Software for Social Network Analysis*

Mark Huisman
Heymans Institute/DPMG

Marijtje A.J. van Duijn
ICS/Statistics & Measurement Theory

University of Groningen

3rd October 2003

Abstract

This chapter gives a state-of-the art overview of available (free and commercial) software for social network analysis as of fall 2003. It reviews and compares six programs, illustrating their functionality with example data. Data manipulation options and available support are also discussed. Furthermore, seventeen other, of which nine special-purpose, software packages and five software routine packages for general statistical software are reviewed briefly. The chapter concludes with some recommendations.

1 Introduction

This chapter reviews software for the analysis of social networks. Both commercial and freely available packages are considered. Based on the software page on the INSNA website (see the references for the URL), and using the main topics in the book on network analysis by Wasserman and Faust (1994), which we regard as the standard text, we selected twenty seven software packages: twenty three stand alone programs, listed in Table 1, and five utility toolkits given in Table 2.

Software merely aimed at visualization of networks was not admitted to the list, since this is the topic of chapter 12 of this book (Freeman, 2004). We do review a few

*We thank Vladimir Batagelj, Ghi-Hoon Ghim, Andrej Mrvar, Bill Richards, and Andrew Seary for making their software available, Wouter de Nooy for his help with the documentation, and the editors Peter Carrington, John Scott, and Stanley Wasserman, as well as Vladimir Batagelj, Steve Borgatti, Ron Burt, Ghi-Hoon Ghim, Andrej Mrvar, Andrew Seary, and Tom Snijders for their valuable comments and suggestions. This research was supported by the Social Science Research Council of the Netherlands Organization for Scientific Research (NWO), grant 400-20-020.

programs with strong visualization properties. Some were originally developed for network visualization, and now contain analysis procedures (like NetDraw, Borgatti, 2002). Other programs were specifically developed to integrate network analysis and visualization (like NetMiner, Cyram, 2003, and visone, Brandes and Wagner, 2003). Two other programs for network visualization are worth mentioning here, because some of the reviewed software packages have export functions to these graph drawing programs, or they are freely distributed together with the social analysis software: KrackPlot (Krackhardt, Blythe, and McGrath, 1994) and Mage (Richardson, 2001).

The age of the software was not a criterion for selection, although the release dates of the last versions of the majority of the reviewed software were within the last two or three years.

Tables 1 and 2 describe the main objective or characteristic of each program. The data format distinguishes three aspects: 1) type of data the program can handle, 2) input format, and 3) whether there is an option to indicate missing value codes for network relations. Next, the functionality is described. For each program we indicate whether the software contains (network) visualization options, for a toolkit its environment (software package or operating system other than Windows), and for both groups of software the kind of analyses it can perform. We use the network terminology and categorization of Wasserman & Faust (1994, Parts 3-6) for the different types of analysis: structural and locational properties, roles and positions, dyadic and triadic methods, and statistical dyadic interaction models. The theoretical background of almost all of the obtainable output can be found there as well. Where necessary, additional references are given. The amount of support is the final characteristic mentioned in the table, distinguishing availability of the program (free or commercial, not listing prices), presence and availability of a manual, and presence of online help during execution of the program.

Section 2 gives an extensive review of six programs (indicated by an asterisk in Table 1). These programs are either regarded as general and well-known (UCINET, Pajek, NetMiner) or as having specific features worth mentioning and illustrating (MultiNet, STRUCTURE, StOCNET). We examine the properties of these packages with respect to data entry and manipulation, visualization, and social network analysis. The software is illustrated by applying a selection of routines to an example data set. A complete reference to a program is given only once, either at the start of the section in which it is reviewed or—for non-reviewed software—at the first mention.

Table 1: Overview of selected programs for social network analysis, with the number of the version that was reviewed, their objectives, data format (type, input format, missing values), functionality (visualization techniques, analysis methods), and support (availability of the program, manual, and online help).

				Data			Functionality		Support		
	Program	Ver.	Objective	Type[1]	Input[2]	Miss.	Visual.	Analyses[3]	Avail.[4]	Manual	Help
	Agna	2.0.7	general	c	m	no	yes	d, sl, sequential	free	yes	yes
	Blanche	4.6.4	network dynamics	c	m	no	yes	simulation	free	yes	yes
	FATCAT	4.2[5]	contextual analysis	c	ln	yes	no	d, s	free[5]	no	yes
	GRADAP	2.0[5]	graph analysis	c	ln	yes	no	d, sl, dt	com[5]	yes	no
	Iknow	–	knowledge networks	e	n	–	yes	d, sl	free	yes	yes
	InFlow	3.0	network mapping	c, e	ln	no	yes	d, sl, rp	com	yes	yes
	KliqFinder	0.05	cohesive subgroups	c	m, ln	no	yes	sl, s	–	yes	no
*	MultiNet	4.24	contextual analysis	c, l	ln	yes	yes[8]	d, rp, s	free	no[11]	yes
	NEGOPY	4.30[5]	cohesive subgroups	c	ln	yes	yes	d, sl, rp	com[5]	yes	yes
	NetDraw	1.0	visualization	c, e, a	m, ln	yes	yes	d, sl	free	yes	no
*	NetMiner II	2.3.0	visual analysis	c, e, a	m, ln	no	yes	d, sl, rp, dt, s	com[9,10]	yes	yes
	NetVis	2.0	visual exploration[6]	c, e, a	m, ln	no	yes	d, sl	free[6,9]	no	yes
*	Pajek	0.94	large data visualization	c, a, l	m, ln	yes[7]	yes	d, sl, rp, dt	free	no	no
	PermNet	0.94	permutation tests	c	m	yes	no	dt,s	free	no	yes
	PGRAPH	2.7	kinship networks	c	ln	–	no	d, rp	free	no[12]	yes
	ReferralWeb	2.0	referral chains	e	ln	–	yes	d	–[9]	yes	yes
	SM LinkAlyzer	2.1	hidden populations	e	ln	–	yes	d	com[10]	yes	yes
	SNAFU	2.0	general for MacOS[6]	c	m, ln	no	yes	d, sl	free	no	no
	Snowball	–[5]	hidden populations	e	ln	–	no	s	free[5]	yes	no
*	StOCNET	1.4	statistical analysis	c	m	yes	no	d, dt, s	free	yes	yes
*	STRUCTURE	4.2[5]	structural analysis	c, a	m	yes[7]	no	sl, rp	free[5]	yes	no
*	UCINET	6.05	comprehensive	c, e, a	m, ln	yes	yes[8]	d, sl, rp, dt, s	com[10]	yes	yes
	visone	1.0b1	visual exploration	c, e	m, ln	no	yes	d, sl	free	no	no

[1] c=complete, e=ego-centered, a=affiliation, l=large networks.
[2] m=matrix, ln=link/node, n=node.
[3] d=descriptive, sl=structure and location, rp=roles and positions, dt=dyadic and triadic methods, s=statistical.
[4] com=commercial product, free=freeware/shareware.
[5] DOS-program which is no longer updated.
[6] Open source software.
[7] Only missing value codes for attributes.
[8] No graph drawing routines.
[9] Freely accessible on the internet (some with reduced functionality).
[10] An evaluation/demonstration version is available.
[11] The manual of some modules is available.
[12] The manual is available after registration.

Table 2: Overview of selected software toolkits for social network analysis, the number of the version that was reviewed, their objectives, data format (type, input format, missing values), functionality (hardware/software environment, analysis methods), and support (availability of the program, manual, and online help).

			Data			Functionality		Support		
Program	Ver.	Objective	Type[1]	Input[2]	Miss.	Envir.	Analyses[3]	Avail.[4]	Manual	Help
JUNG	1.0	modeling graphs	c	ln	–	Java	d, sl, vis	free	yes	–
MatMan	1.0	structural analysis	c, a	m	no	Excel	d, sl, ethological	com	yes	yes
SNA	0.41	general	c	m	no	R/S	d, sl, rp, dt, s, vis	free	yes	–
SNAP	2.5	general	c	m	no	Gauss	d, sl, rp, dt, s	com	yes	–
yFiles	2.1	visual exploration	c	ln	–	Java	d, sl, vis	com	yes	–

[1] c=complete, e=ego-centered, a=affiliation, l=large networks.

[2] m=matrix, ln=link/node, n=node.

[3] d=descriptive, sl=structure and location, rp=roles and positions, dt=dyadic and triadic methods, s=statistical, vis=visualization.

[4] com=commercial product, free=freeware/shareware.

We consider the remaining software to be more specialized and discuss their objectives and properties to a limited extent in Section 3[1]. In this section we also review some routines that were developed to perform social network analysis in general software or on operating systems other than Windows.

The chapter concludes with a section comparing the routines and support offered by the various programs discussed in Section 2, and some general recommendations. This section is by no means final, because by definition a chapter like this becomes outdated with publication.

2 Social network software - a closer look

In this section the programs UCINET, Pajek, NetMiner II, STRUCTURE, MultiNet, and StOCNET are investigated in more detail with the help of an example data set. The order in which the packages are presented is based on age, as well as on generality. We start with three general packages, covering a wide range of analysis methods. They are presented according to age: UCINET, Pajek, NetMiner II. Next, the program STRUCTURE is presented. We consider STRUCTURE as a general program featuring a limited number of methods. Although it has become somewhat outdated, STRUCTURE has some unique features worth presenting. Finally, two more specialized packages are presented: MultiNet and StOCNET.

In the presentation we focus on five groups of procedures the software does or does not possess.

1. Data entry and data manipulation.
2. Visualization techniques.
3. Social network analysis routines, divided into three types of methods:
   (a) descriptive methods to calculate (simple) network statistics (e.g., centrality or transitivity),
   (b) procedure-based analysis based on more complex (iterative) algorithms (e.g., cluster analysis or eigendecompositions), and
   (c) statistical modeling based on probability distributions (e.g., exponential random graph models or QAP correlation).

The choice of social network analysis routines that were inspected is based on the categorization of methods given by Wasserman and Faust (1994) explained in

[1]Except FATCAT

the introduction, and on the analysis methods presented in earlier chapters in this book.

- Structure and location: centrality (Everett and Borgatti, 2004) and cohesive subgroups (cliques).
- Roles and positions: structural equivalence, blockmodeling (Doreian, Batagelj, and Ferligoj, 2004), eigendecompositions.
- Dyadic and triadic methods.
- Statistical methods: exponential random graph models (Wasserman and Robins, 2004), QAP correlation, statistical analysis of network evolution (Snijders, 2004).

2.1 Example data

The example data used are Freeman's EIES network (Freeman and Freeman, 1979), three one-mode networks with two relations on a set of actors ($n = 32$) that is frequently used by social network researchers. The data come from a computer conference among social network researchers and were collected as part of a study of the impact of the Electronic Information Exchange System (EIES). Two relations were recorded: the number of messages sent and acquaintanceship. The acquaintanceship relation is longitudinal, measured at two time points, ranging from 0 (did not know the other) to 4 (close personal friend). For some analysis procedures the data need to be binary (relation absent or present). The following dichotomization is used for the acquaintanceship networks: 1 for values larger than 2 (friend, close friend), 0 for other values (not knowing, not having met, having met). The data set contains two actor attribute variables: primary disciplinary affiliation (sociology, anthropology, statistics and mathematics, psychology), and the number of citations (social science citation index). The complete data set can be found in Wasserman and Faust (1994, p. 745–748) and is one of the standard data sets distributed with UCINET.

2.2 UCINET

UCINET 6.0 (Version 6.05; Borgatti, Everett, and Freeman, 2002) is a comprehensive program for the analysis of social networks and other proximity data. It is probably the best known and most frequently used software package for the analysis of social network data and contains a large number of network analytic routines. The program is a commercial product, but a free evaluation version is available, which can be run for 30 days without registering. The manual consists of two parts: a user's guide

(data management and manipulation) and a reference guide (network analysis). It also available online through the help function.

UCINET is a menu-driven Windows program, and, as the developers say themselves, "is built for speed, not for comfort" (Borgatti, Everett, and Freeman, 1999). Choosing procedures from the menus usually results in opening a parameter form where the input for the algorithms is specified. Speedbuttons are available for data management, export to Pajek and Mage, and launching NetDraw, which three programs are distributed with UCINET. Two kinds of output are generated: textual output, saved in log files and displayed on the screen (see Figure 2 for an example), and data sets that can be used as input for other procedures.

Data entry and manipulation

UCINET is matrix oriented, that is, data sets are collections of one or more matrices. A single UCINET data set consists of two files: one containing the actual data (extension ##D) and one containing information about the data (##H). UCINET data sets can be created by importing data or by entering data directly via the built-in spreadsheet. The spreadsheet editor, containing the EIES data, is shown in Figure 1. The import function can process several types of network data: raw ASCII data, ASCII data saved in DL format, Excel data sets, and data formats from the programs KrackPlot, NEGOPY, and Pajek.

UCINET provides a large number of data management and transformation tools like selecting subsets, merging data sets, permuting, transposing, or recoding data. It has a full-featured matrix algebra language, it can handle two-mode (affiliation) data as well as derive one-mode data sets from two-mode data. There is an option to enter attribute data and to specify missing values. It should be noted, however, that only a few procedures can handle missing values properly. UCINET is distributed with a large number of example data sets, including Freeman's EIES data.

Visualization techniques

UCINET contains graphical tools to draw scatterplots, dendrograms, and tree diagrams (see Figure 3), which can be saved as bitmap files (BMP). The program itself does not contain graphical procedures to visualize networks, but it has a speedbutton to execute the program NetDraw (Borgatti, 2002), which reads UCINET files natively. NetDraw, developed for network visualization, has advanced graphical properties and is further discussed in Section 3.1.1. In addition to export functions to Pajek and Mage, data can be exported for visualization in KrackPlot.

Descriptive methods

The program contains a large number of network analytic routines for the detection

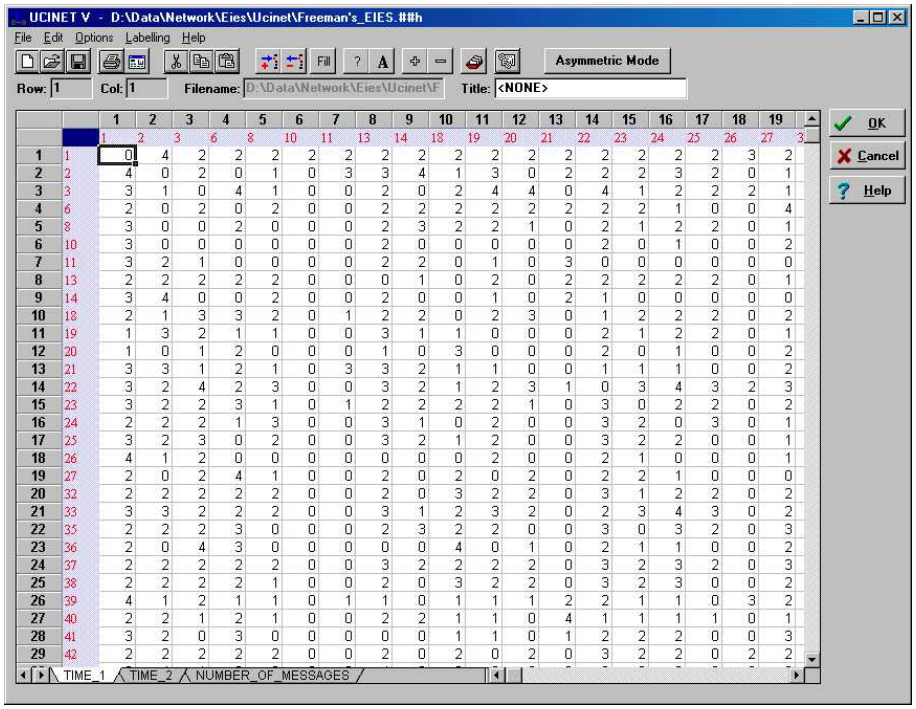

Figure 1: UCINET spreadsheet editor containing the EIES data.

of cohesive subgroups (cliques, clans, plexes) and regions (components, cores), for centrality analysis, for ego network analysis, and for structural holes analysis. As an example, the output of a centrality analysis is presented in Figure 2. For each node, it contains the in- and out-farness (the sum of the lengths of the geodesics to and from every other node), and the in- and out-closeness centrality (the reciprocal of farness times $g - 1$, with g the number of actors), some descriptive statistics, as well as Freeman's group closeness index (Freeman, 1979). The in-closeness for the EIES acquaintanceship data is 43.9% and 68.6%, and the out-closeness is 15.6% and 53.7% for time point 1 and 2, respectively. The data were dichotomized (see Section 2.1) before the analysis. If the user does not dichotomize and symmetrize the network, default options are used (all entries larger than 0 are given value 1, and the data are symmetrized by using the maximum value in a dyad). The default symmetrization was used here.

