UNCERTAINTY IN SPATIAL DATA: DEFINING, VISUALIZING, AND MANAGING DATA ERRORS

Charles R. Ehlschlaeger and Michael F. Goodchild
National Center for Geographic Information and Analysis (NCGIA)
3510 Phelps Hall
University of California
Santa Barbara, CA 93106

ABSTRACT

There is a considerable body of literature on techniques describing and modeling spatial database uncertainty. Unfortunately, spatial error modeling still isn't available to the general users of spatial databases. Spatial error modeling will only become viable when spatial errors are easily defined, visualized, and made available to spatial applications. Easing spatial error modeling into the user domain will be far easier if error definition, visualization, and management are based on the same fundamental principles. Accordingly, this paper will describe and will demonstrate a suite of public domain tools developed at NCGIA that allow for the definition, visualization, and management of spatial data error in a unified manner.

INTRODUCTION

Geographic information systems (GIS) are used to answer thousands of spatial questions everyday. Many spatial answers require extensive data collection, data analysis, and/or visualization tasks that are often divided among different workers, if not separate organizations. Even with simple applications, knowledge about the accuracy of data is often lost among the different people working on different aspects of spatial applications. How many GIS practitioners know how the errors in a USGS digital elevation model affect their applications? Furthermore, traditional spatial applications have well-defined data standards and error risks well-known by experienced GIS practitioners, but newly developed spatial applications and new data collection techniques will make uncertainty analysis extremely difficult. The only practical solution to these problems is to establish a clear useable error modeling methodology that will work for all spatial applications.

Maps are representations of reality (Monmonier 1975), and thus databases constructed from maps are also representations. The quality of a spatial database depends on how well it represents reality, which depends in turn on numerous factors in the processes used to create the database, such as measurement error, uncertainty in interpretation, and digitizing error. In this paper we focus on errors in one particular type of spatial database, specifically the digital elevation model, where data quality is determined by the differences between elevations at each point in the database, and corresponding elevations in reality.

Most of the existing GIS research on spatial errors describes the magnitude of the various types of error (Fisher 1992; Hunter & Goodchild, in press) while the spatial autocorrelative component of error has received less attention. The First Law of Geography states: "Everything is related to everything else, but near things are more related than distant things." (Tobler 1979) To put it more succinctly, spatial autocorrelation exists and is positive. Many map errors are spatially autocorrelated. For example, consider an elevation map made by traditional survey techniques. First, a surveyor chooses a set of available bench marks. Each bench mark has a minute amount of error. A surveyor then builds several transects connecting bench marks. Each transect station has errors from the bench marks, as well as individual surveying errors. From each station, the surveyor locates a set of point elevations that will be used to interpolate the elevation surface. Thus, the errors at each stage of the map generation process will propagate through later stages of map generation. Every error made has a spatial domain, i.e., each point elevation error will influence the region interpolated using that point. The transect station's error will propagate to all the point elevations. Finally, each bench mark error will propagate through each transect station to each elevation point. If the First Law did not apply to the real world, spatial error modeling would be a trivial task. GIS practitioners would be able to choose a set of random points in

the study area and stochastically compare the results of an analysis to the real world. They could see for themselves whether the application answers the questions asked. Unfortunately, spatial autocorrelation exists, and errors are spatially autocorrelated. Unless map construction metadata exists, issues such as product uncertainty and sensitivity, risk analysis and others are conceptually difficult.

Ongoing research at NCGIA is creating a suite of public domain software tools that explicitly define error in maps, propagate map errors through all spatial analyses, and visualize the implications of map error on any and all spatial analyses. This paper will discuss the implementation of error modeling software tools. First, this paper describes how map error descriptors, magnitude and spatial autocorrelation, can generate random fields that represent potential realizations of error. Second, various animation techniques will be discussed, comparing their advantages and limitations for specific problem solving goals. Finally, spatial data management issues affected by the tools described in this paper will be addressed.

