Markov Chain Monte Carlo Estimation of Exponential Random Graph Models

Tom A.B. Snijders
ICS, Department of Statistics and Measurement Theory
University of Groningen *

April 19, 2002

*Author's address: Tom A.B. Snijders, Grote Kruisstraat 2/1, 9712 TS Groningen, The Netherlands, email <t.a.b.snijders@ppsw.rug.nl>.

I am grateful to Paul Snijders for programming the JAVA applet used in this article. In the revision of this article, I profited from discussions with Pip Pattison and Garry Robins, and from comments made by a referee.

This paper is formatted in landscape to improve on-screen readability. It is read best by opening Acrobat Reader in a full screen window. Note that in Acrobat Reader, the entire screen can be used for viewing by pressing `Ctrl-L`; the usual screen is returned when pressing `Esc`; it is possible to zoom in or zoom out by pressing `Ctrl--` or `Ctrl-=`, respectively. The ⇐ sign in the upper right corners links to the page viewed previously.

Abstract

This paper is about estimating the parameters of the exponential random graph model, also known as the p^* model, using frequentist Markov chain Monte Carlo (MCMC) methods. The exponential random graph model is simulated using Gibbs or Metropolis-Hastings sampling. The estimation procedures considered are based on the Robbins-Monro algorithm for approximating a solution to the likelihood equation.

A major problem with exponential random graph models resides in the fact that such models can have, for certain parameter values, bimodal (or multimodal) distributions for the sufficient statistics such as the number of ties. The bimodality of the exponential graph distribution for certain parameter values seems a severe limitation to its practical usefulness.

The possibility of bi- or multimodality is reflected in the possibility that the outcome space is divided into two (or more) regions such that the more usual type of MCMC algorithms, updating only single relations, dyads, or triplets, have extremely long sojourn times within such regions, and a negligible probability to move from one region to another. In such situations, convergence to the target distribution is extremely slow. To be useful, MCMC algorithms must be able to make transitions from a given graph to a very different graph. It is proposed to include transitions to the graph complement as updating steps to improve the speed of convergence to the target distribution. Estimation procedures implementing these ideas work satisfactorily for some data sets and model specifications, but not for all.

Keywords: p^* model; Markov graph; digraphs; exponential family; maximum likelihood; method of moments; Robbins-Monro algorithm; Gibbs sampling; Metropolis-Hastings algorithm.

1. Introduction

Frank and Strauss (1986) proposed Markov graphs as a family of distributions for undirected and directed graphs with non-trivial dependence between edges, but with certain appealing conditional independence properties. More specifically, a random graph is a Markov graph in this sense if the number of nodes is fixed (say, at g) and nonincident edges (i.e., edges between disjoint pairs of nodes) are independent conditional on the rest of the graph. Frank and Strauss elaborated on Besag (1974) in their use of the Hammersley-Clifford theorem to characterize Markov graphs.

As an important subfamily they defined the *triad model* for undirected graphs. The symbol y is used to denote an undirected graph with g nodes represented by the adjacency matrix $y = (y_{ij})_{1\leq i,j\leq g}$, where $y_{ij} = 1$ or 0 indicates that there is, or there is not, an edge between i and j. Note that y being an undirected graph implies that $y_{ii} = 0$ and $y_{ij} = y_{ji}$ for all i, j. The probability function of the triad model is

$$\mathrm{P}_\theta\{Y = y\} = \exp\Big(\theta_1\, u_1(y) \;+\; \theta_2\, u_2(y) \;+\; \theta_3\, u_3(y) \;-\; \psi(\theta)\Big) \qquad (1)$$

where the parameter is $\theta = (\theta_1, \theta_2, \theta_3)$, the sufficient statistic $(u_1(y), u_2(y), u_3(y))$ is defined by

$$\begin{array}{lcll} u_1(y) & = & \sum_{1\leq i<j\leq g} y_{ij} & \text{number of edges} \\ u_2(y) & = & \sum_{1\leq i<j\leq g} \sum_{k\neq i,j} y_{ik}\, y_{jk} & \text{number of twostars} \\ u_3(y) & = & \sum_{1\leq i<j<k\leq g} y_{ij}\, y_{ik}\, y_{jk} & \text{number of triangles,} \end{array} \qquad (2)$$

and $\psi(\theta)$ is a normalizing constant. For $\theta_2 = \theta_3 = 0$ this distribution reduces to the Bernoulli graph, i.e., the random graph in which all edges occur independently and have the same probability $\mathrm{P}(Y_{ij} = 1) = \exp(\theta_1)/(1 + \exp(\theta_1))$ for $i \neq j$.

This model was extended by Frank (1991) and by Wasserman and Pattison (1996) to arbitrary statistics $u(y)$, with a focus on directed graphs (digraphs). This led to the family of probability functions

$$\mathrm{P}_\theta\{Y = y\} = \exp\left(\theta' u(y) - \psi(\theta)\right) \qquad (3)$$

where y is the adjacency matrix of a digraph and the sufficient statistic $u(y)$ is any vector of statistics of the digraph. They called this family the p^* model. This formula can in principle represent any probability distribution for digraphs, provided that each digraph has a positive probability. Subsequent work

(among others, Pattison and Wasserman, 1999; Robins, Pattison, and Wasserman, 1999) elaborated this model and focused on subgraph counts as the statistics included in $u(y)$, motivated by the Hammersley-Clifford theorem (Besag, 1974). Models for directed or undirected graphs defined by (3) will be called here exponential random graph models and, equivalently, p^* models.

1.1. Parameter estimation

Frank and Strauss (1986) commented on the difficulties of estimating parameters in the Markov graph model. They presented a simulation-based method to approximate the maximum likelihood (*ML*) estimate of any one of the three parameters θ_k, given that the other two are fixed at 0. They also proposed a kind of conditional logistic regression method to estimate the full vector θ. This method, of which the principle was proposed in a different context by Besag (1975), was elaborated for random digraph models by Strauss and Ikeda (1990), Frank (1991), and Wasserman and Pattison (1996). It is a pseudolikelihood method which operates by maximizing the so-called pseudologlikelihood defined for digraphs by

$$\ell(\theta) = \sum_{i,j} \ln\Big(\mathrm{P}_\theta\left\{Y_{ij} = y_{ij} \mid Y_{hk} = y_{hk} \text{ for all } (h,k) \neq (i,j)\right\}\Big). \quad (4)$$

Although this method is intuitively appealing and easily implemented using any statistical package for logistic regression analysis, the properties of the resulting estimator for exponential graph models are unknown. (Corander, Dahmström and Dahmström (1998) gave some some simulation-based comparisons between the maximum pseudolikelihood estimator and the ML estimator.) For statisticians, an obvious drawback of the pseudolikelihood method is that it is not a function of the complete sufficient statistic $u(Y)$ which implies that it is not an admissible estimator for a squared-error loss function (cf. Lehmann, 1983).

Dahmström and Dahmström (1993) proposed a simulation-based (Markov chain Monte Carlo, *MCMC*) method for estimating the parameters of the Markov graph distribution. They concentrated on the estimation of θ_2, assuming that $\theta_3 = 0$, and proposed a stepwise simulation method. This work was extended by Corander et al. (1998) to estimating a parameter of more than one dimension. Crouch, Wasserman, and Trachtenberg (1998)

also elaborated an MCMC estimation method for the p^* model. All these authors followed the approach of Geyer and Thompson (1992) to construct Monte Carlo-based algorithms for approximating the maximum likelihood estimate. Their methods rely on Monte Carlo simulations of the Markov graph at the current parameter value. It will be discussed below, however, that simulation algorithms for exponential random graph distributions can suffer from severe convergence problems, not signaled by these authors, and these problems can invalidate the estimation algorithms in which the simulations are used.

1.2. Overview of the present paper

This paper is about the simulation and MCMC estimation of exponential random graph models. Sections 2 to 4 demonstrate a basic problem in the simulation of exponential random graph distributions. Stated briefly, the conclusion is that in the parameter space for the triad model and for many differently specified models, there is a large region in which the demarcation between the subset of parameters θ leading to graphs with relatively low expected densities, and the subset of parameters θ leading to graphs with relatively high expected densities, is quite sharp; and for parameters in or near this demarcation zone, the distribution of the graph density can have a bimodal shape. The demarcation becomes more marked as the number g of nodes increases. Section 5 proposes an algorithm augmented with bigger updating steps to improve the convergence propoerties.

Sections 6 to 8 are about MCMC estimation methods for exponential random graph models. When fitting such models to empirically observed networks, the parameter estimates often are very close to the demarcation zone mentioned above, which leads not only to instability of the estimation algorithm but also to a poor representation of the data by the estimated model. This situation leads to poor convergence properties for the more usual MCMC procedures for simulating exponential random graph distributions. Some proposals are discussed to try and circumvent the simulation problems explained in sections 2 to 4. These proposals sometimes are effective, but not always. Finally, Section 9 discusses the implications for the use of exponential random graph models in network analysis.

2. Simulation of exponential random digraphs

A graph (directed or undirected) will be represented in this paper by its adjacency matrix $y = (y_{ij})_{1 \leq i,j \leq g}$; the number of vertices is denoted by g; the diagonal of the adjacency matrix is zero ($y_{ii} = 0$ for all i). Random variables, vectors, and matrices are indicated by capital letters. Replacing an index by a + sign indicates summation over this index. The focus is on directed graphs.

One of the convenient ways to approximate such random draws is by Gibbs sampling (Geman and Geman, 1983) applied to the elements of the adjacency matrix. This means that an initial matrix $Y^{(1)}$ is chosen, and the elements of this matrix are stochastically updated. This updating mechanism circles through the whole matrix again and again, thus defining a stochastic process $Y^{(t)}$ which is a Markov chain; the distribution of $Y^{(t)}$ tends asymptotically to the desired random graph distribution. This procedure implies that the matrices $Y^{(t)}$ and $Y^{(t+1)}$ differ at most in only one element (the one that is updated in step t).

