SOCIAL NETWORKS

ELSEVIER

Social Networks 20 (1998) 291–299

Matrix methods for calculating the triad census

James Moody *

CB# 31210, Department of Sociology, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599, USA

Abstract

The triad census, T, of a directed network summarizes much of the structural information in a network. Thus, it has been very useful in analyzing structural properties within social networks. This paper presents a set of simple matrix formulas for calculating T. Previous work with the triad census has required enumerating each triad in the graph, which can be time consuming for very large networks. The formulas presented in this paper increase the efficiency of calculating T by an order of magnitude. Thus, these formulas provide researchers with very large networks, or the need to calculate T many times, an efficient tool for studying underlying structural patterns.

1. Introduction

Important structural properties of a social network, such as the level of transitivity, tendencies toward clustering and hierarchy, or the extent to which friends agree on thirds, have long been of interest to sociologists (Cartwright and Harary, 1956; Davis and Leinhardt, 1972; Holland and Leinhardt, 1970, 1971; Johnsen, 1985, 1986). Linking micro-level theories of action to macro level outcomes rests on understanding how individuals negotiate local relations and how those local relations cumulate into structures (Johnsen, 1985, 1986). Empirically, the most direct measure of such tendencies is through the distribution of triads in the network. Thus, one of the most useful tools for understanding the structure of a network is T, the triad census. Fig. 1 presents the 16 possible triads in a directed network.

Researchers can test structural network hypotheses by comparing linear combinations of the triad census to that expected under a random (or conditionally random) model

* Corresponding author. E-mail: moody@email.unc.edu

0378-8733/98/$ - see front matter
PII: S0378-8733(98)00006-9

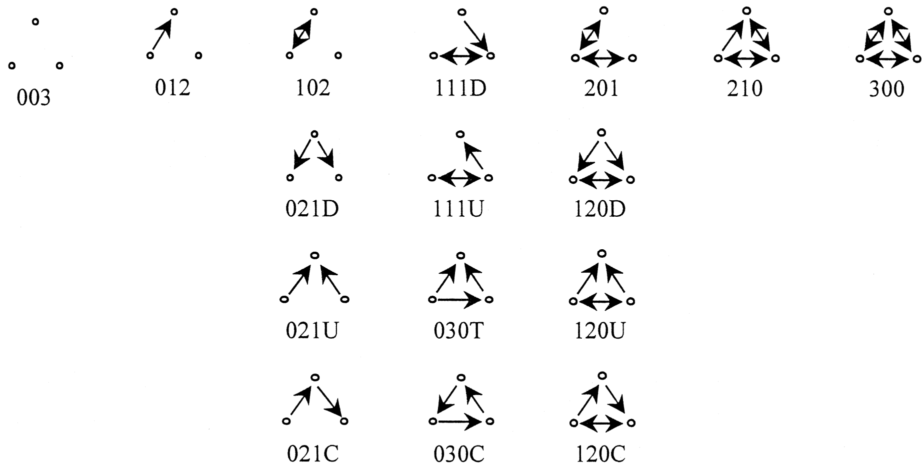

Fig. 1. The triad census.

using formulas developed by Holland and Leinhardt (1970, 1976), Fershtman (1985), Wasserman (1977), and Snijders and Stokman (1987). [1]

Historically, finding T has been cumbersome. Wasserman and Faust (1994) point out that, ''...the 16 components of T are difficult to calculate. There are no simple formulas. In fact, one must examine all $\binom{g}{3}$ triads, and place each into its proper category...'' (p. 567). Computationally, finding and classifying all triads in a network requires $O(N^3)$, which for large networks (many hundreds to thousands of nodes) can be quite time consuming. If one needs to calculate T multiple times, as would be required to use elements of T in a $p*$ framework (Wasserman and Pattison, 1996), the computational requirements are magnified.

In this paper, I present matrix formulas for calculating the 16 elements of T. These formulas allow one to compute T without enumerating every triad and thus reduce the complexity of finding T by an order of magnitude to $O(N^2)$. [2] Additionally, information contained in the matrices used to construct T can be used to study the inter-relations of sub-structures in a network. For example, one can identify a network which counts the number of times actors i and j are linked through a structural hole (White et al., 1976; Burt, 1984), such as the 201 triad.

