Snowball sampling for estimating exponential random graph models for large networks

Alex D. Stivala[a,*], Johan H. Koskinen[b], David A. Rolls[a], Peng Wang[a], Garry L. Robins[a]

[a] *Melbourne School of Psychological Sciences, The University of Melbourne, Australia*
[b] *The Mitchell Centre for SNA, and Social Statistics Discipline Area, University of Manchester, United Kingdom*

[*] Corresponding author at: Melbourne School of Psychological Sciences, The University of Melbourne, Victoria 3010, Australia. Fax: +61 3 9347 6618. Tel.: +61 3 8344 7035.
Email address: stivalaa@unimelb.edu.au (Alex D. Stivala)

Preprint submitted to Social Networks *November 13, 2015*

Abstract

The exponential random graph model (ERGM) is a well-established statistical approach to modelling social network data. However, Monte Carlo estimation of ERGM parameters is a computationally intensive procedure that imposes severe limits on the size of full networks that can be fitted. We demonstrate the use of snowball sampling and conditional estimation to estimate ERGM parameters for large networks, with the specific goal of studying the validity of inference about the presence of such effects as network closure and attribute homophily. We estimate parameters for snowball samples from the network in parallel, and combine the estimates with a meta-analysis procedure. We assess the accuracy of this method by applying it to simulated networks with known parameters, and also demonstrate its application to networks that are too large (over 40 000 nodes) to estimate social circuit and other more advanced ERGM specifications directly. We conclude that this approach offers reliable inference for closure and homophily.

Keywords: Exponential random graph model (ERGM), Snowball sampling, Parallel computing

1. Introduction

Exponential random graph models (ERGMs), first introduced by Frank and Strauss (1986), are a class of statistical model that are useful for modelling social networks (Lusher et al., 2013). Since their introduction, a body of work has been developed around ERGM theory and practice, including the introduction of new specifications for modelling social networks (e.g., Snijders et al., 2006; Robins et al., 2007; Goodreau, 2007), and more sophisticated methods for estimating ERGM parameters (e.g., Snijders, 2002; Handcock et al., 2008; Wang et al., 2009; Caimo and Friel, 2011; Hummel et al., 2012). Originally, the most common method for estimating ERGM parameters was maximum pseudo-likelihood (Strauss and Ikeda, 1990). More recently, Markov chain Monte Carlo maximum likelihood estimation (MCMCMLE) (Corander et al., 1998; Snijders, 2002; Corander et al., 2002; Hunter and Handcock, 2006) has become the preferred method (Robins et al., 2007). These techniques have several advantages over maximum pseudo-likelihood:

if the estimation does not converge, a degenerate model is likely (a situation that maximum pseudo-likelihood does not indicate); converged estimates can be used to produce distributions of graphs in which the observed graph is typical for all effects in the model; reliable standard errors for the estimates are obtained (Robins et al., 2007); and point estimates are more accurate (Snijders, 2002).

Such MCMCMLE techniques require the generation of a distribution of random graphs by a stochastic simulation process. This process, which requires a number of iterations to "burn in" the Markov chain, as well as a large number of iterations to generate samples that are not too auto-correlated, is computationally intensive, and scales (at least) quadratically in the number of nodes in the network. This limits the size of networks to which an ERGM can be fitted in a practical time. Furthermore, this process is inherently sequential (although several chains can be run in parallel, they each must be burned in), which limits the ability to take advantage of the parallel computing power available in modern high performance computing resources.

In this paper, we show how to fit ERGMs to certain large networks where model fitting using standard MCMC procedures would be impractical or impossible. The key idea takes advantage of recent developments in conditional estimation for ERGMs (Pattison et al., 2013) to take multiple snowball samples, estimate ERGM parameters for each sample in parallel, and combine the results with meta-analysis.

To the best of our knowledge, the work of Xu et al. (2013) is the first to take a similar approach. Xu et al. use a special data-intensive supercomputer to estimate an ERGM for a Twitter "unfollow" network with over 200 000 nodes, estimating each of nearly 400 samples in parallel (by running statnet (Handcock et al., 2008) independently on each sample), and combining the results with meta-analysis (Snijders and Baerveldt, 2003). However, as Pattison et al. (2013) show, simply estimating the parameters of snowball samples without taking account of the snowball sampling structure, and assuming they are estimates of the full network, can lead to quite incorrect estimates. The issue is that, for a large class of models, standard parameter estimates for a graph are dependent on the number of nodes N and do not scale up in a consistent manner as N increases (Rolls et al., 2013; Shalizi and Rinaldo, 2013). Further, the Xu et al. (2013) method is applied only to the single Twitter unfollow network, for which the true values are not known, so there can be no comparison of true and estimated parameters, and therefore the reliability of the parameters obtained from their meta-analysis could not

be assessed.

The motivations for fitting ERGMs to data are several. Usually, the aim is to infer whether certain well-established network processes that lead to tie creation are consistent with the data, and to parse apart different processes that might be operating simultaneously. This can be done by parameterizing competing explanatory processes and then inferring which of these are significant (Lusher et al., 2013). But, further, with precise parameter estimates, simulation of the model results in a distribution of graphs that can be interpreted as consistent with the data (at least in regards to the fitted effects). This distribution can be treated as the range of plausible graphs in a population of networks, from which a number of conclusions may be drawn. For instance, the population might relate to school classrooms or to communities of drug users (Rolls et al., 2013).

With very large data, however, the second motivation is often of less concern, because the idea of a "population" of large data is not always coherent. (There is for instance only one world wide web, not a population.) In this case, the interest is more typically on understanding the network processes within the data, such as closure and homophily. In this article, then, we are most interested in the validity of statistical inference for our procedure and hence we focus on type I and type II errors in our results.

2. Exponential random graph models

Under a homogeneity assumption whereby parameters are equated for all structurally identical subgraphs, an ERGM is a probability distribution with the general form

$$\Pr(X = x) = \frac{1}{\kappa} \exp\left(\sum_A \theta_A z_A(x)\right) \tag{1}$$

where

- $X = [X_{ij}]$ is a 0-1 matrix of random tie variables,
- x is a realization of X,
- A is a *configuration*, a (small) set of nodes and a subset of ties between them,
- $z_A(x)$ is the network statistic for configuration A,

- θ_A is a model parameter corresponding to configuration A,
- κ is a normalizing constant to ensure a proper distribution.

In the present work we will be using only undirected graphs, so the matrix X is symmetric. Assumptions about which ties are independent, and therefore the configurations A allowed in the model, determine the class of model.

In the simplest case, where all tie variables are assumed to be independent, the ERGM reduces to a Bernoulli random graph distribution, otherwise known as a simple random graph or Erdős-Renyi random graph (Gilbert, 1959). In such a model only one configuration is used, an edge between two nodes, with the network statistic $z_L(x)$, the number of edges in the network, and the corresponding parameter θ_L.

The Markov dependence assumption, that two tie variables are conditionally independent unless they have a node in common, leads to the class of *Markov random graphs* (Frank and Strauss, 1986). In such models, the subgraph configurations include stars (in which a node has ties to two or more other nodes) and triangles (three mutually connected nodes). Stars can be further categorized as 2-stars, a subset of three nodes in which one node is connected to each of the other two, 3-stars, a subset of four nodes in which one node is connected to each of the other three, and so on, in general giving k-stars. Note that configurations are nested inside each other, for example a triangle contains three 2-stars. Associated with these is the *alternating k-star* statistic (Snijders et al., 2006), which is a weighted sum of the number of k-stars from $k = 2$ to $k = N - 1$ (where N is the number of nodes), with the sign alternating:

$$z_{AS} = \sum_{k=2}^{N-1} (-1)^k \frac{S_k}{\lambda^{k-2}} \tag{2}$$

where S_k is the number of k-stars and $\lambda \geq 1$ is a damping parameter which reduces the impact of higher order stars as it is increased. The alternating star parameter provides modelling flexibility in fitting node degree distributions, and alleviates model degeneracy. Throughout we use $\lambda = 2$, as suggested by Snijders et al. (2006) and modelling experience.

A more general class of model is based on *social circuit dependence* (Snijders et al., 2006; Robins et al., 2007) and often parameterized with *higher order parameters* such as the *alternating k-triangle* and *alternating k-two-path* (or alternating two-path) statistics. A k-triangle is a combination of k

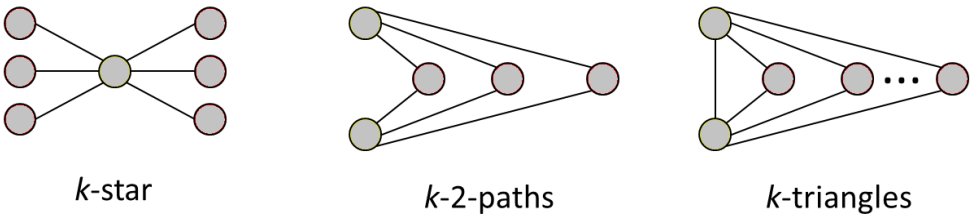

Figure 1: The structural configurations related to the alternating k-star (AS), alternating k-two-path (A2P) and alternating k-triangle (AT) statistics.

individual triangles which all share one edge, useful for modelling transitivity in the network. The alternating k-triangle statistic was defined in Snijders et al. (2006), and can be expressed as:

$$z_{AT} = 3T_1 + \sum_{k=1}^{N-3} (-1)^k \frac{T_{k+1}}{\lambda^k} \tag{3}$$

where T_k is the number of k-triangles. Again, we set the damping parameter to be $\lambda = 2$ throughout.

The k-two-path configuration is the number of distinct paths of length two between a pair of nodes, equivalent to the k-triangle configuration without the common (or "base") edge. Analogous to the alternating k-star and alternating k-triangle statistics, the alternating k-two-path statistic was defined in Snijders et al. (2006), and can be expressed as:

$$z_{A2P} = P_1 - \frac{2P_2}{\lambda} + \sum_{k=3}^{N-2} \left(\frac{-1}{\lambda}\right)^{k-1} P_k \tag{4}$$

where P_k is the number of k-two-paths. We use $\lambda = 2$ throughout.

These configurations are illustrated in Figure 1. Software to fit and simulate ERGMs using these configurations includes PNet (Wang et al., 2009) and statnet (Handcock et al., 2008).

The alternating k-triangle and alternating k-two-path statistics can also be expressed in terms of edgewise and dyadic shared partners as the "geometrically weighted edgewise shared partner" (GWESP) and "geometrically weighted dyadic shared partner" (GWDSP) statistics, respectively (Hunter, 2007). The statnet software package (Handcock et al., 2008) uses GWESP and GWDSP rather than alternating k-triangle and alternating k-two-path statistics by default (Hunter, 2007).

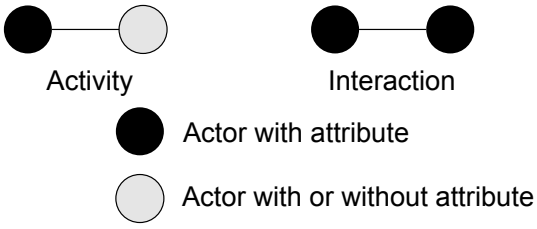

Figure 2: The binary attribute effects activity (ρ) and interaction (ρ_B).

All the configurations discussed so far have been structural, without the consideration of nodal attributes. In addition we wish to consider how an attribute (covariate) on a node can affect the network structure. Attributes may be used in order to relax homogeneity assumptions (Robins et al., 2001). Here we will restrict ourselves to a single binary attribute on each node, that either takes the value True or False, and consider two attribute related configurations: *activity* and *interaction*.

Activity measures the increased propensity for a node with the True value of the attribute to form a tie (regardless of the attribute of the other node). Hence its statistic is a count of the number of nodes which have a tie and have the True value of the attribute. We denote its corresponding parameter as ρ.

Interaction measures the increased propensity for a node with the True value of the attribute to form a tie to another node also with the True attribute value. Its statistic is therefore a count of the number of ties between two nodes both with the True value of the attribute. We denote its corresponding parameter as ρ_B. The relationship between the activity and interaction parameters is discussed in Section 6. The activity and interaction configurations are shown in Figure 2.

In this paper we will work under the social circuit dependence assumption, and use the parameters Edge (θ_L), alternating k-star (AS), alternating k-triangle (AT), alternating two-path (A2P), activity (ρ) and interaction (ρ_B).

Normally, all six (or four when there are no node attributes) parameters are estimated. However, there are situations in which it is useful to condition on density, that is, the density is fixed at a constant value (the observed value of the snowball sample being estimated) in the estimation procedure. In these cases, therefore, the Edge (θ_L) parameter is not estimated, and we refer to the estimate being done with "fixed density".

3. Snowball sampling and conditional estimation

Snowball sampling (Coleman, 1958; Goodman, 1961), as a technique to generate a sample of nodes in a network using the network structure itself, can be simply described as follows. First we take a random sample of N_0 nodes from the network, which we will refer to as the *seed* nodes, or *wave 0* of the snowball sample. Then wave 1 of the sample consists of all nodes that have a tie to a node in wave 0, but are not themselves in wave 0. In general, wave l of the snowball sample consists of all those nodes that have a tie to a node in wave $l-1$ of the sample (but are not themselves in waves $0, \ldots, l-1$). The totality of all nodes in the sample obtained (and the ties between them — that is, the subgraph induced by the nodes in the sample) is then known as an l-wave snowball sample.

This form of snowball sampling, as used in Pattison et al. (2013), differs from the frequently used form, described in Goodman (1961) for example, in which a fixed maximum number of ties for each node can be followed to find a node in the next wave. The origin of the latter form is that a sampled individual in wave l is asked to name up to m individuals, and those who are not already present in waves $0, \ldots, l$ then form wave $l+1$. In the form we are using, there is no such limit m, so *all* ties from a sampled node are followed. This form, in which there is no limit on the number of ties to follow, is equivalent to l steps of breadth-first search (BFS) in the network, and is therefore also known as *BFS sampling*, as described for example in Kurant et al. (2011). Work that uses this definition of snowball sampling includes, for example, Newman (2003) and Lee et al. (2006). The distinction between different usages of the term "snowball sampling" was discussed in a set of three papers (Goodman, 2011; Heckathorn, 2011; Handcock and Gile, 2011).

Our approach relies on forming snowball samples of a large network. Previous work to estimate network properties based on snowball sampled data include Thompson and Frank (2000); Handcock and Gile (2010); Pattison et al. (2013) (see Illenberger and Flötteröd (2012) for an overview). Handcock and Gile (2010) were the first to describe a method to estimate higher order ERGM parameters from a snowball sample. Their method requires that the size of the full network (the number of nodes, N) is known, and that estimation over the entire set of random tie variables X is feasible (so that N needs to be small for practical purposes). The recently developed conditional maximum likelihood estimation procedure (Pattison et al., 2013) also estimates ERGM parameters from snowball samples, but can be used

even when the size of the complete network is unknown. It makes less efficient use of the data than the method of Handcock and Gile (2010), so the latter method is to be preferred in cases where N is known and not too large. In this paper N is known, and we consider cases where N is possibly extremely large, and snowball samples include only a very small fraction of the total number of nodes.