SPATIALLY AUTOCORRELATED UNCERTAINTY

Random fields can be used to represent spatially autocorrelated error. In digital elevation models and similar types of spatial data, the error in elevation at a point can be regarded as a number added to or subtracted from the true elevation - in other words, the observed

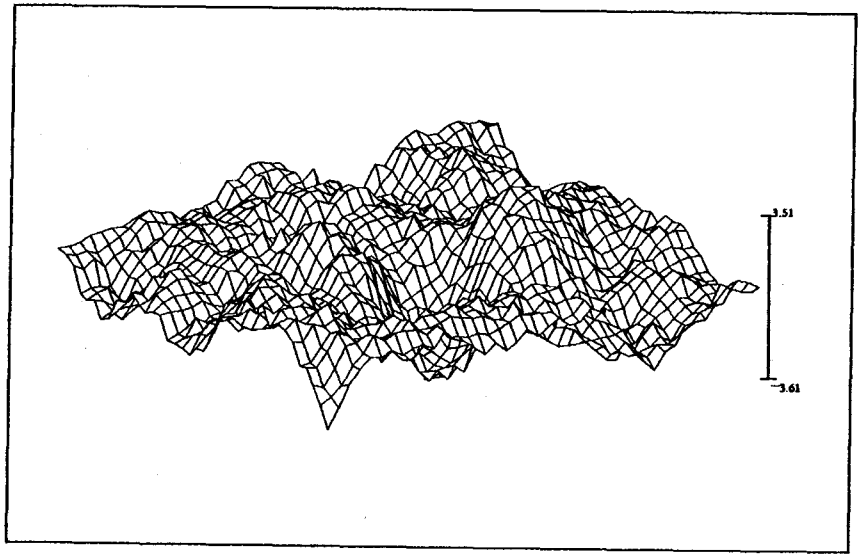

Figure 1. Random Field. The bar on the right represents the number of standard deviations from the mean.

elevation is equal to the true elevation with a positive or negative distortion. The entire set of errors in any one DEM form an error field, superimposed on the true elevation field. At any point the errors that affect elevation have a mean which is probably close to zero, and a range that is described in this paper by a variance, defined as the average squared deviation from the mean. In practice, the specific errors affecting a DEM are unknown, but information on their average magnitude is often available in the form of a variance, or a related statistic such as the root mean square error, and estimates can often be made of their degree of spatial autocorrelation. For example, a surveyed elevation map also contains the locations of bench marks, transect stations, and elevation points. Errors in the map building

process could be estimated and propagated to the resulting elevation map. In many cases, the best a cartographer can hope for is to get a qualitative understanding of a map's errors, and an assessment of the mean and variance of errors within that map by ground truthing. If an error field can be defined by the available quantitative descriptors of error, then its impacts can be examined by simulating one or more random error fields consistent with the available descriptors. Each simulated field is termed a realization of the error model. If a random field can be defined by the qualitative descriptors of the error, the random field applied to the map can generate a map realization. Spatially autocorrelated errors can be viewed as a random field. This random field will be a representation of a potential realization of the errors in a map (see Figure 1).

To facilitate making descriptions of errors into realizations of error fields, the following equation is used by *r.random.surface* to generate random fields:

$$R_i = \frac{\sum_j w_{ij}\epsilon_j}{\sqrt{\sum_j w_{ij}^2}}, \text{ if } d_{ij} < D: w_{ij} = \left(1 - \frac{d_{ij}}{D}\right)^E, \text{ otherwise } w_{ij} = 0. \quad (1)$$

where R_i is a point on the random field, w_{ij} is the spatial autocorrelative effect between points i and j, ϵ_j is a random variable with a mean of zero and variance of one, d_{ij} is the distance between i and j, D is the minimum distance of spatial independence, and E is the distance decay exponent. A random field with infinite bounds defined with a finite D will have a theoretical mean of zero and theoretical variance of one. Realizations of random fields, on the other hand, can have any potential mean or variance.

A more familiar form of random field is the spatially autoregressive field defined by the equation:

$$R_i = \sum_j \rho w_{ij} R_j + \epsilon_i \quad (2)$$

where ρ is a parameter (Haining 1983; Goodchild et al. 1992). But solving this equation for values of ρ close to its upper limit can be computationally intensive. The model used in this paper has two parameters rather than one, allowing the user more flexibility in applications, and also avoids the computational problems that affect the autoregressive model when the parameter approaches its upper limit.

