical, however, is the need to transfer to the computer some of the properties of the space we are trying to simulate. In my opinion one of the most important of these properties is the inability of two objects to occupy the same space at the same time. This must be adequately handled.

By inhabiting space we must be able to operate within it. This requires navigational tools, such as proximity detection, but also a concept of time within which the actions take place. A "snapshot" view of time, where the

whole of the map space is reconstructed after a specified time interval, does not correspond to our experience. For us, change over time is a sequence of individual changes, each with specific locations in space and time, or else some continuous variation of a perceived phenomenon. Our spatio-temporal model must accord with this.

With satisfactory models of space and time, simulation in geographic space (and time) becomes better defined. It consists of the modelling within the computer of a temporal sequence of localized spatial changes. These changes may be induced by the intervention of the operator, or by the mathematical simulation of some physical process. Nevertheless, the general process of simulation must incorporate these localized changes. An attempt has been made here to determine what types of changes are required in order to handle a reasonable selection of applications in geographic space/time. These are: change of attribute over space (Z/S); change of attribute over time (Z/T); and change of location over time (S/T). These may relate to the interior of a region, or to a boundary or discrete object - and they may be discrete jumps or smooth changes.

With the ability to preserve these points of change, rather than every possible space/time location, it becomes feasible to simulate a wide variety of spatial processes. The Voronoi approach is compared with these requirements, and is found to be appropriate in many cases.

2. Concepts of space

Space is a concept to which we are so accustomed that we find it difficult, to discuss. When we attempt to model this within a digital computer we transfer our presuppositions and prejudices to this new environment without much reflection. As space is an aspect of many disciplines (as are computers), 'each with their own traditional perspectives, this is a recipe for confusion and misunderstanding.

Geographic space, as commonly used, has several implications. We are talking about Euclidean space, the space within which we, as individuals, are embedded. We are talking about a spatial scale that corresponds to some extent with our own stature - neither cosmic nor atomic. By "scale" we mean some expression of the approximate distance between discrete objects embedded within it - we are interested neither with the spacing of the stars, that fall outside our field of interest, nor with the individual atoms that form,

in aggregate, the objects that interest us. Geographic space may also imply that we are concerned primarily with what affects us, as creatures moving about on the skin of the globe. Frequently this suggests that two dimensions are adequate, since, due to gravity, relatively few differing objects may be observed to be superimposed within the third dimension. (This is notably not true for the oceans, which explains why marine GIS is still in its infancy.)

When we attempt, to move space to a computer, we are of necessity simulating space in a digital environment. Our space was continuous: this is not possible in a discrete digital world. Instead of having one primary coordinate axis, reflecting the primacy of gravity, together with a system of orientation perhaps related to solar "movement," we have to develop a somewhat arbitrary set of axes and units that reflect the not necessarily geocentric notion of Euclidean space. In (or on) the real world we have fundamental constraints preventing two objects being in the same location at the same time. This is not automatically the case within a digital simulation of space. Indeed, one could say that this is the single greatest limitation of many current systems. A closely related property is that of object, proximity. In the real world collisions (two objects attempting to occupy the same space at the same time!) are avoided not by the use of calculations in global coordinate systems, but by one of many kinds of natural or artificial radar systems, regularly estimating the distance between the observer and a selected set of nearby objects. Thus we require local direct measurement systems, and some definition of "nearby" objects. A previous paper described how the Voronoi model of space may be used for this purpose, and may even relate to human spatial perception [3].

A critical issue in the simulation of space in the computer is its necessary discretization. This occurs in two parts. The first is the much-discussed subject of finite-precision arithmetic (with or without Pentium errors). The simplest example of this problem is that the calculated point of intersection of two lines will rarely fall on either line. More seriously, this may cause different decisions to be made for the same situation if, for example, it is based on which side of a line a particular point falls. This problem may prove to be exaggerated, as what is required is consistency, rather than precision. It does not matter if a point-in polygon query returns the "wrong" answer for a point very close to the boundary between a pair of polygons - the difference is, in practice, invisible (and certainly less than the data precision). What does matter is that the query must return with *one* polygon, and not zero or two. The moral: much spatial analysis is based on the analysis of

graphs (something computers do very well). What is critical is the conversion between coordinates and graphs. Another moral: in the real world such high precision has no meaning, and within such close proximity to the line the point being evaluated would be deemed to be "on" the line. This requires that a previous statement be emphasized: proximity is as, or more, important than intersection.