The essence of the conditional estimation method of Pattison et al. (2013) is to do conditional estimation using the snowball sample, respecting the snowball sampling structure. A conditional probability model of the ties in all waves but the outermost, that is, waves $0, \ldots, l-1$, has the same parameters as the ERGM for the entire network. However it can be estimated conditionally by fixing three things: the ties connecting a node in the outermost wave l to a node in wave $l-1$, the ties between nodes in the outermost wave l, and for every wave in the snowball sample, the wave to which each node belongs. Given a social circuit or Markov dependence assumption, a tie between nodes in some wave i is conditionally independent of ties not between nodes in waves i, $i-1$ or $i+1$. Hence in the MCMC simulation used in the estimation procedure, ties in the outermost wave l are fixed, and ties in the inner waves can change but must respect the snowball sampling wave structure. This means that, in the simulation procedure, a tie cannot be deleted if it is the last remaining tie to a node from an earlier wave, and a tie cannot be added if it would "skip over" a wave. More precisely, in the simulation, a tie that connects a node in wave $i-1$ to a node in wave i cannot be removed if it is the only tie between the node in wave i and *any* node in wave $i-1$, and a tie that would connect a node in wave i to a node in wave $i+\delta$ or $i-\delta$ cannot be added unless $\delta \in \{0, 1\}$.

4. Meta-analysis

As foreshadowed in Pattison et al. (2013), conditional estimates from multiple random snowball samples of a single network can be carried out in parallel. Our method does so, and combines the estimates in order to obtain both a pooled estimate and a confidence interval for each parameter of an ERGM for the entire network. To combine the parameter estimates for the snowball samples into an estimate for the parameters of the entire network we use two different schemes, and compare their performance. First, as a robust point estimator for the entire network that relies on few assumptions

we take the median of the snowball sample point estimates, as is done in Pattison et al. (2013).

Second, estimates can be combined using the weighted least squares (WLS) estimator (Snijders and Baerveldt, 2003) as in Lubbers and Snijders (2007) and Xu et al. (2013):

$$\hat{\mu}_\theta^{\text{WLS}} = \frac{\sum_j \left(\hat{\theta}_j/(\hat{\sigma}_\theta^2 + s_j^2)\right)}{\sum_j \left(1/(\hat{\sigma}_\theta^2 + s_j^2)\right)} \tag{5}$$

where

- $j \in 1, \ldots, N_s$ are the N_s snowball samples,
- $\hat{\theta}_j$ is the estimate for sample j,
- $\hat{\sigma}_\theta^2$ is the estimated between-sample variance,
- s_j is the estimated standard error for sample j.

In Snijders and Baerveldt (2003) the standard error of the WLS estimator is calculated as

$$\text{se}(\hat{\mu}_\theta^{\text{WLS}}) = \frac{1}{\sqrt{\sum_j 1/(\hat{\sigma}_\theta^2 + s_j^2)}} \tag{6}$$

This expression is derived under the assumption that θ_j and s_j^2 are independent across samples, something that may be a reasonable approximation in many circumstances (there is also a test of independence in the package `siena08` (Ripley et al., 2014) for meta-analysis of network models). Since this assumption is not necessarily true in all cases, for robustness we use instead the non-parametric bootstrap adjusted percentile (BCa) method (Efron, 1987; Davison and Hinkley, 1997) to estimate the confidence interval. The BCa method adjusts for bias and skewness in the bootstrap distribution, and we use the BCa bootstrap error estimation to estimate the standard error for the median point estimator. The bootstrap replicates are constructed by taking random resamples of size N_s with replacement from the N_s estimates (one for each snowball sample).

An assumption of the WLS estimator is that $\{\theta_j,\, j = 1, \ldots, N_s\}$ are independent across snowballs with expected value θ and variance σ_θ^2. Independence may seem implausible given that estimates are all based on the same

network. We may however proceed such that our collection of parameter estimates are conditionally independent by relying on the notion of 'separation' developed in Pattison et al. (2013). We refer to two subsets as *well separated regions* if they are conditionally independent, conditional on a separating set. Under social circuit dependence, and using at least two waves, two snowball samples will form well separated regions if the only nodes in common between the two samples (if any) are in the outermost wave of both samples. Thus if all snowball samples are pair-wise well separated by conditioning regions, then the estimates will be (conditionally) independent.

In particular, if the snowball samples are saturated (and connected), Snijders (2010) showed that a more efficient estimator, assuming a *component independent* ERGM (weaker than social circuit dependence assumption), may be obtained by conditioning only on components being connected. This may be interpreted as if there is no "action at a distance" — ties between nodes in one component of the network cannot be affected by changes in ties in another component. Then estimators based on different components yield independent estimates of the same ERGM (with the proviso that the estimators are still conditional).

Strictly enforcing that snowball samples be well separated regions would thus allow us to obtain conditionally independent estimates. But we are not so strict in the following examples for two main reasons. Firstly, for the purposes of comparing the sampling distributions of the pooled estimates, it is desirable to keep the number of estimates fixed. This may be achieved by fixing the number of seed sets and waves. Requiring snowball samples to be strictly well separated restricts the number of snowball samples that are possible. Secondly, finding those snowball samples is computationally demanding. For the snowball sampling used here, seed sets are formed as random samples from the set of graph nodes, with the condition that the seed sets are disjoint. Dependence between snowball samples can arise from overlap of subsequent zones. In practice, for large graphs we expect that the effect of overlap is small even if we do not strictly apply the complete separation criterion, as long as the seed sets are chosen randomly and the snowball samples are small compared to the entire graph. Pathological counter-examples are clearly possible, such as a network with a strong hub where many snowball samples overlap via the hub. It is recommended, therefore, to measure the degree of overlap between the inner waves of snowball samples to ensure that the well separated region assumption is reasonable for a given set of snowball samples.

5. Implementation

The conditional estimation procedure is implemented in the C programming language as a modified version of the PNet program (Wang et al., 2009), which uses the Robbins-Monro algorithm as described in Snijders (2002). Both an MPI version for parallel estimation on a cluster system, and an OpenMP version for parallel estimation on a single multicore machine are implemented. The cluster system we used is an SGI Altix XE Cluster with 1088 Intel Nehalem cores (8 per node), 2.66 GHz, running CentOS 5 and OpenMPI.

In all of the results that follow, unless otherwise noted, we run the MPI version of the program with 20 tasks; one task for each of the snowball samples. Hence the ERGM parameters for all 20 snowball samples from a network are estimated in parallel. Only estimates that converge are included in the pooled estimates. Hence N_s, as used in the previous section and in the tables in the next section, is the number of *converged* estimates, which may be less than 20. The convergence criterion, now standard in the ERGM literature (Lusher et al., 2013), is that the magnitude of the convergence statistic (also known as the t-ratio) for each network statistic corresponding to a model parameter is less than 0.1. The t-ratio is determined using 1000 networks simulated with the parameter estimates, by calculating the difference between the observed and mean (over the 1000 networks) value of the network statistic divided by the standard deviation (over the 1000 networks) of the network statistic.

Scripts for sampling in large networks, visualization, and bootstrap error estimation are written in R (R Core Team, 2013) using the `igraph` package (Csárdi and Nepusz, 2006). Bootstrap confidence intervals are estimated with 20000 replicates using the R `boot` package (Davison and Hinkley, 1997). Boxplots are generated with the R `ggplot2` package (Wickham, 2009). Confidence intervals, shown in graphs and used in inference, are computed as 2 or 3 standard errors as indicated.

6. Simulation studies

In order to evaluate the performance of our method we require networks with known parameters. For each of our sets of parameters shown in Table 1, we generate 100 samples from a network distribution with those parameters using PNet (Wang et al., 2009). Each of these networks is simulated with

a fixed number of nodes (N), edge parameter (θ_L), and parameters for alternating k-star (AS), alternating k-triangle (AT) and alternating two-path (A2P) statistics. These structural parameters are the same as those used in Pattison et al. (2013).

In addition, some of the networks have a binary attribute on each node. For this paper we have done the most extensive exploration on binary attributes. Our preliminary results for categorical and continuous attributes are in Appendix C. For those attributes shown as "50/50" in Table 1, 50% of the nodes have the True value for the attribute, and the other 50% False. Similarly, for the networks labeled "70/30", 70% of the nodes have the True value for the attribute. For those networks with an attribute, there are an additional two parameters, activity (ρ), and interaction (ρ_B). The networks are sampled from a MCMC simulation, with sufficient burn-in (of the order of 10^7 iterations for 5000 node networks and 10^8 iterations for 10000 node networks) to ensure initialization effects are minimized, and samples are taken sufficiently far apart (here separation of the order of 10^6 iterations) to ensure that they are essentially independent. Table 2 shows summary statistics of the simulated networks, Figure 3 shows a visualization of a single snowball sample from one of the 5000 node simulated graphs, and Figure 4 shows the distribution of snowball sample sizes across 2000 samples from the simulated 5000 node networks, while Figure 5 shows the distribution of the number of nodes in the inner waves (that is, all nodes except those in the outermost wave of the snowball sample) of the same samples. This latter number is important as the ties in the outermost wave are fixed in the simulations used during the estimation procedure, so it is the size of the inner waves of the sample that is most relevant for this process.

We refer to a network with a binary attribute as "balanced" when $\rho_B = -2\rho$. That is, there is no "differential homophily". Table 3 shows the conditional log-odds for tie formation between two nodes with a binary attribute, and illustrates how "differential homophily" arises. If $\rho_B = -2\rho$ then there is no "differential homophily", since then the parameter for homophily between two nodes without the binary attribute (top left quadrant) is equal to the parameter for homophily between two nodes with the binary attribute (bottom right quadrant).

Deciding on the snowball sampling parameters to use involves a tradeoff between the size of the samples and the number of samples to take. Is it better to have a large number of small samples, or a smaller number of larger samples? One parameter, the number of samples, is chosen directly, but the

N	Attributes	Edge (θ_L)	AS	AT	A2P	ρ	ρ_B
5000	None	-4.0	0.2	1.0	-0.2		
5000	50/50	-4.0	0.2	1.0	-0.2	0.2	0.5
5000	70/30	-4.0	0.2	1.0	-0.2	0.2	0.5
5000	50/50 balanced	-4.0	0.2	1.0	-0.2	-0.25	0.5
10000	None	-4.0	0.2	1.0	-0.2		

Table 1: Parameters of the simulated networks. The value in the Attributes column shows the percentage of nodes which have, respectively, the values of True and False for their binary attribute, and whether the attribute parameters are "balanced". For networks with attributes, the ρ and ρ_B columns then show, respectively, the activity and interaction parameter values.

N	Attributes	Mean components	Mean degree	Mean density	Mean global clustering coefficient
5000	None	1.00	8.76	0.00175	0.02451
5000	50/50	1.00	9.54	0.00191	0.02661
5000	70/30	1.00	9.99	0.00200	0.02762
5000	50/50 balanced	1.01	8.51	0.00170	0.02428
10000	None	1.00	10.04	0.00100	0.01553

Table 2: Statistics of the simulated networks.

	0	1
0	θ_L	$\theta_L + \rho$
1	$\theta_L + \rho$	$\theta_L + 2\rho + \rho_B$

Table 3: Contingency table showing the conditional log-odds for tie formation on a node depending on the False (0) or True (1) value of its binary attribute. θ_L is the edge (density) parameter, ρ is the activity parameter, and ρ_B is the interaction parameter. If $\rho_\beta = -2\rho$ there is no "differential homophily".

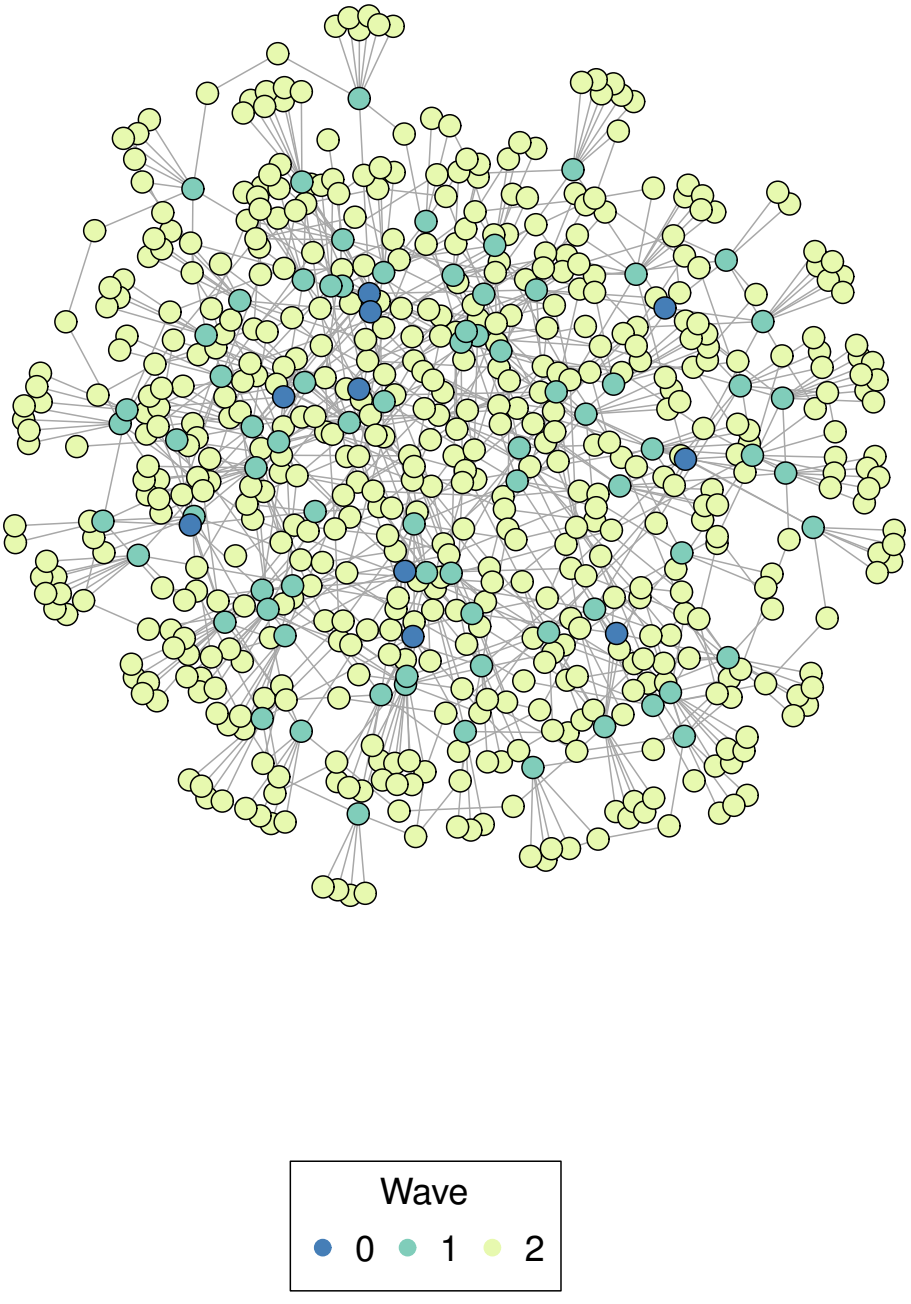

Figure 3: A snowball sample with 10 seeds and 2 waves from a graph drawn from the 5000 node network simulations. This snowball sample has 629 nodes, of which 10 are seed nodes (wave 0), 75 are in wave 1, and 544 are in wave 2.