The Gibbs sampling procedure specifies that all elements Y_{ij} in turn are randomly updated. If, in step t, element Y_{ij} is the one being updated, then its new value is generated according to the conditional distribution given all the other elements,

$$\mathrm{P}_\theta \left\{ Y_{ij}^{(t+1)} = a \mid Y^{(t)} = y^{(t)} \right\} = \tag{5}$$

$$\mathrm{P}_\theta \left\{ Y_{ij} = a \mid Y_{hk} = y_{hk}^{(t)} \text{ for all } (h,k) \neq (i,j) \right\} \qquad (a = 0, 1) \ .$$

In this step, all other elements are left unchanged, i.e., $Y_{hk}^{(t+1)} = Y_{hk}^{(t)}$ for all $(h,k) \neq (i,j)$. Note that the left hand side of this equation is the transition probability to be defined, and the right hand side is the conditional distribution in the target distribution (3). This is the same conditional distribution used also in the pseudolikelihood procedure based on (4), but now used for an entirely different purpose. A general theorem (Geman and Geman, 1983) implies that the distribution of the digraph $Y^{(t)}$ converges for $t \rightarrow \infty$ to the exponential random graph distribution.

Frank (1991) and Wasserman and Pattison (1996) discussed how to obtain the conditional probabilities (5) for the exponential random graph model (3). For a given adjacency matrix y, define by $y^{(ij1)}$ and $y^{(ij0)}$, respectively, the adjacency matrices obtained by defining the (i,j) element as $y_{ij}^{(ij1)} = 1$ and $y_{ij}^{(ij0)} = 0$ and

leaving all other elements as they are in y. Then the conditional distribution (5) is defined by

$$\begin{aligned}&\text{logit}\Big(\mathrm{P}_\theta\left\{Y_{ij}=1 \mid Y_{hk}=y_{hk} \text{ for all } (h,k)\neq(i,j)\right\}\Big)\\ &\qquad = \theta'\Big(u(y^{(ij1)})-u(y^{(ij0)})\Big)\ .\end{aligned} \tag{6}$$

In words this means that the conditional distribution is defined by a logistic regression model where the sufficient statistics are given by the difference between the values for $u(y)$ obtained when letting $y_{ij}=1$ or $y_{ij}=0$, and leaving all other elements of y as they are.

2.1. Instability of the simulation algorithm

Scrutinizing experimental results of the Gibbs sampling procedure for various parameter values shows that, depending on the initial state of the digraph and the parameter values used, it can take extremely long (e.g., in the order of a million steps or more) before the Gibbs sampler seems to converge to a stable distribution. This can be intuitively understood from the following example.

Consider an exponential random graph model (3) for a directed graph with two effects: the number of ties and the number of transitive triplets. This means that the sufficient statistics are

$$\begin{aligned}u_1(y) &= y_{++}=\textstyle\sum_{i,j}\ y_{ij}\\ u_2(y) &= \textstyle\sum_{i,j,h}\ y_{ij}\,y_{jh}\,y_{ih}\ .\end{aligned} \tag{7}$$

Some calculations on the basis of (6) show that the conditional probability (5) used for Gibbs sampling is here defined by

$$\begin{aligned}&\text{logit}\Big(\mathrm{P}_\theta\left\{Y_{ij}=1 \mid Y_{hk}=y_{hk} \text{ for all } (h,k)\neq(i,j)\right\}\Big)\\ &\qquad = \theta_1+\theta_2\,(\mathrm{IS}_{ij}+\mathrm{OS}_{ij}+\mathrm{TP}_{ij})\end{aligned} \tag{8}$$

where IS_{ij}, OS_{ij}, and TP_{ij} are the number of in-twostars, out-twostars, and twopaths ('mixed stars') connecting i and j,

$$\begin{aligned}\mathrm{IS}_{ij} &= \sum_{\substack{h=1\\ h\neq i,j}}^{g} y_{ih}\,y_{jh}\\ \mathrm{OS}_{ij} &= \sum_{\substack{h=1\\ h\neq i,j}}^{g} y_{hi}\,y_{hj}\\ \mathrm{TP}_{ij} &= \sum_{\substack{h=1\\ h\neq i,j}}^{g} y_{ih}\,y_{hj}\ .\end{aligned}$$

Suppose that θ_1 is negative and θ_2 positive, and the initial state $Y^{(1)}$ has a low density. When, in a given updating step for element Y_{ij}, the current digraph has no twostars or twopaths connecting points i and j, the transitive triplets parameter θ_2 will have no effect, however large it is. If the initial digraph has a low density and θ_1 is negative, then the number of twostars and twopaths will remain low for a long time and θ_2 will hardly have an effect on the simulation results. However, when by chance some more twostars and twopaths appear, then the positive θ_2 value will lead to an increase in the number of ties, and possibly to an explosion of ties and an extremely quick transition to a high-density graph.

Whether this explosion occurs depends on the current number of twostars and twopaths and their positions – a matter of chance – and on the two parameter values θ_1 and θ_2. If, for a given value of θ_2, parameter θ_1 is negative and sufficiently large in absolute value, then the explosion will never occur. It is plausible that there exists a non-empty region of values of the parameters (θ_1, θ_2), such that for parameters in this region, if one starts with a low-density digraph the explosion will occur with probability 1. The expected waiting time until the explosion will be a decreasing function of θ_1 and θ_2. Although the theory of finite Markov chains implies that the probability is also 1 that the simulations will at some time return to a low-density state, I conjecture that for the parameter values where an explosion can occur, the expected waiting time for a high-density graph to return to a low-density state is much and much higher than the expected waiting time for a low-density graph to explode to a high-density state.

Example 1.

This property can be examined by running the JAVA applet "pstardemo" provided with this article. This applet runs the Gibbs sampler for parameter values and a number of nodes which can be determined by the user. For the exponential random graph model with sufficient statistics (7), the applet can be used to experimentally determine for which parameter values the explosion effect occurs. The parameters θ_1 and θ_2 of the current model are denoted 'ties' and 'transitivity' in the applet; the parameters called 'reciprocity', 'similarity', and 'two-stars' should be equal to 0 to obtain the model currently under discussion.

Have a look now at the *pstardemo* applet in the file *pstar.htm*. This applet can be viewed in a web browser that supports JAVA.

(The preceding paragraph links to the file `pstar.htm`*.*
The best readability is obtained by opening the present pdf text and the JAVA applet in separate windows.)

When the applet is opened, it shows a digraph with $g = 12$ nodes, a 'ties' parameter of $\theta_1 = -4.0$, and a 'transitivity' parameter of $\theta_2 = 2.0$. Clicking on 'Random' generates a random low-density graph. Click on 'Simulate' and the random Gibbs sampling steps will start. For some time the generated graph retains a low density, but after some random waiting time (usually less than a minute) the density will increase to 1.0: the explosion effect. Once the complete graph has been reached, the probability defined by (8) is so large that it is virtually impossible for the simulation process to go to a lower density. Click subsequently on 'Stop', 'Random', and 'Simulate' to restart Gibbs sampling with a low-density graph.

If the same procedure is followed still with $\theta_1 = -4.0$ but with a lower transitivity parameter θ_2, as long as θ_2 is larger than 1.4, the explosion will still occur, but after a longer average waiting time. This can be tried out by specifying the 'Reciprocity' parameter at 0.1.

If one starts with a high-density graph, still with $\theta_1 = -4.0$, the digraph will quickly go to a low-density state for $\theta_2 \leq 0.28$, but stay at a high-density state for $\theta_2 \geq 0.29$.

If, in addition to the mentioned two effects, there is also a positive reciprocity effect corresponding to the statistic

$$u_3(y) = \sum_{i<j} y_{ij}\, y_{ji} \ ,$$

then the explosive effect will occur more quickly because the reciprocity effect will support the increase in ties when there are sufficiently many twostars or twopaths.

This demonstrates that for certain parameter values of the exponential random graph model, the Gibbs sampling procedure has two distinct states, or regimes, defined by subsets of the outcome space. In the preceding example with, e.g., a 'ties' parameter of -4.0 and 'transitivity' parameter of 1.5, one state consists of digraphs with a high density and is almost stable in the sense that the expected time before leaving this state is astronomically long; the other state consists of low-density digraphs and is semi-stable in the sense that the procedure will after some time leave this state (the explosion effect), while the expected time until this occurs is long but not astronomically so. For some other parameter values the Gibbs sampler will practically permanently remain in a low-density state; and for still other values, it will practically permanently remain in a high-density state. The next section demonstrates a model with parameter values where the Gibbs sampler switches evenly between the low-density and the high-density regime.

3. Simple cases: independent dyads or two-stars

Theoretical understanding of the exponential random graph can be promoted by considering special cases for which some properties can be deduced analytically.

3.1. The reciprocity p^* model

A first model for which exact calculations are possible is the reciprocity p^* model, where the only effects are number of ties and reciprocity,

$$\begin{array}{rcl} u_1(y) & = & y_{++} = \sum_{i,j}\, y_{ij} \\ u_2(y) & = & \sum_{i<j}\, y_{ij}\, y_{ji}\ . \end{array} \tag{9}$$

The $\binom{g}{2}$ dyads (Y_{ij}, Y_{ji}) for $i < j$ are independent and the probabilities for the dyads to be of the types mutual $(y_{ij} = y_{ji} = 1)$, asymmetric $(y_{ij} + y_{ji} = 1)$, or null $(y_{ij} = y_{ji} = 0)$, are given by

$$\begin{array}{rcl} \mathrm{P}\{Y_{ij} = Y_{ji} = 1\} & = & e^{2\theta_1+\theta_2-\psi(\theta)} \\ \mathrm{P}\{Y_{ij} + Y_{ji} = 1\} & = & 2e^{\theta_1-\psi(\theta)} \\ \mathrm{P}\{Y_{ij} = Y_{ji} = 0\} & = & e^{-\psi(\theta)} \end{array} \tag{10}$$

where

$$\psi(\theta) \;=\; \log\left(e^{2\theta_1+\theta_2} \;+\; 2e^{\theta_1} \;+\; 1\right) \;.$$

Generating random draws from this model is straightforward. The independence between the dyads precludes the explosion effect discussed in the preceding section.

