**Software for general similarity network analysis**

**Background**

The ideas behind the phrase “general similarity networks” are described in Östborn and Gerding (2014).

In this note, we discuss why we judged it necessary to write our own software in order to analyse the diffusion of Hellenistic fired bricks using general similarity networks. We also outline the ideas and structure of software that we created. It is written in Matlab, and it may be downloaded at <http://projekt.ht.lu.se/lateres/network-analysis/>.

Unfortunately, the software is custom-made for our database of fired bricks. However, the ideas and structure might inspire a general-purpose software that allows the analysis of any archaeological database.

The results of our network analysis of the diffusion of Hellenistic fired bricks are presented in Östborn and Gerding (2015). This note may be seen as supplementary material to that study.

**The need for new software**

To perform the network analysis we wanted to do, we needed a single software in which it is possible to 1) create general similarity networks with shorthand commands, 2) display them visually on a spatial map, 3) get basic network quantities in return automatically so that the structure of different networks can be quickly compared, 4) explore networks visually, highlighting components, central nodes and shortest paths, and 5) easily obtain database information about each node. In addition, we needed to work in an environment where it is possible to perform further mathematical and statistical network analysis at will.

We did not find software that fulfils all these needs. Therefore we developed our own network program in Matlab®. This environment allows numerical analysis, general programming, and the construction of graphical interfaces. The latter is needed for the effortless exploratory investigation of a variety of networks, stated as an important mode of investigation in Östborn and Gerding (2014).

Let us discuss the shortcomings of existing alternatives. In our database the geographical location of each context is one its attributes. This form of database is the same as that used in GIS programs (Eiteljorg II 2008). Such programs make it possible to select groups of contexts based on their attributes, to display their geographic location visually, and to easily obtain database information about each displayed context. However, the database management commands available in GIS does not allow the construction of general similarity networks. Another problem is that even if there are network analysis toolboxes available for GIS, these are aimed at the exploration of spatial and topological relations between objects, rather than the relation between the other kinds of attributes of these objects. Network analysis in GIS focuses on things such as route optimisation, travel times and travel costs, rather than structural network analysis and the identification of processes taking place in the networks (Batty 2005).

While GIS offers too limited opportunity to construct and analyse networks, it enables much more spatial analysis than is actually needed in similarity network analysis. The contexts should be accurately located on a map, and the edge lengths should be possible to calculate. That is all. Other kinds of geographical information that the archaeologist judge to be important for the interpretation (for instance whether a settlement is located in a valley or on a hill) should be expressed as context attributes. It should be part of the similarity network analysis, not a spatial analysis.

At the other end of the spectrum, social network analysis programs such as Pajek (De Nooy et al. 2005) offer extensive network analysis, but no geographical representation, since social networks are not spatial. Further, they are not constructed in a way that makes it easy to integrate archaeological databases and construct general similarity networks within the programs.

**The software**

The similarity network program we developed is custom made for our brick database. It may nevertheless be worthwhile to describe its main features, since they may provide inspiration for a more general program that is possible to use together with any archaeological database of a prescribed form.



**Figure 1** The user interface of the network analysis program written in Matlab®. The second largest network component is highlighted in blue. Two contexts in Rhegion (red) are the hubs in this blue component, having the highest degree.

The main function of the program is to create general similarity networks, according to the similarity criteria defined for each attribute in Table 1. To do so, we developed shorthand notation so that the similarity condition that specifies the network can be easily entered in the upper row of the window *Network commands* (Fig. 1).

|  |  |  |
| --- | --- | --- |
| *Attribute* | *Attribute type* | *Possible values* |
| Reliable | Categories | Yes/No |
| Site | Categories | One of 131 ancient site names |
| Location | Numerical vector | Any pair of real numbers that encodes latitude and longitude |
| Dating | Interval | Any interval of years contained in the interval 600–1 BCE |
| Phase | Numerical value | 600, 575, 550, …, 50, 25, 0 (BCE) |
| Structural use | Categories | Casing/Masonry/Pavement |
| Context | Categories | Domestic/Manufacture/Military/Public/Sacred/ Sepulchral |
| Function | Categories | Arch/Basin/Barrel vault/Bench/Bonding course/ Cist/Column/Courtyard/Cover/Door frame/ Floor/Foundation/Half-column/Hypocaust/Inner wall/Niche cover/Pillar/Pillar base/Quoin/Street/ Terrace wall/Underground wall/Upper wall/ Wall/Wall socle/Water channel/Furnace wall |
| Structure | Categories | Brick-faced concrete/Imbrices and mud/Interlaced brickwork/ One-stone brickwork/Mixed materials/ Solid brickwork |
| Binding | Categories | Clay mortar/Mortar/No mortar/Timber bindings |
| Shape | Categories | Circle sectors/Circular/Curved/Grooved/ Imbrex/Pierced circular/Rectangular/ Semicircular/Square/Tegula/Triangular/Voussoir |
| Size category | Categories | Lydion/ Pentadoron/Sesquipedalis/Tetradoron |
| Thickness | Interval | An interval given with half-centimetre precision |
| Plaster | Incidence | Yes |
| Stamp | Incidence | Yes |
| Poorly fired | Incidence | Yes/No |
| Tile bricks | Incidence | Yes/No |
| Combined with mud bricks | Incidence | Yes/No |