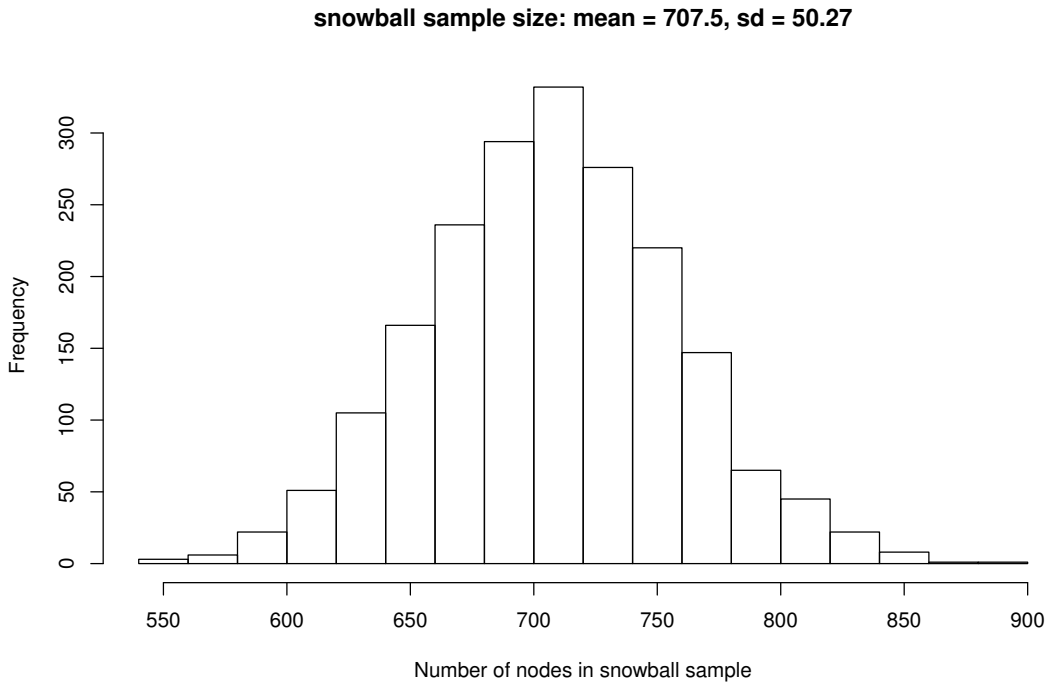

Figure 4: Histogram of 10 seed, 2 wave snowball sample sizes over all 2000 snowball samples (20 samples from each of the 100 graphs) from the 5000 node network simulations.

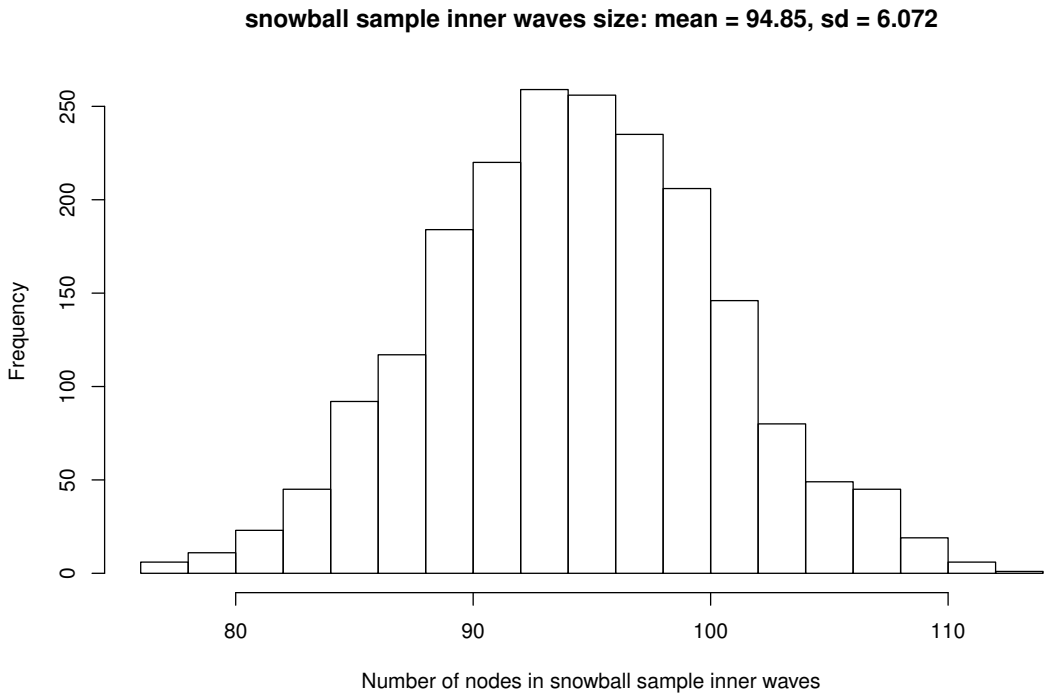

Figure 5: Histogram of the number of nodes in the inner waves for the 10 seed, 2 wave snowball over all 2000 snowball samples (20 samples from each of the 100 graphs) from the 5000 node network simulations.

sample size distribution, for a particular network, is controlled indirectly by the number of seed nodes and the number of waves. In our simulation experiments we fix the number of samples at 20 (each fit by its own parallel task), but we still need to choose the number of seeds and the number of waves. In a situation such as this, where we know the entire network and are performing the snowball sampling computationally, we can explore the consequences of these parameters by varying their values, and observing the results, as shown in Figure 6. This figure shows two outcomes for a given value of the number of waves and number of seeds: in the top row, the number of samples in which a given node appears, and in the bottom row, the size of a sample. Figure 7 shows the same results, but counting only nodes in the inner waves. We want a small number of samples in which a node from the inner waves appears; the more samples in which the same node appears, the more our assumption of well-separated regions is violated (overlap in itself does however not imply that regions are not separated according to the definition of Pattison et al. (2013) — a node may belong to many separating sets). We also want the size of the samples to be small enough to be estimated in reasonable time, but not so small that parameters cannot be estimated.

From Figure 6, we can see that using 3 waves results in samples that are very large, with more than half the network being covered when 5 or more seeds are used. Additionally, Figure 7 shows that the number of samples in which a given inner wave node appears grows quite quickly with the number of seeds, leading to the possibility of an inner wave node appearing in more than half of the samples when 10 seeds are used. When only 1 wave is used, the resulting networks are very small, and do not contain enough structure for useful estimation. Hence it appears most reasonable to use 2 waves.

In the following results, based on these considerations, we take 20 snowball samples, with 10 seed nodes and 2 waves, from each network. The conditional estimation for each of the 20 snowball samples is performed in parallel. Note that when estimations are performed with fixed density (the density of the simulated networks in the MCMC procedure is fixed at the observed value in each snowball sample), no edge (θ_L) parameter is estimated.

Figure 8 shows the elapsed time for estimating the parameters of the 5000 node network (with no attributes) using snowball sampling and conditional estimation. Because each of the 20 snowball samples is estimated in parallel, this elapsed time is the maximum estimation time over the 20 snowball samples. Figure 9 shows the total CPU time taken for the estimation, that is, the sum of the 20 snowball sample estimation times. By way of comparison,

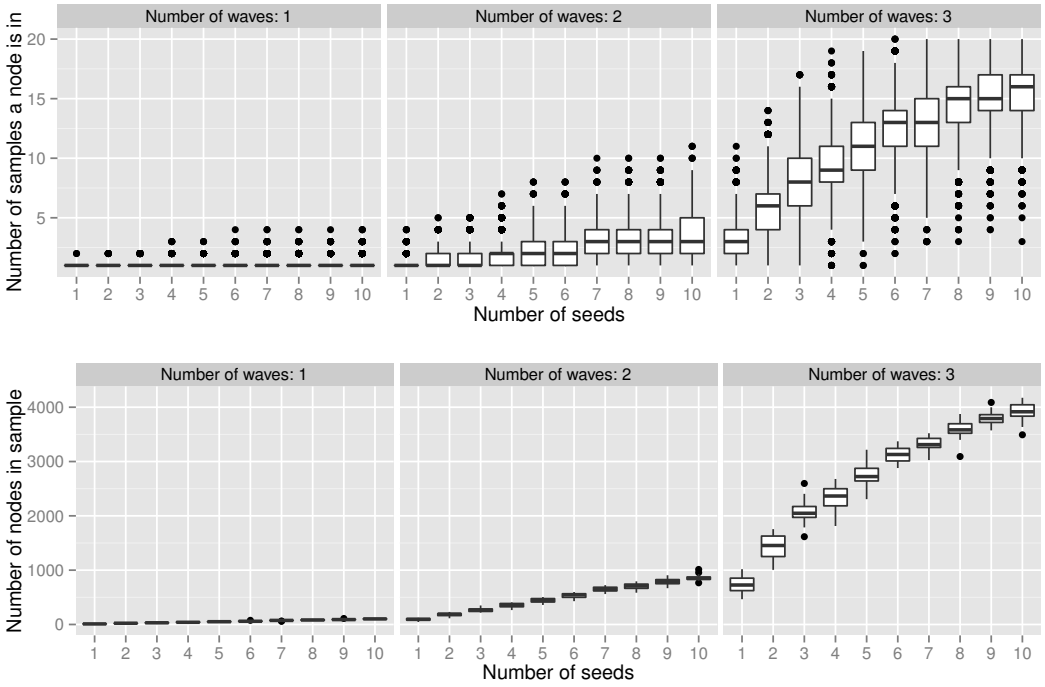

Figure 6: Number of samples in which a node appears (top panel), and the number of nodes in a sample (bottom panel), for a single instance from the simulated 5000 node networks with 50/50 binary attributes, for 20 snowball samples.

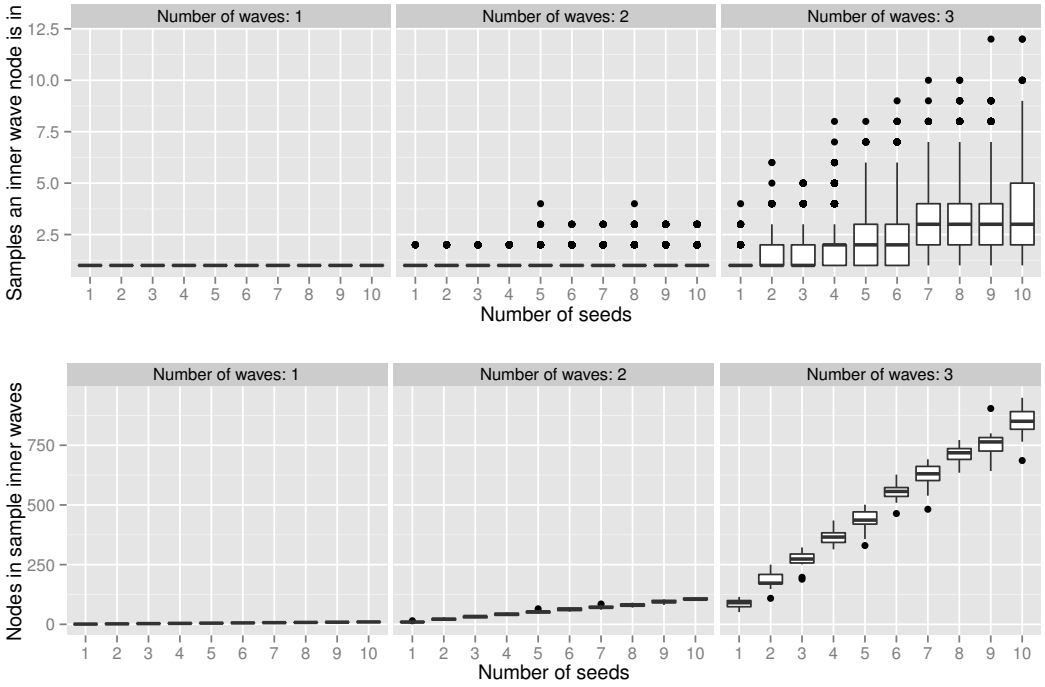

Figure 7: Number of samples in which a node from the inner waves appears (top panel), and the number of inner wave nodes in a sample (bottom panel), for a single instance from the simulated 5000 node networks with 50/50 binary attributes, for 20 snowball samples.

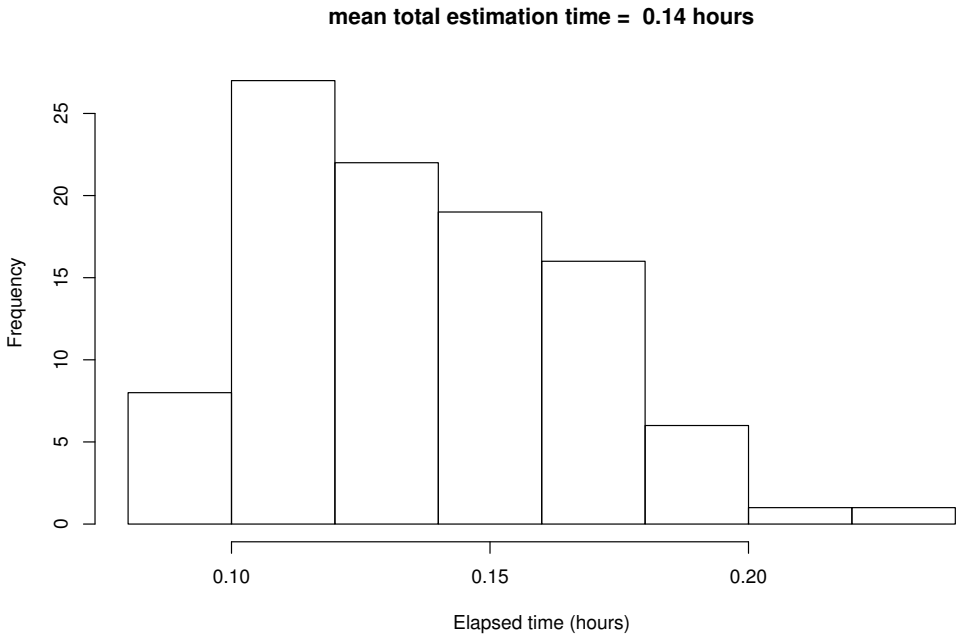

Figure 8: Histogram of elapsed time over the 100 samples from the simulated 5000 node networks (no attributes, fixed density). Each of the 20 snowball samples is estimated in parallel, so the elapsed time here is the longest time that it takes to estimate any of the 20 snowball samples.

using PNet (on the same hardware) to estimate a randomly chosen one of these sampled networks takes on average 83.5 hours (averaged over 8 PNet estimations of the same network). So the total CPU time taken for the estimate using snowball sampling and conditional estimation is on the order of one hour, compared to approximately three days for estimating the entire network with PNet. Further, the elapsed time can be reduced even more by estimating the snowball samples in parallel, as shown in Figure 8, where by using 20 parallel tasks the elapsed time for the estimation is on average less than 10 minutes.