Making realizations of errors from a map is a relatively straightforward task. A random field is created for each source of error. Since the theoretical variance of the random field is one, the random field is multiplied by the error's variance. The minimum distance of spatial independence is the distance at which errors are not related to each other. For example, if a map's values were interpolated from a set of independent measurements, the minimum distance of spatial independence would be the distance between adjacent measurements. The distance decay exponent represents how much the error at one point influences the random field (see Figure 2). For example, if a map's values were interpolated using an inverse distance squared weighting technique as in GRASS's *r.surf.idw* (Koerper 1991) (Isaaks and Srivastava 1989), the distance decay exponent would be two. The locations for the set of random variables are those places where error sources would influence the map. For example, a rectangular raster grid's error sources might be located at the grid points, while a node/arc vector map might only have error sources at the nodes.

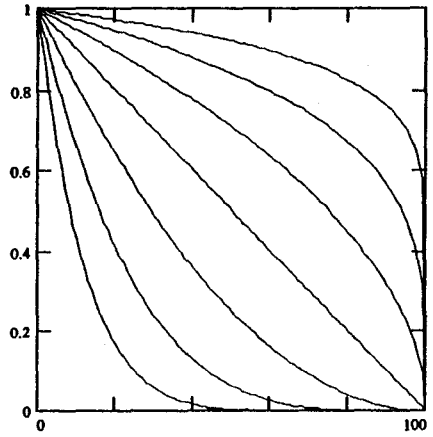

Figure 2. An error source's effect on the random field based on it's distance decay exponent (from .12 to 8 depending on line), and distance of spatial independence (100).

To generate a realization of a DEM, one would use the DEM as the mean value of reality and the random fields created by the DEM's error descriptors as the deviations from the mean. Adding the mean to all deviations will result in one DEM realization. See the *r.random.surface* man page (Ehlschlaeger 1994b) for the details of random field generation of field type maps. Categorical coverage maps can have realizations made with *r.random.model* (Ehlschlaeger 1994a). Work is underway at NCGIA to generate realizations for other types of spatial databases.

VISUALIZING UNCERTAINTY

Even if the explicit description of a map's error is definable, visualizing the map and its potential errors is not a trivial task. For the sake of discussion, consider the following problem: A map user wants to find all locations on a DEM below a specific elevation. Eastman et. al. (1993) described a sample problem predicting what parts of Boston would be flooded from a 1.9 meter increase in ocean level using a DEM for elevations. In GRASS, this would done with *r.mapcalc Flooded = 'if(DEM < OceanHeight)'* (Shapiro 1991). This command will return a map named *Flooded* with the categories zero (not flooded) and one (flooded). However, the uncertainty of the DEM should make the *r.mapcalc* output map vary between 0% (no chance of flooding) to 100% (guaranteed of flooding). While these results are easy to present using traditional GIS visualization software, the results only show the magnitude of the DEM uncertainty. Moreover, such maps can be highly misleading. While the percentage shown at any point is correctly interpreted as the probability that that point will be flooded, the probability that an area of finite size will be flooded is not equal to the probability mapped over the area. The probability that two or more points are flooded depends on the spatial autocorrelation of errors, so to assess the probability of an area being flooded it is necessary to know which DEM points lie in the area, and also to model the spatial structure of errors.

Showing the effects of map uncertainty and the influence of spatial autocorrelative

effects on applications is a more difficult task. Depending on the application, it can be difficult to see how a map will affect that application. It is difficult to imagine the resulting shape of a DEM by viewing the random field modifying it. It is even more difficult to imagine the potential shapes of a random field by knowing the specific error descriptors necessary to construct a representative random field. Unless the error is so minute as to be irrelevant, it is probably impossible to see how the error descriptors will modify the shapes and results of applications. In the case of the Boston flooding problem, the error's spatial autocorrelation effects will cause the potential shoreline to have specific potential shapes. This information might be critical to a spatial application where the spatial location of objects or values are important (e.g., buffer and neighborhood commands).

Considering the infinite variety of potential map error shapes and the lack of a systematic way of cataloging them, the only practical way to demonstrate map error is to show an ensemble of potential maps. This ensemble of map realizations will likely be formed with Monte Carlo sampling techniques due the potentially thousands of random variables needed to create one map realization.