3.2. The twostar p^* models

The second model is the out-twostar p^* model, having as sufficient statistics the number of ties and the number of out-twostars,

$$\begin{aligned} u_1(y) \;&=\; y_{++} \;=\; \sum_{i,j} y_{ij} \\ u_2(y) \;&=\; \tfrac{1}{2} \sum_{\substack{i,j,h=1 \\ j\neq h}}^{g} y_{ij}\, y_{ih} \;=\; \sum_i \binom{y_{i+}}{2} \;. \end{aligned} \tag{11}$$

The probability function for this model can be written as

$$\mathrm{P}\{Y=y\} \;=\; \exp\left(\sum_i \left(\theta_1 y_{i+} \;+\; \theta_2 \binom{y_{i+}}{2}\right) \;-\; \psi(\theta)\right)$$

which indicates that the rows $(y_{i1},\ldots,y_{in})$ of the adjacency matrix are statistically independent for $i=1,\ldots,g$. It may be noted that, if the total number of ties y_{++} is fixed, the number of out-twostars is a linear function of the sum of squared out-degrees, and hence of the out-degree variance. High values of θ_2 therefore imply a high expected out-degree variance.

Similarly the in-twostar p^* model can be defined, for which the columns in the adjacency matrix are independent.

3.3. The reciprocity and twostar p^* model

The reciprocity and the out-twostar p^* models can be combined in a model with density, reciprocity, and out-twostars effects. The model is defined by the three sufficient statistics

$$\begin{aligned} u_1(y) \;&=\; y_{++} \;=\; \sum_{i,j} y_{ij} \\ u_2(y) \;&=\; \sum_{i<j} y_{ij}\, y_{ji} \\ u_3(y) \;&=\; \tfrac{1}{2} \sum_{\substack{i,j,h=1 \\ j\neq h}}^{g} y_{ij}\, y_{ih} \;=\; \sum_i \binom{y_{i+}}{2} \;. \end{aligned} \tag{12}$$

For this model the digraph does not fall apart in independent parts, but some explicit calculations still can be made. The crucial part $\theta' u(y)$ of the logarithm in the probability (3) can be rewritten as follows:

$$\begin{aligned}\theta_1 u_1(y) + \theta_2 u_2(y) + \theta_3 u(y) \;=\; & \left(\theta_1 + \tfrac{1}{2}\theta_2 + \tfrac{1}{2}(g-2)\theta_3\right) y_{++} \\ & + \tfrac{1}{2}\theta_2 \textstyle\sum_{i<j} \left(y_{ij}y_{ji} + (1-y_{ij})(1-y_{ji})\right) \\ & - \tfrac{1}{2}\theta_3 \textstyle\sum_i y_{i+}(g-1-y_{i+}) \;+\; c(\theta)\end{aligned}$$

where $c(\theta)$ is some function of θ not depending on y which therefore is incorporated into the normalizing constant $\psi(\theta)$. The point of this re-expression is that the second and third terms of the right-hand side are invariant upon changing all y_{ij} into $(1-y_{ij})$, i.e., transforming the digraph into its complement. If $\theta_1 + \frac{1}{2}\theta_2 + \frac{1}{2}(g-2)\theta_3 = 0$ the first term vanishes, which implies that under this condition this p^* distribution is invariant for transforming the digraph into its complement. In particular, the expected density must be 0.5. Since the properties (25) and (26) below imply that $\partial \mathrm{E}(u_1(Y))/\partial\theta_1 > 0$, where E denotes the expected value of a random variable, the expected value of the density increases as a function of θ_1. This implies the following.

Property. Under the exponential random graph model with the three sufficient statistics (12), it holds that

$$\mathrm{sign}\left(\frac{\mathrm{E}\, y_{++}}{g(g-1)} - \tfrac{1}{2}\right) \;=\; \mathrm{sign}\left(\theta_1 + \tfrac{1}{2}\theta_2 + \tfrac{1}{2}(g-2)\theta_3\right) . \qquad (13)$$

This property indicates for the reciprocity-and-twostar p^* model, for which parameters the expected density of the stationary distribution is equal to, less than, or greater than 0.5. This is exploited in the next example.

Example 2. Consider again the JAVA applet. The reciprocity and twostar p^* model is obtained by using the 'ties' parameter for θ_1, 'reciprocity' for θ_2, 'two-stars' for θ_3, and setting the 'similarity' and 'transitivity' parameters to 0. The parameter values proposed below all correspond to a digraph distribution with expected density equal to 0.5. In this way we can see experimentally how fast, or slow, can be the convergence to the asymptotic distribution.

Look again at the *pstardemo* applet in the file *pstar.htm*.
Keep the digraph at $g = 12$ vertices, with a constant 'reciprocity' parameter θ_2 of 1.0. Set the 'transitivity' parameter to 0.0, and let the 'similarity' parameter remain 0.0. In this experiment, the 'ties' parameter θ_1 and the 'two-stars' parameter θ_3 are varied in such a way that $\theta_1 = -0.5 - 5\,\theta_3$. Then the expected density of the asymptotic distribution is 0.5.
First let $\theta_3 = 0$, $\theta_1 = -0.5$, which yields the reciprocity model. When the applet is run, the stationary distribution is reached almost immediately. The density of the generated digraph fluctuates about 0.5.
Now increase θ_3 by small steps, which will make the variance of the density rise.
At $\theta_3 = 0.4$, $\theta_1 = -2.5$, the distribution of the density has started to acquire a bimodal shape.
At $\theta_3 = 0.5$, $\theta_1 = -3.0$, the density is clearly bimodally distributed and the random graph process switches between low-density and high-density regimes (which may last thousands of iteration steps).

At $\theta_3 = 0.6$, $\theta_1 = -3.5$, the Gibbs sampler switches between a low-density and a high-density regime, the average number of iterations before switching to the other regime being in the order of a few million.
For $\theta_3 = 0.7$, $\theta_1 = -4.0$, the regimes are so extreme (with densities lower than 0.10 and, respectively, higher than 0.90) that for practical purposes the time before switching to the other regime seems infinitely long.

This example shows that, depending on the parameter values, the exponential random graph distribution can have a bimodal shape in the sense that most of the probability mass is distributed over two clearly separated subsets of the set of all digraphs, one subset containing only low-density and the other subset containing only high-density digraphs. The separation between these two subsets can be so extreme that Gibbs sampling steps, or other stochastic updating steps which change only a small number of arc variables Y_{ij} , have a negligible probability of taking the Markov

process from one to the other subset. This means that for such models simulation results are in practice determined by the initial state, and give us completely misleading information about the expected values of the digraph statistics.

4. Existence of different regimes

It is clear from Examples 1 and 2 that depending on the model specification (sufficient statistics and parameter vector), there may occur different regimes between which the Gibbs sampling algorithm can switch, like a high-density and a low-density regime. A regime is defined here by a subset of the outcome space such that graphs generated consecutively by the algorithm will stay within this subset for very many iterations, and switches between these subsets occur very rarely. The examples demonstrate the possibilities of the existence of a single regime or of two regimes. The existence of two or more regimes is reminiscent of the long-range dependence that is known to occur for certain parameter values in the Ising model (cf., e.g., Besag, 1974, Newman and Barkema, 1999, Besag, 2000), a class of probability distributions for binary variables in a lattice.

This also is related to the degeneracy problem discussed by Strauss (1986) for the transitivity model and other models for interaction. Strauss noted that, for a range of parameter values, the expected graph density will tend to 1 when one considers a sequence of exponential random graph distributions with fixed parameter value and g tending to infinity. However, this concerns asymptotics for an increasing number of vertices, whereas the present paper considers graphs with a constant number of vertices.

Note that the stochastic process $Y^{(t)}$ produced by the Gibbs sampling algorithm is ergodic in the sense that for every two outcomes there is a positive probability to go from one outcome to the other in a finite number of steps. The probability is positive (but tiny) already for such a change to occur in $g(g-1)$ steps . Therefore, if there are two or more regimes, they must communicate in the sense that there is a positive probability to go from one to the other regime in finitely many steps. However, if there are two regimes then it is possible that one is dominant in the sense that the expected sojourn time is much longer in this regime than in the other regime. Further it is possible, whether or not one of

the regimes is dominant, that the expected sojourn time in either regime is extremely long, so that for practical purposes the initial situation determines whether the simulation process will show one regime or the other. This is the case, e.g., for model (12) with parameter values θ_1 = -4.5, θ_2 = 1.0, θ_3 = 0.8; note that both regimes here have probability 0.5.

The existence of two regimes tells us something not only about the Gibbs sampling algorithm for this model, but also about the probability distribution from which it samples. As is illustrated by Example 2, the distribution of the number of ties Y_{++} will have a bimodal shape if there are two regimes, the relative heights and widths of the modes reflecting the probabilities of the two regimes. In Example 2 the distribution of Y_{++} is symmetric about its mean $\frac{1}{2}\binom{g}{2}$, and the two modes are equally high. In Example 1 the existence of two modes, and their relative sizes, depends on the parameter values.

The possibility of two (or perhaps multiple) regimes in the Gibbs sampling algorithm, and the associated bimodality (or multimodality) leads to three problems.

1. For many choices of the vector of sufficient statistics $u(y)$, a subset of the distributions of the family (3) has a bimodal shape. However, a bimodal distribution is very undesirable for modeling a single observation of a social network. For fitting a distribution to a single observation, the major mode of the fitted distribution should be equal, or very close, to the observed data; this is not guaranteed for families of distributions containing bimodal distributions.