The second issue in discretization concerns not computer arithmetic but the very continuity of space itself (as far as we can tell). Yet it must be stored in discrete chunks in the computer used to simulate this space. How do we view space in this context? As unoccupied, except for objects you bump into, or as the continuous variation of some parameter, such as temperature or population? If it is empty except for dangerous objects, then the only thing to do is to navigate within it, - which requires a proximity detector, such as radar, vision or a bat's squeak, in order to change direction or inflate the air bags. If it is a continuously-varying parameter in the real world, then it ceases to be so in the computer - it must be broken up into chunks deemed to have the same value throughout each interior (and an infinite gradient at a border with a cell having a different value). One may pretend that these chunks are atomic - they are not capable of being subdivided further. Image analysis is based on this happy supposition. If reality becomes too intrusive and supposedly atomic chunks must be subdivided further - as in the case of forest stand polygons - then we get the data base problem equivalent to dividing a driving license record into two (presumably after one of these unfortunate attempts of two objects to occupy the same location at the same time). The result is to have two driving license records with the same keys and attributes - which is disapproved of by data base managers, who only have one license. Forest polygons are obviously not atomic objects, as they may be cut in two by an intermediate forest cut - but they are saved in the data base as if they were, until they are not, when problems arise.

The whole concept of a forest map has many problems, and it shares these with the most common type of GIS analysis - the analysis of polygon coverages. These polygon coverages may be completely imaginary, as with political boundaries or zones. These are difficult to detect in the real world, except perhaps by the presence of different-coloured policemen. Thus they may be as precise as one wishes, and with completely arbitrary attributes (the polygons, not the policemen). They may, however, represent some phenomenon observable in the real world, such as temperature. In this case we know that the phenomenon really varies more-or-less continuously, but for

some reason we aggregate by polygon. This may be for good reasons (as when population data was aggregated by census district, and that is all the data you will get), or for less good reasons (as when temperature was observed at some point location, and then assigned a polygon in order to process it with standard software). In either case the variation of the attribute over the map has been made discontinuous to fit the spatial data model in the computer.

In the case of forest maps, as well as census districts, it appears that there is a continuously varying attribute that has been broken into regions. This however is not true: our simulation of reality would (if the data were available) first take a collection of individual spatially located organisms, generalize a point density function of trees or people on the basis of some particular sampling area (very small radius gives a punctual map; very large gives a map average), and then categorize the resulting continuous function as before. Clearly there is significant loss of information through these transformations. Thus again our simulation of space is severely limited by the discrete model required by the computer, as well as by the data collection procedures. (It should be noted that in some applications the "number of objects per unit area" question, with the problems of defining the unit area mentioned above, may be replaced by the "area per object" approach - in other words, the reciprocal, based on the Voronoi diagram of the individual objects.)

3. Simulation of time

The previous statements pose some of the well-known difficulties with modelling space. Modelling time has some of the same constraints. In particular, traversal of the time axis may be compared with traversal of some line across a map. As space may be defined as fixed pieces with attributes attached (e.g. pixels), so may time. This is the "snapshot" approach. As space may be considered as a voyage, encountering obstacles or region boundaries ("changes" or "events") so may time. Indeed, a navigation-based simulation of space involves the concept of time directly, in that a particular observer encounters a particular event at a particular location, at a particular local time. Thus spatio-temporal change involves the documenting of events/changes at particular locations in space/time [4]. One particularly interesting aspect of the Voronoi simulation of space is that (assuming for the moment that database time equals real time) the modification of a map is itself a simulation of the

sequence of events that were required to produce it [6].

In particular, considering the operator constructing the map (or his pen, cursor, etc.) as an agent embedded within the space of the map gives a special context to the concept of scale. If the operator is embedded in the map space he/she may interact with the objects found therein. Attempted collisions have consequences, as in the real world. This is the "boat" interaction model. In the alternative "plane" model there is no interaction, as in spaghetti digitizing, with its subsequent attempts to evaluate the sum of all interactions/intersections. There are in general two types of scale possible:

one in which it is impossible for the observer to be a part of the virtual reality being modelled (because he is too large or too small) or else one where he is. In this latter case the "boat," model may apply, and the observer interacts with the two or three dimensional world already present, and has the ability to modify it at that time (and place).