Figure 10 provides a comparison of the sampling distributions of the pooled estimator $\hat{\mu}_\theta^{\mathrm{WLS}}$ and the complete data MLE based on 100 graphs of size $N = 5000$ from the model in Table 1. The complete data MLEs are calculated using the Geyer-Thompson method of Hunter and Handcock (2006) based on one importance sample of 4100 graphs from the true model (not

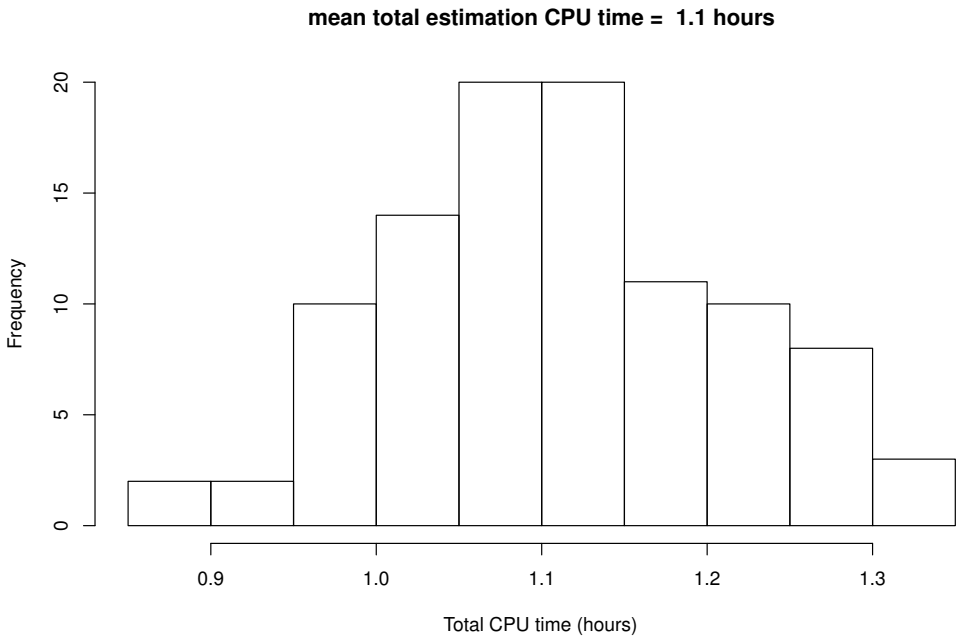

Figure 9: Histogram of total CPU time over the 100 samples from the simulated 5000 node networks (no attributes, fixed density). This is the sum of the CPU time taken for each of the 20 snowball samples, which would be the elapsed time taken if estimated using a single processor, no parallelism and naive initializations.

re-generating graphs saves us considerable computational time but makes estimates more approximate; some of the 100 graphs were on the relative boundary of the convex hull induced by the importance sample and had to be excluded from analysis).

There is generally a good correspondence between the sampling distributions in terms of location and the greater variance of $\hat{\mu}_\theta^{\mathrm{WLS}}$ is what we would expect given that this estimator is less efficient. As noted in Pattison et al. (2013), the bias for the conditional snowball estimator of the density (and alternating star parameter) is somewhat more marked than for other parameters, suggesting that conditioning on density would be a sensible way of removing what is essentially a nuisance parameter. (For a treatment of conditional ERGMs see Snijders and van Duijn (2002)). The sampling fraction for these samples are roughly 20% of nodes (compare Figure 6) or 2% if only inner nodes are considered (compare Figure 7) and the overlap of inner nodes is low (according to top panel of Figure 7, nodes mostly appear in at most one inner node-set). This means that $\hat{\mu}_\theta^{\mathrm{WLS}}$ is pooled across approximately independent samples that together cover a small fraction of the network. This in turn implies that $\hat{\mu}_\theta^{\mathrm{WLS}}$ is very weakly associated with the MCMCMLE that is based on the entire network (see Appendix A for a more detailed illustration of this principle).

In terms of power (one minus the type II error probability) the estimators appear to be comparable for this set of parameters. Both estimators have very low power for the density and alternating k-star parameters.[1] In the sequel we will therefore not attach much importance to the type II errors of these two parameters. For this particular set of estimates (Figure 10) it is not meaningful to compare the bias and root mean square error (RMSE) of the estimators as the range of estimates for the complete data MLE is large and there are a lot of extreme values, something which is presumably largely to do with the approximation (mentioned above).

Table 4 shows the bias ("Bias"), root mean square error ("RMSE") and standard deviation of the estimated value over the 100 simulated networks, for each of the ERGM parameters ("Effect") in the simulated networks. The bias is the mean of the difference between the estimate and the true value,

[1]Further investigation shows that the power also is low for these parameters when using full network MLEs such as those implemented in statnet and PNet. (Data not shown; available on request.)

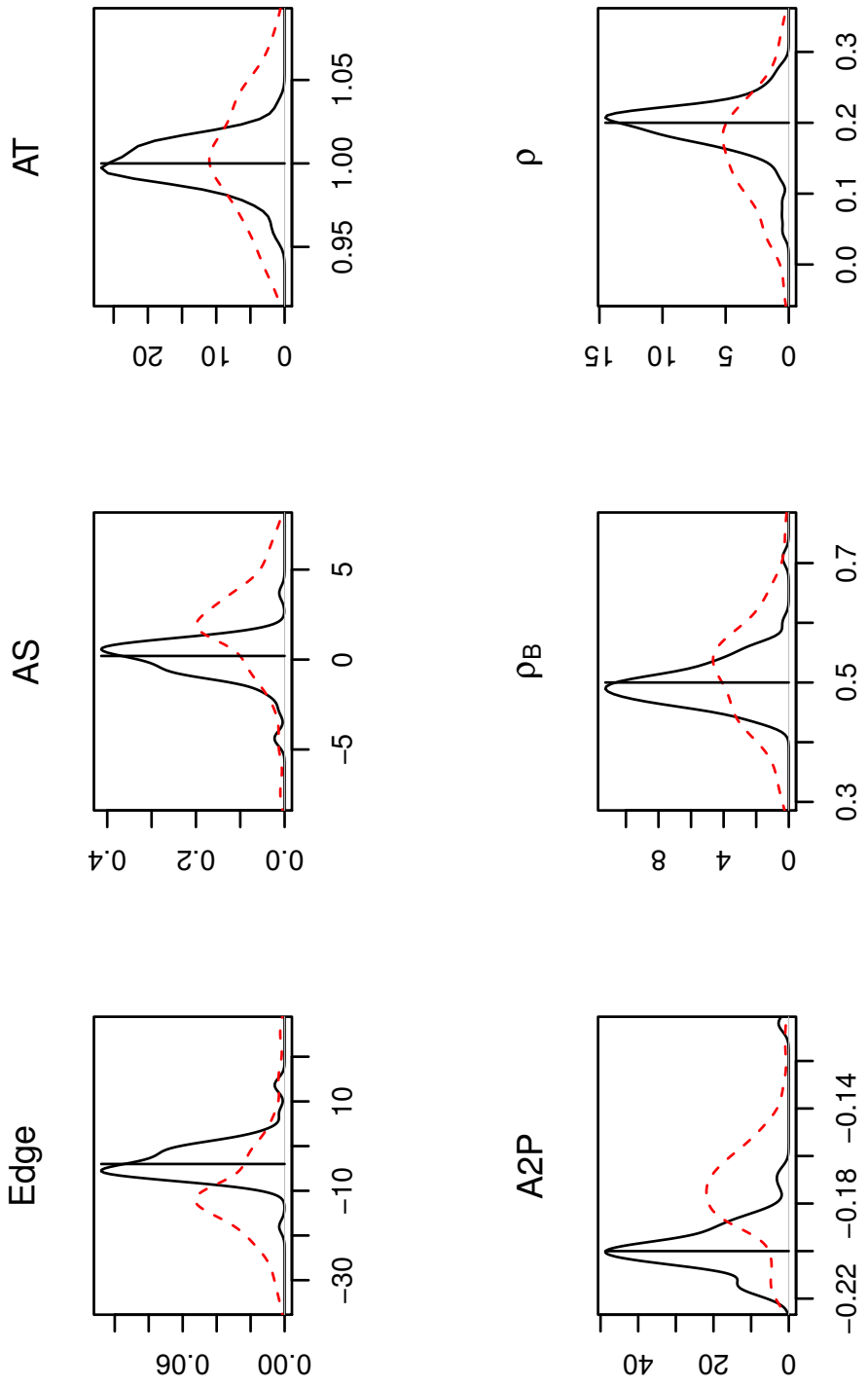

Figure 10: Distribution of the pooled WLS estimates (red, dashed) and MCMC MLE (black, solid) with true values indicated by vertical lines. Population graph $N = 5000$ and 91 estimates (a number of Newton-Raphson estimation runs did not converge).

N	Attributes	Effect	Bias	RMSE	Type II error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	None	A2P	0.0054	0.0177	0	0	4	0.0169	12.74
5000	None	AT	-0.0223	0.0292	0	0	4	0.0190	12.74
5000	None	AS	0.1357	0.2231	78	69	85	0.1781	12.74
5000	70/30	A2P	0.0143	0.0177	0	0	4	0.0106	19.02
5000	70/30	AT	-0.0214	0.0249	0	0	4	0.0128	19.02
5000	70/30	AS	0.1676	0.2102	54	44	63	0.1276	19.02
5000	70/30	ρ	-0.0720	0.0880	66	56	75	0.0508	19.02
5000	70/30	ρ_B	-0.0145	0.0538	0	0	4	0.0520	19.02
5000	50/50 balanced	A2P	0.0002	0.0150	0	0	4	0.0151	15.34
5000	50/50 balanced	AT	-0.0198	0.0267	0	0	4	0.0180	15.34
5000	50/50 balanced	AS	0.1370	0.2324	75	66	82	0.1887	15.34
5000	50/50 balanced	ρ	-0.0138	0.0486	4	2	10	0.0468	15.34
5000	50/50 balanced	ρ_B	0.0068	0.0507	0	0	4	0.0505	15.34
5000	50/50	A2P	0.0130	0.0176	0	0	4	0.0119	17.96
5000	50/50	AT	-0.0237	0.0280	0	0	4	0.0150	17.96
5000	50/50	AS	0.1300	0.1949	58	48	67	0.1460	17.96
5000	50/50	ρ	-0.0704	0.0820	40	31	50	0.0423	17.96
5000	50/50	ρ_B	0.0028	0.0436	0	0	4	0.0437	17.96
10000	None	A2P	-0.0033	0.0179	0	0	4	0.0177	8.56
10000	None	AT	-0.0088	0.0235	0	0	4	0.0219	8.56
10000	None	AS	0.1504	0.2583	82	73	88	0.2111	8.56

Table 4: Error statistics and number of converged snowball samples (out of 20) for the simulated networks, including false negatives (type II error rate) over the 100 simulated networks, using the median point estimator with fixed density. True parameter values for each model are shown in Table 1.

and the RMSE is the square root of the mean squared difference between the estimate and the true value. Table 4 also shows the type II error rate in inference, that is, the false negative rate. This is the percentage of experiments (over the 100 simulated networks) in which the estimate has the wrong sign or the confidence interval covers zero (so we cannot reject the null hypothesis that the parameter for the effect is zero). The 95% confidence interval shown for these estimated error rates is the confidence interval for the binomial proportion computed using the Wilson score interval (Wilson, 1927). We include this information as often when analyzing social networks, we are interested not specifically in the magnitude of an effect, but rather in whether or not it is significant (either positive or negative); the intervals used for Tables 4 through 7 and Appendix B are 3 standard errors. In addition, the table shows the mean number of snowball samples (out of 20) for which the parameter estimates converge. Table B.13 shows the same data, but using the WLS point estimator rather than the median point estimator.

We can see that, with two exceptions, the bias and RMSE are small (relative to the effect size), and correspondingly the type II error (false negative rate) is small or, usually, zero. The exceptions are the alternating k-star parameter (AS) for all networks, and the activity (ρ) parameter for the networks with binary attributes that are not "balanced". For the AS parameter, this is what we would expect based on the evidence in Figure 10. Furthermore, the high RMSE and low power on the Edge and alternating k-star parameters are not an issue specific to snowball sampling or conditional estimation, but also occurs in the full network MLE methods implemented in PNet and statnet.

The results just discussed are for estimations in which the density is fixed. Table B.14 and Table 5 show the corresponding results when density is not fixed in estimation. Now, in addition to AS, the Edge parameter (not estimated when density is fixed) also has large error (bias, RMSE, false negative rate) values. As Edge is essentially a nuisance parameter this is not a cause for concern. Note that the AS (as a parameter which helps control the degree distribution) and Edge parameters are interdependent, and so bias in the Edge parameter causes the AS parameter to be biased also. Hence when conditioning on density, the bias and RMSE for AS is considerably reduced (Table B.13 and Table 4 show much lower values for bias and RMSE on the AS parameter than Table B.14 and Table 5).

However, one more parameter has a very large false negative rate, namely activity (ρ), particularly when the binary attribute is not balanced. Hence

in estimating real networks in which a binary attribute is not balanced, it seems likely that an activity effect on this attribute may be missed when using this technique, unless the effect size is strong.

Another difference between the results for estimation with fixed density (Table B.13 and Table 4) and when density is not fixed (Table B.14 and Table 5), is that fixing the density results in a lower fraction of converged estimates (78% versus 97% when density is not fixed). Fixing the density results in reduced bias and RMSE, and consequently better power (and, as we shall see, a reduced type I error rate on certain parameters) in inference. Estimating with fixed density results in fewer converged estimates, a fact that itself might lead to bias due to only a small number of converged estimates being used in the meta-analysis. Although this does not appear to be too problematic for our simulated networks, for which by definition we know the most appropriate model, this could be an important issue in estimating parameters of a model for empirical networks. As we shall see in Section 7, in such cases it may well be necessary to not fix the density in order to obtain a reasonable number of converged estimates.

The previous simulations allow us to measure errors in the estimation, and the rate of type II errors in inference, that is, the false negative rate. In order to measure type I errors in inference, that is, the false positive rate, for an effect, we need simulated network data that does not have that effect present (its corresponding parameter is zero). Hence for each of the simulated networks, we simulate, for each of the effects, another network distribution in which the effect's parameter is set to zero. This then gives us another set of 100 networks from a distribution, but with a null effect, so in estimating the parameters for these networks we can test for false positives with respect to that effect. The false positive rate is then the percentage of experiments in which, for an estimate of a null effect, the confidence interval did not include zero.