2. If there are two (or more) regimes the convergence of the Gibbs sampler to the target distribution can be so slow that this algorithm is unsuitable for generating a random draw from the distribution.

3. For the parameter values where the distribution has a bimodal shape, the expected values of the sufficient statistics $u(Y)$ are extremely sensitive to the parameter values θ, i.e., some of the derivatives $\partial \mathrm{E}_\theta u_k(Y)/\partial\theta_k$ are very large. (This is a consequence of property (26) in Section 6.) This can cause instability of algorithms for parameter estimation, and requires extra care in setting up the algorithms.

The first problem can only be solved by specifying the model, as defined by the vector of sufficient statistics, in such a way that the observed data set is fitted by a unimodal distribution. More research is needed to investigate whether, and how, this is possible. The last two problems can be solved perhaps by an appropriate construction of algorithms.

4.1. Graphs with fixed densities

For exponential random graph distributions with a given number of ties, the situation is different. The density is constant by definition. However, there still can be several regimes with very low probabilities of transitions between them. As an example consider again model (12), now under the condition that Y_{++} is constant, so that the first sufficient statistic and the parameter θ_1 are irrelevant. Suppose that $\theta_2 = 0$. If θ_3 is high, there is a tendency to have a high number of out-twostars. Suppose, to simplify the discussion, that Y_{++} is constrained to be equal to $c(g-1)$ for some integer c with $1 \leq c \leq g-1$. Then the maximum number of out-twostars is obtained by the digraphs for which c vertices have out-degrees $g-1$ and the other $g-c$ have out-degrees 0. This implies that if θ_3 is high, there will be $\binom{g}{c}$ regimes, each corresponding to a subset of c vertices all having very high out-degrees, the others having very low out-degrees. Depending on the initial situation the Markov chain algorithm will quickly enter in one of these regimes, and it may take a large number of steps before a transition to another regime is observed. If θ_3 is large enough, for practical purposes the waiting time is infinite.

This suggests that the behavior of MCMC algorithms to simulate exponential random graphs is more complicated if the restriction is made that the number of ties is fixed. The reason is that the paths for communicating between different outcomes are so much more restricted.

5. Other iteration procedures for simulating random graphs

In addition to Gibbs sampling, various other procedures can be used to construct a Markov chain $Y^{(1)}, Y^{(2)}, Y^{(3)}, \ldots, Y^{(t)}, \ldots$ of which the distribution converges to the exponential random graph distribution (3). The main technique used for constructing such chains is *detailed balance* (see, e.g., Norris, 1997, Newman and Barkema, 1999). To explain this concept, the set of all adjacency matrices (either all symmetric adjacency matrices, or all adjacency matrices) on g vertices is denoted $\mathcal{Y}_g$.

Denote the transition probabilities of the Markov chain by

$$P(y^a, y^b) = \mathrm{P}\left\{Y^{(t+1)} = y^b \mid Y^{(t)} = y^a\right\} \tag{14}$$

for $y^a, y^b \in \mathcal{Y}_g$. If there exists a probability distribution π on $\mathcal{Y}_g$ such that

$$\pi(y^a)\, P(y^a, y^b) = \pi(y^b)\, P(y^b, y^a) \quad \text{for all } y^a, y^b \in \mathcal{Y}_g \tag{15}$$

then P and π are said to be in detailed balance and π is a stationary distribution of the Markov chain with transition probabilities $P(y^a, y^b)$. If all states communicate (i.e., for each pair of states y^a, y^b there is a positive probability to go from one to the other state in finitely many steps) then π is the unique stationary distribution and and also the asymptotic distribution. For a proof see any textbook on discrete Markov chains, e.g., Norris (1997). This is applied here to $\pi(y)$ given by (3), so that the requirement for the transition probabilities is

$$\log\left(\frac{P(y^a, y^b)}{P(y^b, y^a)}\right) = \theta'\Big(u(y^b) - u(y^a)\Big)\,. \tag{16}$$

Note that this expression does not involve the problematic norming constant $\psi(\theta)$. Many definitions of transition probabilities satisfying (16) are possible. The following sections indicate some possibilities.

5.1. Small updating steps

The smallest updating steps are those where $Y^{(t)}$ and $Y^{(t+1)}$ differ in at most one element Y_{ij}. To define the updating step from $Y^{(t)}$ to $Y^{(t+1)}$, randomly determine a cell (I_t, J_t) (i.e., randomly choose any of the $\binom{g}{2}$ possibilities (i, j)) and determine the cell value

$Y_{I_t,J_t}^{(t+1)}$ according to probability distribution (6). For any pair of matrices y^a, y^b differing in only one element (i,j), it follows that

$$P(y^a, y^b) = \binom{g}{2}^{-1} \frac{\exp\left(\theta' u(y^b)\right)}{\exp\left(\theta' u(y^a)\right) + \exp\left(\theta' u(y^b)\right)}$$

which indeed satisfies (16). This procedure uses the same updating probabilities as Gibbs sampling but the order in which the cells Y_{ij} are visited is random rather than deterministic. This distinction is referred to as mixing vs. cycling (see Tierney, 1994).

An alternative is known as the *Metropolis-Hastings algorithm*. In this algorithm, the new value $Y_{I_t,J_t}^{(t+1)}$ depends also the previous value $Y_{I_t,J_t}^{(t)}$. Under the condition $(I_t, J_t) = (i,j)$, the Metropolis-Hastings probability for changing the cell value Y_{ij} is

$$\begin{aligned} &\mathrm{P}_\theta\left\{Y_{ij}^{(t+1)} = 1 - y_{ij}^{(t)} \mid Y^{(t)} = y^{(t)}\right\} \\ &= \min\left\{1, \exp\left(\theta'(u(y^{(ijc)}) - u(y))\right)\right\} \quad , \end{aligned} \tag{17}$$

where $y^{(ijc)}$ is the adjacency matrix obtained from y by changing element y_{ij} into $1 - y_{ij}$ and leaving all other elements as they are. This algorithm changes $Y^{(t)}$ more frequently than Gibbs sampling, and therefore often is more efficient.

Another Monte Carlo method for approximating draws from the exponential random graph distribution follows from the theorem in Snijders (2001) which gives a specification of a model for a continuous-time network evolution process converging in distribution to the exponential random graph model. This is a different approach because the approximation theorem holds for the continuous time process, so the stopping rule must be specified as a time point rather than as an iteration number. However, the calculations for each step in this algorithm are much more time-consuming than the steps of the procedures mentioned above, and this is not offset by a more rapid approximation of the limiting distribution. Therefore this does not seem a practically useful option.

5.2. Updating by dyads or triplets

Another possibility is to update stochastically not a single element Y_{ij} , but several elements at once. The most natural groups of elements to be changed together are *dyads* (Y_{ij}, Y_{ji}) or *triplets* defined here as triples of elements of the form (Y_{ij}, Y_{ih}, Y_{jh}),

(Y_{ij}, Y_{ih}, Y_{hj}), and/or (Y_{ij}, Y_{jh}, Y_{hi}).

These groupwise updating steps can be defined as follows with updating probabilities analogous to Gibbs sampling. (Metropolis-Hastings versions also are possible.) In the first place, choose a random dyad or triplet to be updated. Denote by I, a subset of $\{(i,j) \mid i \neq j\}$, the set of elements of the adjacency matrix to be changed (updated) in a given step, and denote by $\mathcal{Y}^{(t+1)}(I)$ the set of adjacency matrices with elements equal to $y_{ij}^{(t)}$ for all $(i,j) \notin I$. The set $\mathcal{Y}^{(t+1)}(I)$ is the set of all allowed outcomes of $Y^{(t+1)}$. This set has $2^{|I|}$ elements, where $|I|$ is the number of elements of the set I (note that $|I| = 2$ for a dyad and $|I| = 3$ for a triplet). The groupwise updating probabilities are

$$\mathrm{P}_\theta\left\{Y_{ij}^{(t+1)} = y \mid Y^{(t)} = y^{(t)}\right\} = \frac{\exp\left(\theta' u(y)\right)}{\sum_{\tilde{y}\in\mathcal{Y}^{(t+1)}(I)} \exp\left(\theta' u(\tilde{y})\right)} \qquad (18)$$

for $y \in \mathcal{Y}^{(t+1)}(I)$.

5.3. Big updates

Item (2.) in Section 4 stated that the Gibbs sampler sometimes converges very slowly. The procedures mentioned above suffer from the same problem, because the steps taken are so small. Better convergence properties may be obtained if also updating steps are included that imply bigger 'jumps' in the space $\mathcal{Y}_g$.

Analogous to cluster-flipping algorithms for the Ising model (cf. Newman and Barkema, 1999), it is possible to update Y by switching variables Y_{ij} from 0 to 1 or vice versa not just in a few cells (i,j), but in a big set of cells. Such a set could be defined by rows and/or columns. The biggest step is changing the graph to its complement, called here an *inversion*. It is proposed here to use one of the algorithms described above, augmented with inversion steps as follows. At each step t, with a rather small probability, say, 0.01, an inversion step is taken instead of the step of the basic algorithm. The inversion step is governed by a probability $p_{\mathrm{r}}(y)$. In an inversion step, with probability $p_{\mathrm{r}}(y)$ the current matrix $Y^{(t)} = y$ is replaced by its complement $Y^{(t+1)} = \mathbf{1} - y$ defined by

$$(\mathbf{1} - y)_{ij} = 1 - y_{ij} \quad \text{for all } (i,j)\ ; \qquad (19)$$

with probability $1 - p_{\mathrm{r}}(y)$ the current matrix is left as it is, $Y^{(t+1)} = y$. Gibbs and Metropolis-Hastings versions of the up-

dating probabilities are, respectively,

$$p_{\mathrm{r}}(y) \;=\; \frac{\exp\left(\theta' u(\mathbf{1}-y)\right)}{\exp\left(\theta' u(\mathbf{1}-y)\right)+\exp\left(\theta' u(y)\right)} \tag{20}$$

and

$$p_{\mathrm{r}}(y) \;=\; \min\left\{1,\, \exp\Big(\theta'(u(\mathbf{1}-y)-u(y))\Big)\right\} \;. \tag{21}$$

Inversion steps using these probabilities satisfy the detailed balance equation (16) and therefore still yield the correct stationary distribution.

