

# Implementation Issues of Variogram Cloud Plots and Spatially Lagged Scatterplots in the Linked ArcView 2.1<sup>TM</sup> and XGobi Environment

Jürgen Symanzik<sup>1</sup>, Inna Megretskaia<sup>1</sup>, James J. Majure<sup>2</sup>, Dianne Cook<sup>1</sup>

<sup>1</sup> Department of Statistics, Iowa State University, Ames, IA 50011, USA

<sup>2</sup> GIS Support and Research Facility, Iowa State University

symanzik@iastate.edu

**Abstract.** In Symanzik et al. (1995) and Majure et al. (1995) we have presented a bidirectional link between a Geographic Information System (GIS), ArcView, and an interactive dynamic graphics program, XGobi. Since then, the link has been extended to support additional methods of spatial statistics such as variogram cloud plots and spatially lagged scatterplots.

While the first features of the link (handling of multivariate attribute data and spatial cumulative distribution functions) were straightforward to implement as far as data structures were involved, the new features require efficient handling and storage of large data sets. For example, the  $n$  observations in a data set can yield up to  $n^2$  points in the variogram cloud plot, depending on a selected cutoff distance  $d$ . However, quite often  $d$  is selected in such a way that only a small portion of all possible  $n^2$  points is considered. Also, the issue of linked brushing between  $n$  points in ArcView 2.1<sup>TM</sup> and (a portion of) the  $n^2$  points in XGobi will be discussed in this paper.

## 1 Introduction

In 1994 we started work on a bidirectional link between a Geographic Information System (GIS), ArcView (then version 2.0, now version 2.1), and an interactive dynamic graphics program in the X Window System<sup>TM</sup> environment, XGobi (Swayne et al., 1991). The first features of the link were the handling of multivariate attribute data and spatial cumulative distribution functions. The usefulness of the link, which allows us to display spatial locations and concomitant geographic variables within the GIS while visualizing and exploring the corresponding data space within XGobi simultaneously, has been highlighted for several different applications such as satellite imagery (Symanzik et al., 1995), forest health monitoring (Majure et al., 1995, Cook et al., 1996), and precipitation data (Majure et al., 1996).

A technical description of the implementation of the link via Remote Procedure Calls (RPCs) has been given in Symanzik et al. (1995) as well. The data structures for linking of these initial features were straightforward.

However, the additional spatial plots, i. e., variogram cloud plots and spatially lagged scatterplots, require efficient handling and storage of large data sets. For example, up to  $n^2$  points may appear in either of these plots, even though there are only  $n$  observations in the data set. A cutoff distance  $d$ , selected by the user, may considerably reduce the number of points that appear in either of these plots, but efficient linking is paramount.

Variogram cloud plots and spatially lagged scatterplots will be introduced in Section 2. In Section 3, we will deal with implementation issues of the link such as data structures, linked brushing, and formerly unsolved security hazards of ArcView 2.0<sup>TM</sup>. We conclude this article in Section 4 with an overview on the future directions of our work.

## 2 Variogram Cloud Plots and Spatially Lagged Scatterplots

The spatially lagged scatterplot and the variogram cloud plot are both graphics used to explore spatial dependence in a data set and to search for unusual spatial features. Both types of plots essentially carry the same information but present the information in different ways and emphasize different aspects of the data set.

The lagged scatterplot is widely proposed as an initial tool to investigate spatial data sets (e. g., Rossi et al., 1992; Cressie, 1993). It is generated by plotting the pairs of points  $\{(Z_k(S_i), Z_l(S_j)) \mid (S_i, S_j) \in N(h)\}$  where  $N(h) \equiv \{(S_i, S_j) \mid S_i - S_j = h\}$ .

In practice, the points in the lagged scatterplot are  $Z_k, Z_l$  pairs within a distance interval and/or angle class. These plots can be used to determine the degree of spatial continuity in a data set by looking at the plot for several distance intervals consecutively. If the dispersion of points about the line  $y = x$  increases with increasing distance, then spatial dependence is very likely the cause, since correlation between  $Z_k, Z_l$  decreases as spatial distance increases.