Group centrality options have been recently added (Everett and Borgatti, 2004). The program finds the most central subgroup of fixed size, or tests the (degree) centrality of a specified group. For the dichotomized and symmetrized EIES data (first observation) the most central subgroup of 6 actors consists of sociologists and anthropologists (centrality 87.5%). The degree centrality of the group of sociologists is 24. These results differ from the results of Everett and Borgatti (2004), due to

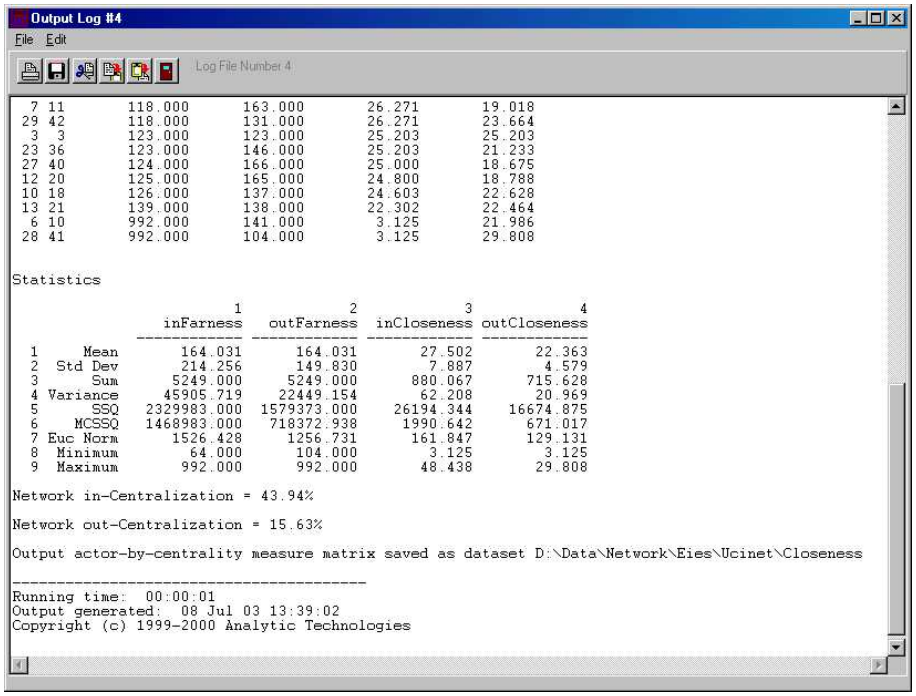

Figure 2: UCINET log file presenting the results of centrality analysis of the EIES acquaintanceship data (first observation).

the different transformations applied. The mean, standard deviation, and p-values based on permutation tests are given: 27.6, 1.54, and 0.97, respectively.

Analyzing the dichotomized and symmetrized (reciprocal relations) EIES data to detect cohesive subgroups based on complete mutuality (i.e., cliques) results in finding 8 and 15 cliques in the EIES data at the first and second time point, respectively. The cliques are presented in Table 3 (the cohesion index is provided by NetMiner, see Section 2.4). UCINET gives the opportunity to further inspect the cliques by calculating the clique overlap with a single link hierarchical cluster procedure (which will be presented in the next paragraph as an example of a procedure-based technique). The cliques found at the second observation are the same or combinations of those found at the first observation.

Procedure-based analysis

UCINET contains a number of routines for procedure-based analysis. One procedure, cluster analysis, was already mentioned. Other procedures are multidimensional scaling (metric or non-metric), two-mode scaling (singular value decompositions, factor analysis, and correspondence analysis), analysis of roles and positions (structural, role, and regular equivalence) and fitting core/periphery models.

Table 3: Cliques in the EIES acquaintanceship data obtained with UCINET.

First observation			Second observation		
Clique	Actors	Cohesion[1]	Clique	Actors	Cohesion[1]
1	14,20,22,24	7.000	1	1,2,31,32	6.222
2	14,16,22,24	7.467	2	1,11,31,32	6.588
3	14,22,24,29	8.000	3	1,13,31	5.118
4	14,20,24,25	9.333	4	1,18,31	5.800
5	2,9,32	14.500	5	1,29,31	4.143
6	1,2,31	10.875	6	1,8,11,32	10.182
7	1,18,31	29.000	7	1,2,9,32	8.615
8	16,21,22	7.909	8	3,14,23	9.667
			9	10,20,29	6.692
			10	14,20,22,24,29	8.438
			11	14,20,24,25	8.615
			12	14,16,22	6.214
			13	14,15,29	5.800
			14	15,29,31	6.214
			15	16,21,22	10.875

[1] Cohesion index of Bock and Husain (1950), provided by NetMiner.

There are hierarchical and non-hierarchical procedures to perform a cluster analysis of the relational data. Using the adjacency matrix as input, the actors are clustered on the basis of their relations. In the analysis of clique overlap mentioned above, the so-called clique overlap matrix is used as input. This matrix indicates for each pair of actors the number of times they occur in the same clique. The result for the first observation of the EIES data, that is, a tree diagram showing the progress of the cluster analysis, is presented in Figure 3 (single link procedure; average and complete link are also available). It shows the level of overlap between the cliques (e.g., actors 14 and 24 are most often together in one clique, followed by the combination of actors 14, 22, and 24).

Several types of structural equivalence procedures can be performed based on the measurement of equivalence (Euclidean distances, correlations, cost functions). The equivalence of the actors is given in a so-called equivalence matrix, which is the input of a hierarchical cluster procedure to find clusters of actors. For example, using the procedure based on comparisons of actor profiles (rows or columns in the adjacency matrix) measured by Euclidean distances (Burt, 1976), actors 12 and 23 are most equivalent, with the minimum distance of 5.8 between them (first observation of the acquaintanceship data). These actors are the first ones to be joined in one cluster. Actor 1 joins this cluster at one of the last stages of the process, having an

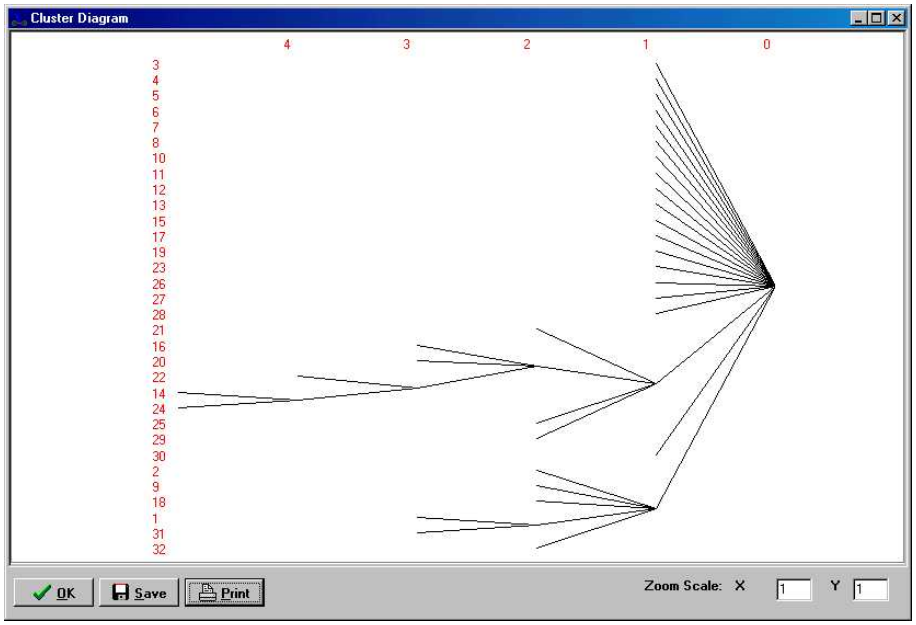

Figure 3: UCINET tree diagram for the single link hierarchical clustering of the clique overlap matrix (first observation of the EIES acquaintanceship data).

equivalence value of 16.3 and 15.9 with actors 12 and 23, respectively.

Statistical modeling

Various statistical routines are available in UCINET, ranging from simple statistics to fitting the p_1 model (Holland and Leinhardt, 1981). There are autocorrelation methods, QAP correlation and regression procedures, and univariate vector methods combined with permutation tests. An example of the latter group of methods is ANOVA with attribute vectors and/or rows or columns of the adjacency matrix, representing a sending or receiving actor, as variables. This is different from procedures where all incoming and outgoing links in an adjacency matrix are used as input for an ANOVA (e.g., MultiNet).

Fitting the p_1 model to the first observation of the dichotomized EIES acquaintanceship data gives estimates of the 'density' and 'reciprocity' parameters (-3.45 and 4.39), and for each actor the expansiveness and popularity parameters (not presented). Expected values and residuals to inspect the fit of the model are given as well. Computation of QAP correlations between the three EIES matrices gives the correlations as presented in Table 4, with p-values indicating the percentage of random correlations that are as large as the observed correlation in 2500 permutations (see Krackhardt, 1987). Besides Spearman correlations, the simple matching coefficient, the Jaccard coefficient and Goodman-Kruskal's gamma are calculated.

Table 4: QAP correlations obtained with UCINET in the EIES data (p-values in parentheses; 2500 permutations).

	Acquaintanceship			
	time 1		time 2	
Acquaintanceship time 2	0.809	(0.00)	—	—
Messages sent	0.240	(0.00)	0.347	(0.00)

2.3 Pajek

Pajek (Version 0.94; Batagelj and Mrvar, 2003a) is a network analysis and visualization program, specifically designed to handle large data sets. The main goals in the design of Pajek are 1) to facilitate the reduction of a large network into several smaller networks that can be treated further using more sophisticated methods, 2) to provide the user with powerful visualization tools, and 3) to implement a selection of efficient network algorithms (Batagelj and Mrvar, 1998). The program can be downloaded free of charge, and its developers are continually updating it. There is no online help, however, and the available documentation is not sufficiently detailed for users who are not experts in network analysis[2].

Pajek can handle multiple networks simultaneously, as well as 2-mode networks, and time event networks. Time event networks summarize the development or evolution of networks over time in a single network (using time indicators). In Pajek very large networks can be analyzed, with more than one million nodes. (The available memory on the computer sets the actual limit. To save memory, names and labels of nodes are not kept for extremely large networks, but these can be attached later to smaller subnetworks.)

Large networks are hard to visualize in a single view. Therefore meaningful substructures have to be identified, which can be visualized separately. The algorithms implemented in Pajek are especially designed for this purpose (see Batagelj and Mrvar, 2003b). Pajek uses six different data structures: 1) networks (nodes and arcs/edges), 2) partitions (classifications of nodes, where each node is assigned exclusively to one class), 3) permutations (reordering of nodes), 4) clusters (subsets of nodes), 5) hierarchies (hierarchically ordered clusters and nodes), and 6) vectors (properties of nodes). Partitions contain discrete attributes of nodes, whereas vectors contain continuous attributes.

The structure of the program is entirely based on these six data structures and on transitions among these structures. The main window presents six drop lists –

[2]A very helpful and well-written textbook by De Nooy, Mrvar, and Batagelj, on using Pajek for exploratory network analysis is forthcoming.

one for each data object – as well as buttons to open, save, and edit the data objects in these lists. The program is menu-driven, where the menu items are ordered according to the data objects to which they apply. The results generated by the procedures are usually presented using the data structures (instead of graphical or tabular output), and can be used as input in other procedures such as visualization methods.

Data entry and manipulation
Network data can be entered in four ways: 1) by defining a (small) network inside the program, 2) by importing ASCII network data from network files (extension NET), 3) by importing data from software packages with other formats (e.g., UCINET DL-files and formats of some visualization programs), and 4) by opening a Pajek project file (PAJ), which combines all different data structures into a single file. The NET-files consist of a node list and arcs/edges list, aimed at entering large networks more efficiently, specifying only the existing ties. For small networks the link list can be replaced by an adjacency matrix. Other data objects can be imported from ASCII data files or generated inside the program. For example, attribute data have to be entered as partitions in ASCII data files (CLU) or as vectors in ASCII data files (VEC). All data objects together can be saved in a PAJ-file.

Pajek contains manipulation options for all its data structures. For example, networks can be transposed, directed graphs changed into undirected graphs and vice versa, lines can be added or removed, or the network can be reduced by shrinking classes or extracting parts. The program also contains basic network operations like recoding or dichotomization. There is no option to specify missing relations, whereas it is possible to specify missing values for attributes (partitions and vectors). Also, there are ample transformations for attributes and options to create other data objects on the basis of the attributes (hierarchies, clusters).

Pajek offers facilities for longitudinal network analysis. Time indicators for the actors' presence in the network at certain observations can be included in the data files, and the user can generate a series of cross-sectional networks. Analyses can be performed on these networks, and the evolution of the network can be examined (e.g., the evolution of balance in a network). These analyses are non-statistical; for statistical analysis of network evolution the module SIENA of the StOCNET package can be used (Section 2.7; see also Snijders, 2004).

Visualization techniques
The graphical properties of Pajek are advanced. The *Draw* window gives the user many options to manipulate the graphs (layout, size, color, spin, etc.). Moreover,

graphical representations of partitions, vectors, and combinations of partitions and vectors can be obtained. The network drawing is based on the principle that distances between nodes should reveal the structural patterning of the network (see also Freeman, 2004). Besides simple layouts (circle, random), Pajek has several automatic procedures to find optimal layouts: procedures using eigenvectors, special procedures for layer drawing of acyclical networks, and spring embedders. The latter procedures are called so, because in those algorithms it is assumed that the nodes are connected by *s*prings, whose stress is to be minimized.

Pajek uses two spring-embedding algorithms to visualize network data: the Kamada-Kawai and the Fruchterman-Reingold algorithms. The former one produces more stable results, but is slower and less suited for large networks. The latter algorithm is faster and can handle large networks. Both are optimization procedures that do not yield the same mapping each time they are run. The graphs, however, should resemble each other largely.

The Kamada-Kawai algorithm is used to draw a graph of the EIES acquaintanceship data at the first observation point, which is presented in Figure 4. In the network drawing, partitions (here: actor's discipline) are depicted by colors and shapes: a blue diamond is sociology, a red circle is anthropology, a magenta circle is statistics, and a green box is psychology. Vector values (here: number of citations) are represented by the size of the nodes, where larger nodes indicate higher citation rates. The nodes can be dragged and dropped to improve the graph, and right-clicking a node shows (textually) to which other nodes it is tied. The programs NetDraw, distributed with UCINET, and NetMiner have the same functionalities.

By creating a super matrix that combines the two acquaintanceship matrices (at the two time points), a visualization of the (dichotomized) EIES data over time can be created (see Everton, 2002). Such a super matrix can be created in, for instance, UCINET, and can be exported to Pajek or opened in NetDraw. Using the Fruchterman-Reingold algorithm to draw the network results in a visualization of the evolution of the network, presented in Figure 5. Networks can also be drawn manually by dragging and dropping nodes with the mouse, as was done to improve the graph in Figure 5. Pajek also supports 3D visualization. The visualizations can be saved using several formats, amongst others (encapsulated) postscript file (EPS), scalable vector graphics file (SVG), kinemages file (KIN), bitmap file (BMP), and virtual reality file (VRML).