This augmentation of the algorithms described in the preceding sections by inversion steps is expected to produce good convergence properties for the reciprocity-and-twostar model of Section 3.3. More generally, this procedure may give good results for graph distributions with a bimodal shape, one mode having low and the other high graph densities. For graph distributions with more than two modes, however, it may be necessary to propose other 'big updating steps'.

5.4. Using inversion steps for variance reduction

The inversion step can be used in an additional way to reduce the estimation variance of expected values of functions of exponential digraph distributions. Let Y be a random graph, produced as the final result of a long MCMC sequence $Y^{(t)}$, such that Y is assumed to be a random draw from the target distribution. For the estimation of $\mathrm{E}f(Y)$ for some function $f(y)$, one will construct many, say N, independent replications of the MCMC sequence, resulting in random draws $Y(1),\ldots,Y(N)$, and estimate $\mathrm{E}f(Y)$ by

$$\frac{1}{N}\sum_{n=1}^{N} f(Y(n)) \;.$$

However, it can be proved that

$$\mathrm{E}f(Y) \;=\; \mathrm{E}\Big(p_{\mathrm{r}}(Y)f(\mathbf{1}-Y) \;+\; (1-p_{\mathrm{r}}(Y))f(Y)\Big) \;, \tag{22}$$

where $p_{\mathrm{r}}(y)$ is defined in (20). The random variable on the right hand side has a smaller variance than $f(Y)$. Therefore it is more efficient to estimate $\mathrm{E}f(Y)$ by

$$\frac{1}{N}\sum_{n=1}^{N}\Big(p_{\mathrm{r}}(Y(n))f(\mathbf{1}-Y(n)) \;+\; (1-p_{\mathrm{r}}(Y(n)))f(Y(n))\Big) \;. \tag{23}$$

This is an example of conditioning, a principle mentioned in Ripley (1987, p. 134).

An example where this yields a considerable reduction in variance is provided by the reciprocity-and-twostar model of Section 3.3 with $\theta_1 + 0.5\theta_2 + 0.5(g-2)\theta_3 = 0$, for which $p_{\mathrm{r}}(y) = 0.5$ for all y. On the other hand, the gain in efficiency will be small if the distribution of $p_{\mathrm{r}}(Y)$ is concentrated strongly on values close to 0.

5.5. Random graphs with fixed densities

Sometimes it is found interesting to simulate exponential random graph distributions conditional on the density or, equivalently, on the total number of ties, Y_{++}. This means that still the probability function (3) is used, but only digraphs with the given number of ties are permitted. The same ideas can then be applied to define stochastic iteration steps yielding a Markov chain converging to the desired distribution, but now the steps must keep the number of ties intact. The simplest procedure is to select two elements (i,j) and (h,k), leave them unchanged if $y_{ij}^{(t)} = y_{hk}^{(t)}$, and if $y_{ij}^{(t)} \neq y_{hk}^{(t)}$ determine their new values with probabilities (18) where $\mathcal{Y}^{(t+1)}(I)$ now is the set of two adjacency matrices, one being $y^{(t)}$ and the other the same matrix but with elements (i,j) and (h,k) interchanged. The Metropolis-Hastings version of the updating step is defined in an analogous fashion.

5.6. Random undirected graphs

Another special situation is obtained when only undirected graphs are considered, i.e., the restriction is made that $y_{ij} = y_{ji}$ for all (i,j). Exponential models for undirected random graphs were studied by Frank and Strauss (1986) and Corander et al. (1998). To generate random draws of exponential models for undirected graphs, the above-mentioned techniques can be applied to the half-matrix of non-redundant elements $(y_{ij})_{i<j}$. Inversion steps still can be used.

6. ML estimation for exponential families

The exponential random graph model is a so-called exponential family of distributions with canonical sufficient statistic $u(Y)$. It is well-known (e.g., Lehmann, 1983) that the maximum likelihood (*ML*) estimate $\hat{\theta}(y)$ for an exponential family is also the solution of the moment equation

$$\mu(\theta) = u(y)\,, \tag{24}$$

where $u(y)$ is the observed value and $\mu(\theta)$ is defined as the expected value

$$\mu(\theta) = \mathrm{E}_\theta\{u(Y)\}\,;$$

that $\mu(\theta)$ is also the gradient of $\psi(\theta)$,

$$\mu_k(\theta) = \partial\psi(\theta)/\partial\theta_k\,; \tag{25}$$

that the covariance matrix $\Sigma(\theta) = \mathrm{cov}(u(Y))$ of $u(Y)$ with elements $\sigma_{hk}(\theta)$ is the matrix of derivatives of $\mu(\theta)$,

$$\sigma_{hk}(\theta) = \frac{\partial\mu_k}{\partial\theta_h} = \frac{\partial^2\psi(\theta)}{\partial\theta_h\,\partial\theta_k}\,; \tag{26}$$

and that the asymptotic covariance matrix of the ML estimator $\hat{\theta}$ is given by

$$\mathrm{cov}_\theta\left(\hat{\theta}\right) = \left(\Sigma(\theta)\right)^{-1}\,. \tag{27}$$

This theory cannot be easily applied to derive ML estimators for exponential random graph models, because the functions $\psi(\theta)$ and $\mu(\theta)$ cannot be easily calculated except for the simplest models. This was a reason in the earlier literature for proposing the pseudolikelihood estimators discussed in Section 1. In the present section first the reciprocity p^* model is briefly discussed, because it allows an explicit comparison of the ML and pseudolikelihood estimators. Subsequently a brief overview is given of algorithms for parameter estimation based on Monte Carlo simulation.

6.1. The reciprocity p^* model

For the reciprocity p^* model (9) it is possible to give explicit formulae for the ML estimates. The numbers of mutual, asymmetric, and null dyads (as defined by Holland and Leinhardt, 1976; also

see Wasserman and Faust, 1994) are given by

$$\begin{aligned} M &= \sum_{i<j} y_{ij}\, y_{ji} \\ A &= \sum_{i,j} y_{ij}\,(1-y_{ji}) \\ N &= \sum_{i<j} (1-y_{ij})\,(1-y_{ji})\ . \end{aligned}$$

The independence of the dyads implies that (M, A, N) has a multinomial distribution with multinomial denominator $\binom{g}{2}$ and probabilities given by (10). General properties of the multinomial distribution imply that ML estimates for this model are given by

$$\begin{aligned} \hat\theta_1 &= \log\left(A/(2N)\right) \\ \hat\theta_2 &= \log\left((4MN)/(A^2)\right)\ . \end{aligned} \tag{28}$$

On the other hand, the pseudologlikelihood (4) for this model is identical to the loglikelihood of $g(g-1)$ independent observations, of which $2M+A$ are Bernoulli trials with odds ratio $\exp(\theta_1+\theta_2)$ among which $2M$ successes are observed, while the other $2N+A$ are Bernoulli trials with odds ratio $\exp(\theta_1)$ among which there are A successes. The parameter estimates maximizing this pseudologlikelihood are also given by (28), but the standard errors given by the pseudolikelihood procedure are lower: in fact they are too low because of the incorrect independence assumptions which are implicit in this loglikelihood. The fact that the ML estimates can be explicitly calculated provides a possibility for checking MCMC algorithms for computing ML estimates.

Example 3.

As an example, consider the friendship relation in Krackhardt's (1987) high-tech managers data as presented also in Wasserman and Faust (1994). This is a directed graph with $g = 21$ vertices. It has 102 ties, with a dyad count of $M = 23$ mutual, $A = 56$ asymmetric, and $N = 131$ null dyads. Table 1 gives the ML and pseudolikelihood estimates for the reciprocity model, with the associated standard errors.

The results confirm that the ML and the pseudolikelihood estimates are identical, but the standard errors as obtained from the pseudolikelihood are considerably too low. The parameter estimates demonstrate a strong tendency toward reciprocity.

Table 1. Parameter estimates for the reciprocity p^* model for the friendship relation among Krackhardt's high-tech managers. Model 1: reciprocity.

	ML		pseudolikelihood	
Effect	estimate	s.e.	estimate	s.e.
Density	-1.54	0.16	-1.54	0.10
Reciprocity	1.35	0.35	1.35	0.18

6.2. Simulation-based estimation

If $\mu(\theta)$ and $\Sigma(\theta)$ would be computable, the ML estimate could be found by the Newton-Raphson algorithm with iteration step

$$\hat{\theta}^{(n+1)} = \hat{\theta}^{(n)} - \left(\Sigma(\hat{\theta}^{(n)})\right)^{-1} \left(\mu(\hat{\theta}^{(n)}) - u(y)\right) . \qquad (29)$$

The fact that $\Sigma(\theta)$ is a positive definite matrix guarantees the convergence to the solution of the ML equation. However, none of the functions $\psi(\theta)$, $\mu(\theta)$, or $\Sigma(\theta)$ can be computed in practice for exponential graph models, unless g is very small or the model is very simple (e.g., the reciprocity p^* model).

There are a variety of ways for solving intractable estimation problems by means of Monte Carlo simulation; see, e.g., McFadden (1989), Pakes and Pollard (1989), Geyer and Thompson (1992), Gilks, Richardson, and Spiegelhalter (1996), and Gouriéroux and Monfort (1996).