Table B.15 and Table 6 show the results of these experiments, when density is fixed in estimation. Note that the alternating two-path effect (A2P) is not present. This is because when the A2P parameter is set to zero, the simulated graphs have very high density (above 0.2, the nodes having a mean degree greater than 1000), and so are both unrealistic as social networks, and also impractical to estimate in reasonable time since the high density results in the snowball sample sizes approaching the size of the full network. Therefore they have been excluded. Using the median point estimator (Table 6), in all cases the type I error rate is less than 10%,

N	Attributes	Effect	Bias	RMSE	Type II error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	None	A2P	0.0345	0.0469	6	3	12	0.0320	20.00
5000	None	AT	-0.0018	0.0440	0	0	4	0.0442	20.00
5000	None	Edge	6.1230	11.4700	99	95	100	9.7510	20.00
5000	None	AS	-1.7240	3.0530	100	96	100	2.5330	20.00
5000	70/30	A2P	0.0214	0.0343	0	0	4	0.0270	19.88
5000	70/30	AT	-0.0039	0.0428	0	0	4	0.0428	19.88
5000	70/30	Edge	13.3700	20.3400	100	96	100	15.4100	19.88
5000	70/30	AS	-3.5420	5.2590	100	96	100	3.9080	19.88
5000	70/30	ρ	0.0495	0.1419	81	72	87	0.1337	19.88
5000	70/30	ρ_B	-0.0339	0.1481	41	32	51	0.1449	19.88
5000	50/50 balanced	A2P	0.0254	0.0424	7	3	14	0.0341	17.68
5000	50/50 balanced	AT	-0.0063	0.0518	0	0	4	0.0517	17.68
5000	50/50 balanced	Edge	8.5920	14.4100	95	89	98	11.6300	17.68
5000	50/50 balanced	AS	-2.3120	3.7550	100	96	100	2.9740	17.68
5000	50/50 balanced	ρ	-0.0100	0.1052	58	48	67	0.1053	17.68
5000	50/50 balanced	ρ_B	0.0020	0.1143	15	9	23	0.1149	17.68
5000	50/50	A2P	0.0199	0.0311	1	0	5	0.0240	20.00
5000	50/50	AT	0.0061	0.0423	0	0	4	0.0420	20.00
5000	50/50	Edge	8.4380	14.2800	100	96	100	11.5800	20.00
5000	50/50	AS	-2.2470	3.6810	100	96	100	2.9300	20.00
5000	50/50	ρ	-0.0121	0.0979	80	71	87	0.0976	20.00
5000	50/50	ρ_B	0.0132	0.1042	8	4	15	0.1039	20.00
10000	None	A2P	0.0101	0.0347	0	0	4	0.0333	19.65
10000	None	AT	-0.0060	0.0510	0	0	4	0.0509	19.65
10000	None	Edge	-0.3986	16.1600	95	89	98	16.2400	19.65
10000	None	AS	-0.0418	4.1230	97	92	99	4.1440	19.65

Table 5: Error statistics and number of converged snowball samples (out of 20) for the simulated networks, including false negatives (type II error rate) over the 100 simulated networks, using the median point estimator, when density is not fixed. True parameter values for each model are shown in Table 1.

N	Attributes	Effect	Bias	RMSE	Type I error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	None	AT	-0.0087	0.0683	3	1	8	0.0681	18.94
5000	None	AS	0.2483	0.4185	6	3	12	0.3386	10.16
5000	70/30	AT	-0.0027	0.0477	3	1	8	0.0479	19.96
5000	70/30	AS	0.2394	0.2925	5	2	11	0.1689	16.08
5000	70/30	ρ	-0.0419	0.0753	6	3	12	0.0629	18.17
5000	70/30	ρ_B	-0.0099	0.0646	4	2	10	0.0641	18.17
5000	50/50 balanced	AT	-0.0091	0.0706	3	1	8	0.0703	19.16
5000	50/50 balanced	AS	0.1648	0.4174	7	3	14	0.3854	10.48
5000	50/50 balanced	ρ	-0.0245	0.0500	7	3	14	0.0438	16.96
5000	50/50 balanced	ρ_B	-0.0022	0.0639	2	1	7	0.0642	15.17
5000	50/50	AT	-0.0128	0.0480	0	0	4	0.0465	19.87
5000	50/50	AS	0.2369	0.2949	5	2	11	0.1764	14.00
5000	50/50	ρ	-0.0245	0.0500	6	3	12	0.0438	16.96
5000	50/50	ρ_B	0.0103	0.0526	4	2	10	0.0519	17.25

Table 6: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with a zero effect to find false positives (type I error rate) over the 100 simulated networks, using the median point estimator with fixed density.

however the results for the WLS point estimator (Table B.15) are not as good, with four cases in which the type I error rate is 10% or greater.

Table B.16 and Table 7 show the corresponding results when density is not fixed in estimation. It is clear that the median point estimator (Table 7) is a better choice in this case, although there are still four cases in which the type I error rate is 10% or greater.

We have found that the median estimator generally performs better than the WLS estimator currently used in the network literature. In terms of type I errors, results appear to be excellent for a median estimator with fixed density models. Even when density is not fixed, the type I error rates are less than 10%, except for network closure effects (AT) which may extend out past 15%. We need to do more work to determine whether larger seed set size or more samples will enable us to lower the AT type I error rates.

In terms of power (i.e. one minus the type II error probability), median estimators again perform better, especially in fixed density models. Nevertheless, power for the star effect was very poor. Power was also poor for attribute activity, perhaps due to small effect size. However, power for the interaction parameter is generally good, except when density is not fixed and

N	Attributes	Effect	Bias	RMSE	Type I error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	None	AT	-0.8790	1.0310	39	30	49	0.5419	18.16
5000	None	AS	-0.8238	1.7170	1	0	5	1.5140	19.78
5000	70/30	AT	-0.1851	0.5475	10	6	17	0.5179	16.71
5000	70/30	AS	-1.9680	2.7820	1	0	5	1.9760	18.70
5000	70/30	ρ	0.0307	0.1503	5	2	11	0.1478	19.13
5000	70/30	ρ_B	-0.0328	0.1395	0	0	4	0.1362	19.25
5000	50/50 balanced	AT	-0.4453	0.8633	16	10	24	0.7433	13.28
5000	50/50 balanced	AS	-0.9863	1.7070	3	1	9	1.4000	14.13
5000	50/50 balanced	ρ	-0.0101	0.0983	1	0	5	0.0983	18.71
5000	50/50 balanced	ρ_B	-0.0086	0.1293	2	1	7	0.1297	17.22
5000	50/50	AT	-0.2859	0.6725	12	7	20	0.6118	15.94
5000	50/50	AS	-1.3370	2.3210	1	0	5	1.9070	17.03
5000	50/50	ρ	-0.0101	0.0983	1	0	5	0.0983	18.71
5000	50/50	ρ_B	-0.0089	0.1388	3	1	8	0.1392	18.07

Table 7: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with a zero effect to find false positives (type I error rate) over the 100 simulated networks, using the median point estimator, when density is not fixed.

the attribute value is not evenly distributed across nodes (70/30 in our case).

In these experiments, we have used 20 snowball samples, each with 2 waves from 10 seed nodes, giving a total of 200 seed nodes. We might then ask how these results compare with using the conditional estimation technique of Pattison et al. (2013) directly, with the same number of seeds: that is, taking one large snowball sample using the same number of waves and same total number of seeds, with no pooling and hence no meta-analysis. We attempted to perform such a comparison using the 5000 node simulated networks with 50/50 binary attributes (not balanced). Taking a single snowball sample with 2 waves and 200 seeds results in an average snowball sample size of 4845 nodes (that is, almost the entire network) and an average of 1736 nodes in the inner waves. These are very large networks to estimate; no such sample could be estimated within a seven day time limit. This is one aspect of the difference between the pooled estimation method described here and using the Pattison et al. (2013) conditional estimation method directly: the former can be used to obtain estimates, using the same total number of seed nodes, for networks far larger than the latter.

Since 5000 node networks are too large to use conditional estimation

directly with this many seeds, we performed the experiments on 1000 node networks, simulated with the same parameters. Again using 20 samples with 10 seeds each and 2 waves (so 200 seeds for the single snowball sample using conditional estimation directly with no pooling), the single snowball sample has average size 998 nodes (again nearly the whole network), and the average number of nodes in the inner waves is 809, so conditional estimation is possible. By comparison, using 20 samples with 10 seeds each for pooling, the average snowball sample size is 386 nodes with only 76 nodes in the inner waves. Using the median point estimator and BCa error estimator, we found, comparing the pooled estimator to using conditional estimation directly, that the results for AT and AS are similar, however using conditional estimation directly results in reduced bias and RMSE for A2P, Edge, ρ, and ρ_B, giving significantly better power for the latter three effects (Edge, Activity and Interaction). Regarding the type I error rate, using conditional estimation directly gives significantly better results on the alternating k-triangle parameter, however there is no significant difference for the other parameters.

From this example it seems that using the same number of seeds with conditional estimation directly (no pooling and hence no meta-analysis) can reduce bias and increase power and reduce the type I error rate on some parameters. However for larger networks the pooling technique allows a large number of seeds to be used, taking advantage of parallel computing. Using the same number of seeds with conditional estimation directly becomes computationally impractical, allowing the use of only a single thread of computation and requiring an impractically long time to converge due to the larger size of the single snowball sample.

In summary, then, without strong effect sizes, star effects and actor activity effects may be difficult to detect. Homophily effects may be difficult to detect when density is not fixed and attributes are not distributed equally across nodes. Of course, if an effect is significant, the issue of power becomes irrelevant. In that case, based on these results, we can regard our inference of an effect as reliable when we use median estimators and fixed density, with the caveat that this might be subject to a bias due to too small a fraction of samples reaching a converged estimate when density is fixed. When density is not fixed, the problem of samples not reaching a converged estimate is reduced, however the type I error on the AT parameter is increased. A recommended procedure therefore, would be to attempt estimation with fixed density, and if too many samples fail to reach a converged estimate, repeat

with the density not fixed.

7. Example application to empirical networks

7.1. Parameter Estimation

In the previous section we applied our method to simulated networks. Since those networks were generated from an ERGM, we would expect that an ERGM can be fitted, and the assumption that configurations occur homogeneously across the network holds, so this is the best case for any ERGM estimation method. With empirical networks, however, it is not guaranteed to be so easy to fit models. For a method to be useful it is important that we demonstrate that it can also be applied to such networks. As an illustrative example, we apply our method to some collaboration networks from the igraph Nexus network data set repository (`http://nexus.igraph.org`). We do not fix the density in estimation, as doing so was found to result in more non-converged estimates than when density is not fixed, and use the median point estimator and BCa bootstrap error estimation. For the simulated networks, converged estimates are obtained for 87% of snowball estimates, so although we have been excluding those samples that do not converge, in fact this happens relatively infrequently. For empirical networks, however, quite often we do not obtain a converged estimate for a snowball sample, so in results we also show N_s, the number of snowball samples for which a converged estimate was obtained.

The first network is the `netscience` network science collaboration network (Newman, 2006). This network has 1589 nodes, so it is small enough that we can also estimate ERGM parameters directly for the full network with PNet, with the results shown in Table 8. This estimation took approximately 9.5 days to converge on the same hardware used for the snowball sampling experiments. This table also shows the parameters estimated using snowball sampling (2 waves, 20 seeds, 20 snowball samples) and conditional estimation, with 2 and 3 standard error confidence intervals. Running all 20 estimations in parallel, the elapsed time for this estimation is only 5 hours, and the total CPU time used by all 20 tasks is only 15 hours. As can be seen from this table, there is a good agreement between the PNet estimation and snowball sampling estimation. The snowball 2 standard error confidence intervals overlap a considerable part of the intervals given by the MLE $\pm$ 2 std. error. The most noteworthy result is the significant and very strong

Effect	PNet				Snowball							
	Estim.	Std. error	Conv. stat.		N_s	Estim.	2 std. err. C.I. lower	upper		3 std. err. C.I. lower	upper	
A2P	-0.0216	0.0085	0.0493	*	20	-0.0451	-0.1183	-0.0098	*	-0.1281	0.0380	
AT	3.8091	0.0602	-0.0401	*	20	3.7423	3.2890	4.0379	*	3.1692	4.3155	*
Edge	-7.3165	0.1335	-0.0321	*	20	-7.9978	-8.4463	-7.4144	*	-8.7875	-7.2081	*
AS	-0.7046	0.0635	-0.0784	*	20	-0.4655	-0.5938	-0.3013	*	-0.6893	-0.2417	*

Table 8: ERGM parameter estimates for the network science collaboration network (1589 nodes), estimated by PNet on the full network, and by snowball sampling and conditional estimation. The convergence statistic for PNet is obtained by simulating from the parameter estimates, extracting a sample of 1000 graphs, and comparing the observed value of each statistic with the mean from the sample distribution. For PNet, asterisks indicate significance according to the approximate Wald test for the nominal 95% confidence interval. For snowball sampling, asterisks indicate significance according to the 2 and 3 standard error confidence intervals. N_s is the number of snowball sample estimates that converged, out of 20.

alternating k-triangle (AT) effect, meaning that the network has a high degree of transitivity (closure), as we would expect in a collaboration network. When using the 3 standard error confidence interval that we have used in the previous sections, the alternating k-two-path effect (A2P) is no longer considered significant.

The other two collaboration networks we use as examples are the `condmatcollab2005` condensed matter physics collaboration network (Newman, 2001, 2004), and the `astrocollab` astrophysics collaboration network (Newman, 2001). These networks have 40 421 and 16 706 nodes, respectively, and so are too large to allow estimation of ERGM parameters for the entire network in reasonable time.

For the condensed matter collaboration network, we use the same snowball sampling parameters we have used so far (2 waves, 10 seeds, 20 snowball samples). The results are shown in Table 9. Similarly to the network science collaboration network, the result, as expected for a collaboration network, shows a significant strong alternating k-triangle effect. This estimation took approximately 23 days (with 20 parallel tasks), and as can be seen from Table 9, a converged estimate was obtained for only 15 of the 20 snowball samples.

For the astrophysics collaboration network, we use 8 seeds for each of the

Effect	N_s	Estimate	C.I.		
			lower	upper	
A2P	15	-0.0043	-0.0107	0.0020	
AT	15	5.2625	3.8186	6.7063	*
Edge	15	-10.3796	-14.3140	-6.4452	*
AS	15	-0.4868	-0.7092	-0.2644	*

Table 9: ERGM parameters for the condensed matter physics collaboration network (40 421 nodes), estimated using snowball sampling and conditional estimation. N_s is the number of snowball sample estimates that converged, out of 20. Asterisks indicate significance according to the 3 standard error confidence interval.