Descriptive methods

Each data object in Pajek has its own descriptive methods. The largest number of methods is available for networks, for instance, computation of degrees, depths,

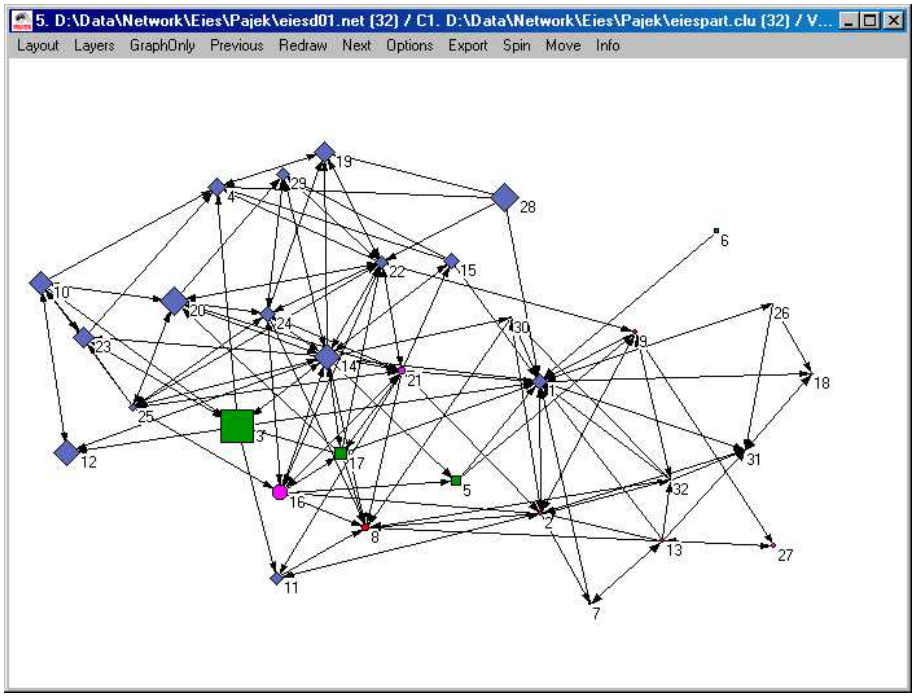

Figure 4: Pajek draw window presenting the graph of the dichotomized EIES acquaintanceship network (first observation) using the Kamada-Kawai spring embedder.

cores, or cliques (output is a partition), centrality (closeness, betweenness), detection of components (weak, strong, biconnected, symmetric), paths, or flows, structural holes, and some binary operations on two networks. The menu *Info* gives general characteristics of each data structure.

Computing closeness centrality with Pajek is straightforward. The network has to be dichotomized before calculating the closeness. For directed graphs, the in- or out-closeness can be calculated as well as the closeness for the symmetrized network (default using the maximum of the two links) by choosing the command *All*. This latter option gives 0.390 and 0.515 for closeness for time points 1 and 2, respectively.

Identifying cliques in large networks is difficult, because of the large number of cliques. Therefore, unlike UCINET, Pajek has no direct procedures for detecting cliques. There is, however, an indirect way of finding cliques by looking for complete triads (cliques of size 3) in a network (De Nooy, Mrvar, and Batagelj, 2002). Using the option to search for particular fragments (in this case triads) in the first observation of the dichotomized and symmetrized (based on reciprocated relations) acquaintanceship network, cliques of size 3 are found. The output, presented in Figure 6, consists of several data objects, one of them being a subnetwork made of the

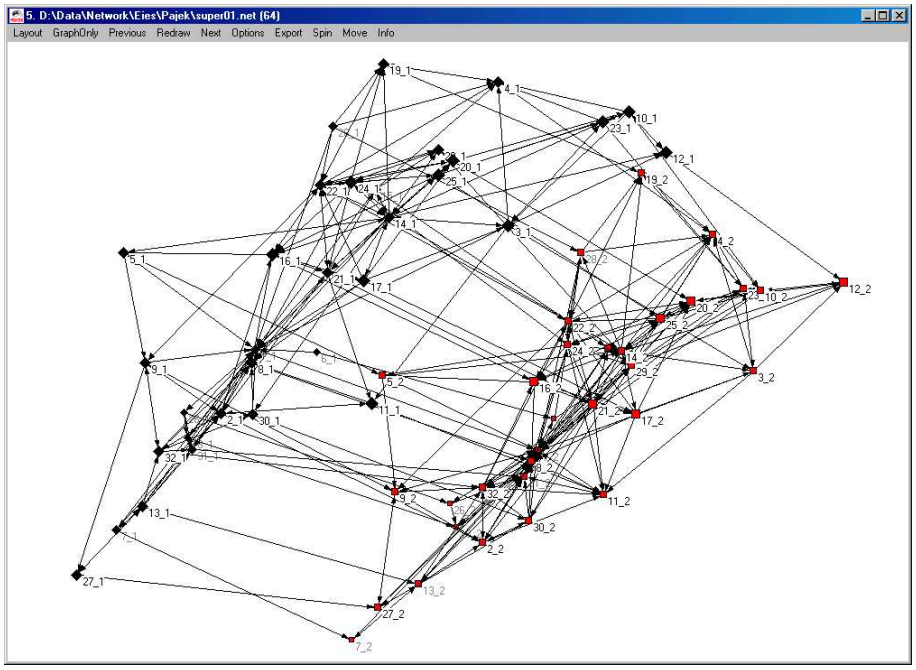

Figure 5: Pajek draw window presenting the simultaneous drawing of the dichotomized EIES acquaintanceship networks (both observations) using the Fruchterman-Reingold spring embedder.

desired cliques. Figure 6 shows the triads (cliques of size 3), as well as the cliques of size 4, which were also found with UCINET (see Table 3). In addition, a hierarchy is generated to inspect the overlap of triads, as well as a partition to identify the number of triads to which a node belongs (not shown here).

Instead of a clique-procedure, Pajek contains the procedure *p-cliques*. This procedure results in a partition of the network nodes into clusters such that the nodes within one cluster have at least a proportion of p neighbors inside the cluster (cf. NEGOPY, Section 3.2.1). For large networks it is preferable to use k-cores instead of cliques, because of the computing time. Dense parts of large networks can be found using k-cores.

Procedure-based analysis

Pajek contains several procedure-based methods, for instance, for detecting structural balance and clusterability, hierarchical decomposition, and blockmodeling (structural, regular equivalence). For the analysis of structural equivalent actors, dissimilarities between nodes can be computed in several ways. In its pull-down menu, Pajek indicates if the network is too large for calculating dissimilarities, in view of the computational complexity and the amount of time involved. For the first ob-

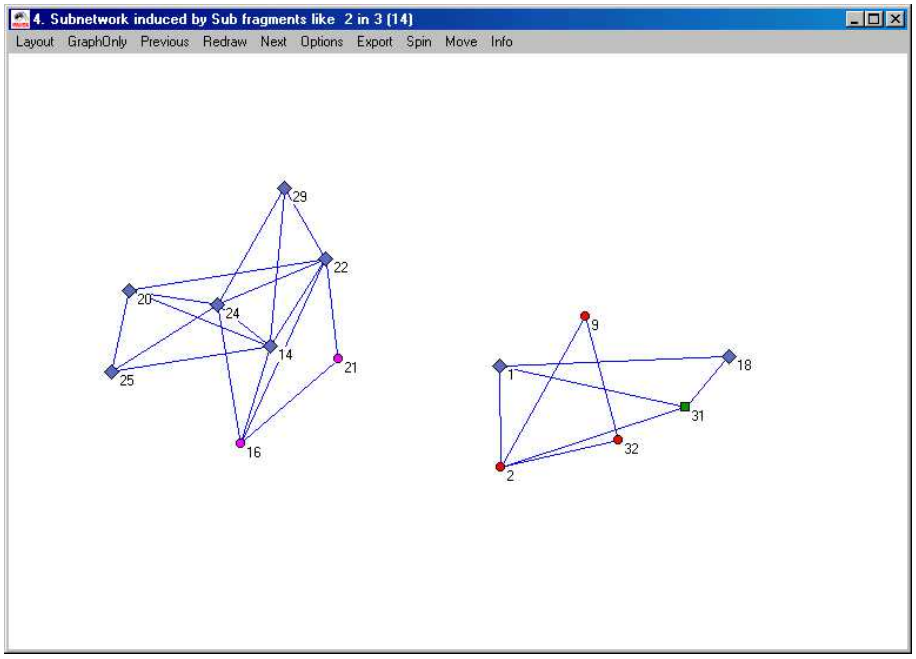

Figure 6: Pajek draw window presenting the subnetwork consisting of triads (cliques) in the first observation of EIES acquaintanceship network.

servation of the acquaintanceship data dissimilarities between actors are calculated using Euclidean distances, and the resulting matrix is used in a hierarchical cluster analysis, using Ward's linking method to combine clusters (the default option out of six). The resulting clusters are presented as a hierarchy and the corresponding dendrogram is saved in an EPS-file. The dendrogram, presented in Figure 7, shows two very dissimilar clusters: one containing the actors 1, 2, 6, 7, 9, 13, 26, 27, 30, 31, and 32, i.e., few sociologists, and actors with low citation rates, who were all positioned on the right side of the graph in Figure 4, the other containing the remaining actors. An almost identical solution was found with UCINET that employs the single linkage method.

Blockmodeling the dichotomized acquaintanceship data in which the block types are defined using structural equivalence, does not yield statisfactory results. Starting from random partitions, the final, best fitting partitions (in 2, 3, or 4 blocks) still had large associated error scores (125, 115, 111, respectively). Besides blockmodeling based on structural and regular equivalence, Pajek can be used for generalized blockmodeling, where combinations of permitted block types can be defined by the user (see Doreian, Batagelj, and Ferligoj, 2004).

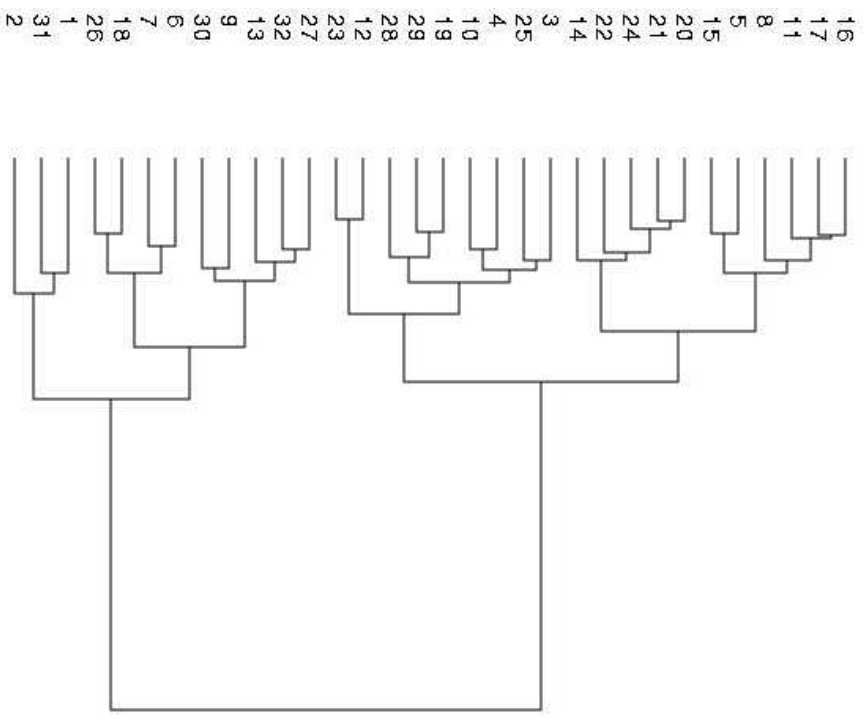

Figure 7: Dendrogram of the hierarchical cluster analysis (Ward linkage) of the EIES acquaintanceship data (first observation) obtained with Pajek.

Statistical modeling

The program contains only a few basic statistical procedures. Attributes of nodes (including structural properties that can be expressed as attributes), which are available as partitions and vectors, can be included in statistical analyses: computation of correlations, linear regression, and cross-tabulation (including some measures of association). However, the statistical package R can be called with Pajek data structures (networks and vectors) and the statistical procedures available in R can be used (see Section 3.3).

2.4 NetMiner II

NetMiner II (Version 2.0.5; Cyram, 2003) is a software tool that combines social network analysis and visual exploration techniques. It allows users to explore network data visually and interactively, and helps to detect underlying patterns and structures of the network. Two versions of the program are available for users: NetMiner II for Windows (commercial) and NetMiner II for Web (online freeware with reduced functionality compared to the commercial product). Both versions are Java-based applications. A free evaluation version is available, which can be used for 21 days

without registering. NetMiner offers good support providing online help and a user's manual that can be downloaded from the Cyram website.

The program is especially designed for the integration of exploratory network analysis and visualization. In order to facilitate this integration the main window of the program contains a map frame in which the results of the analysis are graphically presented and a separate map control toolbar (apart from the main toolbar). Moreover, the *Explore* panel can be activated to inspect the results of the analysis. In Figure 8 the main NetMiner window and its features are presented.

Data entry and manipulation

NetMiner adopts a network data model that is optimized for integrating analysis and visualization. It combines three types of variables: adjacency matrices (called *layers*), affiliation variables, and actor attribute data. The data can be entered in three ways: 1) directly via the built-in matrix editor (a spreadsheet editor similar to the one that is available in UCINET, see Figure 1), 2) by importing Excel datasheets, comma-separated ASCII values files (CSV), or UCINET DL files, or 3) by opening a NetMiner data file (NTF), which contains the values of the three types of variables. Data sets are saved as NTF-files or can be exported in Excel, CSV, or UCINET DL format.

The program contains ample data manipulation options (transformation, recoding, symmetrizing, dichotomizing, selection, normalization, etc.), facilitated by the data manager that contains the transformation history. It is possible to create random graphs (including scale-free networks) and to edit text files. A drawback, however, is that the program does not allow the specification of missing values.

Visualization techniques

Like Pajek and NetDraw, NetMiner has advanced graphical properties. Moreover, almost all results are presented both textually and graphically, contrary to both other programs, where the user needs to request visualization of the results of a certain analysis. In NetMiner graphical and textual results are directly obtained via the *Explore* function of the main menu. The other two functions of the main menu produce either textual results in report form (*Analyze*), or graphs (*Visualize*) with various options.

The *Analyze* function has reduced computing time in comparison to the *Explore* function and contains more analysis methods. Network drawing can be based on spring-embedding algorithms, multidimensional scaling, so-called applied procedures based on analysis procedures (e.g., centrality vectors or clustering combined with spring embedders), and simple procedures (circle, random).

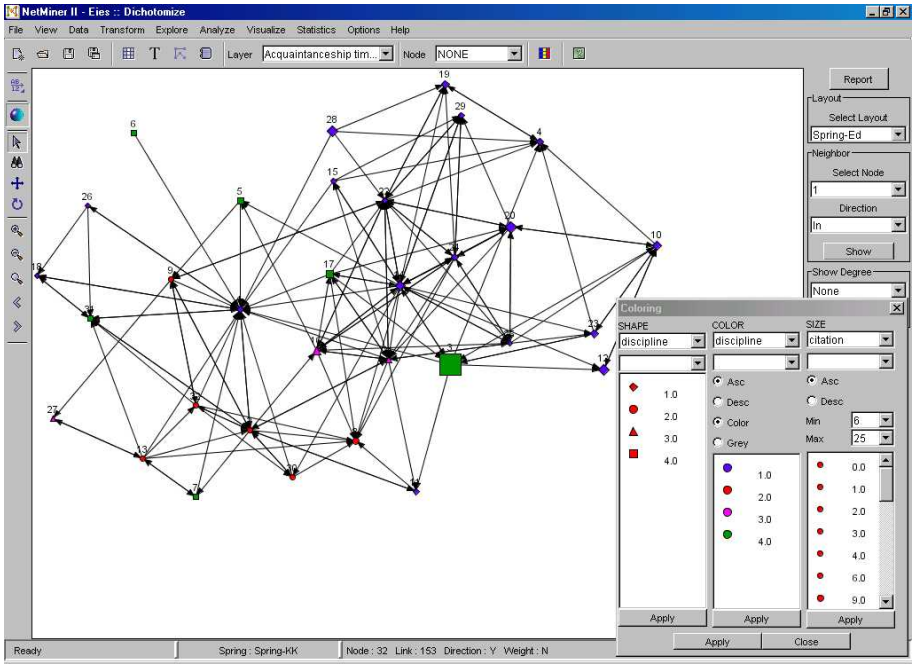

Figure 8: NetMiner II user interface presenting the graph for the dichotomized EIES acquaintanceship network (first observation) using the Kamada-Kawai spring embedder.

The Kamada-Kawai and Fruchterman-Reingold algorithms are the spring embedders that are implemented in NetMiner, as well as two algorithms based on the spring embedder by Eades. In Figure 8 the user interface of NetMiner is presented, in which the map frame contains a graph of the first observation of the EIES acquaintanceship network obtained with spring embedding algorithm of Kamada-Kawai. The aim of the Kamada-Kawai algorithm is to find a set of coordinates in which, for each pair of nodes, the Euclidean distance is approximately proportional to the geodesic distance between two nodes (e.g., see Everton, 2002, and Freeman, 2004). Although the procedure does not produce exactly the same mapping each time it is used, the graphs obtained with the Kamada-Kawai algorithm in Pajek (Figure 4) and NetMiner (Figure 8) largely resemble each other.