Monte Carlo sampling has been proposed in error modeling as an analytical tool (Openshaw 1989) because it can find the range of an application's variability easily. Openshaw suggests that a limited number of application runs (as few as thirty) is enough to give a statistical understanding of the application. For visualization purposes, thirty potential realizations are more than enough to show the general shapes caused by the spatial autocorrelation of errors. Animations including only eight realizations highlighting potential DEM errors around Boston Harbor have been created at NCGIA. These animations can give the viewer a good understanding of the DEM error's spatial autocorrelative effects.

The map animations were carefully designed to maximize the viewer's understanding of the error's spatial autocorrelative effects. First, eight realizations showing a 1.9 meter rise in ocean level were made. Second, five interpolations were constructed between each pair of adjacent realizations. Playing the interpolations one after another produces a gradual fade between one realization (or original DEM), and the next realization. Third, an appropriate view of Boston Harbor was chosen using *SG3d*, GRASS's three dimensional graphics program for Silicon Graphics workstations. Fourth, a script was made in *SG3d* to display the realizations and interpolations in the 3-D view and save a visual copy of the results (also known as a screen dump). Fifth, these visual copies were sequenced into animations using *moviemaker*, a freely available animation program for Silicon Graphics workstations. Finally, these animations were played using *movieplayer*. While this process was implemented on a Silicon Graphics workstation, animation making routines are commonly available for most workstations and personal computers powerful enough to run GIS software.

Fading between realizations is important for the ease of spatial error understanding. The fades allow viewers to see specific shapes caused by specific spatial error descriptors form and disappear. Figure 3 shows the "timelines", or animation scripts, used to make several animations at NCGIA. The length of time chosen to pause at each realization and travel between realizations was determined by the complexity of the map being displayed (for pause length), and the complexity or magnitude of errors (for fade time). Simpler maps and smaller errors would have shorter times while more complex maps or errors would lengthen pauses and fades.

The upper timeline has the animation showing each realization for two seconds, with a two second fade to go from one realization to the next. The lower animation is of similar format except the animation will backtrack, re-showing a realization, before moving onto the next realization. The lower animation allows the viewer to focus on an error shape during the first fade between realizations. The next two fades will allow the viewer to analyze that error shape. The lower animation timeline is useful in situations where the error descriptors are complex and/or plentiful while the upper animation timeline will provide a more rapid viewing of many realizations.

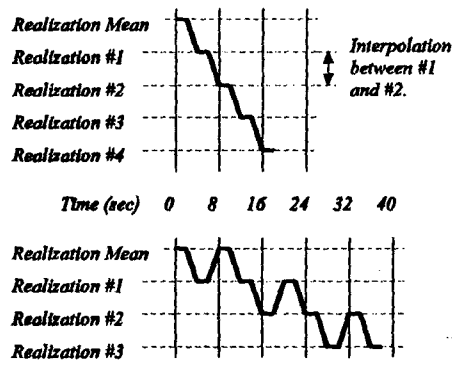

Figure 3. Animation timelines showing potential map realizations. Top timeline cycles through realizations, while bottom timeline backtracks to demonstrate individual error sources.

The temporal dimension of the animations described in this paper is very important. The length of time a viewer sees an effect in the animation is supposed to represent the probability that the condition will exist in the real world. For example, if an island in Boston Harbor is completely underwater 30% of the time in the animation, there is a 30% chance that the island will be under water after the water rises 1.9 meters.* The fades between realizations, not being realizations themselves, are not statistically sound and can cause an error effect to be displayed slightly longer or shorter than it should. However, information on the probability of flooding at each point is readily available from a static display of probabilities, as discussed earlier.

MANAGING UNCERTAINTY

These visualization techniques are conceptually compatible with Monte Carlo sampling as well as easy to understand. They allow the viewer to see how the errors in maps affect the quality of their application's results without *requiring* the viewer to explicitly understand the individual errors. By visualizing the application's results realizations and the data realizations, the viewer can make more informed management decisions about the likelihood that their application's results are representative of reality.