4. Simulating process in space and time

Given reasonable responses to the problems posed above, modelling geographic processes may become relatively simple. If a map changes over time, the representation in the computer simulates this if it tracks the changes in the real world, subject to the constraints described above. At the minimum one redraws only those portions of the map that have changed. In the case of a simple vector map this requires a locally updatable topological structure. For example, if the data states that a particular forest polygon boundary changes between one time step and another, the old arc must be moved (or deleted and re-drawn) at the appropriate simulation time. This gives the "map as a movie" approach. While this may not be simulation in the usual sense, this readily becomes so if the boundary movement may be expressed as the result of some function describing a process (such as forest expansion).

We have proposed above [4] that simulation (or change) over space, with time, may be modelled within the computer not by snapshots of the whole map at regular time intervals, but by local update of the location of spatial change, at the time that this event occurred. In addition, in order to preserve our spatial model, the topological structure must be maintained at all times. This differs from other workers [?], [9], [11] where the emphasis has been on the management and query of spatio-temporal information within the data base.

To clarify our terms, we can define several types of "change" within the map.

The first is change in attribute over space (Z/S). A simple continuous example is the gradient, at any location, of a terrain model. A discrete example is the jump in the attribute value at a polygon boundary in a choropleth map. The second type of change is the change in spatial location over time (S/T). This would be a discrete jump when a map boundary is updated by redigitizing, or a continuous change over time if a point movement simulation was being performed, for example for robot navigation or free-Lagrange fluid flow modelling [6]. The third type of change is a change in attribute value over time (Z/T), as with the growth of the interior of a forest stand over time.

These three types of change in space/time may apply to an object (such as a polygon boundary, or a point) ("b"), or to a field (such as the interior of a polygon, or a grid square) ("i"). They may be discrete ("d") (non-differentiable) changes (e.g. at the boundary between two polygons with differing attributes), or continuous ("c") (differentiable) change, for example at any location on a surface defined by a mathematical function. Not all combinations make sense with any particular type of map, but, since the alternative is to store all values of the attribute for all combinations of spatial location and time, it is desirable to store only non-zero *changes* at particular locations in space/time, and these will be subsets of the classes just described. Some examples will hopefully clarify the concept. Note that in some cases, especially after the necessary discretization of a continuous spatial function, there may be more than one way to represent the change.

5. Examples of change in maps

- A static polygon map, classifying a region into various categories: save Z/S(b,d).
A traditional choropleth map illustrates discrete change over space at the polygon boundaries - "walking" across the map, an observer would notice differences only at boundaries. Thus saving each boundary and the values on each side would suffice to describe the map. Of course, other topologically-equivalent alternatives, such as pointers to the polygon interior associated with each attribute, are possible.
- A forest map, with simulation of forest growth within each polygon: save Z/S(b,d) and Z/T(i,c).

Here a complete model of change within the map would require the boundary, as marking the location of spatial change, as well as the polygon interior, associated with the change of the attribute (forest growth). The topological link between the boundary and the polygon interior would be the most desirable way of identifying change across the boundary at any particular moment in time.

- The forest map as above, but with simulation of boundary spread: save $S/T(b,c)$ as well.

The rate of movement of the boundary of one forest type encroaching on another would need to be preserved for each boundary. Note that at some moment in the simulation individual boundaries will appear or disappear, giving "topological events" as well [12]. In the case of the approximation of continuous boundary movement, by the re-digitizing of forest stand boundaries every few years during forest inventories, the continuous change $S/T(b,c)$ will become discrete $S/T(b,d)$.

- A contour map, or other representation of a smooth surface: save $Z/S(i,c)$ throughout.

Here the gradient of the surface should be preserved at all possible locations. This not being possible, except implicitly for mathematically defined functions, approximations are made. In one case a grid is used to partition space into polygons, permitting the surface to be represented as a choropleth map, but using implicit boundaries. The other common alternative is the contour map, where the contour lines themselves, implying zero change along their length, indicate a maximum change (of the attribute over space) perpendicular to themselves.

- An undulating surface, changing over time: save $Z/S(i,c)$ and $Z/T(i,c)$ throughout.

Here, as before, in the ideal model of space there would be no discretization but changes of attribute over space, and attribute over time, would be continuous within the map interior. In the grid approximation $Z/T(i,c)$ would be saved at each grid cell, whereas in the contour representation $S/T(b,c)$ would preserve the displacement of contour lines over time.

- A finite-difference flow model, with water movement between fixed cells: save $Z/T(i,c)$ and $Z/S(b,d)$.