The link that has been developed between ArcView 2.1

<sup>TM</sup>ArcView 2.1 is a trademark of Environmental Systems Research Institute, Inc.

<sup>TM</sup>X Window System is a trademark of MIT.

<sup>TM</sup>ArcView 2.0 is a trademark of Environmental Systems Research Institute, Inc.

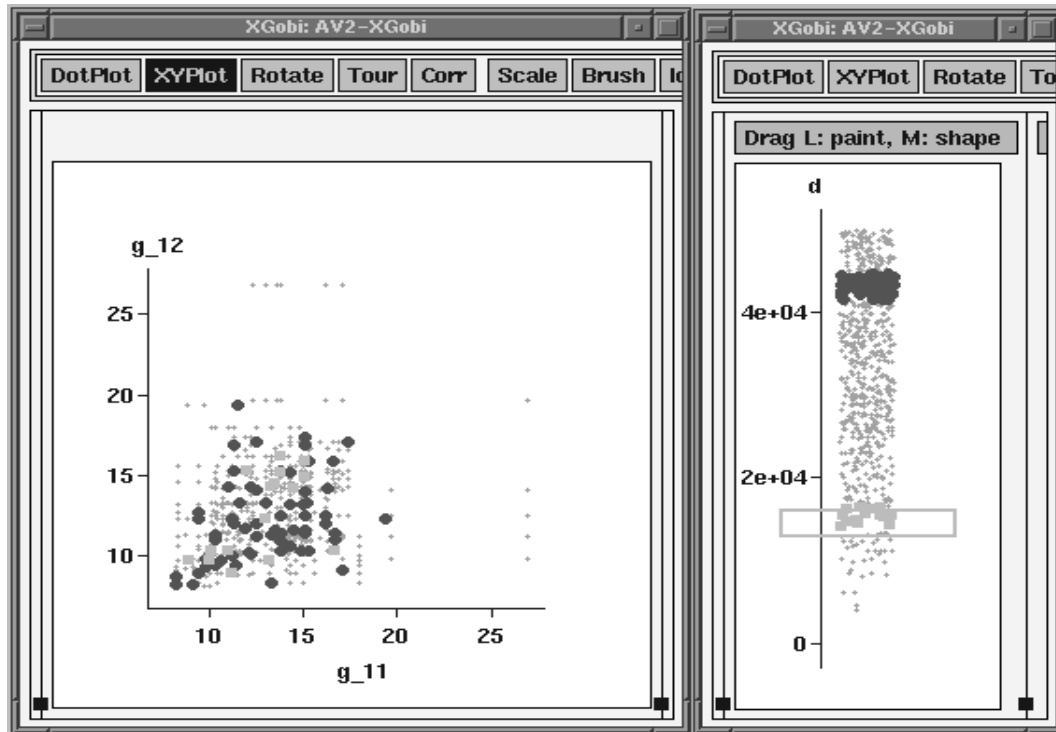


Figure 1: *Spatially Lagged Scatterplot (left), Dotplot of Spatial Distance (right).*

and XGobi provides a truly dynamic lagged scatterplot. With interactive brushing, the distances can be brushed from low to high. If the data set exhibits spatial dependence, the dispersion of points simultaneously increases. In addition, changing the brush size allows different sizes of distance interval to be studied. (See Cook et al., 1994, for further explanation.) The lagged scatterplot in Figure 1 was created using precipitation data from sites in the northeastern United States. All pairwise points are shown in small pluses and two distance intervals have been brushed. The dark filled circles (corresponding to a distance interval of approximately 42–48 km) appear to be more dispersed than the light filled squares (corresponding to a distance interval of approximately 13–18 km).

The variogram cloud plot (Bradley and Haslett, 1992) is also used to assess spatial structure in a data set. It is generated by plotting the square-root absolute difference  $|Z(S_i) - Z(S_j)|^{1/2}$  for each pair of locations in a given angle class versus the distance between the locations. The variogram cloud plot, then, provides a simultaneous plot for all distance classes. In a variogram cloud plot that exhibits spatial dependence, the differences will become more variable as the distance between pairs of locations increases.