[1] The purpose of this paper is to present a more efficient way of calculating T. Calculating the mean, variance and covariance T, used to statistically test structural hypotheses under conditionally random distributions, is not changed. An example SAS-IML program for calculating T is given in the appendix. A complete set of SAS-IML programs which calculate T, and test linear combinations of T under the U|MAN distribution are available from the author.

[2] Thanks to an anonymous reviewer for identifying the complexity of the two approaches to finding T. This difference can be dramatic. If it takes an $O(N^3)$ process 27 s to find T in a 300 node network, and approximately 2 min in a 500-node network, it will take over 1.44 days to calculate T for a 5000 node network. The comparable $O(N^2)$ procedure will take 9 s for the 300 node network, 25 s for the 500 node network, and 42 min for the 5000 node network.

2. Definitions

The *adjacency matrix*, $\mathbf{A}$, of a graph with g points has entry $\mathbf{A}_{ij} = 1$ if i nominates j $(i \rightarrow j)$, and 0 otherwise. The diagonal of $\mathbf{A}$ refers to self-nominations, and for purposes here is assumed to be 0. A number of other matrices derived from $\mathbf{A}$, are needed to identify the elements in T. The symmetric matrix, $\mathbf{E}$, is created when any arc in $\mathbf{A}$ is changed to an edge, i.e., $\mathbf{E}_{ij} = 1$ if $\mathbf{A}_{ij} = 1$ or $\mathbf{A}_{ji} = 1$. The symmetric matrix $\mathbf{M}$ is created from the mutual arcs in $\mathbf{A}$, i.e., $\mathbf{M}_{if} = 1$ if $\mathbf{A}_{ij} = 1$ and $\mathbf{A}_{ji} = 1$, else $\mathbf{M}_{ij} = 0$, and $\mathbf{C}$ is the matrix of only asymmetric arcs, i.e., $\mathbf{C} = \mathbf{A} - \mathbf{M}$. The complement of $\mathbf{E}$, $\overline{\mathbf{E}}$, is created when all 0's (except the diagonal) in $\mathbf{E}$ are changed to 1s and 1s changed to 0s. Thus, $\overline{\mathbf{E}}_{ij} = 1$ if $\mathbf{E}_{ij} = 0$, $\overline{\mathbf{E}}_{ij} = 0$ if $\mathbf{E}_{ij} = 1$ and $\overline{\mathbf{E}}_{ii} = 0$. Standard matrix operators are used, so $\mathbf{Tr}(\mathbf{X}) = \text{Trace}(\mathbf{X}) = \Sigma \mathbf{X}_{ii}$, the transpose of a matrix is given by prime ($'$) (i.e., the transpose of $\mathbf{X} = \mathbf{X}'$), and '$\times$' refers to element-wise multiplication ($\mathbf{A} \times \mathbf{B} = a_{ij} b_{ij}$). The symbol $\rightarrow$ is used to indicate an arc ('i sends to j' $= i \rightarrow j$), and 'i does not send to $j' = i \wedge\!\!\rightarrow j$. The elements of T are found by identifying particular combinations of $\mathbf{A}$, $\mathbf{C}$, $\mathbf{M}$ and $\mathbf{E}$ which uniquely identify the triad.

3. Formulas for triad counts

In this section, I present formulas for calculating 14 of the 16 elements of T, using the matrices directly derived from $\mathbf{A}$. These formulas are then used in the next section to calculate the number of 111D and 111U triads in the network.

$$T_{003} = \text{Tr}(\overline{\mathbf{E}}^3)/6 \tag{1}$$

Proof: An 003 triad occurs when there are 3 null dyads. $\overline{\mathbf{E}} = 1$ if $\mathbf{A}_{\text{ij}} = \mathbf{A}_{\text{ji}} = 0$. $\overline{\mathbf{E}}^3_{ii}$ counts the number of paths of length 3 starting and ending with i. Each complete triad contributes 6 such cycles, thus dividing the trace by 6 gives the number of empty triads in $\mathbf{A}$.

$$T_{012} = \Sigma\left(\overline{\mathbf{E}}^2 \times (\mathbf{C} + \mathbf{C}')\right)\Big/2 \tag{2}$$