The main drawback to this visualization technique occurs when the time required to show a representative sampling of realizations exceeds the attention span of the viewer. This can be a real problem if maps contain complex error descriptors or if unlikely potential realities can significantly affect the analyses of an application. For example, suppose a significant event only has a 1% chance of occurring. If the Monte Carlo simulation only has 100 realizations, there is $.99^{100}$ or 36.6% chance that the significant event will not be in the sampling. If the viewer is seeing one realization every four seconds, he or she might not to wait for the nearly seven minutes to view the relevant realizations. There are work-a-rounds to these problems, but they require the GIS practitioner to understand the ramifications of the potential misuse of information presented from a statistical perspective.

---

*This assumes accurate data and a well-formed GIS application. The error descriptors used by the animations for this paper are for illustrative purposes only and are not measured and tested.

The worse thing that can happen is that the user can fall into a false sense of security due to the "pretty graphics" of animations technique. The authors hope that these visualization techniques are used to understand error, not hide it.

CONCLUDING REMARKS

This paper has discussed spatial data uncertainty issues, especially as they relate to DEMs. The most difficult part in defining spatial data uncertainty is determining the spatial autocorrelation and being able to represent this in a GIS. Random fields can be used to represent the various sources of spatial error, allowing realizations of data to be simulated. Applying the realizations of data to a GIS application creates an ensemble of application results. A set of ensembles for a GIS application gives the user a statistical perspective on the application's possibilities. The user can get a visual understanding of their application's uncertainties by displaying the ensembles in an animation where time dimension represents the likihood of occurance. Finally, the advantages and drawbacks of this Monte Carlo scheme were discussed.

ACKNOWLEDGEMENTS

The authors wish to thank the following people and organizations: The Institute for Computational Earth Systems Science at UCSB for the access to their Video Imaging Lab; Helena Mitasova, of the Construction Engineering Research Labs, for her assistance with *SG3d*, *moviemaker*, and *movieplayer*; and especially Steve Cullen for his assistance with the computer-to-video hardware and software.

REFERENCES

Eastman, R., Kyem, P., Toledano, J., & Jin W., 1993, GIS and decision making, Explorations in Geographic Information Systems Technology, Vol. 4, pp. 21-24, Clark University, Worcester MA.

Ehlschlaeger, C. 1994a, *r.random.model* man page, GRASS 4.2 User Manual, Office of GRASS Integration, Champaign, IL.

Ehlschlaeger, C. 1994b, *r.random.surface* man page, GRASS 4.2 User Manual, Office of GRASS Integration, Champaign, IL.

Fisher, P. 1992, First experiments in viewshed uncertainty: Simulating the fuzzy viewshed, Photogrammetric Engineering & Remote Sensing, Vol. 58, pp. 1345-52.

Goodchild, M., Guoquing, S. & Shiren Y. 1992, Development and test of an error model for categorical data, International Journal of Geographical Information Systems, Vol. 6, No. 2, pp. 87-104.

Haining, R., Griffith, D., & Bennett, R. 1983, Simulating Two-dimensional Autocorrelated Surfaces, Geographical Analysis, Ohio State University Press, Vol. 15, No. 3, pp. 247-255.

Hunter, G. & Goodchild, M. (in press), Dealing with uncertainty in spatial databases: a simple case study, Photogrammetric Engineering and Remote Sensing.

Isaaks, E. & Srivastava, R. 1989, Applied Geostatistics, Oxford University Press.

Koerper, G. 1991, *r.surf.idw* man page, GRASS 4.0 User Manual, Office of GRASS Integration, Champaign, IL.

Monmonier, 1975, Maps, Distortion, and Meaning, Association of American Geographers Resource Paper No. 75-4, Washington D.C.

Openshaw, S., 1989, Learning to live with errors in spatial databases, Accuracy of spatial databases, pp. 263-76, Taylor & Francis.

Shapiro, M. & Westervelt, J. 1991, r.mapcalc: An Algebra for GIS and Image Processing, U.S. Army Construction Engineering Research Laboratory, Champaign, IL.

Tobler, W. 1979, Cellular Geography, Philosophy in Geography, Dordrecht, pp. 379-86.

Veregin, H. 1989, Error Modeling for the Map Overlay Operation: Accuracy of Spatial Databases, Taylor & Francis, pp. 3-18.