In the ArcView–XGobi variogram cloud link, the angle class can easily be changed using the linked brush-

ing environment. In this way data sets can be explored for anisotropic variation. Figure 2 displays a variogram cloud plot, along with a plot of the sine vs. the cosine of the angle between locations, and a plot of the sampling locations. Two angle classes, corresponding to the NE–SW direction (dark filled circles) and the SE–NW direction (light filled squares), have been brushed.

In both the lagged scatterplot link and the variogram cloud link, the points displayed in XGobi are dynamically linked to both sampling locations with which they are associated. Brushing points in the XGobi window or sampling locations in the ArcView 2.1 window causes corresponding points to be brushed in the other application, keeping the analysis in its geographic frame of reference. In Figure 2, the lines displayed on the map correspond to the angle classes that have been brushed in XGobi.

### 3 Implementation Issues

Our link has been developed for DEC<sup>TM</sup> alphastations. It also has been tested on other platforms such as Sun<sup>TM</sup>/Sparc<sup>TM</sup> workstations and SGI<sup>TM</sup> workstations. A current version of the software can be downloaded

<sup>TM</sup>DEC is a trademark of Digital Equipment Corporation.

<sup>TM</sup>Sun is a trademark of Sun Microsystems, Inc.

<sup>TM</sup>Sparc is a trademark of Sun Microsystems, Inc.

<sup>TM</sup>SGI is a trademark of Silicon Graphics, Inc.