Proof: An 012 triad occurs whenever an asymmetric arc links two null dyads. $\overline{\mathbf{E}}^2_{ij}$ = the number of 2-step paths from i to j, indicating two null dyads in $\mathbf{A}$ which have ($\overline{\mathbf{E}}^2_{ij}$) nodes in common. The ijth element of $\mathbf{C} + \mathbf{C}' = 1$ if $\mathbf{C}_{ij} = 1$ or $\mathbf{C}_{ji} = 1$, indicating an asymmetric arc between i and j in $\mathbf{A}$. Element-wise multiplication of $\overline{\mathbf{E}}^2$ and $\mathbf{C} + \mathbf{C}'$ yields a non-zero value whenever there is non-zero element in both $\overline{\mathbf{E}}^2$ and $\mathbf{C} + \mathbf{C}'$, indicating the number of paths from i to j (in $\overline{\mathbf{E}}$) connected by an asymmetric arc in $\mathbf{C}$. Because $\overline{\mathbf{E}}^2$ and $\mathbf{C} + \mathbf{C}'$ are symmetric, each such arc is counted twice and dividing the sum by 2 gives the number of 012 triads in $\mathbf{A}$.

$$T_{102} = \Sigma(\overline{\mathbf{E}}^2 \times \mathbf{M})\Big/2 \tag{3}$$

Proof: A 102 triad occurs whenever a symmetric arc bridges two null dyads. Thus, the proof mirrors that of T_{012}. Substituting $\mathbf{M}$ for $\mathbf{C} + \mathbf{C}'$ in Eq. (2) identifies the places where two linked null dyads are connected by a symmetric arc.

$$T_{021\text{D}} = \Sigma(\mathbf{C}'\mathbf{C}) \times \overline{\mathbf{E}}\Big/2 \tag{4}$$

Proof: An 021D triad exists whenever one node sends to both members of a null dyad. $\mathbf{C}'\mathbf{C}_{ij}$ = number of common thirds who send asymmetric ties to both i and j. Element-wise multiplication by $\overline{\mathbf{E}}$ returns a n on-zero value for null pairs in $\mathbf{A}$. Because both $\mathbf{C}'\mathbf{C}$ and $\overline{\mathbf{E}}$ are symmetric, each pair is counted twice, and dividing the sum by 2 gives the number of 021D triads.

$$T_{021\mathrm{U}} = \Sigma(\mathbf{CC}') \times \overline{\mathbf{E}}\big/2 \tag{5}$$

Proof: An 021U occurs whenever both members of a null dyad send asymmetric arcs to the same third. $\mathbf{CC}'_{\mathrm{ij}}$ = the number of times i and j send asymmetric arcs to the same node. Element-wise multiplication by $\overline{\mathbf{E}}$ returns non-zero values for null dyads only. Because $\mathbf{CC}'$ and $\overline{\mathbf{E}}$ are symmetric, each pair is counted twice, and dividing the sum by 2 gives the number of 021U triads.

$$T_{021\mathrm{C}} = \Sigma(\mathbf{C}^2) \times \overline{\mathbf{E}} \tag{6}$$

Proof: An 021C occurs whenever $i \to j$, $j \to k$ and all other arcs in the ijk triad are null. This is a two-path through only asymmetric arcs linking both members of a null dyad. Because $\mathbf{C}$ contains only the asymmetric elements of $\mathbf{A}$, $\mathbf{C}^2$ provides all 2-paths through asymmetric arcs only. Element-wise multiplication by $\overline{\mathbf{E}}$ returns a non-zero value for 2-paths between null dyads.

$$T_{030T} = \Sigma(\mathbf{C}^2) \times \mathbf{C} \tag{7}$$

Proof: An 030T occurs whenever the starting node of an asymmetric 2-path also sends an asymmetric tie to the third, i.e. whenever $i \to j$, $j \to k$, $i \to k$, and all other arcs in the ijk triad are null. All 2-paths through only asymmetric arcs are identified by $\mathbf{C}^2$. Element-wise multiplication by $\mathbf{C}$ returns a non-zero value whenever there is both an asymmetric 2-path and an asymmetric arc between i and k.

$$T_{030C} = \mathrm{Tr}(\mathbf{C}^3)/3 \tag{8}$$

Proof: An 030 triad is a directed cycle of length 3. $\mathbf{C}^3_{ii}$ = the number of asymmetric paths of length 3 starting and ending with i. Since each 030C contains 3 such paths, dividing the sum by 3 gives the number of 030C triads.