NetMiner has the functionality to set node shape, color, and size according to three attribute variables (both categorical and continuous), like Pajek and NetDraw. In Figure 8 the nodes are colored and shaped according to the attribute discipline: a blue diamond is sociology, a red circle is anthropology, a magenta triangle is statistics, and a green box is psychology. The size of the nodes reflects the value of the second attribute, the number of citations, where larger nodes reflect higher citation rates.

The multidimensional scaling algorithms for drawing graphs in NetMiner can be metric or non-metric. For instance, Torgerson-Gower's classical metric multidimensional scaling (principal coordinate analysis), based on an eigenvalue decomposition of which only the first two positive eigenvalues and eigenvectors, can be applied.

NetMiner supports various 3D visualizations and contains a graph editor which can be used to generate new graphs (random placement of nodes or positioning by user) or edit existing graphs (adding new nodes or links). All visual displays can be saved in a wide variety of formats (including EPS, GIF, JPEG, PDF, PNG, EMF, etc.).

Descriptive methods

The network statistics available in NetMiner include methods to analyze the connection and neighborhood structure of the network (e.g., influence, structural holes) and subgraph configurations (dyad and triad census), to calculate centrality measures (e.g., closeness, betweenness), and to analyze subgroup structures (cliques, clans, cores). To show the integration of standard network methodology and visualization in NetMiner, the closeness centrality index was calculated for the dichotomized EIES acquaintanceship data (first observation). NetMiner, like Pajek, has the option to calculate the in- and out-closeness of directed graphs. UCINET only calculates closeness for undirected graphs.

Via the *Explore* menu the in-closeness centrality was calculated. The output consists of two parts: a report containing the closeness indices (at actor and network level) and a graphical presentation of the calculated closeness, the so-called centrality map, presented in Figure 9. The figure shows the NetMiner user interface and the visual presentation of in-closeness statistics in the map frame, in which also the centralization index (the in-closeness for directed graphs) is given: 0.439. The out-closeness is given in the textual output (obtained by clicking the Report-button): 0.156. For the second observation of the EIES data, the in- and out-closeness equal 0.686 and 0.537, respectively. The same values were found with UCINET and Pajek.

Figure 9 shows one of the interactive features of NetMiner: right-clicking a node opens a context-sensitive menu with which network properties of the node can be obtained (in-degree, out-degree, egonet size and density) or the neighborhood of the selected node can be drawn (in a new submap window). For actor 6 the network properties are presented (note that the egonet density cannot be calculated because node 6 is only connected to one other node).

Figure 10 displays the result of the analysis of cohesive subgroups: the visualization of cliques in the EIES data, dichotomized and symmetrized as before. It presents the cliques labeled G1 to G8 and its members. The cliques found by NetMiner are

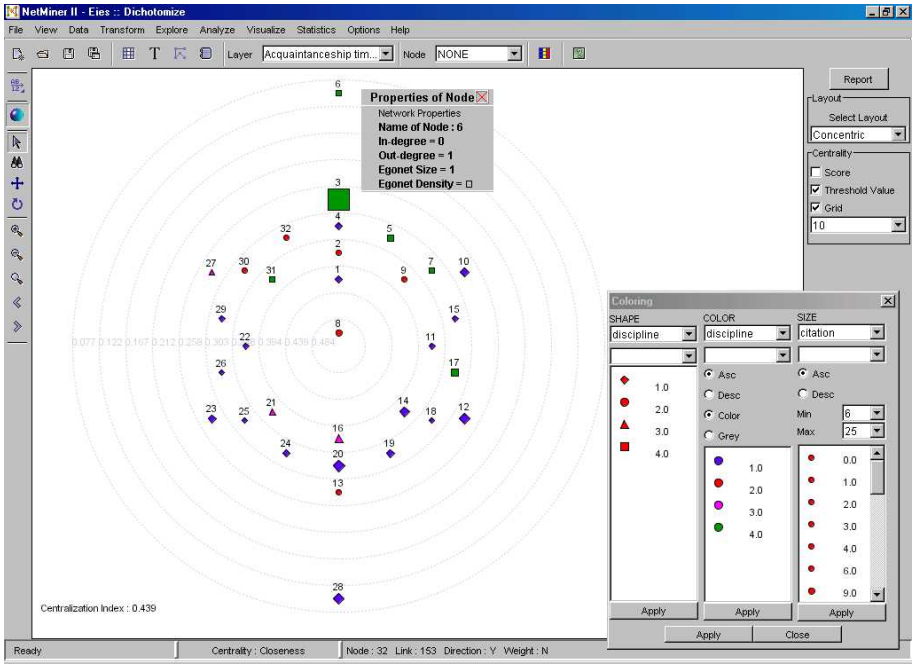

Figure 9: NetMiner II user interface showing the closeness index and centrality map for the EIES acquaintanceship data (first observation).

shown in Table 3, and are the same as those found by UCINET. Additionally, NetMiner reports for each clique the cohesion index by Bock and Husain (1950). This index measures the degree to which strong ties are within rather than outside the clique. If the index is equal to 1 the strength of ties does not differ within the subgroup compared to outside the subgroup. If the ratio is larger than 1 the ties within the subgroup are more prevalent than the ties outside the subgroup.

Right-clicking a clique in the map opens a menu with which properties of the group, group member lists, or group networks can be obtained. In Figure 10 the member list of clique G7 is shown, as well as the group network of clique G3. Previous versions of NetMiner (Version 1.x) had the option to draw directly bipartite, co-member, and overlap maps of the cliques. Unfortunately, in NetMiner these features can only be obtained indirectly. For example, the clique bipartite map can be obtained by adding the clique affiliation matrix to the data set (via the analysis report), selecting the affilition mode (in the *Transform* menu), and choosing the bipartite method. The node-clique bipartite graph for the EIES data is presented is Figure 11. The cliques are represented by yellow boxes labelled K1 to K8.

Procedure-based analysis

NetMiner contains routines for multidimensional scaling, correspondence analysis,

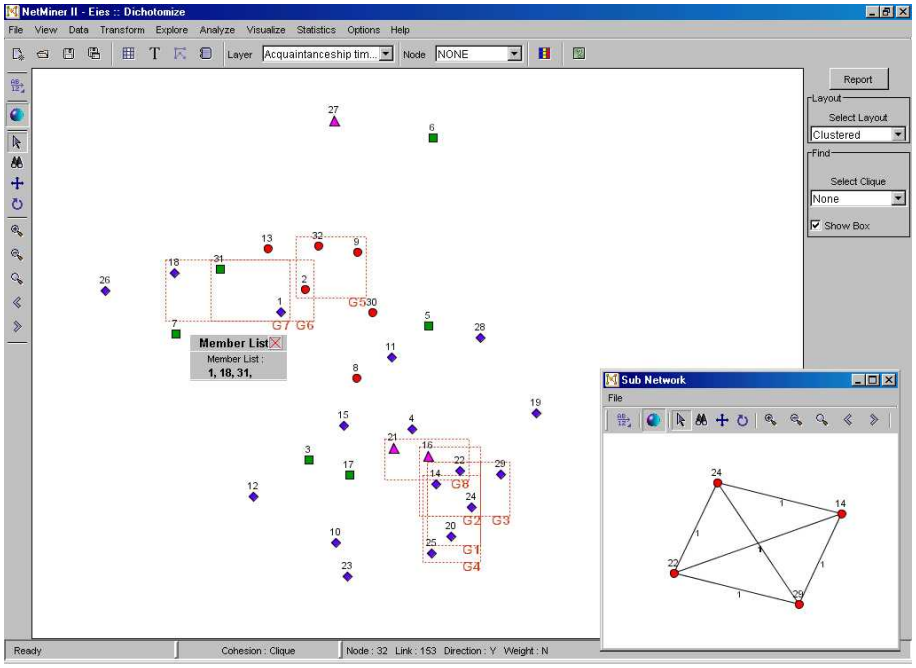

Figure 10: NetMiner II user interface showing the cliques (at least size 3) for the EIES acquaintanceship data (first observation).

cluster analysis, and matrix decompositions (eigen, singular, spectral). These procedures are integrated in the *Explore/Analyze* submenus and are available as separate options in the *Statistics* menu. The program also contains some procedure-based routines to explore the role-set structure of a network (structural, role, and regular equivalence). Finally, blockmodel routines are available, including goodness-of-fit statistics and permutation tests of significance.

The structural equivalence procedure is used to analyze the first observation of the acquaintanceship network, based on the similarity-of-tie-profiles among the actors. For all pairs of actors the structural equivalence is computed using Euclidean distances (Burt, 1976). The diagonal values are specified to be ignored. The mean distance between pairs is 11.75 (S.D. 2.39).

Subsequent hierarchical clustering of the equivalence matrix gives a cluster diagram and the possibility to show the different clusters in a map. NetMiner gives four possible cluster linkage methods (single, complete, average, and Ward), whereas UCINET gives three, and Pajek six. For the comparison of different linkage methods, a hierarchical cluster analysis with average linkage is performed. The equivalence map is presented in Figure 12. In this map the different clusters are shown by giving them different colors (the number of clusters is chosen to be 4; the colors are assigned by the program). In the top of the map actors 1 and 31 form one cluster,

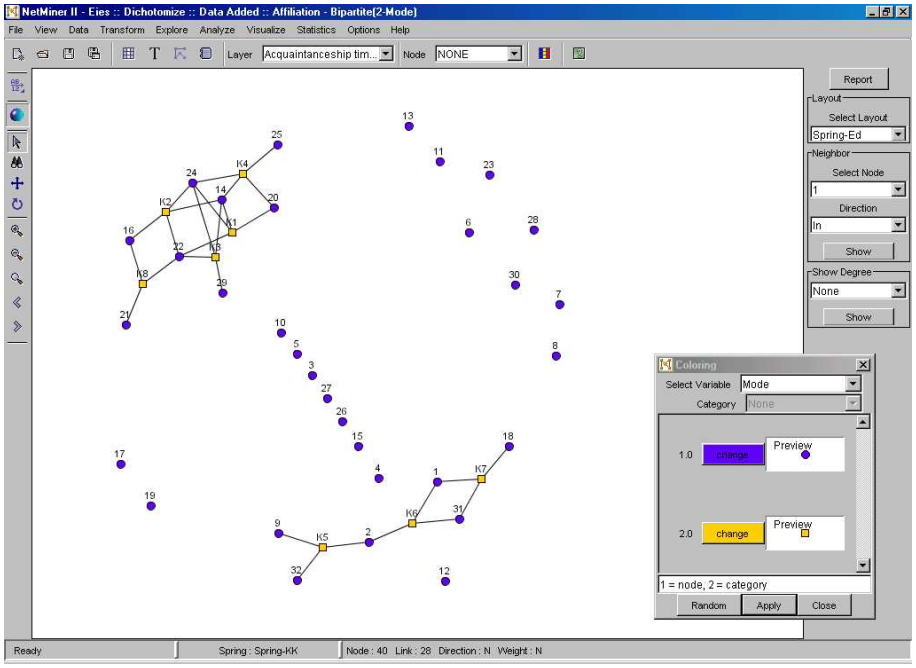

Figure 11: NetMiner II user interface showing the cliques bipartite map for the EIES acquaintanceship data (first observation).

while actor 2 constitutes a cluster by itself; on the right a cluster of nine actors is found, on the left a cluster of 20 actors.

Comparing the cluster methods of the three packages UCINET, Pajek, and NetMiner, similar results are found (given the different linkage methods). In all programs the equivalence measure was based on Euclidean distances, and actors 12 and 23 were found to be most equivalent, and actor 1 least equivalent with these two actors (UCINET, Section 2.2). Inspecting Figure 12 confirms this finding, where NetMiner locates actors 12 and 23 on one side and actor 1 on the other side of the map. The dendrogram presented by Pajek (Figure 7) shows a similar clustering.

Statistical modeling

NetMiner supports a number of standard statistical routines: descriptive statistics, ANOVA, correlations and regression. All of these routines can be applied to both attribute vectors and (adjacency) matrices. The statistics are given with conventional significance tests (based on independence and normality, which may not always be appropriate) and random permutation tests. For adjacency matrices QAP permutation is adopted (see Krackhardt, 1987). Besides, NetMiner provides Markov chain Monte Carlo simulation tests for several network measures based on the on the $\mathcal{U} \mid X_{i+}, X_{+j}$ and $\mathcal{U} \mid X_{i+}, X_{+j}, M$ distributions (cf. the module ZO in StOCNET).

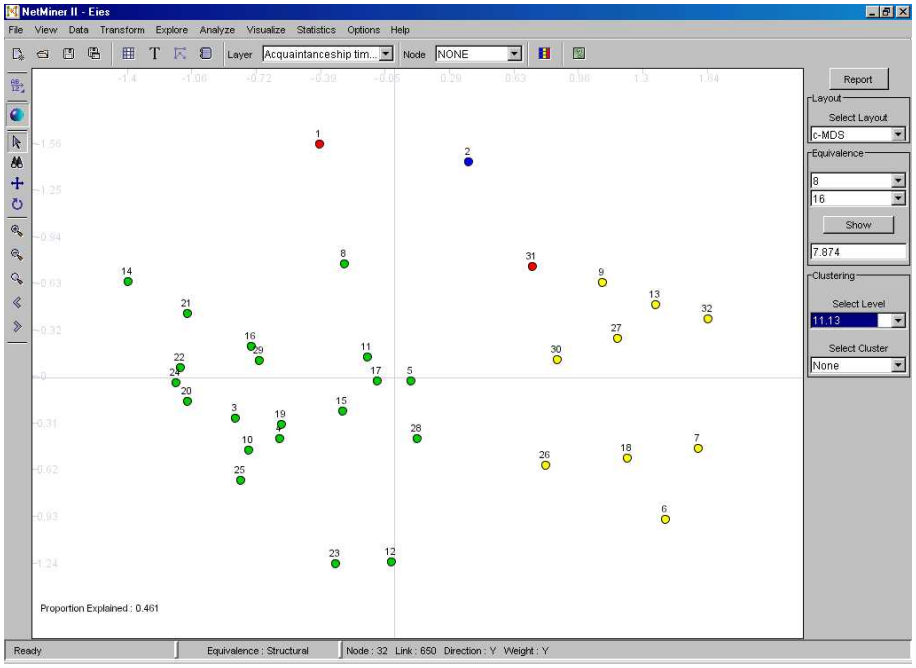

Figure 12: NetMiner II user interface showing the equivalence map and cluster map for the EIES acquaintanceship data (first observation).

The QAP-correlation found between the two time points of the acquaintanceship data is 0.809 (significant at $p = 0.001$ level, 1000 simulations). This is the same result as found by UCINET, but, unlike UCINET, NetMiner provides no additional information on the test.

2.5 STRUCTURE

STRUCTURE (Version 4.2; Burt, 1991) is a program "providing sociometric indices, cliques, structural and role equivalence, density tables, contagion, autonomy, power and equilibria in multiple network systems" (reference manual, p. 1). It is a command-driven DOS program that needs an input file containing commands for data management and network analysis. After opening the input file, the program executes the required routines without the possibility of user interaction. The program can be downloaded free of charge together with a comprehensive manual including introductions to network analysis, network data, and network models.

STRUCTURE supports network models within five types of network analysis. These are autonomy (analysis of structural holes), cohesion (detection of cliques), contagion, equivalence (analysis of structural or role equivalence and blockmodeling), and power (analysis of network prominence and equilibrium). The programs UCINET, Pajek, and NetMiner contain procedures to perform analyses of one or more

of these types. Most procedures in STRUCTURE, however, are unique and cannot be found in the other general programs. These procedures are discussed here.

Data entry and manipulation
STRUCTURE distinguishes four types of data: 1) direct measures of relations, 2) binary choice data (obtained with a name generator), 3) sociometric rank order data (where actors ranked their relations with others), and 4) (two-mode) joint involvement data (actors' involvement in the same events or affiliations with the same groups). The first three types have to be presented as adjacency matrices in ASCII data files with fixed positions. For the joint involvement data networks are created by reading events in each network and aggregating the weight of events in which each pair of actors is involved. Actor attributes are entered as ASCII values. Output data files are written in ASCII fixed-column format (WRT).