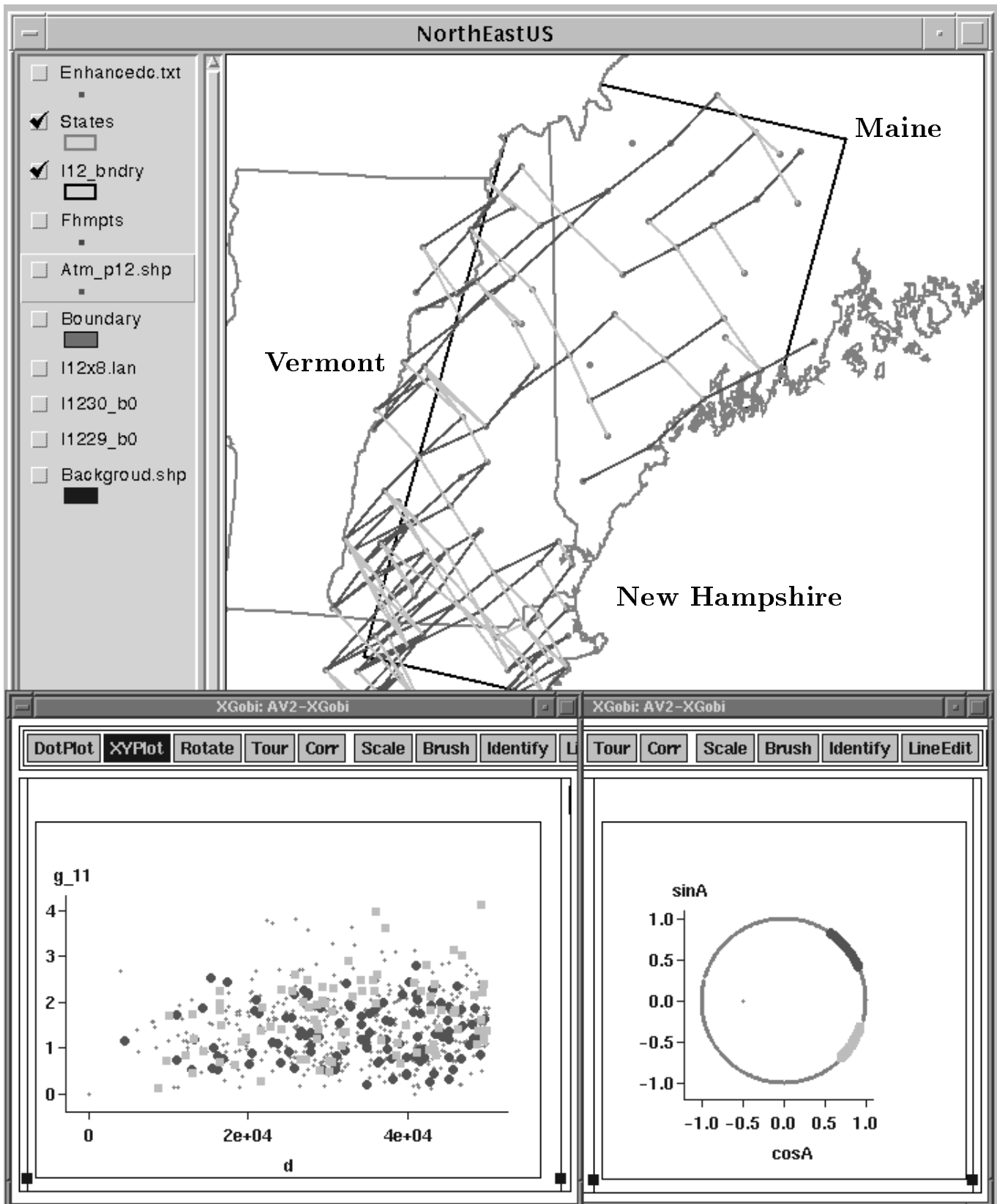


Figure 2: Variogram Cloud Plot (bottom left), Angle between Locations (bottom right), Map View of Sampling Sites in the Northeastern United States (top).

from the WWW at the URL <http://www.gis.iastate.edu/XGobi-AV2/XGobi-AV2.html>.

### 3.1 Data Structures

Data structures for the variogram cloud link and the spatially lagged scatterplot link are exactly the same, so only one of them will be described in this paper. A small example in Figure 3 is an illustration of data used for the variogram cloud plot.

Originally, we have an array of *Initial Data* passed from ArcView 2.1. In this example, the number of points,  $n$ , is 3.  $X$  and  $Y$  are the coordinates of each point (we will call this point  $S_i, i = 1, \dots, n$ ) in the 2-dimensional plane.  $Z_1$  and  $Z_2$  (in the general case  $Z_1, \dots, Z_k$ ) are the measures taken at point  $S_i = (X_i, Y_i)$ . They may be of the same variable taken at different times, or they may not be related at all. Parameter  $d$ , the cutoff distance, is also passed from ArcView 2.1. In this example,  $d$  is 1.1.

The *Array of Pairs* is the main data structure used to support this part of the link. It consists of records corresponding to each pair  $(S_i, S_j)$  of points satisfying the cutoff condition  $\sqrt{((X_i - X_j)^2 + (Y_i - Y_j)^2)} \leq d$ . Each record consists of the point numbers ( $i$  and  $j$ ), the angle between  $S_j - S_i$  and the horizontal axis measured in degrees ( $A$ ), the set  $\gamma_{lm}, l = 1, \dots, k, m = 1, \dots, k$ , where  $\gamma_{lm} = |(Z_l(S_i) - Z_m(S_j))|^{1/2}$ , and  $D = \sqrt{(\sum (Z_l(S_i) - Z_l(S_j))^2)}$ . -90 is assigned to  $A$  for identical points ( $i = j$ ), and -45 is assigned to  $A$  for observations from the same point ( $S_i = S_j, i \neq j$ ). The record also contains  $\sin(A)$  and  $\cos(A)$  which are useful for brushing in the arbitrary angle class, but they are not displayed in Figure 3.

All other arrays shown in Figure 3 are designed to create and support the link between points in XGobi and ArcView 2.1 and are required to do brushing in both directions. A *Binary Map* is created upon initialization of the link. It is a 2-dimensional array of size  $n \cdot \text{ceil}(n/32)$ . A one at the  $j$ th bit in the  $i$ th row indicates that the distance between  $S_i$  and  $S_j$  is smaller than  $d$  (e. g., it is 1 between  $S_1$  and  $S_3$ ). A zero indicates that the distance between these two points is bigger than  $d$  (e. g., it is  $\sqrt{2}$  between  $S_1$  and  $S_2$ ).