$$T_{201} = \Sigma\mathbf{M}^2 \times \overline{\mathbf{E}}\big/2 \tag{9}$$

Proof: A 201 triad occurs whenever a symmetric 2-path through mutual dyads bridges a null dyad. $\mathbf{M}^2_{ij}$ = number of mutual 2-paths connecting i and j. Element-wise multiplication by $\overline{\mathbf{E}}$ returns a non-zero value for all such 2-paths linking members of a null dyad. Again, because $\overline{\mathbf{E}}$ and $\mathbf{M}^2$ are symmetric, dividing the sum by 2 gives the number of 201 triads.

$$T_{120\mathrm{D}} = \Sigma\mathbf{C}'\mathbf{C} \times \mathbf{M}\big/2 \tag{10}$$

Proof: $T_{120\mathrm{D}}$ is the same as $T_{021\mathrm{D}}$, except the null dyad in $T_{021\mathrm{D}}$ is mutual in $T_{120\mathrm{D}}$. Thus, substitute $\mathbf{M}$ for $\overline{\mathbf{E}}$ in Eq. (4), to get $T_{120\mathrm{D}}$.

$$T_{120\mathrm{U}} = \Sigma\mathbf{CC}' \times \mathbf{M}\big/2 \tag{11}$$

Proof: $T_{120\mathrm{U}}$ is the same as $T_{021\mathrm{U}}$, except the null dyad in $T_{021\mathrm{U}}$ is mutual in $T_{120\mathrm{U}}$. Thus, substitute $\mathbf{M}$ for $\overline{\mathbf{E}}$ in Eq. (5), to get $T_{120\mathrm{U}}$.

$$T_{120\mathrm{C}} = \Sigma\mathbf{C}^2 \times \mathbf{M} \tag{12}$$

Proof: T_{120C} is the same as T_{021C}, except the null dyad in T_{021C} is mutual in T_{120C}. Thus, substitute **M** for $\bar{\mathbf{E}}$ in Eq. (6), to get T_{120C}.

$$T_{210} = \Sigma \mathbf{M}^2 \times (\mathbf{C} + \mathbf{C}') \big/ 2 \tag{13}$$

Proof: A 210 triad occurs whenever an asymmetric dyad is linked through a mutual 2-path. Thus, it is the same as 201, substituting an asymmetric dyad for the null. The proof then follows that for T_{201}.

$$T_{300} = Tr(\mathbf{M}^3)/6 \tag{14}$$

Proof: A 300 triad occurs whenever each dyad is mutual, giving a symmetric cycle of length three. $\mathbf{M}^3_{ii}$ = number of cycles of length 3 starting and ending with i. Each complete triad contributes 6 such cycles, and dividing the trace by 6 gives the number of complete triads in the network.

T_{111D} and $\mathbf{T}_{111U}$.

Fourteen down, two to go. Unfortunately, the strategy used above, which identifies a particular arc or dyad in the triad, will not work with the 111 triads. The 111 triads are unique in that the complement of the triad is isomorphic to the triad itself, and the null dyad is bridged by both a common node, and an intransitive triple. Thus, a simple two-path or common third method will be ineffective for differentiating 111D from 111U.

The distinguishing characteristic of the 111D from the 111U is the direction of the asymmetric arc. Which can be conceived of as the number of arcs each null dyad member jointly sends to or receives from the third. Each member (i, j) of the null dyad in a 111 triad either sends an arc to (if it is a 111D) or receives an arc from (if it is a 111U) the third node in the triad. Five triad types contain null dyads that jointly send to or receive from the third: {201, 021D, 021U, 111U, and 111D}. We can thus distinguish the 111Ds from the 111Us by subtracting from the total number of common sent or common received nodes those coming from 201, 021D, or 021U. In 201, i and j both send and receive from the same node. In 021D, i and j receive from the same node, and in 021U they send to the same node. If i and j are in a 111D, they both send to the third, and one receives from the third. If they are in the 111U they both receive from the same third. Thus, the number of 111D triads a given null pair is involved in is the total number of thirds they jointly send to, minus the number of 201s and 021Us they are involved in. Similarly, the number of 111U triads a given null pair is involved in is the total number of thirds they commonly receive from, minus the number which come from 201s and 021Ds.