The program has a few data manipulation options, which are only available for directed relations: using diagonal elements as measures of strength of self-relations, symmetrizing relations, and transforming relations (converting to row or column marginals, eliminating negative relations, making networks row and/or column stochastic). For joint involvement data the weights can be defined in different ways.

Visualization techniques
STRUCTURE has no procedures to visualize networks.

Descriptive methods
The analysis of structural holes is the single descriptive method available.

Procedure-based analysis
The procedure-based analysis methods offered by STRUCTURE are hierarchical cluster analysis (detection of cliques, structural equivalence) and eigenvalue decomposition (to compute power measures). STRUCTURE can detect different kinds of cliques, depending on how relations are measured from the raw data and how cohesion is defined from the relations (Scott, 1991). Detection of cliques by STRUCTURE is based on hierarchical clustering of the matrix of cohesion, and is therefore different from clique finding procedures in UCINET, Pajek, and NetMiner.

In STRUCTURE cohesion can be defined in several ways. If cohesion is defined by the weakest relation between actors (default) and if cohesion between clusters is defined by the minimum cohesion between the actors in the clusters (cliques), then the clustering procedure will merge clusters if the minimum cohesion within the clusters remains positive. Thus, cliques are found in which the actors are completely

Table 5: Cliques in the EIES acquaintanceship data (first observation) obtained with STRUCTURE.

Valued, unsymmetrized			Dichotomous, symmetrized		
Clique	Actors	Cohesion	Clique	Actors	Cohesion
1	11,16,21	2	1	4,19	1
2	10,20,23,25	2	2	10,23	1
3	3,14,18,26	2	3	21,25	1
4	4,19,22,24,28,29	2	4	13,27	1
5	8,13,27,30	2	5	14,22,24,29	1
6	1,2,15,17,31	2	6	1,18,31	1
7	5,9,32	2	7	2,9,32	1

connected and have reciprocated relations. This also holds for cliques found by other programs, but the difference in STRUCTURE is that an actor can appear in only one clique. Other definitions of cohesion and other clustering methods result in different kinds of cliques (see Scott, 1991).

Applying the algorithm to the acquaintanceship data (first observation), without dichotomizing and symmetrizing the network, results in the detection of seven cliques. These cliques are presented in the left part of Table 5, which also gives the minimum cohesion within the clusters, here equal to 2 for all cliques. As a result of the clustering procedure there is no clique overlap. Applying the algorithm to the dichotomized and symmetrized (only reciprocated relations) EIES data again results in seven cliques, presented in the right part of Table 5. Comparing these results with the cliques in Table 3 shows that the solution is different, although cliques 5, 6, and 7 are found in both analyses.

Statistical modeling

STRUCTURE contains two routines for statistical modeling of the network data: contagion analysis and analysis of network equilibrium. The analysis of contagion in STRUCTURE is based on the principle that the structure of the network is such that the behavior (attribute) of one actor is influenced by other actors. This means that attribute values of actors are correlated, due to the structure of the network. Stated otherwise, an attribute that is affected by contagion results in network correlation. In STRUCTURE this is modeled with a regression equation in which the dependent variable is the attribute value of one actor (ego) and the independent variable is the weighted average of the values of the same attribute of the other actors (alters), where the weights reflect the structure of the network. This kind of contagion analysis is not directly available in the programs described earlier.

Table 6: STRUCTURE output of the contagion analysis for the EIES acquaintanceship data (first observation) with the attribute citation.

Observed responses	mean:	22.906
	S.D.:	31.737
Expected responses from contagion	mean:	24.088
	S.D.:	8.129
Contagion effect (32 observations)		
	regression intercept:	-7.426
	regression slope:	1.259
	correlation:	0.323
	jackknife t-test (31 df):	1.508

The program has two options to define the network weights: by equivalence (Euclidean distances) or by cohesion (relation values). Given these weights the regression equation is estimated with ordinary least-squares (OLS). If the input data are a random sample from a population, OLS gives inconsistent and inefficient estimates, and other estimation procedures must be used (Ord, 1975; Doreian, 1980). If the data are population data, however, then OLS is accurate. This is typically the case in network analysis (Scott, 1991). The significance of the contagion effect (the slope of the regression equation, i.e., the network correlation) is tested with a jackknife t-test. A contagion analysis was performed on the acquaintanceship data (first observation), with citation as the attribute affected by contagion and the weights defined by structural equivalence. The results are presented in Table 6.

The observed (ego) and expected (alters) citation rates are given together with the results on contagion. The network correlation of 0.323 is not significant according to the jackknife t-test (with $g - 1 = 31$ degrees of freedom), which indicates that structurally equivalent actors (researchers) do not tend to have the same citation rates.

The analysis of network equilibrium in STRUCTURE is based on the distribution of power, which is obtained with eigenprocedures (Katz, 1953; Bonacich, 1976). An actor is defined to be powerful if he receives many exclusive relations from powerful others. The scores range from 1 (most powerful) to 0 (weakest). Analysis of the first observation of the acquaintanceship data reveals that actor 1 is the most powerful (1.00) and actor 6 (0.05) is the weakest actor (see the graph of the network in Figures 4 or 8).

Network equilibrium is analyzed by predicting how relations in a network will change if powerful actors could initiate any relation they want. This prediction is

Table 7: Turnover table to equilibrium in the EIES acquaintanceship data (first observation) obtained with STRUCTURE.

	Equilibrium				
Observed	None	Weak	More	Strong	Total
None ($z = 0$)	222	120	0	0	342
Weak ($z < 0.1$)	112	496	17	0	625
More	8	8	8	1	25
Strong ($z > 0.5$)	0	0	0	0	0
Total	342	624	25	1	992

based on a linear regression model that predicts the value of equilibrium relations from observed relations (Scott, 1991). The equilibrium relations from actor i to j are defined by $z_{ij}\left(\frac{p_i}{p_j}\right)$, where z_{ij} is the relation from i to j divided by the row sum (row stochastic adjacency matrix), and p_i is the power of actor i. The analysis of the first acquaintanceship network results in a regression equation which predicts 42.4% of the variation in the equilibrium relations (the correlation is 0.65). A high correlation means that equilibrium relations and observed relations are alike, which implies that the inclination to change relations is small.

The program gives a so-called turnover table to equilibrium (presented in Table 7), showing the association between observed relations and equilibrium relations. It is used to determine stability and locate unstable classes of relations. The relations are divided into four classes. From the table it follows that change is primarily zero strength relations becoming weak and vice versa. This indicates that the network is relatively stable.

STRUCTURE provides an option for Monte Carlo network analyses. In such analyses networks can be simulated according to the uniform, (nearly) normal, or lognormal probability distribution. With these simulated networks, studies of any of the network models in the program can be carried out.

2.6 MultiNet

MultiNet (Version 4.24 for Windows; Richards and Seary, 2003) is a program suitable for the analysis of large data sets and sparse network data. The program is designed for contextual analysis, that is, analyzing network data with nodal attributes. Besides network data, the program contains some methods to analyze attribute data (crosstables, ANOVA, correlations). It is menu-driven, where higher level menus and extra menu items become available after the necessary options are specified. It

has context sensitive-online help and, like NetMiner, gives both graphical representations of the results and textual output. An example of the MultiNet user interface (including an example of some graphical output) is presented in Figure 13.

The program is available from the authors. There is no complete user's manual, which makes it difficult to use and explore MultiNet to its full extent, but the authors provide useful information and some papers on MultiNet modules (Seary and Richards, 2000; Seary, 2003).

Some of the network analysis methods and procedures in MultiNet were originally contained in separate programs. FATCAT (Version 4.2, Richards, 1993), for instance, performs the same type of categorical social network analysis and produces the accompanying contingency tables and panigrams as MultiNet. Although incorporated in MultiNet, FATCAT is still freely available as a stand-alone DOS program that runs under Windows. The program is interactive and menu-driven and it provides context-sensitive online help. Another program integrated in MultiNet is PSPAR (Seary, 1999), which estimates the p^* model (Wasserman and Pattison, 1996) for sparse matrices.

Data entry and manipulation

Because MultiNet is designed for the analysis of large networks, like Pajek it uses node and link lists as data input instead of adjacency matrices. The former is a list of all actors in the network together with the values of the available attributes, the latter is a list of the (existing) relations between the actors. There are three options to enter the data: 1) by opening a MultiNet system file (MNW), 2) by importing ASCII data from node (NOD) and link (LIN) files, or 3) by opening data in comma-delimited files (CSV). In the link file non-existing relations (e.g., the relations with value 0 in the acquaintanceship data) do not have to be specified. Multiple link variables, like the two observations of the EIES data, have to be included in one link file. Data are saved in MNW files or exported to ASCII NOD and LIN files. Distributed with the program are the two stand-alone utilities ADJ2NEG and FREEFIX to create node and link import files.

The program contains some data manipulation options (recoding, grouping variables together) and has a simple data manager. It is possible to a value for missing observations, which has to be the same for all network and attribute variables. There is also an option to treat missing links as zero values (no links) and vice versa.

Visualization techniques

MultiNet contains procedures to give graphical representations of almost all output generated by the analysis routines. It has graphical tools to draw histograms, cu-

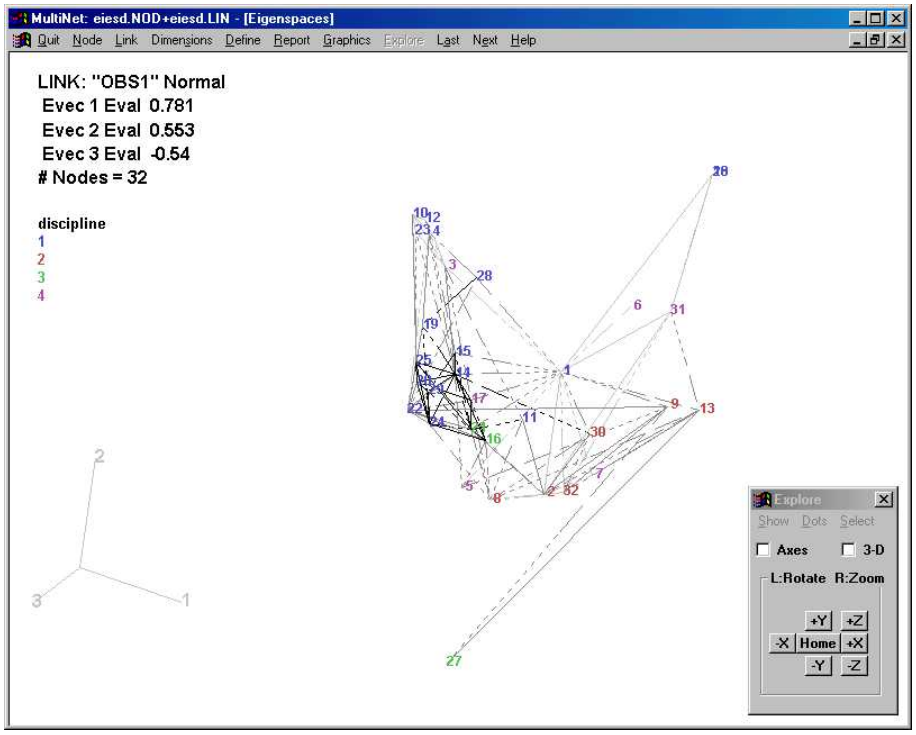

Figure 13: MultiNet *user interface showing a normal eigendecomposition for the dichotomized EIES acquaintanceship data (first observation).*

mulative distribution functions, and line diagrams. Networks are visualized using eigendecompositions (see Figure 13). Crosstables are visualized with so-called panigrams (see Figure 14). Adjacency matrices can be presented visually (see Figure 15), which can be useful to display large networks. To detect clustering one can permute the adjacency matrix according to actor attribute.

All graphical representations are interactive, which means that the user can click on displays to inspect attribute values or probability levels, explore effects, permute displays, or find information on nodes and links. The program also has several options to improve the displays (rotation, translation, magnification). The graphs can be saved, either as postscript (PS) or bitmap (BTM) files.

Descriptive methods

For network data the degree, betweenness, closeness, and components statistics can be computed, together with frequency distributions of these statistics. Frequency distributions and corresponding descriptives, like mean and standard deviation, of the network data (the links) and the attribute data (nodes) can be obtained as well.

Procedure-based analysis

With MultiNet one can analyze the structure of networks with several eigenspace methods. The methods create visual displays of the network such that the location of the actors reveals the structure of the relationships and their patterns (Richards and Seary, 2000; Freeman, 2004). Thus, the eigenmethods pursue the same goal as the spring-embedding algorithms (used in NetDraw, Pajek, and NetMiner) and the multidimensional scaling procedures (used in NetDraw and NetMiner). Pajek also contains some eigenmethods.

Eigenprocedures require dichotomized and symmetrized data. The result of an eigendecomposition is an eigenspace that can be used to visualize the network structure (Seary, 2003). In the visual displays the coordinates of the nodes are based on the coordinates of the first two or three eigenvectors, yielding 2D and 3D displays, respectively. Between the nodes lines are drawn based on the link variable (i.e., the dichotomized and symmetrized links in the original network). Associated with each dimension is a certain amount of variance in the original data, where the largest amount of variance is associated with the first dimension, and so on. A one dimensional display of the network can be generated as well, based on the first eigenvector. This is a so-called virtual adjacency matrix in which only the existing links are shown (using sparse methods; see Seary, 2003).

The results can be rotated, resized and rescaled to obtain a better presentation of the data. The eigenspace methods can also be used to partition the actors on the basis of the network structure. In Figure 13 the three dimensional normal eigendecomposition of the dichotomized EIES acquaintanceship data (first observation) is presented. The actors are colored according to their discipline (1–4: sociology–psychology). For every eigendecomposition a textual report is generated that includes details about the current eigenspace.

Statistical modeling

MultiNet contains four statistical techniques to analyze network data, of which the first three can also be used for the standard analysis of actor attribute data: 1) crosstables and χ^2-tests, 2) ANOVA, 3) correlations, and 4) the p^* exponential random graph model (Wasserman and Pattison, 1996; Seary and Richards, 2000).

Crosstables are visualized using panigrams. An example is presented in Figure 14. The tables and panigrams are used to explore the association within networks (out- and in-degrees, i.e., sender and receiver effects) or the association between networks and an attribute. In Figure 14 a panigram of discipline and incoming links (receiver effects) of the first observation of the acquaintanceship network is presented. The links can take the values 0 to 4 ('have not met' to 'close friends'),

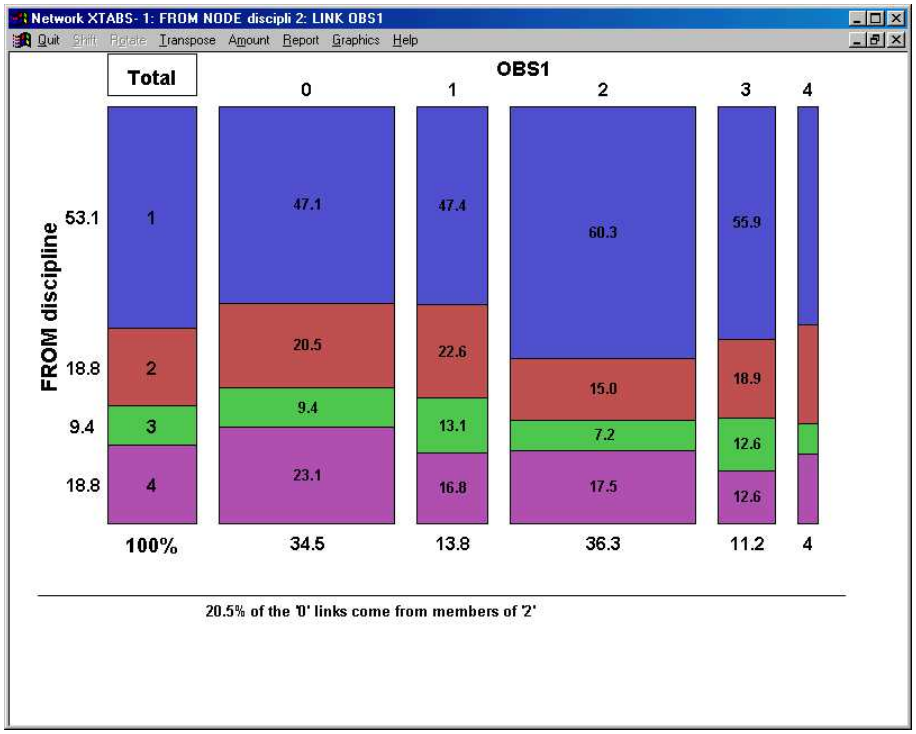

Figure 14: MultiNet user interface presenting the panigram of discipline and the first observation of the EIES acquaintanceship network (incoming links).

discipline the values 1 to 4. Interactive help is available, explaining the meaning of the 'cells'. For example, 20.5% of the links with value 0 ('have not met') come from actors with discipline value 2 (anthropology) and 9.4% from members of discipline 3 (statistics). The χ^2-statistic equals 23.4 ($df = 12$, $p < 0.05$), which indicates a significant association between the variables (with sociologists receiving more friendship choices). The association between discipline and outgoing links (sender effects) of the first acquaintanceship network is also significant (results not reported here).