Effect	N_s	Estimate	C.I.		
			lower	upper	
A2P	11	-0.0091	-0.0199	0.0018	
AT	11	6.7772	2.5540	11.0004	*
Edge	11	-12.1147	-50.4044	26.1750	
AS	11	-0.0518	-3.1544	3.0508	

Table 10: ERGM parameters for the astrophysics collaboration network (16 706 nodes), estimated using snowball sampling and conditional estimation. N_s is the number of snowball sample estimates that converged, out of 20. Asterisks indicate significance according to the 3 standard error confidence interval.

snowball samples (still using 2 waves and 20 snowball samples), in order to obtain snowball samples that are small enough to be estimated in a reasonable time. Trial and error guided by experience are likely to be necessary in choosing snowball sampling parameters. As a general rule, we have tried to find parameters (specifically, the number of seed nodes; in all experiments we have used two waves) so that the mean snowball sample size is around 1000 nodes (or preferably less), with a size distribution so that not too many samples have more than 1500 nodes. For example the mean snowball sample size in the condensed matter collaboration network above is 949 nodes (range: 271–1843), and for the astrophysics collaboration network with the parameters used here, 1456 nodes (range: 554–3569).

The results for the astrophysics collaboration network are shown in Table 10. This estimation took approximately 30 days with 20 parallel tasks. Again, as expected, the most notable feature is the strong significant alter-

nating k-triangle effect. We also note that a converged estimate is obtained for only 11 over the snowball samples, which may lead to bias by discarding such a large fraction of non-converged samples.

7.2. Estimation Time

The fact that estimations are taking several days (or even weeks) and many snowball samples do not lead to converged estimates suggests there may be faster ways to obtain parameter estimates. For example, we may be better off having many more snowball samples, each of which is smaller. Therefore we also try the estimations for both networks with 100 snowball samples, but only 3 seeds in each (still using 2 waves). Note that, unlike the simulated network data (Figure 4), these distributions are very positively skewed.

For these estimations we run all 100 snowball sample conditional estimations in parallel. Table 11 shows the results after all snowball samples have either converged (or failed to converge by reaching the iteration limit or being detected as degenerate), which took 102 hours elapsed time, leading to $N_s = 63$ converged estimates. The results are consistent with those in Table 9.

Table 11 also shows the results after the first seven hours of elapsed time, at which point $N_s = 57$ samples have converged estimates. The results are completely consistent with those obtained after running for the full 102 hours. Hence the additional 95 hours, during which time only 6 more converged estimates were obtained, has not contributed much to the final model estimates.

Table 12 shows the results of a similar experiment with the astrophysics collaboration network, in which it took over 11 days for 42 of the 100 snowball samples to converge (the remaining 58 reaching the iteration limit or being found degenerate). In this case, the Edge and alternating k-star (AS) parameters are found to be significant, when they were not in the earlier experiment with fewer samples (Table 10). This table also shows the results after the first seven hours of elapsed time, at which point $N_s = 32$ samples have converged estimates. The results are completely consistent with those after the full 11 days, but with the confidence intervals (with the exception of that for the Edge parameter) reduced when using more samples. Hence the additional 272 hours of elapsed time has not changed any conclusions, but slightly increased confidence in the estimates.

Effect	Results at completion					Results after seven hours				
	N_s	Estimate	C.I.			N_s	Estimate	C.I.		
			lower	upper				lower	upper	
A2P	63	-0.0004	-0.0044	0.0037		57	0.0001	-0.0045	0.0046	
AT	63	4.3729	3.6009	5.1448	*	57	4.2161	3.4736	4.9586	*
Edge	63	-9.2179	-10.5882	-7.8477	*	57	-8.8726	-10.2970	-7.4482	*
AS	63	-0.6983	-1.1029	-0.2936	*	57	-0.8375	-1.2084	-0.4666	*

Table 11: ERGM parameters for the condensed matter collaboration network (40 421 nodes), estimated using snowball sampling (100 snowball samples, each with 3 seeds and 2 waves) and conditional estimation. N_s is the number of snowball sample estimates that converged, out of 100. Asterisks indicate significance according to the 3 standard error confidence interval. This estimation took 102 hours elapsed time (with 100 parallel tasks).

Effect	Results at completion					Results after seven hours				
	N_s	Estimate	C.I.			N_s	Estimate	C.I.		
			lower	upper				lower	upper	
A2P	42	-0.0032	-0.0084	0.0021		32	-0.0030	-0.0109	0.0049	
AT	42	4.9829	4.0210	5.9448	*	32	4.4372	3.2447	5.6298	*
Edge	42	-10.2733	-12.7103	-7.8363	*	32	-9.5459	-11.3248	-7.7669	*
AS	42	-0.6128	-1.1944	-0.0313	*	32	-0.6751	-1.3127	-0.0374	*

Table 12: ERGM parameters for the astrophysics collaboration network (16 706 nodes), estimated using snowball sampling (100 snowball samples, each with 3 seeds and 2 waves) and conditional estimation. N_s is the number of snowball sample estimates that converged, out of 100. Asterisks indicate significance according to the 3 standard error confidence interval. This estimation took 279 hours elapsed time (with 100 parallel tasks).

Figure 11 and Figure 12 show histograms of the individual snowball sample estimation times in the condensed matter and astrophysics collaboration networks, respectively. The implication of the long tail in these distributions is that, although we cannot estimate in advance how long the estimation will take, we may want to terminate the whole estimation process after some number of samples have converged, if we think we have enough converged estimates, and we do not want to take too much longer. A heuristic for deciding if enough snowball samples have been obtained is to examine a plot of the estimates (with error bars) plotted against the number of snowball samples, as shown for example in Figure 13. If the estimates have settled to a stable value then probably sufficient snowball samples have been estimated. In Figure 13, for example, clearly using fewer than 20 snowball samples is unreliable, but little improvement is obtained once around 40 snowball samples have been estimated.

For these networks, for example, we may have chosen to terminate the estimation after seven hours, or when half of the estimates have converged (or failed to converge by reaching the iteration limit or being found to be degenerate), after approximately 6 and 20 hours respectively for the two networks. Note this is another advantage of parallelism: if we ran all 100 estimations one after the other on a single processor, since we do not necessarily know which ones will take a long time, we might be unlucky and wait dozens of hours before getting any results. However by running all estimations in parallel, we can just stop whenever we have enough estimations, or at some predefined elapsed time limit, and be sure that any estimations that could have converged within that time will have done so.

8. Conclusion

We have shown that it is possible to estimate ERGM parameters for networks of much greater size than has been possible to date. Combining multiple snowball samples from a large network with a recent conditional estimation procedure to fit ERGMs (Pattison et al., 2013) orders of magnitude can be gained in possible network size and estimation time. Further, because estimation of the samples can be done in parallel, it is possible to take advantage of modern high performance computing clusters to accelerate the estimation process using parallelism.

By using simulated networks with known parameters, we have demonstrated valid statistical inference for homophily and closure. We have shown

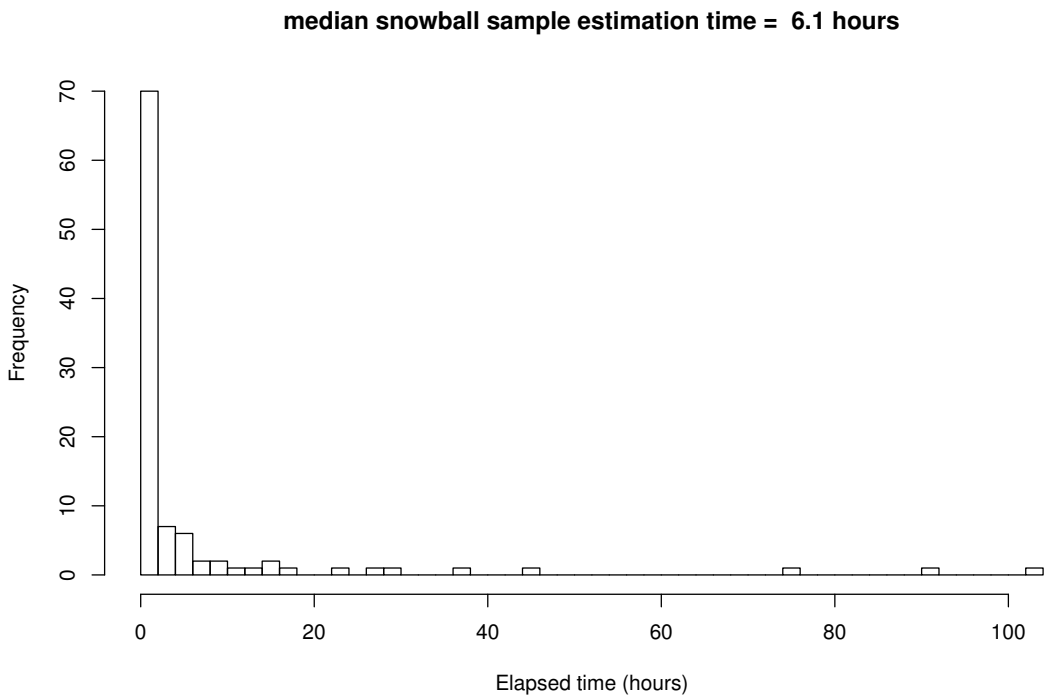

Figure 11: Histogram of snowball sample estimation times in the condensed matter collaboration network (100 samples, 3 seeds, 2 waves).

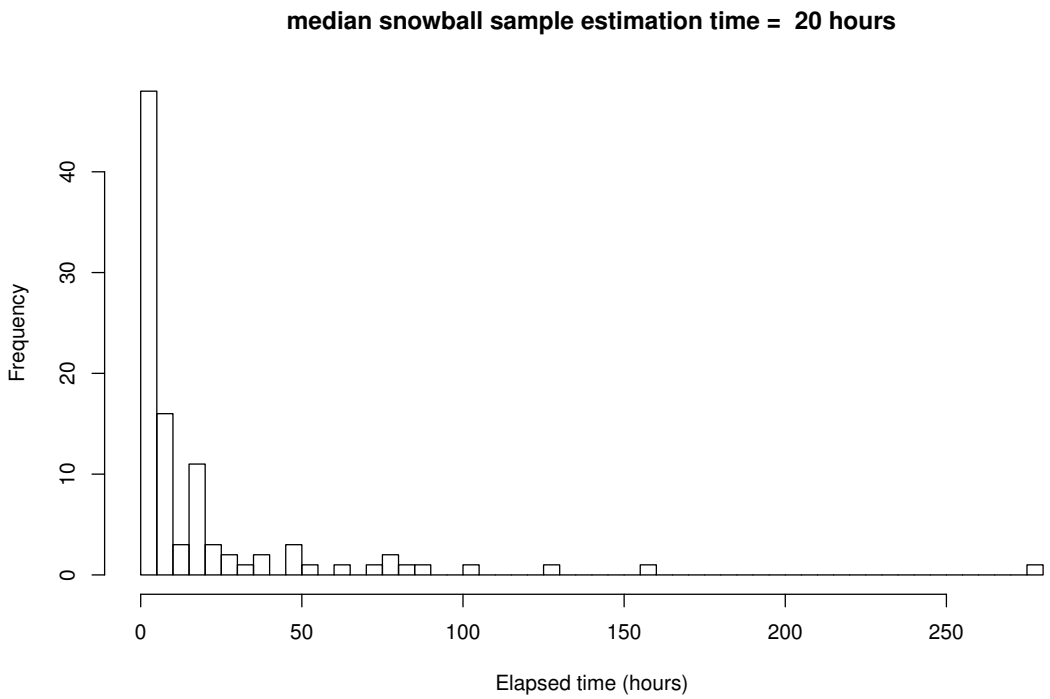

Figure 12: Histogram of snowball sample estimation times in the astrophysics collaboration network (100 samples, 3 seeds, 2 waves).

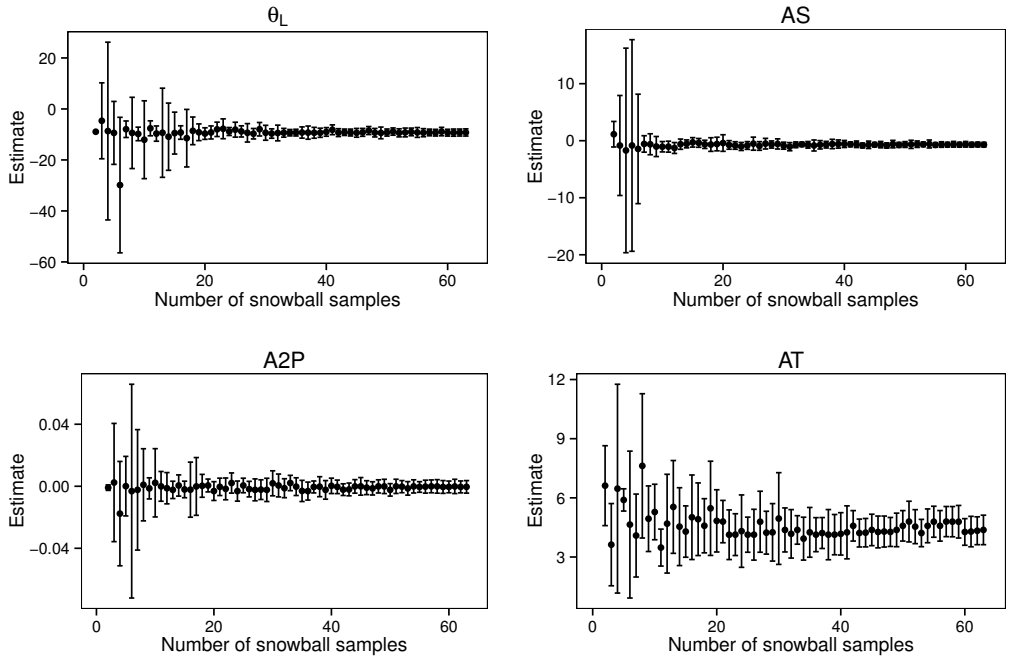

Figure 13: Estimated parameter values (median point estimator) showing 3 standard error confidence interval (BCa standard error estimator), as the number of snowball samples is increased.

that under the conditions of a positive alternating k-star parameter, power may be low, so that if the effect is statistically significant, then it can be treated as reliable, but if not, we cannot draw definite conclusions. There are reasons, both from experience (Snijders et al., 2006) and theory (Chatterjee and Diaconis, 2013), to believe that having a positive k-star parameter is more problematic than a negative one. Even in the absence of snowball sampling, using complete MLE methods such as those implemented in statnet and PNet, the alternating k-star parameter can be problematic in a similar manner.

It should be noted that the bias on some parameter estimates, specifically the alternating k-star parameter, can be very high, especially when density is not fixed. This issue is not specific to this method, but can also occur in full network MLE procedures. In such cases estimation of both the Edge and alternating k-star parameters performs poorly, resulting in low power on these parameters. Conditioning on density, thereby removing the Edge parameter from estimation, results in much improved performance in estimating the AS parameter. Development of a bias correction method would improve the method's suitability for a goodness-of-fit procedure and its applicability outside the context of parameter inference, for example simulating graph populations from estimated parameters.

Fixing the density results in a lower number of converged estimates, which itself could lead to biased results. Non-convergence can occur due to two distinct causes. First, the default settings in the stochastic approximation method (Snijders, 2002) might be inappropriate, and in principle convergence can be improved adjusting these parameters. Second, the non-convergence could be caused by an inappropriate model or a pathological sample, such as one in which there is a very strong hub. This latter situation is much more problematic, and development of a technique to handle networks with such strong hubs is ongoing work.