Geyer and Thompson (1992) give a method which can be used for approximating ML estimates in exponential families. This approach was used for parameter estimation in exponential random graph models by Corander at al. (1998) and by Crouch, Wasserman, and Trachtenberg (1998). The iteration steps in this procedure can be sketched as follows. At the current parameter value $\theta^{(n)}$, a Monte Carlo simulation of the Markov graph is made; this simulation is used to estimate cumulants (or moments) of the distribution, and these cumulants are used to make an expansion approximating $\mu(\theta)$ for θ in a neighbourhood of $\theta^{(n)}$. The moment equation then is solved using this approximation, yielding the updated provisional parameter estimate $\theta^{(n+1)}$.

The present paper uses a slightly different algorithm, which is a version of the Robbins-Monro (1951) algorithm. The Robbins-Monro algorithm may be considered to be a Monte Carlo variant of the Newton-Raphson algorithm. The use of this method for moment estimation in statistical models for networks was proposed also in Snijders (1996, 2001). I suppose the difference between the Geyer-Thompson and the Robbins-Monro approaches is a matter mainly of convenience.

7. MCMC estimation for exponential families using the Robbins-Monro algorithm

The Robbins-Monro (1951) algorithm is a stochastic iterative algorithm intended to solve equations of the form

$$\mathrm{E}\{Z_\theta\} = 0\,, \tag{30}$$

where Z_θ is a k-dimensional random variable of which the probability distribution is governed by a k-dimensional parameter θ, and where realizations of Z_θ can be observed for arbitrary values of θ. Relatively accessible introductions to the Robbins-Monro procedure and its extensions are given by Ruppert (1991) and Pflug (1996).

It is clear from this description that the Robbins-Monro algorithm can be used, in principle at least, to compute moment estimates, and therefore also maximum likelihood estimates in exponential families such as the exponential random graph model. The aim here is to solve (30), where Z_θ is given by

$$Z_\theta = u(Y) - u_0 \tag{31}$$

where $u_0 = u(y)$ is the observed value of the sufficient statistic and Y has probability distribution (3) with parameter θ. The combination of equations (30) and (31) is equivalent to the moment equation (24).

The iteration step in the Robbins-Monro (1951) procedure for solving (30) (the multivariate version is from Nevel'son and Has'minskii, 1973), with step-size a_n, is

$$\hat\theta^{(n+1)} = \hat\theta^{(n)} - a_n D_n^{-1} Z(n)\,, \tag{32}$$

where $Z(n)$ for $n = 1, 2, ...$ are random variables such that the conditional distribution of $Z(n)$ given $Z(1), ..., Z(n-1)$ is the

distribution of Z_θ obtained for $\theta = \hat{\theta}^{(n)}$. The step sizes a_n are a sequence of positive numbers converging to 0. This is called the *gain sequence*.

The stochastic process $\hat{\theta}^{(n)}\,(n = 1, 2, ...)$ generated by this Monte Carlo simulation process is a Markov chain, because for any n_0 the sequence $\{\hat{\theta}^{(n)}\}$ for $n > n_0$ depends on the past values via the last value, $\{\hat{\theta}^{(n_0)}\}$. Therefore this approach may be called a frequentist Markov chain Monte Carlo (MCMC) method, although the name MCMC is used more frequently for Bayesian procedures.

For the discussion in the present section it is assumed that a procedure is available to generate Monte Carlo draws $Y^{(n)}$ from the exponential random graph distribution with arbitrary parameter θ. The preceding sections imply that the availability of such a procedure is not always evident, and will depend on the choice of the vector of sufficient statistics $u(Y)$.

For the classical choice $a_n = 1/n$ in the Robbins-Monro updating step, the optimal value of D_n is the derivative matrix $D_\theta = (\partial E_\theta Z/\partial\theta)$. In our case, this matrix is given by (26). In adaptive Robbins-Monro procedures (Venter, 1967; Nevel'son and Has'minskii, 1973), this derivative matrix is estimated during the approximation process. If D_n is a consistent estimator for D_θ and if certain regularity conditions are satisfied (see Ruppert, 1991), then $\hat{\theta}^{(n)}$ is asymptotically multivariate normal, with the solution of (30) as its mean, and

$$\frac{1}{n}\, D_\theta^{-1}\, \mathrm{cov}_\theta(Z)\, {D'_\theta}^{-1} \tag{33}$$

as its covariance matrix. This is the optimal asymptotic covariance matrix possible for this kind of stochastic approximation problem (see Ruppert, 1991, and Pflug, 1996). A particular property of the present estimation problem, due to the fact that the probability model is an exponential family, is that $\mathrm{cov}_\theta(Z) = D_\theta$ so that the asymptotic covariance matrix (33) of the approximation by the Robbins-Monro algorithm is $1/n$ times the estimation covariance matrix (27) of the ML estimator.

Instead of using an adaptive version of the Robbins-Monro algorithm, one may also follow a procedure proposed by Polyak (1990) and Ruppert (1988) (also see Pflug, 1996, Section 5.1.3, and Kushner and Yin, 1997). They showed that, under conditions which are satisfied here because the matrix of derivatives (26) is

positive definite, the optimal rate of convergence can also be obtained while using a constant positive diagonal matrix $D_n = D_0$. The sequence $a_n > 0$ then must be chosen as n^{-c}, for a c between 0.5 and 1.0. Further, the estimate of θ must be not the last value $\hat{\theta}^{(n)}$, but the average of the sequence $\hat{\theta}^{(n)}$ or the average of the 'last part' of this sequence.

These ideas were used also in Snijders (2001) to construct a Robbins-Monro type algorithm for parameter estimation in an actor-oriented model for network evolution. The same algorithm is used here, but with a simplification possible because in this case the matrix of derivatives can be estimated by the covariance matrix of the generated statistics rather than by a finite difference quotient. The algorithm is presented explicitly in the appendix. A more extensive motivation for the specification of the algorithm can be found in Snijders (2001).

The algorithm consists of three 'phases'. The *first phase* is used to determine the diagonal matrix $D_n = D_0$ (independent of n) to be used later in the updating steps (32). This matrix is a diagonal matrix. Its diagonal elements are estimates of the derivatives $d_{kk} = \partial \mathrm{E}_\theta u_k(Y)/\partial\theta_k$ evaluated in the initial value θ of the estimation algorithm. The *second phase* iteratively determines provisional estimated values according to the updating steps (32). It consists of several subphases, as explained in the appendix. The gain values a_n are constant within subphases and decrease between subphases. Reasonable values for the first gain value a_1 are $0.001 \leq a_1 \leq 0.1$. A large value of a_1 will lead the process quickly from almost any reasonable starting value into the vicinity of the solution of the moment equation, but also leads to bigger steps, and therefore perhaps greater instability, once this vicinity is reached. Therefore it is advisable to use a rather large a_1 (e.g., $a_1 = 0.1$) if the initial value $\hat{\theta}_1$ could be far from the ML estimate, and a rather small value (e.g., 0.01) if the initial value is supposed to be close already.

In the *third phase* the parameter value is kept constant at $\hat{\theta}$, the presumably found approximate solution of the moment equation (24). A large number of steps is carried out to check the approximate validity of this equation. This is done by estimating from these simulations the expected value and standard deviation

of each statistic $u_k(Y)$ and from this computing the t-ratios

$$t_k = \frac{\widehat{\mathrm{E}}_{\hat\theta}(u_k(Y)) - u_0}{\widehat{\mathrm{SD}}_{\hat\theta}(u_k(Y))} \,. \tag{34}$$

If $|t_k| \leq 0.1$, the convergence is considered to be excellent; for $0.1 < |t_k| \leq 0.2$ it is good and for $0.2 < |t_k| \leq 0.3$ it is fair.

Moreover, Phase three is used to estimate the covariance matrix $\Sigma(\hat\theta) = \mathrm{cov}_{\hat\theta}(u(Y))$, which then is used to estimate the standard errors of the estimate $\hat\theta$ according to (27).

7.1. Measures to improve convergence

The combination of the Monte Carlo simulation methods explained above with the Robbins-Monro method yields, at least in theory, a procedure which convergences with probability 1 to the ML estimate of θ, provided that we are willing to continue long enough with the Robbins-Monro algorithm and, within each step in this algorithm, continue long enough with the simulations to approximate a sample from the exponential random graph distribution. Whether this is a satisfactory procedure will depend on the data set and the model specification. In my experience, the use of inversion steps as proposed in Section 5.3 has proved important to obtain reasonable convergence in models involving triplet counts.

The existence of multiple regimes and the possibility of very long sojourn times in these regimes, however, indicates that one must be wary of convergence difficulties. Any MCMC procedure for this estimation problem that does not take into account the possibility of these regime changes may give unreliable results for certain data sets.

In the specification of the algorithm as given in the appendix, a small value is used for a_1 , not greater than 0.1, to avoid instability of the algorithm. If the result of Phase 3 of the algorithm is that the found value $\hat\theta$ is rather close to being a solution of the likelihood equation (24) but not close enough, then it is advisable to restart the algorithm from this initial value $\hat\theta$ with a very small step size such as given by $a_1 = 0.01$ or even smaller.

The example of model (7) with the updating steps depending on the conditional probability (8) illustrates that for models depending on some subgraph count (in this case, counts of transitive

triplets), the 'sub-subgraphs' which are part of this subgraph (in this case, twostars and twopaths) can function like a lubricant to help the process moving in the good direction. This is an algorithmic motive for the rule, which also is sensible from the point of view of statistical modeling, that whenever some subgraph count is included in the sufficient statistic $u(Y)$, also the counts for the corresponding sub-subgraphs should be included in $u(Y)$.

Experience up to now with this algorithm is rather mixed. For some data sets and models, good estimates (as judged by the t-ratios (34)) have been obtained. The next section presents some examples. For many other data sets the algorithm was less succesful. As far as this is due to the unknown ML estimate being in the subset of the parameter space where the graph has a bimodal distribution, there is no easy way out. In such a case, one might look for a model specification, obtained by extending the vector of sufficient statistics, for which the unknown ML estimate corresponds to a unimodal distribution. However, there may be other reasons for lack of convergence. In the first place, the MCMC simulations may have been too short to approximate a random draw from the exponential graph distribution with the given parameter value. This may be solved by using more updating steps, or updating by more efficient steps. It may be possible to obtain improvement here by developing other 'big steps' in addition to inversions.