In Table 8 the results of two analyses of variance for the first observation of the acquaintanceship data are presented. The independent grouping-variable is the nature of the relation between two actors (sender and receiver) at the first time point. The dependent variables are the mean citation rates of the senders and the receivers. A graphical display of the citations per relation-group is produced as well (not shown). From the table it follows that there is a significant difference between the mean citation rates of receivers, but not between the senders. The means show that receivers are on average less often cited in the 'did not know' relation-group.

The analyses differ from those performed by UCINET, where only one row or column of the adjacency matrix is used in an ANOVA. By using all links in the

Table 8: ANOVA results for the EIES acquaintanceship data (first observation) obtained with MultiNet.

Relation	n	Mean citation of	
		receiver	sender
Did not know	342	15.0	20.6
Had not met	137	29.8	23.2
Had met	360	27.8	23.5
Was friend	111	22.5	23.3
Was close friend	42	24.1	34.6
ANOVA: p-value		< 0.01	> 0.10

analyses MultiNet assumes independence between all relations, whereas UCINET assumes independence between actors. The former will generally not be the case, and the user should therefore be very cautious interpreting the results.

MultiNet comprises PSPAR, an earlier program by Seary (1999), designed to fit p^* models to large networks by pseudo-likelihood based on sparse methods. The method fits the model parameters to triad statistics selected by the user. Blockparameters can be obtained by fitting models of which the blockstructure is defined by one or more (categorical) actor attributes. Figure 15 shows the p^* graphic display window obtained for the EIES acquaintanceship data (first observation). The effects included in the model are density, reciprocity, transitivity, and the block parameter 'choice within blocks' with the blocks defined by the attribute discipline. All estimates were significant and are reported in Table 10 in Section 2.7, together with the estimates obtained in StOCNET. The p^* graphic display shows the adjacency matrix with correctly predicted links (green), the false negatives (blue), and false positives (red).

2.7 StOCNET

StOCNET (Version 1.4; Boer, Huisman, Snijders, and Zeggelink, 2003) is an open software system, in a Windows environment, for advanced statistical analysis of social networks. It provides a platform to make available a number of statistical methods, presented in separate modules, and allows new routines to be easily implemented (Huisman and Van Duijn, 2003). The program is freeware and can be downloaded from the StOCNET website. A user's manual describing the operation of the StOCNET system is available, as well as a programmer's manual, which describes the main procedures and functionalities of the system to facilitate the in-

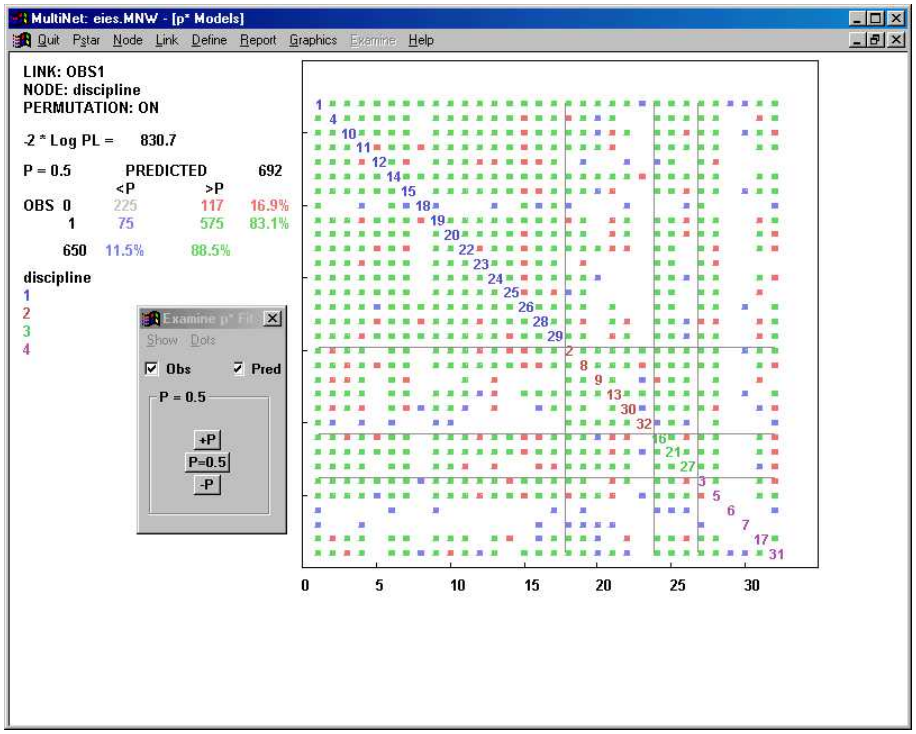

Figure 14: MultiNet user interface with the p graphic display window showing the results for the EIES acquaintanceship data (first observation).*

clusion of new statistical methods. On the website user's manuals of all modules and programmer's manuals together with source codes of some modules can be found.

Analyses take place within *sessions*. A session consists of (a cyclical process of) five steps: 1) data definition, 2) transformation, 3) selection, 4) model specification and analysis, and 5) inspection of results. A typical StOCNET window is presented in Figure 16 showing the user interface for the module SIENA for longitudinal network data (see Snijders, 2004).

Data entry and manipulation

Network data have to be presented as adjacency matrices saved in ASCII format with the values separated by blanks. Actor attributes have to be presented as ASCII files as well, with blanks separating the values. Data sets are saved as ASCII data files and StOCNET sessions are saved in session files (SNS). Export functions to MultiNet, NetMiner, Pajek, and STRUCTURE are available. StOCNET contains a recoding, symmetrizing, and selection option. Missing values can be specified, both for network data and attributes. The handling of missing observations depends on the statistical model selected in the modeling step.

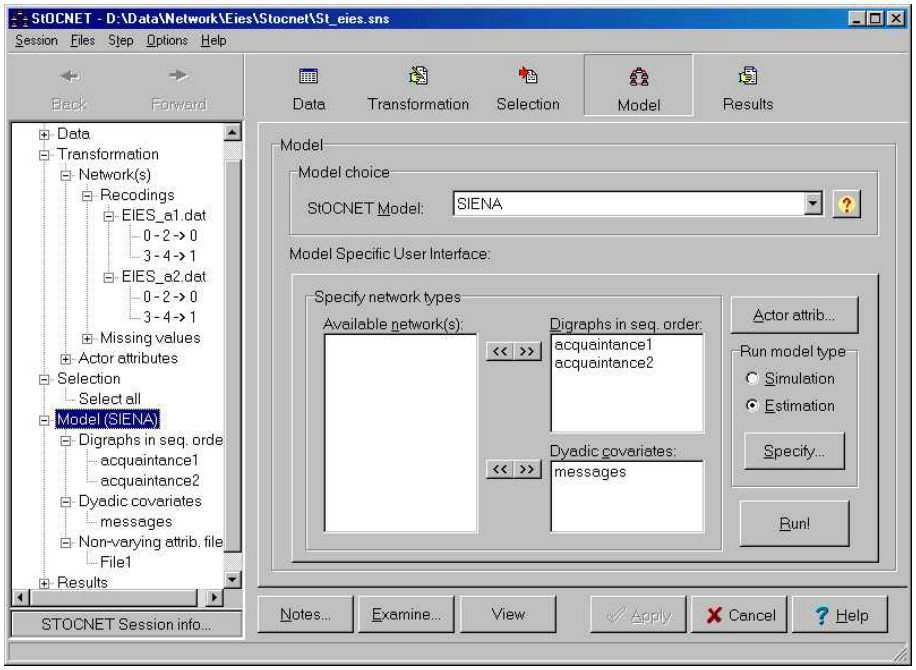

Figure 16: StOCNET user interface of the SIENA module for longitudinal analysis of the EIES acquaintanceship data.

Visualization techniques

StOCNET does not contain procedures for the visualization of networks.

Descriptive methods

In four of the five steps in a StOCNET session, descriptive analyses of the available data can be performed by clicking the *Examine* button. This button is available in the main windows of all steps (see Figure 16), except in the last step (i.e., inspection of results). Degree variances, index of heterogeneity, dyad and triad census, degree of reciprocity and transitivity, and segmentation are some of the network statistics that are calculated for separate network data sets. For longitudinal analysis of networks, change statistics are calculated.

Procedure-based analysis

There are no procedure-based routines available in StOCNET.

Statistical modeling

StOCNET contains five statistical modules: 1) BLOCKS, for stochastical blockmodeling (Nowicki and Snijders, 2001); 2) ULTRAS, for estimating latent transitive structures using ultrametrics (Schweinberger and Snijders, 2003); 3) P2, for fitting the

exponential random graph model p_2 (Van Duijn, Snijders, and Zijlstra, 2004); 4) SIENA, for the analysis of longitudinal network data (Snijders, 2001, 2004); and 5) ZO for determining probability distributions of statistics of random graphs based on the $\mathcal{U} \mid X_{i+}, X_{+j}$ and $\mathcal{U} \mid X_{i+}, X_{+j}, M$ distributions (Snijders, 1991; Molloy and Reed, 1995). Other exponential random graph models can also be fitted in StOCNET: the p_1 model (Holland and Leinhard, 1981) as *Examine* option in P2, and the p^* model (Wasserman and Pattison, 1996) in SIENA where MCMC estimation with the Robbins-Monro algorithm is applied to a single network observation instead of repeated observations (Snijders, 2002; Snijders and Van Duijn, 2002),

The results of applying modules SIENA and P2 to the EIES data are shown in Tables 9 through 11. Figure 16 shows the model-specific user interface for the SIENA module. Both time points of the acquaintanceship networks are analyzed with the dynamic actor-oriented model of Snijders (2001, 2004). The first observation of the network is analyzed with the p_2 model and with the p^* model. For all models the dichotomized data were used.

The estimated effects of the SIENA model are presented in Table 9 (see also Snijders, 2004, for a discussion on the interpretation of the parameters). The rate parameter shows that on average the actors made about 2.5 relationship changes in the period between the observations. In the evolution of the acquaintanceship network, a clear reciprocity effect and a transitivity-type effect are present, the latter being specified as a tendency away from indirect relations. There is also a tendency for popular others (i.e., others who receive many choices). No significant attribute effects were found[3].

In the SIENA module, MCMC estimation with the Robbins-Monro algorithm of p^* model is implemented. As Snijders (2002) notes, both the pseudo-likelihood estimation (as implemented in MultiNet but which can also be done with standard software for logistic regression), and MCMC estimation using the Geyer and Thompson (1992) method as used by Crouch, Wasserman, and Trachtenberg (1998) and Corander, Dahmström and Dahmström (1998) are unsatisfactory. Since it is not a function of the complete statistic, the pseudo-likelihood estimate has unknown properties. This leads in any case to underestimation of the standard errors of the estimates. MCMC estimation is not satisfactory either, because the simulation of random graph distributions turns out to be a complicated matter due to bimodality, which leads to convergence problems. See Wasserman and Robins (2004) for

[3]Snijders and Van Duijn (1997) analyzed another dichotomization of the EIES data: not knowing/having met vs. having met/being friends. They found different effects (especially effects of the attribute citation) influencing the evolution of the 'meeting' network.

Table 9: Estimated (significant) effects for the evolution of the EIES acquaintanceship data obtained with the SIENA module in StOCNET.

Effect	Est.	S.E.
Constant change rate	2.47	
Density (out-degree)	-1.80	0.52
Reciprocity	2.06	0.39
Indirect relations	-0.27	0.13
Popularity	6.40	1.05

an extended discussion of pseudo-likelihood and MCMC estimation of p^* models. Snijders (2002) and Snijders and Van Duijn (2002) propose several alternative simulation methods to improve convergence, based on single relations, dyads and triplets, using Gibbs or Metropolis Hastings steps, making small or large updates (through inversion steps), and/or on conditional simulation (fixing the number of relations, or the in- and outdegrees and thus limiting the outcome space). More developments in this area are expected.

In Table 10 the results are given of fitting the p^* model to the first observation of the EIES data. Maximum pseudo-likelihood were obtained with MultiNet (see Section 2.6). MCMC estimates with the Robbins-Monro algorithm were obtained with the SIENA model in StOCNET. It was not possible to estimate the p* model unconditionally. As soon as the transitivity effect was added to the model, no convergence was obtained. It was possible to obtain estimates of the p^* model conditional on the number of ties, which means that no density effect is estimated. The convergence of the model with the dissimilarity (or block) effect of discipline was unsatisfactory as well, which shows in the large standard error for this effect, given in Table 10. The convergence of the conditional model with only reciprocity and transitivity was acceptable. The estimates for reciprocity and even their standard errors are similar for pseudo-likelihood and MCMC. The estimates for transitivity and the similarity (block) effect of discipline are quite different.

The p_2 model is a random effects model with the dyadic ties as the dependent variable (Van Duijn, Snijders, and Zijlstra, 2004). The sender and receiver parameters, fixed in the p_1 model, are regressed on available – categorical or continuous – nodal attributes (*actor covariates*). If no attributes are available, the regression model reduces to random sender and receiver effects. Likewise, the density and reciprocity parameters, can be linked to other available networks (*dyadic covari-*

Table 10: Pseudo-likelihood estimates obtained with MultiNet and Markov Chain Monte Carlo Robbins Monro p^*-estimates obtained with StOCNET for the EIES acquaintanceship data (first observation).

	Pseudo-likelihood		MCMC Robbins Monro conditional on ties			
Effect	Est.	S.E.	Est.	S.E.	Est.	S.E.
Density	-3.61	0.22				
Reciprocity	1.94	0.23	2.15	0.31	2.20	0.30
Transitivity	0.32	0.036	0.17	0.01	0.17	0.01
Dissimilarity discipline	0.55	0.22	0.25	3.32		

ates), without a random component. Dyadic covariates can also be computed from the nodal attributes, for instance by taking their difference or absolute difference, which are both standard options in the P2 module. Thus, dissimilarity matrices are created. If the nodal attribute is categorical, one can construct dichotomous (dis)similarity matrices, comparable to the block-parameters in MultiNet. Unlike the p^* model, the p_2 model does not contain network effects other than reciprocity.

Table 11 contains the parameter estimates for the fixed and random effects of the model. Dissimilarity with respect to citation has a significant negative effect on density, in two ways: expressed as the absolute difference of the actors' number of citations, and expressed as the simple difference of the actors' number of citations. The first effect implies that the probability of an acquaintance relation decreases the more actors differ with respect to their citations; the second indicates a directional effect that actors whose citations are high tend to choose less often actors whose citations are low. The second effect can be viewed as a refinement of the positive sender effect for citation which indicates that the probability of an outgoing acquaintanceship relation (irrespective of the receiver attributes) increases with the number of citations. The positive effect of similarity with respect to discipline indicates that actors tend to choose more within their own discpline group, which effect was also found for p^* model in MultiNet. There is a general reciprocity effect, but this is not differentiated according to dyadic attributes.

Analysis of the first observation of the acquaintanceship data with the stochastic blockmodeling routine BLOCKS (results not shown here) reveals some classes of stochastically equivalent actors (i.e., they have the same probability distribution of their relations to other actors). The fit of the models, like the blockmodeling results obtained with Pajek, however, is not very good. The blocks found do not coincide

Table 11: p_2-estimates for the EIES acquaintanceship data (first observation; only significant effects) obtained with StOCNET.