Information about 32 pairs of points is packed into one unsigned integer. Simultaneously with the creation of *Binary Map*, we count the number of points that lie within the cutoff distance for every initial point. After we finish looking at *Initial Data*, we know exactly how many pairs with point  $S_i$  are going to be in *Array of Pairs* for every  $i = 1, \dots, n$ . This allows us to allocate memory for *Array of Pairs* and *Brushing List 1 & 2*. *Brushing List 1* corresponds to all pairs for which  $S_i$

is the first point in the pair (e. g., 1 and 2 for  $S_1$ ), and *Brushing List 2* corresponds to all pairs for which  $S_i$  is the second point in the pair (e. g., 1 and 5 for  $S_1$ ). For many purposes one list is sufficient.

Now we describe the logical part overlaying these data structures. Using *Binary Map* we pick proper pairs of points, add information to *Brushing List 1 & 2*, and fill the record for this pair in the *Array of Pairs*. As for *Brushing List 1 & 2*, we know at each step exactly how many “boxes” are filled (there exists a special array of counters for those). So, we place the number of the record in the first free positions of *Brushing List 1 & 2* and increment the corresponding counters by one.

After we have gone through *Binary Map*, we have *Array of Pairs* and *Brushing List 1 & 2* completely filled. Now we can brush points in two directions. If we take a record from *Array of Pairs* we immediately know which points this entry consists of. If we pick a point in the array of *Initial Data*, say number  $i$ , the  $i$ th line of *Brushing List 1 (2)* contains all numbers of the pairs that contain this point  $i$  as first (second) component.

Finally, we want to comment on the efficiency of our data structures. Assume there are  $n = 128$  points in ArcView 2.1. This might yield up to  $128^2 = 16384$  points in XGobi. However, XGobi is suited best for small to medium size data sets. It might be desirable to have a cutoff distance  $d$  selected such that only 1024, i. e., 1/16th of the possible number of points, are displayed in XGobi. The additional memory required for *Binary Map* ( $128^2/32 = 512$  unsigned integers) can be neglected in comparison to the gain of memory for not having allocated 16384 records of *Array of Pairs*. Each line of *Brushing List 1 & 2* is reasonably small ( $1024/128 = 8$  in average) to allow efficient handling and brushing of points.

### 3.2 Linked Brushing

Linked brushing between ArcView 2.1 and the XGobi displaying variogram cloud plots and spatially lagged scatterplots works slightly different than the standard linked brushing. There, exactly one point in one application is related to one point in the other application. For each point brushed, the related point is brushed as well. Here, for every point in XGobi, there are two related points in ArcView 2.1. Actually, there might be up to  $n^2$  points in XGobi even if there are only  $n$  points in ArcView 2.1. We assume we brush only one point at a time in either application. Then, we make use of the following strategy for linked brushing:

**Brushing in XGobi:** Mark the point in XGobi with the selected color, glyph, and symbol. Determine in XGobi which are the two related points (using *Array*