To calculate this, we first need the count matrices for the 201, 021D and 021U triad types. These matrices count the number of times nodes i and j are linked by the given triad type. They have already been identified above, as the matrices that are summed over to get the triad counts. Thus,

$$\mathbf{t201} = (\mathbf{M}^2 \times \bar{\mathbf{E}}) \tag{15}$$

where $\mathbf{t201}_{ij}$ = number of times i and j are members of the null dyad in a 201 triad

$$\mathbf{t021D} = (\mathbf{C}'\mathbf{C}) \times \bar{\mathbf{E}} \tag{16}$$

where $\mathbf{t021D}_{ij}$ = number of times i and j are members of the null dyad in an 021D triad and

$$\mathbf{t021U} = (\mathbf{CC}') \times \overline{\mathbf{E}} \tag{17}$$

where $\mathbf{t021U}_{ij}$ = number of times i and j are members of the null dyad in an 021D triad. Subtracting these from the total common sent nodes between a null dyad gives:

$$\mathbf{t111D} = (\mathbf{AA}' \times \overline{\mathbf{E}}) - \mathbf{t201} - \mathbf{t021U} \tag{18}$$

Summing over the matrix and dividing by 2 gives us the total number of 111D triads, $T_{111\mathrm{D}}$.

$$T_{111D} = \Sigma \mathbf{t111D}/2 \tag{19}$$

and similarly for $T_{111\mathrm{U}}$.

$$\mathbf{t111U} = (\mathbf{A}'\mathbf{A} \times \overline{\mathbf{E}}) - \mathbf{t201} - \mathbf{t021D} \tag{20}$$

$$T_{111\mathrm{U}} = \Sigma \mathbf{111U}/2 \tag{21}$$

4. Discussion

These formulas can be used to calculate T efficiently for any network, and greatly reduce the computational time needed to calculate T. This computational gain will be of use to those studying very large networks, and those who want to calculate T multiple times under slight variations in the original network.

The matrix approach used here rests on identifying particular patterns within the network, then counting the number of occurrences of that pattern. Global counts, however, are simply one type of information contained in these matrices. For example, **t201** (Eq. (15)) is an indicator matrix for the number of times persons i and j share links to a given third, but have no links between themselves. This pattern is a classic instance of a structural hole (White et al., 1976; Burt, 1984). The matrix **t201**, then, can be seen as a network of actors connected by the joint occupancy of a structural hole, and could be analyzed as such directly. Researchers interested in the structural linkages among sub-patterns in a network might extend the matrix approaches used here to other sub-graphs of particular interest.

Acknowledgements

Funding for work on this paper is from the National Longitudinal Study of Adolescent Health (Add Health), a program project designed by J. Richard Udry and Peters. Bearman, and funded by a grant HD31921 from the National Institute of Child Health and Human Development to the Carolina Population Center, University of North Carolina at Chapel Hill, with cooperative funding participation by the following agencies: The National Cancer Institute; The National Institute of Alcohol Abuse and Alcoholism; the National Institute on Deafness and other Communication Disorders; the National Institute on Drug Abuse; the National Institute of General Medical Sciences; the National Institute of Mental Health; the Office of AIDS Research, NIH; the Office of

Director, NIH; The National Center for Health Statistics, Centers for Disease Control and Prevention, HHS; Office of Minority Health, Centers for Disease Control and Prevention, HHS, Office of the Assistant Secretary for Planning and Evaluation, HHS; and the National Science Foundation. Thanks to Peter S. Bearman, Lisa A. Keister, Hyojoung Kim, and the *Social Networks* reviewers for comments on earlier drafts of this paper.