Effect	Parameter	Est.	S.E.
Density	μ	-2.79	0.29
	Dissimilarity citation (abs. diff.)	-0.017	0.005
	Dissimilarity citation (diff.)	-0.013	0.003
	Similarity discipline	0.64	0.18
Reciprocity	ρ	2.36	0.32
Sender	Variance σ_A^2	1.01	0.24
	Citations	0.028	0.0082
Receiver	Variance σ_B^2	0.98	0.23
Sender-receiver	Covariance σ_{AB}	-0.40	0.18

with the partitions based on actor attributes.

ULTRAS, aimed at finding groups according to a latent structure based on ultrametrics (i.e., triadic distances between actors), was also applied to the first network of the valued acquaintanceship network, using a Poisson distribution for the network ties. The groups can be presented as a tree, branching further with larger distances. The number of ultrametrics needs to be determined using a Bayesian model selection process. Here we found a solution with 7 ultrametrics that was not completely stable (results not shown here). Some of the cliques found in UCINET (see Table 3) were also found in the ULTRAS analysis.

More examples of statistical analyses with the StOCNET modules are given by Huisman and Van Duijn (2003).

3 Social Network Software - other packages and routines

In this section other available software for social network analysis is briefly discussed, without illustrations. We distinguish general packages and five types of special purpose packages: for identification of subgroups, for knowledge networks, for hidden populations, for kinship networks, and for statistical testing. Only the most important features are mentioned. The final subsection treats routines and utilities for the analysis of social networks developed to be used in a general statistical software package or in a programming language.

3.1 General packages

In this section seven general packages are mentioned (in alphabetical order). One of them, GRADAP, is well-known since it has been around for more than fifteen years. We consider GRADAP, although outdated, worth mentioning because it contains routines and statistics not available in packages like UCINET or Pajek. The other programs are quite new and regularly updated. We distinguish two kinds of general programs: programs intended for data analysis that have visualization options (Agna and SNAFU), and programs intended for network visualization that feature analysis procedures (so-called visual exploration; InFlow, NetDraw, NetVis, and visone).

Agna (Version 2.0.7; Benta, 2003)
The platform-independent application Agna (Applied Graph & Network Analysis) is designed for social network analysis and sequential analysis. Sequential analysis deals with behavioral chains, which are modeled in order to find rules that govern the inner structure of behavior. This inner structure is represented by dyad transitions. Agna is designed to study communication relations in groups, kinship relations and the structure of animal behavior. The analysis methods include general descriptives, shortest path analyses, and centrality and sociometric coefficients. The program has ample visualization options.

GRADAP (Version 2.0; Sprenger and Stokman, 1989)
The software package GRADAP (GRAph Definition and Analysis Package), an environment for analyzing graphs and networks, is an organized set of programs explicitly developed to analyze network data represented as graphs, and includes a wide range of cohesive subgroup and centrality methods, and models for the distribution of in- and outdegrees. It is only available as a DOS application and will not be updated to a Windows environment.

SNAFU–MacOS (Version 2.0; Hagen, 2003)
SNAFU (Social Network Analysis Functional Utility) is a general-purpose network analysis tool for MacOS systems, which is distributed "as-is" with no warranties or support beyond reasonable requests. It imports and exports to UCINET, InFlow and some visualization programs, and is generally oriented toward connected graphs of a few hundred nodes. It includes network editing features, descriptive techniques, some matrix algebra, visualization tools, and multiple example data sets.

3.1.1 Visual exploration

InFlow (Version 3.0; Krebs, 2002)
InFlow is a commercial software package for network mapping, especially aimed at organizational applications. It was originally developed for Macintosh, but has been updated to Windows. Interactively, it carries out network analysis and network visualization simultaneously (with ample graphical export options). Thus, it is possible to express changes in the network directly in terms of network measures. It features a number of descriptive and procedure-based routines, but no statistical methods.

NetDraw (Version 1.0.0.21; Borgatti, 2002)
NetDraw is a program for drawing networks. It is a free, stand-alone program, but is also distributed together with UCINET. This reflects its close relation with UCINET: it can be executed within UCINET and reads UCINET files natively without the need for import and export functions.

NetDraw uses several different algorithms for displaying nodes in a two-dimensional space, using a circle layout or layouts obtained with multidimensional scaling or spring embedding. These layouts are based on geodesic distance (see Freeman, 2004, and Sections 2.3 and 2.4). It has tools for grouping and automatically recoloring, resizing, or reshaping of nodes, ties and labels to represent these groups. Graphs can be rotated, flipped, resized, and saved in several formats, amongst others, as bitmap (BMP) and JPEG files. Export functions to Mage and Pajek are available. NetDraw includes some analysis procedures, for example, identification of isolates, components, or k-cores, the results of which are displayed graphically.

NetVis (Version 2.0; Cummings, 2003)
With advances in open source software, social network researchers have new opportunities for analyzing and visualizing network data. One such possibility is the NetVis module, a web-based tool to analyze and visualize social networks using data from CSV files, online surveys, and dispersed teams. It is available online, where data can be uploaded, analyzed, and output and (3D) graphs are generated, which can be downloaded. For all algorithms, the source code is available.

visone (Version 1.0beta1; Brandes and Wagner, 2003)
The visone project team is developing models and algorithms to integrate and advance the analysis and visualization of social networks. It facilitates the visual exploration of network data by experts as well as novices. Its origins lie in an interdisciplinary cooperation with researchers from mathematics, computer and information science, and political science. visone is a research platform that is not

intended to become a standard tool and is in development and therefore subject to change.

visone contains several different algorithms for drawing graphs and representing results of analysis. It uses spring embedders, spectral layouts, layered layouts, and radial layouts to present networks. It has many options to improve the (layout of the) graphs and visualizations can be exported in SVG or postscript format. The analysis methods include local measures (degrees), distance measures (e.g., betweenness, closeness), and feedback measures (e.g., status, eigenvector, authority).

3.2 Special purpose packages

In this section we discuss nine packages, divided into five specific areas of social network analysis: identification of subgroups, knowledge networks, hidden populations, kinship networks, and statistical testing.

3.2.1 Identification of subgroups

KliqFinder for Windows (Version 0.05; Frank, 2003)
KliqFinder is the Windows version of the Fortran and SAS-based program KliqueFinder (adapted for Windows by Richard Congdon). It is aimed at identifying cohesive subgroups and produces a so-called crystallized subgroup representing the subgroups and their relations within and between the clusters. The subgroups are identified in an iterative algorithm maximizing the log odds of a tie within the group (Frank, 1995, 1996). For the graphical representation of the subgroups, the program SAS is called in KliqFinder,

NEGOPY (Version 4.30, Richards, 1995)
The main purpose of the DOS-based program NEGOPY is to find cohesive subgroups. To this end, it defines a number of role categories, such as groups, isolates, participants on the basis of their linkage with other nodes, more or less similar to the p-cliques discussed in the section on Pajek. The reader is referred to the manual for exact definitions and how these definitions may be adapted, and short references in Wasserman and Faust, 1994. The result is a discrete categorization of the nodes in the network. NEGOPY uses partial decomposition methods, to approximate eigendecomposition methods unfeasible for large networks, whereas MultiNet calculates exact eigenpairs (see Richards and Seary, 2000).

3.2.2 Knowledge networks

Blanche (Version 4.6.4, Hyatt, Contractor, Ferrone, Han, Hsu, Kochhar, Palazzo,

Su, and Willard, 2002)
Since knowledge of the knowledge network causes changes and further evolution of the knowledge network, the program Blanche was designed to create and simulate models of network dynamics. It uses a system of nodes and links, and (non-linear difference) equations that describe how the strengths of links and the attributes of nodes change over time. It consists of three modules to create models, to create data, and to run the model and output the results, respectively.

Iknow (Contractor, O'Keefe, and Jones, 1997)
Iknow is specialized Java-based software to collect and present data on communication and knowledge networks. In this kind of knowledge networks the nodes are actors (individuals or organizations) and the links the knowledge or information they have about characteristics of the other actors. These characteristics typically concern knowledge of various domains. The software either collects interactively or automatically, from the web, information about the network actors and their links, and then presents this information in various ways.

Referral Web (Kautz, Selman, and Shah, 1997)
This Java-based software was developed in the area of artificial intelligence. It is aimed at research communities and helps users, that is, researchers, explore the social networks in which they participate (such that they can quickly find short referral chains between themselves and experts on arbitrary topics). It either shows the neighborhood of a specified researcher (the node), the path to some specified other node or to an -unknown- expert on a specified topic. The program operates by automatically generating representations of social networks based on evidence gathered from publicly available documents on the internet. For instance, nodes who are found to be co-authors, are linked. The definition of association on which the linking is based may be difficult, and therefore the resulting networks may be incorrect and/or incomplete.

3.2.3 Hidden populations

SNOWBALL (Snijders, 1994)
SNOWBALL is a DOS program for the estimation of the size of a hidden population from a one-wave snowball sample, implementing the estimates proposed by Frank and Snijders (1994). Snowball sampling is a term used for sampling procedures that allow the sampled units to provide information not only about themselves but also about other units. This is advantageous when rare properties are of interest.

SocioMetrica LinkAlyzer (Version 2.1; MDLogix, 2002)
SocioMetrica LinkAlyzer is aimed at constructing a network from data obtained from (a sample from) a difficult or hidden population. The program was developed to investigate HIV links between drug users. The typical problem is that many actors in the network are difficult to identify because of their use of, possibly various, nicknames. To construct a network from the data that are usually collected as egocentric networks (by interviewers) it is necessary to find out which nominees are the same. The software tries to identify these actors by matching them on various possible attributes such as gender, age, appearance, location(s), etc. Although the software is commercial (available in two versions for smaller and larger networks), a demo version can be downloaded from the web. It is possible to work with example data or with other data (containing not more than 50 actors) and thus to get an impression of the features of the program that also contains some standard network measures like centrality. It has import and export possibilities to common other packages such as UCINET, SPSS, and Excel.

3.2.4 Kinship networks

PGRAPH (Version 2.7 for Windows; White and Skyhorse, 1997)
PGRAPH is software for kinship and marriage networks, where P stands for parent or parental. On the webpage, the authors call it a "toolkit for structural analysis of genealogical data and kinship and marriage data". The p-graph is a concept for a representation of networks, where the vertices are not individuals but intersections between individuals (as in marriage), or between groups and individuals, where graph-theoretic cycles and blocks are relevant units of analysis (see also Harary and White, 2001). Networks can be analyzed using p-graphs with either the PGRAPH package, or with Pajek software in combination with some utility programs that pre-analyze the data and convert it to Pajek input format (White, Batagelj, and Mrvar, 1999).

3.2.5 Statistical testing

PermNet (Version 0.94; Tsuji, 1997)
The program PermNet (PERMutation NETworks) contains a set of permutation tests for social network data. It provides symmetry tests, transitivity tests for real-valued data, and a triad census test for binary data (cf. NetMiner and the module ZO of the StOCNET software).

3.3 Utilities and routines

We mention five software toolkits with utilities available for programming, either in general software (Excel, Gauss, R/S) or in a common programming language (Java). The routines developed for Gauss and especially those developed for R are the most general and complete. The Excel routines are specifically aimed at ethological applications, and the Java-based libraries of procedures are largely aimed at visualization.

Next to these routines some other data preparation utilities are available. Some of them (ADJ2NEG and FREEFIX) were already mentioned in section 2.6. Another, PREPSTAR (Version 1.0; Crouch and Wasserman, 1998) has been developed to perform p^* analyses in SPSS or SAS.

JUNG–Java Library (Version 1.0; White, O'Madadhain, Fisher, and Boey, 2003)
The Java Universal Network/Graph (JUNG) framework is a software library that provides a common and extendible language for the modeling, analysis, and visualization of data that can be represented as a graph or network. JUNG supports a variety of representations of graphs (e.g., directed, undirected) and the current version includes algorithms for clustering, decomposition, random graph generation, statistical analysis, and calculating of network distances, flows, and importance measures. It also provides a visualization framework to construct tools for data exploration.

MatMan–Microsoft Excel (Version 1.0 for windows; Noldus, 2001)
An add-in for Microsoft Excel, MatMan is aimed at performing specific matrix manipulations, common in ethological research, for sociomatrices, behavioral profile data, and transition matrices. Furthermore, social dominance and correlation analyses can be performed.

SNA–R-routines for S (Version 0.41; Butts, 2002)
This collection of routines to be used in R or S ('Carter's archive'), contains many well-documented procedures for performing various kinds of social network analyses ranging from general analyses such as mutuality, betweenness or centrality to specific analyses such as QAP and p^* analyses, or blockmodeling. It also contains visualization routines. The R routines can be called from the program Pajek (see Section 2.3).

SNAP–GAUSS (Version 2.5; Friedkin, 2001)
Like SNA, a collection of network analysis routines that include procedures for calculating many graph theoretical properties of graphs and nodes, and for fitting social influence models.

yFiles–Java library (Version 2.1; *y*Works, 2003)
The Java class package yFiles provides efficient and effective visualization algorithms. It is a class library for viewing, editing, optimizing, layouting, and animating graphs. Since it is written in Java, yFiles is fit for platform independent applications. It has a graph viewer and supports many functionalities, like labels for nodes and edges or multiple views of a graph. Furthermore, yFiles has some routines for exploration and descriptive analysis of networks (e.g., bipartitions, shortest paths, transitivity).

4 Recommendations

We conclude this section with a summary of the packages presented in Section 2. We scored the software at 1) functionality, using the earlier-defined categorization of procedures: data manipulation (data entry was found not to be a problem for any program), network Visualization, descriptive methods, procedure-based methods, and statistical methods; 2) support: the availability of a manual and a online help-function; and 3) user-friendliness. The scores are given in Table 12. A + is used to indicate that it is good (or at least sufficient), ++ that it is very good or strong, a − that it has shortcomings, a 0 that it is lacking, and a +− that it is undecided (having both good and bad parts). We will explain the scores, especially the negative ones, further below.

Obviously, we try to present an objective, substantiated view, but we admit that we cannot give a completely unbiased opinion. We also stress that it is impossible to make a fair comparison between the packages, because their objectives are different, which leads to different functionalities. For instance, the aim of StOCNET is not to compete with but to be an addition to existing software, and therefore it contains no procedure based methods. Likewise, STRUCTURE is too old to offer any visualization.

Therefore, we advise reading the table vertically as well: for instance, if one is looking for a package with the primary aim to obtain many descriptive network measures, UCINET or NetMiner would be a good candidate. On the other hand, if network visualization is an important objective, Pajek and NetMiner are competing packages, where MultiNet and UCINET (with NetDraw) also give the opportunity for visual exploration.

In two of the six programs, MultiNet and StOCNET, data manipulation obtained the score +− because they contain relatively few options. STRUCTURE received a negative score, because it contains hardly any options for data manipulation.

StOCNET does not have any visualization options, but this is compensated via

Table 12: Scores for the packages presented in Section 2.

	Functionality					Support		User-
	Data	Visual.	Descr.	Proc.	Stat.	Manual	Help	friendliness
MultiNet	+−	+	+−	+	+−	+−	++	+
NetMiner	++	++	++	++	+−	+	+	++
Pajek	+	++	+	++	0	−	0	+−
StOCNET	+−	0	+−	0	++	+	+	+
STRUCTURE	−	0	+−	++	+	++	0	+−
UCINET	++	+[1]	++	++	+−	+	+	+

[1] The program NetDraw for network visualization is distributed with UCINET

export possibilities to NetMiner and Pajek that score very well with respect to visualization.

The scores for the descriptive, procedure-based, and statistical methods, are indicative of the number of different features. The descriptive methods are rather sparse in MultiNet, StOCNET, and STRUCTURE. They are most comprehensive in NetMiner and UCINET. These programs contain many procedure-based methods as well, whereas STRUCTURE has some unique procedures. StOCNET does not contain any procedure-based methods, but has many statistical methods, more, and more advanced, than the other programs. The statistical methods in Pajek are so limited that they score a 0 (although there is the possibility to call statistical routines in R). The statistical methods in STRUCTURE are limited as well, but exclusive. The other three programs do contain a number of – sometimes exclusive – statistical methods, but they are presented uncritically whereas some warning would definitely be warranted for the ANOVA procedures, estimation of the p^* model, and QAP regression.