In the second place, the Robbins-Monro algorithm may be unstable, e.g., because of high correlations between the sufficient statistics $u_k(Y)$, or because of long-tailed distributions of these statistics, or because the covariance matrix $\mathrm{cov}_\theta\,(u(Y))$ changes very quickly as a function of θ. In this case it is advisable to restart the algorithm from the currently found parameter value but with a smaller initial step size a_1 .

8. Examples

This section presents examples for a directed graph and an undirected graph. The presented results have all t-ratios (34) less than 0.1, indicating good convergence.

The first example continues the investigation of the friendship relation among Krackhardt's (1987) high-tech managers as presented in Wasserman and Faust (1994). The density of this graph is 0.24. The MCMC estimation of the reciprocity model converged quickly to the ML estimates presented in Table 1, thus providing some confidence in the method and the computer program.

As a next result, the estimates are presented for the model containing the following seven effects (sufficient statistics): numbers of ties, of mutual dyads, of out-twostars, of in-twostars, of twopaths, of transitive triplets, and of three-cycles. The mathematical definitions of the corresponding sufficient statistics are

$$
\begin{aligned}
u_1(y) &= y_{++} = \sum_{i,j} y_{ij} \\
u_2(y) &= \sum_{i<j} y_{ij}\, y_{ji} \\
u_3(y) &= \sum_i \binom{y_{i+}}{2} \\
u_4(y) &= \sum_i \binom{y_{+i}}{2} \\
u_5(y) &= \sum_i y_{+i}\, y_{i+} \\
u_6(y) &= \sum_{i,j,h} y_{ij}\, y_{jh}\, y_{ih} \\
u_7(y) &= \sum_{i<j<h} y_{ih}\, y_{hj}\, y_{ji} \ .
\end{aligned}
$$

The simulations were carried out with $60,000$ Metropolis-Hastings updating steps in each simulation to generate a digraph. Simulations with Gibbs steps or with updates of dyads or triplets did not differ greatly from the Metropolis-Hastings steps.

Table 2 presents the MCMC and pseudolikelihood estimates. They are quite different; for the three-cycles effect, the signs of the estimates even are opposite. The pseudolikelihood results appear to be totally unreliable for this dataset. For this small data set, and controlling for all these effects, the MCMC results show evidence only of the mutuality effect (more reciprocated ties

than if this effect were nil) and the out-twostars effect (higher dispersion of out-degrees than if this effect were nil).

Table 2. MCMC and pseudolikelihood parameter estimates for the friendship relation in Krackhardt's high-tech managers. Model 2: structural effects.

	MCMC		pseudolikelihood	
Parameter	estimate	s.e.	estimate	s.e.
Number of ties	-2.066	0.656	-2.538	0.399
Mutual ties	2.035	0.437	2.569	0.291
Out-twostars	0.219	0.049	0.210	0.035
In-twostars	-0.025	0.110	0.079	0.058
Twopaths	-0.104	0.066	-0.208	0.040
Transitive triplets	0.070	0.087	0.205	0.054
Three-cycles	-0.004	0.225	0.168	0.121

When the values of some of the coordinates of these estimates are increased only slightly, the graphs generated according to these changed parameters have densities close to 1. This shows that the ML estimate is very close to the region where the expected density $\mathrm{E}_\theta(u_1(Y))$ increases extremely rapidly from a low to a high value. This phenomenon was found for quite many data sets for which this type of model was fitted.

As a next example, the symmetrized version of the Sampson (1969) monastery data is used that was also considered by Frank and Strauss (1986), and presented in their Figure 7. The Markov undirected graph model, defined by (1) and (2), was fitted. The sufficient statistics are the numbers of edges, two-stars, and triangles. For this undirected graph with $g = 18$ nodes, there are 61 edges, 393 twostars, and 70 triangles. The MCMC and pseudolikelihood estimates are presented in Table 3. The pseudolikelihood estimates appear to differ strongly from the MCMC estimates.

The three statistics are extremely highly correlated. This is reflected by the estimated correlation matrix of the estimates,

$$\widehat{\mathrm{cor}}\begin{pmatrix}\hat\theta_1\\ \hat\theta_2\\ \hat\theta_3\end{pmatrix} = \begin{pmatrix} 1.0 & -0.982 & 0.947\\ -0.982 & 1.0 & -0.990\\ 0.947 & -0.990 & 1.0\end{pmatrix}.$$

Table 3. MCMC parameter estimates for the undirected Sampson data set, Markov graph model.

	MCMC		pseudolikelihood	
Effect	estimate	s.e.	estimate	s.e.
Edges	-0.77476	0.88	-0.0810	0.66
Twostars	-0.04875	0.11	-0.2285	0.06
Triangles	0.34520	0.20	0.9754	0.12

An eigenvalue decomposition of the estimated covariance matrix of the three parameter estimates shows that the eigenvector with the smallest eigenvalue corresponds to the linear combination $(0.0515\,\hat{\theta}_1 + 0.9523\,\hat{\theta}_2 + 0.3007\,\hat{\theta}_3)$ which has standard error 0.0028, much smaller than the standard errors of the three individual estimates. Thus, this linear combination is known more precisely than is suggested by the standard errors of the individual estimates. This implies that, in order to get a parameter estimate which accurately solves the likelihood equation, the coordinates should be retained in at least 4 and preferably 5 decimal figures, notwithstanding the large standard errors of the individual estimates. It also implies that quite different values could be obtained as approximate ML estimates, but this particular linear combination would have to be almost the same (making some allowance for the inaccuracy with which these eigenvectors and eigenvalues are estimated).

9. Discussion

This article considered MCMC simulation and parameter estimation procedures for exponential random graph models, also called p^* models. In this discussion, a distinction must be made between MCMC *simulation* algorithms, which employ a fixed parameter value and have the purpose to simulate a random draw from the exponential graph distribution with this parameter value, and MCMC *estimation* algorithms, which have the purpose to estimate the parameter for a given data set and thereby repeatedly use MCMC simulation algorithms.

Consideration was given first to applying the usual type of

MCMC algorithms for *simulating* a sequence converging to a given exponential random graph distribution, such as the Gibbs sampler and the Metropolis-Hastings algorithm, using small steps where only one arc variable (one cell entry of the adjacency matrix) is updated in each step. It was shown that for certain specifications of the sufficient statistics and the parameter values, the graph density and many other statistics have bimodal or perhaps multimodal distributions. This happens especially if the sufficient statistics include subgraph counts such as triad counts. In rather many cases there are two modes, one for a high graph density and the other for a low graph density. MCMC algorithms that change one or just a few arc variables per step may require an exceedingly high number of iteration steps to go from outcomes near one mode to outcomes near the other mode. However, such switches are necessary to estimate the relative probabilities of the several modes. This rules out such small-step MCMC algorithms as a way to estimate expected values of statistics of exponential random graphs for these specifications of sufficient statistics and parameter values. It was proposed to augment small-step MCMC algorithms by occasionally changing the graph to its complement ('inversion steps'), using probabilities that still guarantee asymptotic convergence to the desired graph distribution. For some model specifications this solves the convergence problems posed by bimodal distributions. Whether one, two, or three arc variables are changed in the 'small steps' did not have a great effect on the efficiency of the algorithm.

MCMC algorithms for parameter *estimation* in these models are even more complex, because they employ the MCMC simulation algorithm many times, with different parameter values. This paper considered an MCMC estimation method based on the Robbins-Monro algorithm. If the true ML estimate is within or near the part of the parameter space where the graph density has a bimodal distribution, the algorithm may be unstable. Then it is advised to use very small step sizes.

The quality of the estimates produced by the algorithm can be evaluated by the t-statistics (34). Judging by this evaluation, it appears that the proposed algorithm performs well for some data sets and models, but there are other data sets for which it fails to find an estimate that solves the moment equation to a satisfactory degree of precision. More research is needed to find good practical

methods for estimating the parameters of this type of model.

Another problem for the estimation procedure is posed by the extremely high correlations of the subgraph counts that are often used in applications of the p^* model. This property leads, however, to accuracy problems rather than essential problems. It is conceivable that approximate orthogonalisation of the sufficient statistics could be used to solve this problem.

The algorithm proposed in this article is included in the `SIENA` program (version 1.94 and higher) which is available on the web at `http://stat.gamma.rug.nl/snijders/socnet.htm`.

Some attention was paid also to the pseudolikelihood estimates which are currently the most often used method for modeling social networks by p^* models. In some examples, these estimates were quite far from the maximum likelihood estimates. In another example, the estimates were identical but the standard errors obtained for the pseudolikelihood method were considerably too large. It seems to me that the basis for the use of the pseudolikelihood estimators is so weak that their use should not be advocated until more research has been done supporting their quality.

In addition to posing technical problems for parameter estimation, however, the bimodality of many exponential distributions for random graphs also poses a general conceptual problem in their use for modeling networks. Bimodal distributions do not in general create a problem if a sample from the distribution is available, but they do if only one outcome is observed, such as is usual for applications in social network analysis. When these exponential models for random graphs are applied to a single observed social network, it is important to establish that the parameter estimates represent the observed network, and do not just as well represent a totally different network (as could happen if the parameter estimates correspond to a bimodal distribution). A counter-argument could be that social reality often is equivocal, and that a given set of local rules for relationship preferences, as embodied in the parameter values, could correspond to very different network structures, as embodied in the observed data. This would be a social network version of the long-range dependence known from the Ising model. To take this argument seriously, however, more research is needed investigating the relation between micro-processes and network macro-structures, and how

these two aspects can be reflected in statistical models for social networks.