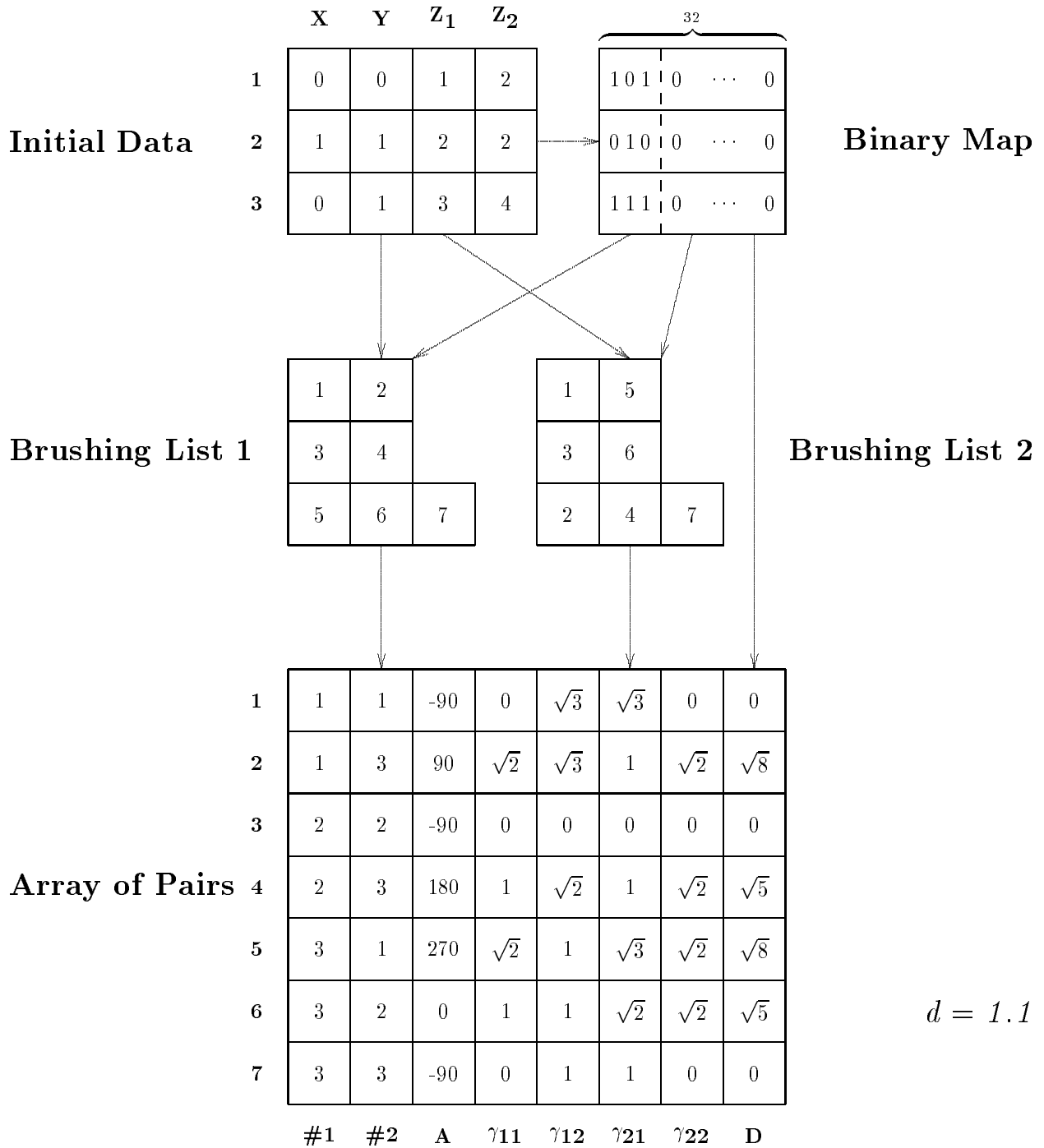


Figure 3: Data Structures.

of Pairs and pass this information to ArcView 2.1. Connect these points in ArcView 2.1 with a line of the selected color (provided by XGobi).

**Brushing in ArcView 2.1:** This can be described as a three phase process, where ArcView 2.1, XGobi, and again ArcView 2.1 proceed. First, brush the point in ArcView 2.1 (nothing visible happens) and

pass the information, which point has been brushed and which color, glyph, and symbol have been selected, to XGobi. Then, in XGobi, determine all points that are related to this point (there might be  $2n-1$  such points). For each of these points, assume it actually has been brushed in XGobi. Follow the instructions on brushing in XGobi, using the color, glyph, and symbol provided by ArcView 2.1. Finally, this will result in lines drawn in ArcView 2.1 between the selected point and all points that are not farther away than the cutoff distance  $d$ .

### 3.3 Security Issues

RPCs and their security issues have been described in the technical literature, e. g., Corbin (1991). For this discussion, it is enough to know that there exist different authentication levels that restrict access to the provided service. In Symanzik et al. (1995) we complained that ArcView 2.0 only provides null authentication for its RPC mechanism. Running the ArcView 2.0 RPC server causes a total lack of data security. ArcView 2.1 provides at least Unix authentication where additional fields such as a time stamp, the name of the local host, and the client's effective user and group IDs, are transmitted with every RPC request. In our link, we make use of this additional information by only processing those requests where local host and user ID match between ArcView 2.1 and XGobi (which is started from within ArcView 2.1 — thus IDs will match for valid requests).

Unfortunately, there is no verifier for Unix authentication, thus the previously mentioned credentials are still easy to fake. Therefore, we suggest that ESRI provides an RPC mechanism for DES (Data Encryption Standard) authentication in addition to the current null and Unix authentication for future releases of ArcView.