In our opinion, the manual of STRUCTURE is the best, since it contains both good practical information and a theoretical background. The completeness of the manual shows that it was developed in the pre-internet era, and that it was – and still is – used for educational purposes. MultiNet's manual is, at the time of writing, incomplete, but the program has good, interactive, online help. Pajek's manual is so little instructive, that we scored it negatively. Without additional information, provided to us via the forthcoming book by De Nooy, Mrvar and Batagelj (2002), it is very difficult to use Pajek to its full extent. The fact that Pajek does not have an online help function is a further drawback.

We see some connection between the support offered in the various packages and their authors and development period. Except for NetMiner, the developers of all packages are or were rather active in the social network analysis community.

Authors with a social science background (UCINET, STRUCTURE) are very able and experienced in communicating their methods and incorporating them in social theories. Packages with authors with a mixed background (both social and mathematical/computational; MultiNet, StOCNET) offer less social theory. The more mathematical orientation of the authors shows in Pajek, where the user is supposed to know what s/he wants. The most commercial – non-academic – developers of NetMiner have been able to profit from the experience of previously developed software to join completeness and user-friendliness.

The insufficient manual and lack of online help is the reason of the +− score for Pajek's user-friendliness. STRUCTURE obtains a +− score because of its age. We find that it would be worthwhile to upgrade STRUCTURE or to incorporate it in one of the existing programs. The same applies to GRADAP. With respect to user-friendliness, NetMiner stands out, because of its interface where visualization, data, and procedures are integrated.

It remains, however, hard to compare the different packages, as we already pointed out at the beginning of this section. We leave it to the reader of this chapter to decide which software to use for the social network analysis s/he wishes to do.

5 References

Table 12 gives the URLs of all reviewed programs and software toolkits. Most packages, or information on the software, can also be obtained via the software pages of the INSNA website at `http://www.sfu.ca/~insna/INSNA/soft_inf.html`.

Batagelj, V., and Mrvar, A. (1998). Pajek: A program for large network analysis. *Connections. 21*, 47–57.

Batagelj, V., and Mrvar, A. (2003a). *Pajek: Package for large networks. Version 0.92.* Ljubljana: University of Ljubljana.

Batagelj, V., and Mrvar, A. (2003b). Pajek. Analysis and visualization of large networks. In Jünger, M., and Mutzel, P. (eds.), *Graph Drawing Software*, pages 77–144. New York: Springer.

Benta, I.M. (2003). *Agna. Version 2.0.7.* Cork: University College Cork, Ireland.

Bock, R.D., and Husain, S.Z. (1950). An adaptation of Holzinger's B-coefficients for the analysis of sociometric data. *Sociometry. 13*, 146–153.

Boer, P., Huisman, M., Snijders, T.A.B., and Zeggelink, E.P.H. (2003). *StOCNET: An open software system for the advanced statistical analysis of social networks. Version 1.4.* Groningen: *Pro*GAMMA/ICS, University of Groningen.

Bonacich, P. (1972). Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology. 2*, 113–120.

Table 12: URLs of all reviewed programs and software toolkits.

Program	Ver.	URL
Agna	2.0.7	http://www.geocities.com/imbenta/agna/index.htm
Blanche	4.6.4	http://www.spcomm.uiuc.edu/Projects/TECLAB/BLANCHE/
FATCAT	4.2	http://www.sfu.ca/~richards/Pages/fatcat.htm
GRADAP	2.0	http://www.assess.com/Software/GRADAP.htm
Iknow	–	http://www.spcomm.uiuc.edu/Projects/TECLAB/IKNOW/
InFlow	3.0	http://www.orgnet.com/
KliqFinder	0.05	http://www.msu.edu/~kenfrank/software.htm
MultiNet	4.24	http://www.sfu.ca/~richards/Multinet/Pages/multinet.htm
NEGOPY	4.30	http://www.sfu.ca/~richards/Pages/negopy4.html
NetDraw	1.0	http://www.analytictech.com/downloadnd.htm
NetMiner II	2.3.0	http://www.netminer.com/NetMiner/home_01.jsp
NetVis	2.0	http://www.netvis.org/
Pajek	0.94	http://vlado.fmf.uni-lj.si/pub/networks/pajek/default.htm
PermNet	0.94	http://www.meijigakuin.ac.jp/~rtsuji/en/software.html
PGRAPH	2.7	http://eclectic.ss.uci.edu/~drwhite/pgraph/
ReferralWeb	2.0	http://www.cs.washington.edu/homes/kautz/referralweb/
SM LinkAlyzer	2.1	http://www.md-logic.com/id142.htm
SNAFU	2.0	http://innovationinsight.com/networks.html
Snowball	–	http://stat.gamma.rug.nl/snijders/socnet.htm
StOCNET	1.4	http://stat.gamma.rug.nl/stocnet/
STRUCTURE	4.2	http://gsbwww.uchicago.edu/fac/ronald.burt/teaching/
UCINET	6.05	http://www.analytictech.com/ucinet_5_description.htm
visone	1.0b1	http://www.visone.de/
JUNG	1.0	http://jung.sourceforge.net/index.html
MatMan	1.0	http://www.noldus.com/products/index.html?matman/index
PREPSTAR	1.0	http://kentucky.psych.uiuc.edu/pstar/index.html
SNA	0.41	http://legba.casos.ri.cmu.edu/R.stuff/
SNAP	2.5	http://www.soc.ucsb.edu/faculty/friedkin/Software/Software.htm
yFiles	2.1	http://www.yworks.de/en/products_yfiles_about.htm
KrackPlot	3.0	http://www.andrew.cmu.edu/~krack/
Mage	2.1	http://kinemage.biochem.duke.edu/kinemage/kinemage.html

Borgatti, S.P. (2002). *NetDraw 1.0: Network visualization software. Version 1.0.0.21.* Harvard: Analytic Technologies.

Borgatti, S.P., Everett, M.G., and Freeman, L.C. (1999). *UCINET 5 for Windows: Software for social network analysis. Version 1.00.* Natick: Analytic Technologies.

Borgatti, S.P., Everett, M.G., and Freeman, L.C. (2002). *UCINET 6 for Windows: Software for social network analysis.* Harvard: Analytic Technologies.

Brandes, U., and Wagner, D. (2003). visone. Analysis and visualization of social networks. In Jünger, M., and Mutzel, P. (eds.), *Graph Drawing Software*, pages 321–340. New York: Springer.

Burt, R.S. (1976). Positions in networks. *Social Forces. 55*, 93–122.

Burt, R.S. (1991). *STRUCTURE. Version 4.2.* New York: Columbia University.

Butts, C.T. (2002). *Package SNA: Tools for social network analysis. Version 0.41.* Irvine: University of California.

Contractor N.S., O'Keefe, B.J., and Jones, P.M. (1997). *Iknow: Inquiring knowledge networks on the web.* Urbana-Champaign: University of Illinois.

Crouch, B., and Wasserman, S. (1998). A practical guide to fitting p^* social network models. *Connections. 21*:87-101.

Crouch, B., Wasserman, S., and Trachtenberg, F. (1998). Markov Chain Monte Carlo maximum likelihood estimation for p^* social network models. Paper presented at the Sunbelt XVIII and Fifth European International Social Networks Conference, Sitges (Spain), May 28-31, 1998.

Cyram (2003). *Cyram NetMiner II. Version 2.0.5.* Seoul: Cyram Co., Ltd.

Cummings, J.N. (2003). *NetVis module - Dynamic visualization of social networks.* Cambridge: Massachusetts Institute of Technology.

Corander, J., Dahmström, K., and Dahmström, P. (1998). Maximum likelihood estimation for Markov graphs. Research Report 1998:8. Stockholm: Department of Statistics, University of Stockholm.

De Nooy, W., Mrvar, A., and Batagelj, V. (2002). *Exploratory social network analysis with Pajek.* Cambridge: Cambridge University Press. forthcoming.

Doreian, P. (1980). Linear models with spatially distributed data. *Sociological Methods & Research. 9*, 29–60.

Doreian, P., Batagelj, V., and Ferligoj, A. (2004). Positional analysis of sociometric data. In Carrington P.J., Scott, J., and Wasserman, S. (eds.), *Models and methods in social network analysis.* Cambridge: Cambridge University Press.

Everett, M.G., and Borgatti, S.P. (2004). Extending centrality. In Carrington, P.J., Scott, J., and Wasserman, S. (eds.), *Models and methods in social network analysis.* Cambridge: Cambridge University Press.

Everton, S.F. (2002). A guide for the visually perplexed: visually representing social networks. Stanford: Stanford University.
`http://www.stanford.edu/group/esrg/siliconvalley/sivnap.html`

Frank, K.A. (1995). Identifying cohesive subgroups. *Social Networks. 17*:27-56.

Frank, K.A. (1996). Mapping interactions within and between cohesive subgroups. *Social Networks. 18*:93-119.

Frank, K.A. (2003). *KliqFinder for Windows. Version 0.05.* East Lansing, MI: Michigan State University.

Frank, O., and Snijders., T.A.B. (1994). Estimating the size of hidden populations using snowball sampling. *Journal of Official Statistics. 10*, 53–67.

Freeman, L.C. (1979). Centrality in social networks: conceptual clarification. *Social Networks. 1*, 35–41.

Freeman, L.C. (2004). Graphical techniques for exploring social network data. In Carrington, P.J., Scott, J., and Wasserman, s. (eds.), *Models and methods in social network analysis*. Cambridge: Cambridge University Press.

Freeman, S.C., and Freeman L.C. (1979). The networks network: A study of the impact of a new communications medium on sociometric structure. Social Science Research Reports No. 46. Irvine, CA: University of California.

Friedkin, N.E. (2001). *SNAP: Social network analysis procedures for GAUSS. Version 2.5*. Maple Valley, WA: Aptech Systems, Inc.

Geyer, C.J., and Thompson, E.A. (1992). Constrained Monte Carlo maximum likelihood for dependent data. *Journal of the Royal Statistical Society B. 54*, 657–699.

Hagen, G. (2003). *Social Network Analysis Functional Utility (SNAFU). Version 2.0*. Wesley Chapel, Florida: Innovation Insight.

Harary, F., and White, D.R. (2001). P-Systems: a structural model for kinship studies. *Connections. 24*, 22–33.

Holland, P.W., and Leinhardt, S. (1981). An exponential family of probability distributions for directed graphs. *Journal of the American Statistical Association. 76*, 33–50.

Huisman, M., and Van Duijn, M.A.J. (2003). StOCNET: Software for the statistical analysis of social networks. *Connections. 25*, 7–26.

Hyatt, A., Contractor, N., Ferrone, T., Han, K.K., Hsu, F., Kochhar, S., Palazzo, E., Chunke, S., and Willard, B. (2001). *Blanche. Version 4.6.4*. Urbana-Champaign: University of Illinois.

Katz, L. (1953). A new status index derived from sociometric analysis. *Psychometrika. 18*, 39–43.

Kautz, H., Selman, B., and Shah, M. (1997). The hidden web. *American Association for Artificial Intelligence Magazine. 18*, 27–36.

Krackhardt, D. (1987). QAP partialling as a test of spuriousness. *Social Networks. 9*, 171–186.

Krackhardt, D., Blythe, J., and McGrath, C. (1994). KrackPlot 3: An improved network drawing program. *Connections. 17*: 53–55.

Krebs, V.E. (2002). *InFlow. Version 3.0*. Cleveland: Orgnet.com.

MDLogix (2002). *SocioMetrica LinkAlyzer. Version 2.1*. Towson, MD: Medical Decision Logic Inc.

Molloy, M., and Reed, B. (1995). A critical point for random graphs with a given degree sequence. *Random Structures and Algorithms. 6*, 161–179.

Noldus Information Technology (2001). *MatMan. Version 1.0 for Windows*. Wageningen: Noldus Information Technology bv.

Nowicki, K., and Snijders, T.A.B. (2001). Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association. 96*, 1077–1087.

Ord, K. (1975). Estimation methods for models of spatial interaction. *Journal of the American Statistical Association. 70*, 120–126.

Richards, W.D. (1995). *NEGOPY. Version 4.30.* Burnaby: Simon Fraser University.

Richards, W.D., and Seary, A.J. (1993). *FATCAT. Version 4.2.* Burnaby: Simon Fraser University.

Richards, W.D., and Seary, A.J. (2000). Eigen analysis of networks. *Journal of Social Structure. 1.*
`http://www2.heinz.cmu.edu/project/INSNA/joss/ean.html`.

Richards, W.D., and Seary, A.J. (2003). *MultiNet. Version 4.24 for Windows.* Burnaby: Simon Fraser University.

Richardson, D.C. (2001). *Mage. Version 6.00.* Durham: Duke University.

Schweinberger, M., and Snijders, T.A.B. (2003). Settings in social networks: a measurement model. Submitted. `http://stat.gamma.rug.nl/Settings.pdf`

Scott, T. (1991). Network models. In Burt, R.S. (ed.) *STRUCTURE. Version 4.2 reference manual.* New York: Columbia University.

Seary, A.J. (1999). *PSPAR: Sparse matrix version of PSTAR.* Burnaby: Simon Fraser University.

Seary, A.J. (2003). The MultiNet eigenspaces module. Report for SAR 897-5. Burnaby: Simon Fraser University.

Seary, A.J., Richards, W.D. (2000). Fitting to p^* models in MultiNet. *Connections. 23*, 84–101.

Snijders, T.A.B. (1991). Enumeration and simulation models for 0-1 matrices with given marginals. *Psychometrika. 56*, 397–417.

Snijders, T.A.B. (1994). *SNOWBALL.* Groningen: University of Groningen.

Snijders, T.A.B. (2001). The statistical evaluation of social network dynamics. In Sobel, M.E., and Becker, M.P. (eds.), *Sociological Methodology*, pages 361–395. London: Basil Blackwell.

Snijders, T.A.B. (2002). Markov chain Monte Carlo estimation of exponential random graph models. *Journal of Social Structure. 3.*
`http://zeeb.library.cmu.edu:7850/JoSS/snijders/Mcpstar.pdf`

Snijders, T.A.B. (2004). Models for longitudinal network data. In Carrington, P.J., Scott, J., and Wasserman, S. (eds.), *Models and methods in social network analysis.* Cambridge: Cambridge University Press.

Snijders, T.A.B., and Van Duijn, M.A.J. (1997). Simulation for statistical inference in dynamic network models. In Conte, R., Hegselmann, R., and Terna, P. (eds.), *Simulating social phenomena*, pages 493–512. Berlin: Springer.

Snijders, T.A.B., and Van Duijn, M.A.J. (2002). Conditional maximum likelihood estimation under various specifications of exponential random graph models. In Hagberg, J. (ed.), *Contributions to Social Network Analysis, Information Theory, and Other Topics in Statistics; A Festschrift in honour of Ove Frank*, pages 117–134. Stockholm: University of Stockholm, Department of Statistics.

Sprenger, C.J.A., and Stokman, F.N. (1989). *GRADAP: Graph definition and analysis package. Version 2.0.* Groningen: iec. *Pro*GAMMA.

Tsuji, R. (1997). PermNet Version 0.94. Sapporo: Hokkaido University.

Van Duijn, M.A.J., Snijders, T.A.B., and Zijlstra, B.J.H. (2004). p_2: a random effects model with covariates for indirected graphs. *Statistica Neerlandica. 58*, in press.

Wasserman, S., and Faust, K. (1994). *Social Network Analysis. Methods and applications.* Cambridge: Cambridge University Press.

Wasserman, S., and Pattison, P. (1996). Logit models and logistic regression for social networks: I. An introduction to Markov graphs and p^*. *Psychometrika. 61*, 401–425.

Wasserman, S., and Robins, G. (2004). An introduction to random graphs, dependence graphs, and p^*. In Carrington, P.J., Scott, J., and Wasserman, S. (eds.), *Models and methods in social network analysis.* Cambridge: Cambridge University Press.

White, D.R., and Skyhorse. (1997). *PGRAPH: Representation and analytic program for kinship and marriage networks. Version 2.7 for Windows.* Irvine: University of California.

White, D.R., Batagelj, V., and Mrvar, A. (1999). Analyzing large kinship and marriage networks with PGRAPH and Pajek. *Social Science Computer Review. 17*, 245–274.

White, S., O'Madadhain, J., Fisher, D., and Boey, Y-B. (2003) *JUNG. Version 1.0.* Irvine, California: University of California.

*y*Works (2003). *yFiles. Version 2.1.* Tübingen, Germany: *y*Works GmbH.