For the appropriateness of using exponential random graph models for modeling single observations of social networks, in any case more research is required to obtain insight in whether and how it is possible to specify such models so that fitted distributions are unimodal; or, at least, have a dominant mode that is close to the observed network.

APPENDIX: STOCHASTIC APPROXIMATION ALGORITHM

The algorithm consists of three phases. It is very similar to the algorithm of Snijders (2001), except that Phases 1 and 3 are simplifications possible here due to the special structure of the estimation problem (the exponential family), and in all phases inversion is used to reduce estimation variance as indicated by (23). For further motivation and some background discussion, the reader is referred to the mentioned publication.

Steps in the iteration process are indicated by n. The digraph $Y(n)$ is assumed in each step to be generated by a Monte Carlo method according to the exponential random graph model with parameter θ dependent on the current step. The MCMC algorithm used to generate each $Y(n)$ starts with a random graph in which each arc variable Y_{ij} is determined independently with a probability 0.5 for the values 0 and 1; it uses Gibbs or Metropolis Hastings updating steps by single arcs, dyads, or triplets; and it has a probability of 0.01 for inversion steps. The number of steps for generating each $Y(n)$ is in the order of $100\,g^2$. Depending on

the success of the algorithm as exhibited in Phase 3, this number can be decreased or increased. Given $Y(n)$, denote

$$P(n) \;=\; p_{\mathrm{r}}(Y(n))$$

as defined in (20).

The initial value for the algorithm is denoted $\theta^{(1)}$. The observed value of the sufficient statistic is u_0.

Phase 1. In this phase a small number N_1 of steps are made to estimate $D(\theta^{(1)}) = \mathrm{cov}_{\theta^{(1)}} u(Y)$.
Generate N_1 independent networks $Y(n)$ according to parameter $\theta^{(1)}$ and define

$$\begin{aligned}
\bar{u} \;&=\; \frac{1}{N_1}\sum_{n=1}^{N_1}\Big(P(n)u(\mathbf{1}-Y(n)) \;+\; (1-P(n))u(Y(n))\Big)\;,\\
D \;&=\; \frac{1}{N_1}\sum_{n=1}^{N_1}\Big(P(n)u(\mathbf{1}-Y(n))'u(\mathbf{1}-Y(n))\\
&\qquad\qquad +\; (1-P(n))u(Y(n))'u(Y(n)) \;-\; \bar{u}'\bar{u}\Big)\;,
\end{aligned}$$

and $D_0 = \mathrm{diag}(D)$.

At the end of this phase, an optional possibility is to make one partial estimated Newton-Raphson step,

$$\hat{\theta}^{(N_1)} \;=\; \theta^{(1)} - a_1\, D^{-1}\,(\bar{u} - u_0)\;.$$

Phase 2. This is the main phase. It consists of several subphases. In each iteration step within each subphase, $Y(n)$ is generated according to the the current parameter value $\hat{\theta}^{(n)}$ and after each step this value is updated according to the formula

$$\hat{\theta}^{(n+1)} = \hat{\theta}^{(n)} \;-\; a_n\, D_0^{-1}\, Z(n) \tag{35}$$

where

$$Z_k(n) \;=\; P(n)\,u(\mathbf{1}-Y(n)) \;+\; (1-P(n))\,u(Y(n)) \;-\; u_0\;.$$

The value of a_n is constant within each subphase.
The number of iteration steps per subphase is determined by a stopping rule, but bounded for subphase k by a minimum value N_{2k}^{-} and a maximum value N_{2k}^{+}. The subphase is ended after less than N_{2k}^{+} steps as soon as the number of steps in this

subphase exceeds N_{2k}^- while, for each coordinate k, the sum within this subphase of successive products $Z_k(n+1)\,Z_k(n)$ is negative. If the upper bound N_{2k}^+ is reached, then the subphase is terminated anyway.

At the end of each subphase, the average of $\hat{\theta}^{(n)}$ over this subphase is used as the new value for $\hat{\theta}^{(n)}$.

The value of a_n is divided by 2 when a new subphase is entered. The bounds N_{2k}^- and N_{2k}^+ are determined so that $n^{3/4}a_n$ tends to a finite positive limit.

The average of $\hat{\theta}^{(n)}$ over the last subphase is the eventual estimate $\hat{\theta}$.

Phase 3. Phase 3 is used only for the estimation of $\Sigma(\theta)$ and the covariance matrix of the estimator, and as a check for the approximate validity of (24). Therefore the value of $\hat{\theta}^{(n)}$ is left unchanged in this phase and is equal to the value $\hat{\theta}$ obtained after last subphase of phase 2. The procedure further is as in phase 1. The number of iterations is N_3. The covariance matrix of $u(Y)$, required for the calculation of (27), is estimated in the same way as D in Phase 1.

For the numbers of steps, similar values are proposed as in Snijders (2001): $N_1 = 7 + 3p$, $N_{2k}^- = 2^{4(k-1)/3}(7+p)$, $N_{2k}^+ = N_{2k}^- + 200$, and $N_3 = 1000$. The initial value of a_n in phase 2 is 0.1 and the number of subphases is proposed to be 4. If the initial value $\theta^{(1)}$ is known to be rather close to the solution, it is advised to use $a_1 = 0.01$.

REFERENCES

Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society*, ser. B, 36, 192 – 225.

Besag, J. (1975). Statistical analysis of non-lattice data. *The Statistician*, 24, 179 – 195.

Besag, J. (2000). *Markov chain Monte Carlo for statistical inference.* Center for Statistics and the Social Sciences,

University of Washington, Working Paper No. 9.
Obtainable from http://www.csss.washington.edu/Papers/.

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

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.

Dahmström, K., and Dahmström, P. (1993). *ML-estimation of the clustering parameter in a Markov graph model*. Stockholm: Research report, Department of Statistics.

Doreian, P., and Stokman, F.N. (eds.) (1997). *Evolution of Social Networks*. Amsterdam etc.: Gordon and Breach.

Frank, O. 1991. Statistical analysis of change in networks. *Statistica Neerlandica*, 45, 283 – 293.

Frank, O., and D. Strauss. 1986. Markov graphs. *Journal of the American Statistical Association*, 81, 832 – 842.

Geman, S., and Geman, D. (1983). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6, 721 – 741.

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.

Gilks, W.R., Richardson, S., and Spiegelhalter, D.J. (1996). *Markov Chain Monte Carlo in Practice*, London: Chapman and Hall.

Gouriéroux, C., and Monfort, A. (1996). *Simulation-based econometric methods*. Oxford: Oxford University Press.

Holland, P.W., and Leinhardt, S. (1976). Local structure in social networks. In D. Heise (ed.), *Sociological Methodology*. San Francisco: Jossey-Bass.

Krackhardt, D. (1987). Cognitive social structures. *Social Networks*, 9, 109 – 134.

Kushner, H.J., and Yin, G.G. (1997). *Stochastic Approximation: Algorithms and Applications*. New York: Springer.

Lehmann, E.L. (1983). *Theory of Point Estimation*. New York: Wiley.

McFadden, D. (1989). A method of simulated moments for estimation of discrete response models without numerical integration. *Econometrica*, 57, 995 – 1026.

Nevel'son, M.B., and Has'minskii, R.A. (1973). "An adaptive Robbins-Monro procedure", *Automatic and Remote Control*, 34, 1594 – 1607.

Newman, M.E.J., and Barkema, G.T. (1999). *Monte Carlo methods in statistical physics.* Oxford: Clarendon Press.

Norris, J.R. (1997). *Markov Chains*. Cambridge: Cambridge University Press.

Pakes, A., and Pollard, D. (1989). The asymptotic distribution of simulation experiments, *Econometrica*, 57, 1027 – 1057.

Pattison, P., and Wasserman, S. (1999). Logit models and logistic regressions for social networks: II. Multivariate relations. *British Journal of Mathematical and Statistical Psychology*, 52, 169 – 193.

Pflug, G.Ch. (1996). *Optimization of Stochastic Models*. Boston: Kluwer.

Polyak, B.T. (1990). New method of stochastic approximation type. *Automation and Remote Control*, 51, 937 – 946.

Ripley, B.D. (1987). *Stochastic Simulation*. New York: Wiley.

Robbins, H., and Monro,S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, 22, 400 – 407.

Robins, G., Pattison, P., and Wasserman, S. (1999). Logit models and logistic regressions for social networks, III. Valued relations. *Psychometrika*, 64, 371 – 394.

Ruppert, D. (1988). *Efficient estimation from a slowly convergent Robbins-Monro process*. Technical Report no. 781, School of Operations Research and Industrial Engineering, Cornell University.

Ruppert, D. (1991). Stochastic approximation. In *Handbook of Sequential Analysis* edited by Gosh, B.K., and P.K. Sen. New York: Marcel Dekker.

Snijders, T.A.B. (1996). Stochastic actor-oriented models for network change. *Journal of Mathematical Sociology*, 21, 149 – 172. Also published in Doreian and Stokman (1997).

Snijders, T.A.B. (2001). The Statistical Evaluation of Social Network Dynamics. *Sociological Methodology - 2001*. Obtainable from http://stat.gamma.rug.nl/snijders/socnet.htm.

Strauss, D. (1986). On a general class of models for interaction. *SIAM Review*, 28, 513 – 527.

Strauss, D., and Ikeda, M. (1990). Pseudolikelihood estimation for social networks. *Journal of the American Statistical Association*, 85, 204 – 212.

Tierney, L. (1994). Markov chains for exploring posterior distributions (with Discussion). *Annals of Statistics*, 22, 1701 – 1762.

Venter, J.H. (1967). An extension of the Robbins-Monro procedure. *Annals of Mathematical Statistics*, 38, 181 – 190.

Wasserman, S., and Faust, K. (1994). *Social Network Analysis: Methods and Applications*. New York and 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.