Appendix A. A SAS-IML function module to calculate *T*

The following function module, MTCEN, will calculate the triad census for the input adjacency matrix, **A**. It can be stored and included, or typed directly in the SAS program.

```
start mtcen(a);
   e = (a + a`) > 0; /* arcs to edges */
   eb = choose(e = 1,0,1); /* eb_ij = 1 if A_ij = A_ji = 0 */
   eb = eb-diag(eb); /* remove the diagonal */
   free e; /* to save memory, free mats which are not going to be used below */

   M = (a + a`) > 1; /* mutual edges only */
   C = a-m; /* asymmetric arcs only */
   T = j(16,1,0); /* initialize the 1 by 16 triad census vector */

   ccp = c*c`;
   t021u = ccp#eb;
   t[13,1] = sum(ccp#m)/2; /* 120U */
   free ccp;
   t[5,1] = sum(t021u)/2; /* t021U */

   cpc = c`*c;
   t021d = cpc#eb;
   t[12,1] = sum(cpc#m)/2; /* 120D */
   free cpc;
   t[4,1] = sum(t021d)/2; /* t021d */
   t[1,1] = trace(eb**3)/6; /* t003 */

   cmax-c + c`;
   eb2 = eb**2;

   t[2,1] = sum(eb2#(cmax))/2; /* t012 */
   t[3,1] = sum(eb2#m)/2; /* t102 */
   free eb2;

   m2 = m**2;
   t[15,1] = sum(m2#cmax)/2; /* t210 */
   free cmax;
```

```
  t201 = m2#eb;
  free m2;
  t[11,1] = sum(t201)/2; /* t201 */

  c2 = c**2;
  t021c = c2#eb;
  t[9,1] = sum(c2#c); /* 030T */
  t[14,1] = sum(c2#m); /* 120C */
  free c2;
  t[6,1] = sum(t021c); /* 021c */
  t[10,1] = trace(c**3)/3; /* 030c */
  free c;
  t[16,1] = trace(m**3)/6; /* t300 */
  free m;

  t111 = (a**2)-t201-t021c;

  t111d = ((a*a`)#eb)-t201-t021u;
  t[7,1] = sum(t111d)/2;

  t111u = ((a`*a)#eb)-t201-t021d;
  t[8,1] = sum(t111u)/2;

  return(t);
finish;
```

SAS function modules create new matrices as a function of the arguments. Thus, the following example would generate the triad census for the adjacency matrix, **X**.

The program:

```
Proc Iml;
%include 'c:\<directory>\mtcen.mod';/* call in the function module */

x = {0 1 1 1 0 1 0 1 0,
     0 0 0 0 1 0 0 1 0,
     1 0 0 1 0 0 0 1 0,
     1 0 1 0 0 0 0 0 0,
     1 0 0 0 0 1 0 1 0,
     0 0 0 0 0 0 0 0 0,
     0 0 0 0 0 0 0 0 0,
     1 0 0 0 0 1 0 0 0,
     0 0 0 0 0 1 0 0 0}; /* small graph with each triad type */

xt = mtcen(x);
print xt;
quit;
```

The output:

XT
21
26
11
1
5
3
2
5
3
1
1
1
1
1
1
1

References

Burt, R.S., 1984. Structural holes: The social structure of competition. Harvard Univ. Press, Cambridge, MA.

Cartwright, D., Harary, F., 1956. Structural balance: a generalization of Heider's theory. Psychological Review 63, 277–293.

Davis, J.A., Leinhardt, S., 1972. The structure of positive relations in small groups. In: Berger, J., Zelditch, M., Anderson, B. (Eds.), Sociological Theories in Progress, Vol. 2, Houghton Mifflin, Boston, MA, pp. 218–51.

Fershtman, M., 1985. Transitivity and the path census in sociometry. Journal of Mathematical Sociology 11, 159–189.

Holland, P.W., Leinhardt, S., 1970. A method for detecting structure in sociometric data. Am. J. Sociol. 70, 492–513.

Holland, P.W., Leinhardt, S., 1971. Transitivity in structural models of small groups. Comparative Groups Studies 2, 107–124.

Holland, P.W., Leinhardt, S., 1976. The statistical analysis of local structure in social networks. In: Heise, D.R. (Ed.), Sociological Methodology, Jossey-Bass, San Francisco, 1–45.

Johnsen, E.C., 1985. Network macrostructure models for the Davis-Leinhardt set of empirical sociomatrices. Social Networks 7, 203–224.

Johnsen, E.C., 1986. Structure and process: agreement models for friendship formation. Social Networks 8, 257–306.

Snijders, T.A.B., Stokman, F.N., 1987. Extensions of triad counts to networks with different subsets of points and testing underlying random graph distributions. Social Networks 9, 249–275.

Wasserman, S., 1977. Random directed graph distributions and the triad census in social networks. Journal of Mathematical Sociology 5, 61–86.

Wasserman, S., Faust, K., 1994. Social Network Analysis. Cambridge Univ. Press, Cambridge.

Wasserman, S., Pattison, P., 1996. Logit modules and logitic regressions for social networks: I. An introduction to Markov graphs and P∗. Psychometrika 61, 401–425.

White, H.C., Boorman, S.A., Breiger, R.L., 1976. Social structure from multiple networks: I. Blockmodels of roles and positions. Am. J. Sociol. 81, 730–780.