Merely by assuming that ERGM estimates for an entire large network are meaningful, we have assumed that the configurations of interest occur homogeneously across the network. This assumption is true by construction for our simulated networks, however large they become. For empirical networks, however, as they become larger this homogeneity assumption would become less tenable. It may be possible to test for heterogeneity by measuring the variance of the point estimates of snowball samples, and if it is too large, we might conclude that we must reject the homogeneity assumption. In such a case, it may be possible to find homogeneous subsets by isolating network

communities (Newman, 2004, 2006; Fortunato, 2010) and estimating them separately, under the assumption that individual communities are internally homogeneous, although subnetworks consisting of multiple communities are not.

There are a number of additional ways in which this work could be extended. Currently, network properties such as the density of the entire network are known but unused. How this extra information might be included to improve the estimation is left for future work. Recently, a fast technique for estimating ERGM parameters based on a "stepping" algorithm (Hummel et al., 2012) has been described. It would be interesting to do direct comparisons of estimation time and estimation error. This is left for future work also. Results from applying the method to networks with categorical and continuous node attributes are described in Appendix C; extensions for future work include the implementation of estimation of networks with dyadic covariates, the extension of the method to directed and bipartite graphs, and the development of a goodness-of-fit procedure that can be practically used on very large networks. Work on some of these extensions is currently under way.

Appendix A. Correlation between pooled estimator and full data estimator

For illustration consider the simple example of estimating expected values for n independent and identically distributed variates $X_i \overset{iid}{\sim} X$, with $E(X) = \mu$ and $V(X) = \sigma^2$. Assume that we have one estimator based on all of the n observations, $\bar{X} = \frac{1}{n}\sum X_i$, corresponding to the complete data MLE in our example. Assume further that we have a pooled estimate based on estimators for k mutually exclusive subsets $A_j \subset \{1, \ldots, n\}$, each of size $|A_j| = m$. Let $\bar{Y} = \frac{1}{k}\sum_j Y_j$ be a pooled estimate based on the partial means $Y_j = \sum_{i \in A_j} X_i/m$ (that are clearly independent). The correlation between the pooled estimator and the estimator using all of the data only depends on the sampling fraction. More specifically $\mathrm{cor}(\bar{Y}, \bar{X}) = \sqrt{km/n}$, something which is straightforward to check after noting that we can write

$$\bar{Y}\bar{X} = \frac{1}{nkm}\left(\sum_{i \in A_1} X_i^2 + \sum_{i \in A_1} X_i \sum_{h \neq i} X_h + \cdots + \sum_{i \in A_k} X_i^2 + \sum_{i \in A_k} X_i \sum_{h \neq i} X_h\right)$$

resulting in $E(\bar{Y}\bar{X}) = \sigma^2/n + \mu^2$.

This means that if we have a Bernoulli graph with link-probability μ, then the correlation between a pooled estimate and the density, $\bar{X}$, will not depend on μ. If the sample sizes for the partitions are m_j rather then equal, the same result applies with $km = m_1 + \cdots + m_k$. Similar results may be obtained for estimators that are more general functions of variables that are not necessarily independent within subsets.

The interpretation is that under the assumption that the network is homogenous and that estimates $\hat{\theta}_j$ are independent, then $\hat{\mu}_\theta^{\text{WLS}}$ should not be interpreted as an approximation to the complete data MLE. This was confirmed in pairwise comparisons of estimates in Figure 10. For component ERGMs investigation (obtainable upon request) shows that there may be almost as much variation between the pooled estimate (based on Snijders (2010)) and the complete data MLE as there is variation across pooled estimates even if the latter are unbiased.

Appendix B. Weighted least squares (WLS) point estimator results

Table B.13, Table B.14, Table B.15, and Table B.16 show the results of estimating the simulated networks using the WLS point estimator. The results for the median point estimator are in the main text.

N	Attributes	Effect	Bias	RMSE	Type II error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	None	A2P	0.0126	0.0180	0	0	4	0.0129	12.74
5000	None	AT	-0.0217	0.0272	0	0	4	0.0165	12.74
5000	None	AS	0.1207	0.1780	70	60	78	0.1315	12.74
5000	70/30	A2P	0.0185	0.0205	0	0	4	0.0089	19.02
5000	70/30	AT	-0.0198	0.0226	0	0	4	0.0110	19.02
5000	70/30	AS	0.1432	0.1794	30	22	40	0.1085	19.02
5000	70/30	ρ	-0.0851	0.0945	51	41	61	0.0412	19.02
5000	70/30	ρ_B	-0.0048	0.0434	0	0	4	0.0434	19.02
5000	50/50 balanced	A2P	0.0075	0.0149	0	0	4	0.0130	15.34
5000	50/50 balanced	AT	-0.0190	0.0252	0	0	4	0.0167	15.34
5000	50/50 balanced	AS	0.1198	0.1951	66	56	75	0.1548	15.34
5000	50/50 balanced	ρ	-0.0108	0.0398	0	0	4	0.0385	15.34
5000	50/50 balanced	ρ_B	0.0041	0.0493	0	0	4	0.0493	15.34
5000	50/50	A2P	0.0193	0.0217	0	0	4	0.0098	17.96
5000	50/50	AT	-0.0228	0.0263	0	0	4	0.0131	17.96
5000	50/50	AS	0.1075	0.1607	49	39	59	0.1201	17.96
5000	50/50	ρ	-0.0728	0.0811	16	10	24	0.0360	17.96
5000	50/50	ρ_B	-0.0014	0.0380	0	0	4	0.0382	17.96
10000	None	A2P	0.0034	0.0146	0	0	4	0.0143	8.56
10000	None	AT	-0.0074	0.0218	0	0	4	0.0206	8.56
10000	None	AS	0.1554	0.2380	66	56	75	0.1812	8.56

Table B.13: Error statistics and number of converged snowball samples (out of 20) for the simulated networks, including false negatives (type II error rate) over the 100 simulated networks, using the WLS point estimator with fixed density. True parameter values for each model are shown in Table 1.

N	Attributes	Effect	Bias	RMSE	Type II error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	None	A2P	0.0447	0.0528	1	0	5	0.0283	20.00
5000	None	AT	-0.0088	0.0359	0	0	4	0.0350	20.00
5000	None	Edge	-6.7140	9.5740	50	40	60	6.8590	20.00
5000	None	AS	1.4810	2.2950	81	72	87	1.7620	20.00
5000	70/30	A2P	0.0285	0.0361	1	0	5	0.0222	19.88
5000	70/30	AT	-0.0054	0.0305	0	0	4	0.0302	19.88
5000	70/30	Edge	-7.4800	13.0300	61	51	70	10.7200	19.88
5000	70/30	AS	1.7090	3.2010	78	69	85	2.7200	19.88
5000	70/30	ρ	-0.0235	0.1069	87	79	92	0.1048	19.88
5000	70/30	ρ_B	-0.0040	0.1075	2	1	7	0.1079	19.88
5000	50/50 balanced	A2P	0.0401	0.0511	3	1	8	0.0318	17.68
5000	50/50 balanced	AT	-0.0075	0.0411	0	0	4	0.0406	17.68
5000	50/50 balanced	Edge	-4.5100	7.1260	64	54	73	5.5450	17.68
5000	50/50 balanced	AS	0.9884	1.7490	86	78	91	1.4510	17.68
5000	50/50 balanced	ρ	-0.0192	0.0789	31	23	41	0.0769	17.68
5000	50/50 balanced	ρ_B	-0.0018	0.0974	2	1	7	0.0978	17.68
5000	50/50	A2P	0.0269	0.0341	1	0	5	0.0210	20.00
5000	50/50	AT	0.0019	0.0353	0	0	4	0.0354	20.00
5000	50/50	Edge	-10.3500	13.1200	44	35	54	8.1040	20.00
5000	50/50	AS	2.4510	3.1870	71	61	79	2.0470	20.00
5000	50/50	ρ	-0.0515	0.0908	85	77	91	0.0751	20.00
5000	50/50	ρ_B	0.0155	0.0881	0	0	4	0.0871	20.00
10000	None	A2P	0.0257	0.0388	1	0	5	0.0293	19.65
10000	None	AT	-0.0123	0.0423	0	0	4	0.0406	19.65
10000	None	Edge	-21.6800	33.4400	63	53	72	25.5900	19.65
10000	None	AS	5.2050	8.2230	75	66	82	6.3980	19.65

Table B.14: Error statistics and number of converged snowball samples (out of 20) for the simulated networks, including false negatives (type II error rate) over the 100 simulated networks, using the WLS point estimator, when density is not fixed. True parameter values for each model are shown in Table 1.

N	Attributes	Effect	Bias	RMSE	Type I error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	None	AT	0.0446	0.0674	14	9	22	0.0509	18.94
5000	None	AS	0.2764	0.3781	6	3	12	0.2593	10.16
5000	70/30	AT	0.0293	0.0469	8	4	15	0.0368	19.96
5000	70/30	AS	0.2162	0.2541	6	3	12	0.1342	16.08
5000	70/30	ρ	-0.0510	0.0732	10	6	17	0.0529	18.17
5000	70/30	ρ_B	-0.0039	0.0536	3	1	8	0.0537	18.17
5000	50/50 balanced	AT	0.0486	0.0723	25	18	34	0.0538	19.16
5000	50/50 balanced	AS	0.2493	0.3944	7	3	14	0.3072	10.48
5000	50/50 balanced	ρ	-0.0310	0.0507	9	5	16	0.0404	16.96
5000	50/50 balanced	ρ_B	0.0030	0.0554	4	2	10	0.0556	15.17
5000	50/50	AT	0.0286	0.0475	7	3	14	0.0382	19.87
5000	50/50	AS	0.2349	0.2827	10	6	17	0.1581	14.00
5000	50/50	ρ	-0.0310	0.0507	9	5	16	0.0404	16.96
5000	50/50	ρ_B	0.0109	0.0529	7	3	14	0.0520	17.25

Table B.15: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with a zero effect to find false positives (type I error rate) over the 100 simulated networks, using the WLS point estimator with fixed density.

N	Attributes	Effect	Bias	RMSE	Type I error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	None	AT	0.3156	0.3243	91	84	95	0.0752	18.16
5000	None	AS	0.9794	1.2320	22	15	31	0.7506	19.78
5000	70/30	AT	0.2198	0.2340	71	61	79	0.0809	16.71
5000	70/30	AS	0.4580	1.3980	16	10	24	1.3270	18.70
5000	70/30	ρ	-0.0495	0.1273	3	1	8	0.1178	19.11
5000	70/30	ρ_B	0.0155	0.1041	0	0	4	0.1034	19.23
5000	50/50 balanced	AT	0.3395	0.3484	90	83	94	0.0785	13.28
5000	50/50 balanced	AS	0.4672	1.1050	7	4	14	1.0070	14.13
5000	50/50 balanced	ρ	-0.0289	0.0831	1	0	5	0.0783	18.71
5000	50/50 balanced	ρ_B	0.0114	0.0991	1	0	5	0.0989	17.22
5000	50/50	AT	0.2521	0.2622	76	67	83	0.0723	15.94
5000	50/50	AS	0.9350	1.2650	16	10	24	0.8559	17.03
5000	50/50	ρ	-0.0289	0.0831	1	0	5	0.0783	18.71
5000	50/50	ρ_B	0.0080	0.1076	1	0	5	0.1078	18.07

Table B.16: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with a zero effect to find false positives (type I error rate) over the 100 simulated networks, using the WLS point estimator, when density is not fixed.

N	Attributes	Edge (θ_L)	AS	AT	A2P	Match	Difference
5000	Categorical	-4.0	0.2	1.0	-0.2	0.5	
5000	Continuous	-4.0	0.2	1.0	-0.2		0.5

Table C.17: Parameters of the simulated networks with categorical and continuous attributes.

N	Attributes	Mean components	Mean degree	Mean density	Mean global clustering coefficient
5000	Categorical	1.00	9.18	0.00184	0.02609
5000	Continuous	1.00	10.13	0.00203	0.02754

Table C.18: Statistics of the simulated networks with categorical and continuous attributes.

Appendix C. Continuous and categorical attributes

In the main text we addressed only binary node attributes. Here we describe early results of applying the method to simulated networks with continuous and categorical node attributes. For categorical attributes, we simulated networks with a categorical attribute on each node, taking one of three possible values. The value of the categorical attribute at each node is assigned uniformly at random. For continuous attributes, the attribute value v_i at each node i is $v_i \overset{iid}{\sim} N(0, 1)$. The parameters of the simulated networks are shown in Table C.17. For the categorical attribute, the networks are simulated with the same parameters as the networks described in the main text, but with an additional Match parameter, for homophily on the categorical attribute. For the continuous attribute, the networks are again simulated with the same structural parameters, but with the additional Difference parameter, for heterophily on the continuous attribute (the network statistic for an edge between nodes i and j is equal to $|v_i - v_j|$). The graph summary statistics of the simulated networks are shown in Table C.18.

The estimations described here were run on the Gordon Compute Cluster at the San Diego Supercomputer Center (SDSC), an Extreme Science and Engineering Discovery Environment (XSEDE) facility (Towns et al., 2014). We run each experiment on a single node with 16 parallel tasks thereby using all cores on a node; as 20 snowball samples are estimated, some tasks must process two samples.

N	Attributes	Effect	Bias	RMSE	Type II error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	Continuous	A2P	0.0126	0.0156	0	0	4	0.0091	19.60
5000	Continuous	AT	-0.0146	0.0185	0	0	4	0.0113	19.60
5000	Continuous	Difference	-0.0229	0.0270	0	0	4	0.0145	19.60
5000	Continuous	AS	0.1289	0.1695	36	27	46	0.1106	19.60
5000	Categorical	A2P	0.0112	0.0151	0	0	4	0.0102	19.03
5000	Categorical	AT	-0.0140	0.0201	0	0	4	0.0144	19.03
5000	Categorical	AS	0.1380	0.1823	49	39	59	0.1196	19.03
5000	Categorical	Match	-0.0010	0.0226	0	0	4	0.0227	19.03

Table C.19: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with continuous and categorical attributes, including false negatives (type II error rate) over the 100 simulated networks, using the WLS point estimator with fixed density. True parameter values for each model are shown in Table C.17.

N	Attributes	Effect	Bias	RMSE	Type II error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	Continuous	A2P	0.0075	0.0126	0	0	4	0.0101	19.60
5000	Continuous	AT	-0.0149	0.0197	0	0	4	0.0129	19.60
5000	Continuous	Difference	-0.0211	0.0259	0	0	4	0.0152	19.60
5000	Continuous	AS	0.1519	0.2019	46	37	56	0.1337	19.60
5000	Categorical	A2P	0.0041	0.0134	0	0	4	0.0128	19.03
5000	Categorical	AT	-0.0157	0.0220	0	0	4	0.0155	19.03
5000	Categorical	AS	0.1807	0.2374	64	54	73	0.1548	19.03
5000	Categorical	Match	-0.0024	0.0258	0	0	4	0.0259	19.03

Table C.20: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with continuous and categorical attributes, including false negatives (type II error rate) over the 100 simulated networks, using the median point estimator with fixed density. True parameter values for each model are shown in Table C.17.