**Table 1** The attributes in the database, their type, and their possible values. Out of eighteen attributes, thirteen describe the properties of the brick constructions *per se*, whereas two describe its temporal location and two describe its spatial location.

The database is available at <http://projekt.ht.lu.se/lateres/>.

In the software, each attribute is denoted by a capital letter, and each value is encoded as one or several numbers. Numbers that represent incidences and categories are chosen to be integers. Table 2 describes how similarity commands addressing the values of a single attribute are expressed.

|  |  |  |
| --- | --- | --- |
| *Command* | *Applies to attributes of type* | *Effect: connect all pairs of contexts for which*  |
| *A* | Numerical value, Numerical vector, Abundance, Incidence, Hierarchical categories, Categories | the value of *A* is the same |
| *A* | Interval | the intervals that define their values of *A* overlap |
| *A* = *X* | Numerical value, Abundance, Incidence, Hierarchical categories, Categories | the value of *A* is *X* |
| *dA* <= *X* | Numerical value, Abundance, Hierarchical categories | the difference between their values of *A* is equal to or less than *X* |
| *dA <= X* | Numerical vector | the vectorial distance (defined according to taste) between their values of *A* is equal to or less than *X* |
| *A* = (*X*, *Y*) | Numerical value, Abundance, Hierarchical categories | both has a value of *A* that is contained in the interval (*X*, *Y*) |
| *A* = (*X*, *Y*) | Numerical vector | the intervals that define their values of *A* both overlaps the interval (*X*, *Y*) |
| *A = (X, Y, Z, …)* | Categories | the value of *A* in both is contained in the list |
| *A* =[(*X*1, *X*2), (*Y*1, *Y*2)] | Numerical vector | the vectorial value (*X*, *Y*) of both is contained in the square with corners (*X1*, *Y1*), (*X1*, *Y2*), (*X2*, *Y2*) and (*X2*, *Y1*)  |
| *A* : *n* | Numerical value, Numerical vector, Abundance, Hierarchical categories, | the similarity rank with respect to *A* of context 2 as seen from context 1, or vice versa, is equal to or less than *n*. |

**Table 2** Network specification commands applying to a single attribute *A*. If information about the value of *A* is lacking in one or both contexts, then they are not connected by an edge. Lack of information is not a similarity.

In this study, we do not use commands of the last two types. The next to last type may, for example, correspond to a requirement that only contexts located within a given geographical square are included in the network. The last type may correspond to proximal point analysis.

To create more general conditions for connection, we have to combine the above commands and add some new ones (Table 3).

|  |  |
| --- | --- |
| *Command* | *Effect: connect all pairs of contexts for which* |
| *CO* = *X* | the values of at least *X* attributes are the same (applies to these attribute types: Numerical value, Numerical vector, Abundance, Hierarchical categories, Categories) or overlap (applies to this attribute type: Interval). |
| *CO* = *X/AB* | the above holds true even if we exclude attributes *A* and *B* |
| – C153 | neither is the context with identity number 153; it is excluded from the network |
| & | the commands at both sides of the symbol are satisfied |
| % | at least one of the commands at each side of the symbol is satisfied |

**Table 3** Network specification commands that combine the ones given in Table 2 and add new options. The spatial attributes “Site” and “Location” are not included in the counting of common attribute values in the command *CO* = *X*. The command is used to help infer the spatial nature of a process; therefore we should not by assumption make it easier for close contexts to be connected.