This is the same situation as for the forest growth model, with change across boundaries as well as within each cell. Note that flow across the boundary is a function of the gradient between cells. In finite difference methods this is presumed not to be infinite at an abrupt boundary, but to be defined by the attribute difference between cells divided by the spatial separation of the cells' "centroids".

- A free-Lagrange type of flow simulation [2], where packets of air or water interact with each other and move: as above, but save $S/T(b,c)$ also.

This is the same case as the forest model with moving boundaries. The boundaries are often represented by Voronoi cells generated automatically around the "centroids" representing the centres of mass. Thus $S/T(b,c)$ for each boundary may be generated from the relative motions induced in each centroid. In addition, topological events also occur as cell boundaries appear and disappear.

- Robot navigation through a field of obstacles: save $Z/S(b,d)$ and $S/T(b,c)$ (if using the Voronoi navigation model [6]).

Here again we have the case of the forest model with moving boundaries, but without attribute change over time within each cell. Attributes associated with each object and its surrounding cell are usually object labels. Boundaries are generated automatically, and the primary information required is the set of spatial relationships of pairs of objects with a common boundary (or of edges in the equivalent dual triangulation). This gives the change $Z/S(b,d)$. $S/T(b,c)$, as in the free-Lagrange method, may be represented by the relative motions of the generating objects. For a single moving robot only one object will be moving at a time. Thus the triangulation plus the current movement of the robot suffices.

5.1 Summary

We have identified three forms of change - Z/S , Z/T and S/T . These may refer to continuous or discrete change, and may refer to a region interior (a

field) or an object (often a boundary). Not all combinations appear useful at present. Z/T applies to interiors, not boundaries, while Z/S and S/T apply to boundaries alone within the digital model of space.

Z/T(i,c), Z/S(i,c) and S/T(b,c) are taken to represent the continuity of space and time in the real world. In the previous descriptions of map types all "c" type representations will need to be converted to "d" for actual computer implementations.

Spatial boundaries used for the discretization of space may be explicit (as in digitized census districts) or implicit (as in grid cells or Voronoi cells). Attribute change across boundaries (Z/S) is best represented by referring to the attribute values of the left and right region interiors, using some topological structure. No discussion is given here about temporal (as opposed to spatial) boundaries, nor of topological events [12].

6. Discretization as a transformation

The process of partitioning space and time involves an inevitable loss of information. This may occur at the data collection stage, as in census data aggregation by district, or within the computer, as with gridding. This discretization may be treated as a series of transformations. A preliminary list is given below. Note that each moves from a continuous model towards a discrete representation, and that there may be more than one path between two states.

- Z/T(i,c) may be discretized over T, giving Z/T(i,d). Thus the value of Z within a region will jump between time steps.
- Z/T(i,c) may be discretized over Z, giving Z/T(i,d) again. Here the partition of Z into categories determines the time step used to display the change.
- Z/S(i,c) may be partitioned by S, giving Z/S(b,d). Thus a smooth surface in continuous space becomes a step function with changes at boundaries.
- Z/S(i,c) may be partitioned by Z, giving Z/S(b,c) as a first step. Here contour lines, having continuous normal gradients, partition the surface.

- $Z/S(b,c)$ may be further partitioned by Z , giving $Z/S(b,d)$. In this context, this second step consists of the partitioning of Z into categories, therefore giving a step function.
- $Z/S(b,c)$ may be partitioned by S , giving $Z/S(b,d)$. A typical example is the gridding of a contour map.
- $S/T(b,c)$ may be partitioned by T , giving $S/T(b,d)$. An example is robot movement or free-Lagrange flow modelling with an iteration for each time step.
- $S/T(b,c)$ may be partitioned by S , giving $S/T(b,d)$ again. Here the boundary is moved when it is to be drawn within the next screen pixel, for example.

6.1 The inverse process

We have been talking about the transformation from continuous to discrete representation of space and time as we move from the real world to the computer, based on both data collection and digital data representation issues. The inverse process may also occur: a discrete set of observations within the computer may be used to generate a continuous representation of the phenomenon. This is normally called interpolation. It may assume that the "fields" generated by each observation are non-overlapping (i.e. only one may occupy any spatial location at any one time), or overlapping. In the first case we have a surface precisely matching each generating object [7] whereas in the second case fields may penetrate each other, and the generating objects, giving surface approximations that will not usually correspond to the data point observations. The Voronoi "area-stealing" approach, as it embeds objects and proximity fields within the same spatial structure, is of the first type, whereas traditional "gravity model" interpolation is of the second type. It should be noted that this is not a precise inverse transformation from discrete to continuous representations. The real world has potential observations at *all* possible locations, whereas the interpolation process merely generates a representation at *any* desired location. (Thanks to Geoffrey Edwards for this observation.)