N	Attributes	Effect	Bias	RMSE	Type II error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	Continuous	A2P	0.0289	0.0329	0	0	4	0.0158	19.97
5000	Continuous	AT	0.0045	0.0278	0	0	4	0.0276	19.97
5000	Continuous	Difference	-0.0143	0.0416	0	0	4	0.0392	19.97
5000	Continuous	Edge	-8.7130	14.4500	69	59	77	11.5900	19.97
5000	Continuous	AS	2.0060	3.5690	82	73	88	2.9670	19.97
5000	Categorical	A2P	0.0385	0.0461	0	0	4	0.0255	19.96
5000	Categorical	AT	0.0003	0.0347	0	0	4	0.0349	19.96
5000	Categorical	Edge	-7.8220	11.4500	65	55	74	8.4010	19.96
5000	Categorical	AS	1.7560	2.7610	81	72	87	2.1410	19.96
5000	Categorical	Match	-0.0000	0.0423	0	0	4	0.0425	19.96

Table C.21: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with continuous and categorical attributes, including false negatives (type II error rate) over the 100 simulated networks, using the WLS point estimator, when density is not fixed. True parameter values for each model are shown in Table C.17.

Table C.19 and Table C.20 show the results using the WLS and median point estimators, respectively, when density is fixed in estimation. The type II error rate is extremely good on all parameters except the problematic alternating k-star parameter.

Table C.21 and Table C.22 show the results for the WLS and median point estimators, respectively, when density is not fixed in estimation. Again, the type II error rate is excellent on all parameters except alternating k-star and Edge.

Just as for the structural parameters and binary attributes described in the main text, in order to estimate the type I error rate we simulated networks with the parameter for each effect in turn set to zero. Table C.23 and Table C.24 show the results for the WLS and median point estimators, respectively, when density is fixed in estimation. The type I error rate on parameters other than alternating k-star is good.

Table C.25 and Table C.26 show the results for the WLS and median point estimators, respectively, when density is not fixed in estimation. In addition to the problematic alternating k-star parameter (for the WLS point estimator), the type I error rate for the alternating k-triangle parameter is also very high.

In summary, the method has very good power and low type I error rate

N	Attributes	Effect	Bias	RMSE	Type II error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	Continuous	A2P	0.0196	0.0290	0	0	4	0.0215	19.97
5000	Continuous	AT	0.0078	0.0376	0	0	4	0.0369	19.97
5000	Continuous	Difference	-0.0039	0.0448	0	0	4	0.0449	19.97
5000	Continuous	Edge	9.6560	17.4600	99	95	100	14.6200	19.97
5000	Continuous	AS	-2.5890	4.4840	100	96	100	3.6800	19.97
5000	Categorical	A2P	0.0270	0.0403	2	1	7	0.0301	19.96
5000	Categorical	AT	0.0022	0.0441	0	0	4	0.0443	19.96
5000	Categorical	Edge	9.7830	15.9500	100	96	100	12.6600	19.96
5000	Categorical	AS	-2.6050	4.1610	100	96	100	3.2620	19.96
5000	Categorical	Match	-0.0010	0.0521	0	0	4	0.0524	19.96

Table C.22: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with continuous and categorical attributes, including false negatives (type II error rate) over the 100 simulated networks, using the median point estimator, when density is not fixed. True parameter values for each model are shown in Table C.17.

N	Attributes	Effect	Bias	RMSE	Type I error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	Continuous	AT	0.0213	0.0419	5	2	11	0.0362	19.97
5000	Continuous	Difference	-0.0040	0.0187	8	4	15	0.0184	19.23
5000	Continuous	AS	0.2372	0.2597	12	7	20	0.1063	18.11
5000	Categorical	AT	0.0309	0.0524	7	3	14	0.0425	19.75
5000	Categorical	AS	0.2863	0.3388	13	8	21	0.1820	16.71
5000	Categorical	Match	-0.0010	0.0236	3	1	8	0.0237	18.79

Table C.23: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with continuous and categorical attributes, with a zero effect to find false positives (type I error rate) over the 100 simulated networks, using the WLS point estimator with fixed density.

N	Attributes	Effect	Bias	RMSE	Type I error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	Continuous	AT	-0.0150	0.0490	2	1	7	0.0468	19.97
5000	Continuous	Difference	-0.0026	0.0201	4	2	10	0.0200	19.23
5000	Continuous	AS	0.2654	0.3007	11	6	19	0.1420	18.11
5000	Categorical	AT	-0.0180	0.0631	7	3	14	0.0607	19.75
5000	Categorical	AS	0.2787	0.3627	11	6	19	0.2332	16.71
5000	Categorical	Match	-0.0022	0.0283	4	2	10	0.0284	18.79

Table C.24: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with continuous and categorical attributes, with a zero effect to find false positives (type I error rate) over the 100 simulated networks, using the median point estimator with fixed density.

N	Attributes	Effect	Bias	RMSE	Type I error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	Continuous	AT	0.2184	0.2312	72	63	80	0.0763	17.31
5000	Continuous	Difference	-0.0067	0.0338	2	1	7	0.0333	19.95
5000	Continuous	AS	1.2280	1.7700	26	18	35	1.2800	19.95
5000	Categorical	AT	0.2767	0.2853	86	78	91	0.0700	15.65
5000	Categorical	AS	1.1010	1.5340	20	13	29	1.0730	19.68
5000	Categorical	Match	0.0066	0.0485	2	1	7	0.0483	19.97

Table C.25: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with continuous and categorical attributes, with a zero effect to find false positives (type I error rate) over the 100 simulated networks, using the WLS point estimator, when density is not fixed.

N	Attributes	Effect	Bias	RMSE	Type I error rate (%)			Std. dev.	Mean
					Estim.	95% C.I.		estimate	samples
						lower	upper		converged
5000	Continuous	AT	-0.2874	0.6739	17	11	26	0.6126	17.31
5000	Continuous	Difference	-0.0048	0.0394	1	0	5	0.0393	19.95
5000	Continuous	AS	-1.3880	2.2940	0	0	4	1.8350	19.95
5000	Categorical	AT	-0.5430	0.8852	23	16	32	0.7027	15.65
5000	Categorical	AS	-1.0360	1.9850	1	0	5	1.7020	19.68
5000	Categorical	Match	-0.0074	0.0657	0	0	4	0.0656	19.97

Table C.26: Error statistics and number of converged snowball samples (out of 20) for the simulated networks with continuous and categorical attributes, with a zero effect to find false positives (type I error rate) over the 100 simulated networks, using the median point estimator, when density is not fixed.

for homophily (or heterophily for the continuous attribute) on attribute parameters in our simulated networks. However when density is not fixed the type I error rate for the alternating k-triangle parameter is very high, as noted in Section 6 of the main text.

References

Caimo, A., Friel, N., 2011. Bayesian inference for exponential random graph models. Social Networks 33, 41–55.

Chatterjee, S., Diaconis, P., 2013. Estimating and understanding exponential random graph models. The Annals of Statistics 41, 2428–2461.

Coleman, J.S., 1958. Relational analysis: the study of social organizations with survey methods. Human Organization 17, 23–36.

Corander, J., Dahmström, K., Dahmström, P., 1998. Maximum likelihood estimation for Markov graphs. Technical Report 8. Stockholm University, Department of Statistics.

Corander, J., Dahmström, K., Dahmström, P., 2002. Maximum likelihood estimation for exponential random graph models, in: Hagberg, J. (Ed.), Contributions to Social Network Analysis, Information Theory, and Other Topics in Statistics; A Festschrift in honour of Ove Frank. Department of Statistics, University of Stockholm, pp. 1–17.

Csárdi, G., Nepusz, T., 2006. The igraph software package for complex network research. InterJournal Complex Systems, 1695. URL: `http://igraph.sf.net`.

Davison, A.C., Hinkley, D.V., 1997. Bootstrap Methods and Their Application. Cambridge University Press, Cambridge.

Efron, B., 1987. Better bootstrap confidence intervals. Journal of the American Statistical Association 82, 171–185.

Fortunato, S., 2010. Community detection in graphs. Physics Reports 486, 75–174.

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

Gilbert, E.N., 1959. Random graphs. The Annals of Mathematical Statistics 30, 1141–1144.

Goodman, L.A., 1961. Snowball sampling. The Annals of Mathematical Statistics , 148–170.

Goodman, L.A., 2011. Comment: On respondent-driven sampling and snowball sampling in hard-to-reach populations and snowball sampling not in hard-to-reach populations. Sociological Methodology 41, 347–353.

Goodreau, S.M., 2007. Advances in exponential random graph (p^*) models applied to a large social network. Social Networks 29, 231–248.

Handcock, M.S., Gile, K.J., 2010. Modeling social networks from sampled data. The Annals of Applied Statistics 4, 5–25.

Handcock, M.S., Gile, K.J., 2011. Comment: On the concept of snowball sampling. Sociological Methodology 41, 367–371.

Handcock, M.S., Hunter, D.R., Butts, C.T., Goodreau, S.M., Morris, M., 2008. statnet: Software tools for the representation, visualization, analysis and simulation of network data. Journal of Statistical Software 24, 1548. URL: `http://www.jstatsoft.org/v24/i01`.

Heckathorn, D.D., 2011. Comment: Snowball versus respondent-driven sampling. Sociological Methodology 41, 355–366.

Hummel, R.M., Hunter, D.R., Handcock, M.S., 2012. Improving simulation-based algorithms for fitting ERGMs. Journal of Computational and Graphical Statistics 21, 920–939.

Hunter, D.R., 2007. Curved exponential family models for social networks. Social Networks 29, 216–230.

Hunter, D.R., Handcock, M.S., 2006. Inference in curved exponential family models for networks. Journal of Computational and Graphical Statistics 15, 565–583.

Illenberger, J., Flötteröd, G., 2012. Estimating network properties from snowball sampled data. Social Networks 34, 701–711.

Kurant, M., Markopoulou, A., Thiran, P., 2011. Towards unbiased BFS sampling. IEEE Journal on Selected Areas in Communications 29, 1799–1809.

Lee, S.H., Kim, P.J., Jeong, H., 2006. Statistical properties of sampled networks. Physical Review E 73, 016102.

Lubbers, M.J., Snijders, T.A., 2007. A comparison of various approaches to the exponential random graph model: A reanalysis of 102 student networks in school classes. Social Networks 29, 489–507.

Lusher, D., Koskinen, J., Robins, G. (Eds.), 2013. Exponential Random Graph Models for Social Networks. Structural Analysis in the Social Sciences, Cambridge University Press, New York.

Newman, M.E.J., 2001. The structure of scientific collaboration networks. Proceedings of the National Academy of Sciences of the USA 98, 404–409.

Newman, M.E.J., 2003. Ego-centered netwoks and the ripple effect. Social Networks 25, 83–95.

Newman, M.E.J., 2004. Fast algorithm for detecting community structure in networks. Physical Review E 69, 066133.

Newman, M.E.J., 2006. Finding community structure in networks using the eigenvectors of matrices. Physical Review E 74, 036104.

Pattison, P.E., Robins, G.L., Snijders, T.A.B., Wang, P., 2013. Conditional estimation of exponential random graph models from snowball sampling designs. Journal of Mathematical Psychology 57, 284–296.

R Core Team, 2013. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria. URL: `http://www.R-project.org`.

Ripley, R.M., Snijders, T.A., Boda, Z., Vörös, A., Preciado, P., 2014. Manual for RSIENA. University of Oxford, Department of Statistics; Nuffield College. (RSiena version 4.0, manual version June 26, 2014).

Robins, G., Elliott, P., Pattison, P., 2001. Network models for social selection processes. Social Networks 23, 1–30.

Robins, G., Snijders, T., Wang, P., Handcock, M., Pattison, P., 2007. Recent developments in exponential random graph (p^*) models for social networks. Social Networks 29, 192–215.

Rolls, D.A., Wang, P., Jenkinson, R., Pattision, P.E., Robins, G.L., Sacks-Davis, R., Daraganova, G., Hellard, M., McBryde, E., 2013. Modelling a disease-relevant contact network of people who inject drugs. Social Networks 35, 699–710.

Shalizi, C.R., Rinaldo, A., 2013. Consistency under sampling of exponential random graph models. The Annals of Statistics 41, 508–535.

Snijders, T., van Duijn, M., 2002. Conditional maximum likelihood estimation under various specifications of exponential random graph models, in: Hagberg, J. (Ed.), Contributions to Social Network Analysis, Information Theory, and Other Topics in Statistics; A Festschrift in honour of Ove Frank, pp. 117–134.

Snijders, T.A.B., 2002. Markov chain Monte Carlo estimation of exponential random graph models. Journal of Social Structure 3, 1–40.

Snijders, T.A.B., 2010. Conditional marginalization for exponential random graph models. Journal of Mathematical Sociology 34, 239–252.

Snijders, T.A.B., Baerveldt, C., 2003. A multilevel network study of the effects of delinquent behavior on friendship evolution. Journal of Mathematical Sociology 27, 123–151.

Snijders, T.A.B., Pattison, P.E., Robins, G.L., Handcock, M.S., 2006. New specifications for exponential random graph models. Sociological Methodology 36, 99–153.

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

Thompson, S.K., Frank, O., 2000. Model-based estimation with link-tracing sampling designs. Survey Methodology 26, 87–98.

Towns, J., Cockerill, T., Dahan, M., Foster, I., Gaither, K., Grimshaw, A., Hazlewood, V., Lathrop, S., Lifka, D., Peterson, G.D., et al., 2014. XSEDE: accelerating scientific discovery. Computing in Science & Engineering 16, 62–74.

Wang, P., Robins, G., Pattison, P., 2009. PNet: program for the simulation and estimation of exponential random graph (p^*) models. Department of Psychology, The University of Melbourne.

Wickham, H., 2009. ggplot2: elegant graphics for data analysis. Springer New York. URL: `http://had.co.nz/ggplot2/book`.

Wilson, E.B., 1927. Probable inference, the law of succession, and statistical inference. Journal of the American Statistical Association 22, 209–212.

Xu, B., Huang, Y., Contractor, N., 2013. Exploring twitter networks in parallel computing environments, in: Proceedings of the Conference on Extreme Science and Engineering Discovery Environment (XSEDE13): Gateway To Discovery, ACM. p. 20.

1. Acknowledgments

This research was supported by a Victorian Life Sciences Computation Initiative (VLSCI) grant numbers VR0261 and VR0297 on its Peak Computing Facility at the University of Melbourne, an initiative of the Victorian Government, Australia. This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575. Specifically, we used the Gordon Compute Cluster at SDSC under allocation TG-SES140024 "Exponential Random Graph Models for Large Networks: Snowball Sampling and Conditional Estimation using Parallel High Performance Computing". We also used the University of Melbourne ITS High Performance Computing facilities.