When the logical operators & and % are used, parentheses mark priority of operations, just as in arithmetic. For example, the condition

*“Connect all pairs of brick contexts that are no more than 300 km apart, whose intervals of dating overlap, and in which the function or the shape of the bricks is the same.”*

translates to

(*dD* <= 300) & *N* & (*E* % *H*),

where *D* = location, *N* = dating, *E* = function, and *H* = shape.

A network specification we commonly use is *N* & (*CO* = *X*), which means that two contexts are connected whenever their dating overlap and they have at least *X* attribute values in common. This command sets a general level of similarity and combines it with the possibility of a temporal match, which is necessary if a connection is to qualify as a causal link. We may want to include only reasonably well dated contexts that are assigned a phase O, and argue that a temporal match means that the phase of two contexts is the same, or belong to a neighbour 25 year bin. Then we may construct the network N & (dO <= 25) & (CO = X). Note that if we require that the phase is the same, then the network fragments into components corresponding to time slices that are 25 years wide.

In the lower text field in the window *Network commands* (Fig. 1), it is possible to enter the commands with which to analyse already existing networks. The command *Component X* highlights the *X*:th largest component (like the blue component in Fig. 12). *Degree* >= *X* highlights all nodes with degree equal to or larger than *X*. The command *Diff* highlights the difference between the present network and a previous one, so that the effects of a small change in the network specification condition can be examined.

As soon as a network component is highlighted with the command *Component X*, its characteristics can be studied. *B\_Centres* and *C\_Centres* highlight the nodes with highest betweenness and closeness centrality, respectively. *Hubs* highlight the nodes with the highest degree. *Paths*(*X*, *Y*) shows the shortest paths between the nodes that correspond to contexts with identity numbers *X* and *Y*. (There may be several paths that have the same shortest length.)

Apart from the visual display according to Fig.1, network information is also shown in the Matlab command window. The identity numbers of all contexts that are part of the network are listed, as well as the identities of all nodes that are part of a highlighted component. The commands *B\_Centres*, *C\_Centres* and *Hubs* produce lists of the values of the betweenness centrality, closeness centrality and degree of all contexts in the component.

This identity information can be used to quickly access a description of the context by typing its identity number in the window *Context information* (Fig. 1). It is also possible to click on a site on the network map, and get descriptions in the context information window about the contexts located there. Matlab offers the possibility to zoom in on a network to separate close sites and explore the network relations between them.

The window *Network properties* (Fig. 1) shows statistical and structural network quantities. The *component size distribution* makes it possible to choose components with different sizes to highlight on the map. The *Cumulative edge length realisation ratio* is discussed in Östborn and Gerding (2015) in relation to the statistical analysis. Whenever the data points fall clearly below the smooth blue curve, similar contexts are located closer to each other in the mean than dictated by pure chance. It is a sign that the distribution of the contexts is the result of a causal diffusion process.

A complete list of codes for attributes and attribute values, and of commands that make use of these codes, appears if the user presses the button *Help* in the *Network Commands* window (Fig. 1).

If you type the ID of a context (as defined in the database) in the appropriate box in the *Context information* window (Fig. 1), then the available information about that context appears in plain text.

If you have created a network, you may click on the button *View name & ID* in the *Network Commands* window. Then a hair cross appears on the map, which you may position on top of any site that appears in the network. If you click on that site, the name of the site appears on the map. Also, the ID:s of contexts in that site are listed at the top of the *Context information* window. If you compare that list with the list in the Matlab command window of all contexts that belong to the network, you may decide which contexts in that site that are actually part of the network you have created. Then you may get information about those contexts in plain text as described in the previous paragraph.

**References**

M. Batty (2005). Network Geography: Relations, Interactions, Scaling and Spatial Processes in GIS. In D. Unwin and P. Fisher (eds.), Re-presenting GIS (Chichester).

# W. De Nooy, A. Mrvar, V. Batagelj (2005). Exploratory Social Network Analysis with Pajek, (Cambridge).

H. Eiteljorg II and W. F. Limp (2008). Archaeological computing (Bryn Mawr).

P. Östborn and H. Gerding (2014). Network analysis of archaeological data: a systematic approach, Journal of Archaeological Science 46 (June), 75-88. [dx.doi.org/10.1016/j.jas.2014.03.015](http://dx.doi.org/10.1016/j.jas.2014.03.015)

P. Östborn and H. Gerding (2015). The Diffusion of Fired Bricks in Hellenistic Europe: A Similarity Network Analysis, Journal of Archaeological Method and Theory 22(1). forthcoming