7. Simulation of process within geographic space and time

Once we have created a model of change over space and *time* we may represent the results of our simulation in a fashion that mimics, to a known degree, the behaviour of space in the real world. Simulation of a physical process then becomes the development of numerical models to describe the changes Z/S , Z/T and S/T as required, taking into account, the transformations induced by the discretization process. The mathematical functions describing the geographical processes are application specific, and largely dependent on the spatial partitioning process chosen. Examples have been given above for forest modelling, finite-difference groundwater flow, free-Lagrange flow methods and robot navigation, but this is only a small subset of the possibilities. In many of these cases the forces acting on entities within our spatial model are local, based on the set of neighbours, rather than global. The structures suggested should facilitate the modelling of local interactions - a form of "virtual reality" in two dimensions. This is a developing field, with much to be done.

7.1 The Voronoi implementation

The dynamic Voronoi approach, as outlined in various papers [4], [5], [6] was designed as a locally modifiable spatial structure, based on a well defined set of relationships with neighbouring objects. It is based on the better known static Voronoi data structure [1], [10] Being dynamic it can accommodate object displacement over time. This makes it relatively attractive as a structure for simulating geographic processes, as object displacement over space (S/T) is readily available. The area-stealing method of interpolation using the Voronoi method [7] allows the local discretization of space to be used to reconstruct change in attribute over space (Z/S). The change in attribute value over time (Z/T) has traditionally caused few problems. Thus the Voronoi model is one possible spatial structure for representing a wide range of spatial processes within a digital environment.

8. Acknowledgements

This research was made possible by the founding of an Industrial Research Chair in Geomatics at Laval University, jointly funded by the Natural Sciences and Engineering Research Council of Canada and the Association de l'industrie forestiere du Quebec.

References

- Aurenhammer, P., 1991. Voronoi diagrams - a survey of a fundamental geometric data structure. *ACM Computing Surveys*, v. 23, pp. 345-405.
- Fritts, M.J., W.P. Crowley and H.E. Trease, 1985. *The Free-Lagrange Method*. Lecture Notes in Physics 238. Springer-Verlag, New York.
- Gold, C.M., 1992. The meaning of "neighbour". *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*. Lecture Notes in Computer Science 639, Springer-Verlag, Berlin, pp. 220-235.
- Gold, C.M., 1993. An outline of an event-driven spatial data structure for managing time-varying maps. *Proceedings, Canadian Conference on GIS*, Ottawa, pp. 880-888.
- Gold, C.M., 1994. The interactive map. In: *Advanced Geographic Data Modelling: Spatial Data Modelling and Query Languages for 2D and 3D Applications* (M. Molenaar and S. de Hoop, eds.). Netherlands Geodetic Commission Publications in Geodesy, New Series, No. 40, pp. 121-128.
- Gold, C.M., and A.R. Condal, 1994. A spatial data structure integrating GIS and simulation in a marine environment. *Marine Geodesy* (in press).
- Gold, C.M. and T. Roos, 1994. Surface modelling with guaranteed consistency - an object-based approach. *IGIS'94: Geographic Information Systems*. Lecture Notes in Computer Science 884, Springer-Verlag, Berlin, pp. 70-87.
- Langran, G., 1992. *Time in Geographic Information Systems*. Taylor and Francis, New York.

Langran, G., 1993. Manipulation and analysis of temporal geographic information. Proceedings, Canadian Conference on GIS, Ottawa, pp. 869-879.

Okabe, A., B. Boots and K. Sugihara, 1992. Spatial Tessellations: Concepts and Applications of Voronoi Diagrams. John Wiley and Sons, Chichester.

Peuquet, D.J. and N. Duan, 1995. An event-based spatio-temporal data model (ESTDM) for temporal analysis of geographical data. International Journal of Geographical Information Systems, v. 9, pp. 7-24.

Roos, T., 1993. Voronoi diagrams over dynamic scenes. Discrete Applied Mathematics, v. 43, pp. 243-259.