## 4 Future Directions

Currently, our work is completed as far as it involves the incorporation of methods of spatial statistics into the link. Nevertheless, since the generic work required to link ArcView 2.1 with XGobi can be adapted to any graphical method of spatial statistics which can be displayed within XGobi itself, it is easy to add such methods whenever desired.

Instead, we focus our work on modifications of XGobi, e. g., different types of smoothers (helpful for the variogram cloud plot) and a new type of linked brushing, called “hierarchical” linked brushing. This linked brushing is required for a future environment where ArcView 2.1 communicates with several XGobis, each of them displaying a different feature of the link, e. g., one XGobi and a cloned child for the attribute data, one XGobi for the spatial cumulative distribution functions, and one

XGobi for the variogram cloud plot. The current linked brushing in XGobi does not reasonably support such an environment. Extensions to our Avenue code within ArcView 2.1 are required as well to allow the communication with multiple XGobis through RPCs. It is also planned to extend the link towards other statistical packages such as S — PLUS<sup>®</sup>.

### Acknowledgements

Research related to this article was supported by an EPA EMAP grant under cooperative agreement # CR822919. The article has not been subjected to the review of the EPA and thus does not necessarily reflect the view of the agency and no official endorsement should be inferred. Symanzik's research was also partially supported by a German “DAAD-Doktorandenstipendium aus Mitteln des zweiten Hochschulsonderprogramms”.

### References

- Bradley, R. and Haslett, J. (1992). Interactive Graphics for the Exploratory Analysis of Spatial Data — The Interactive Variogram Cloud. In Dowd, P. A. and Royer, J. J., editors, *2nd CODATA Conference on Geomathematics and Geostatistics, Sci. de la Terre, Ser. Inf.*, volume 31, pages 373–386, Nancy.
- Cook, D., Cressie, N., Majure, J., and Symanzik, J. (1994). Some Dynamic Graphics for Spatial Data (with Multiple Attributes) in a GIS. In Dutter, R. and Grossmann, W., editors, *COMPSTAT 1994: Proceedings in Computational Statistics*, pages 105–119, Heidelberg. Physica-Verlag.
- Cook, D., Majure, J. J., Symanzik, J., and Cressie, N. (1996). Dynamic Graphics in a GIS: Exploring and Analyzing Multivariate Spatial Data Using Linked Software. *Computational Statistics, Special Issue: Computer Aided Analyses of Spatial Data*, to appear.
- Corbin, J. R. (1991). *The Art of Distributed Applications: Programming Techniques for Remote Procedure Calls*. Springer, New York, Berlin, Heidelberg.
- Cressie, N. A. C. (1993). *Statistics for Spatial Data (revised edition)*. Wiley, New York, NY.
- Majure, J. J., Cook, D., Cressie, N., Kaiser, M., Lahiri, S., and Symanzik, J. (1995). Spatial CDF Estimation and Visualization with Applications to Forest Health Monitoring. *Computing Science and Statistics*, 27: to appear.
- Majure, J. J., Cressie, N., Cook, D., and Symanzik, J. (1996). GIS, Spatial Statistical Graphics, and Forest Health. In *Third International Conference/Workshop on Integrating GIS and Environmental Modeling, Santa Fe, New Mexico, USA, January 21-25, 1996*. to appear.
- Rossi, R. E., Mulla, D. J., Journel, A. G., and Franz, E. H. (1992). Geostatistical Tools for Modeling and Interpreting Ecological Spatial Dependence. *Ecological Monographs*, 62:277–314.
- Swayne, D. F., Cook, D., and Buja, A. (1991). XGobi: Interactive Dynamic Graphics in the X Window System with a Link to S. In *ASA Proceedings of the Section on Statistical Graphics*, pages 1–8, Alexandria, VA. American Statistical Association.
- Symanzik, J., Majure, J. J., and Cook, D. (1995). Dynamic Graphics in a GIS: A Bidirectional Link between ArcView 2.0 and XGobi. *Computing Science and Statistics*, 27: to appear.

---

<sup>®</sup>S-PLUS is a registered trademark of StatSci, a division of MathSoft, Inc.